Latent Space Representations of Hypergraphs

Kathryn Turnbull[1], Simón Lunagómez[2], Christopher Nemeth[2], and Edoardo Airoldi[3]

[1]STOR-i Centre for Doctoral Training, Lancaster University
[2]Department of Mathematics and Statistics, Lancaster University
[3]Fox School of Business, Temple University

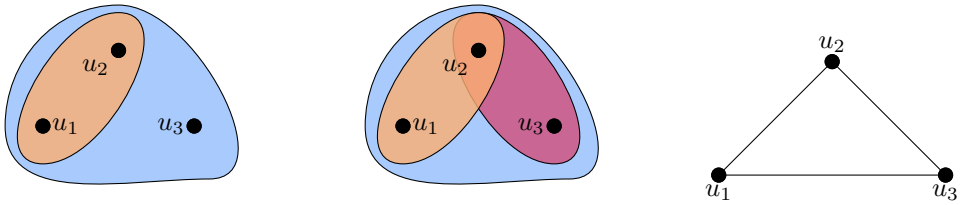
Abstract

The increasing prevalence of relational data describing interactions among a target population has motivated a wide literature on statistical network analysis. In many applications, interactions may involve more than two members of the population and this data is more appropriately represented by a hypergraph. In this paper we present a model for hypergraph data which extends the latent space distance model of Hoff et al. (2002) and, by drawing a connection to constructs from computational topology, we develop a model whose likelihood is inexpensive to compute. We obtain posterior samples via an MCMC scheme and we rely on Bookstein coordinates to remove the identifiability issues associated with the latent representation. We demonstrate that the latent space construction imposes desirable properties on the hypergraphs generated in our framework and provides a convenient visualisation of the data. Furthermore, through simulation, we investigate the flexibility of our model and consider estimating predictive distributions. Finally, we explore the application of our model to a real world co-occurrence dataset.

Keywords: Hypergraphs, Latent Space Networks, Simplicial complex, Bayesian Inference, Statistical Network Analysis.

1 Introduction

In this paper we present a model for relational data which describe interactions involving several members of a target population. Our focus is on modelling hypergraphs comprised of N nodes and M hyperedges, where a hyperedge corresponds to a set of nodes, and we assume throughout that hyperedges are modelled randomly given a fixed collection of labelled nodes. A common approach to modelling relationships involving more than two nodes is to project the hypergraph onto a graph in which the connections are assumed to occur between pairs of nodes only. Representing the data as a graph, so that each hyperedge is replaced by a clique, allows the data to be analysed according to an extensive graph modelling literature (see Kolaczyk (2009), Barabási and Pósfai (2016) and Salter-Townshend et al. (2012)) which includes the stochastic blockmodel (Holland et al. (1983)), exponential random graphs (Holland and Leinhardt (1981)), random graph models (Erdös and Rényi (1959), Barabási and Albert (1999)), and latent space network models (Hoff et al. (2002)). However, it is clear that representing a hypergraph as a graph results in a loss of information (see Figure 1) and, although several models for hypergraphs have been introduced (see Stasi et al. (2014), Ng and Murphy (2018), and Liu et al. (2013)), this literature is currently



(a) $\{1,2\},\{1,2,3\}$ (b) $\{1,2\},\{2,3\},\{1,2,3\}$ (c) $\{1,2\},\{1,3\},\{2,3\}$

Figure 1: Figures 1a and 1b show examples of a hypergraph on the nodes u_1, u_2 and u_3. A shaded region represents a hyperedge, and a node belongs to a hyperedge if it lies within the associated shaded region. The graph obtained by connecting i and j if they are contained within the same hyperedge is given in Figure 1c for the hypergraphs in Figures 1a and 1b. Note that we cannot recover the hyperedges in Figures 1a and 1b from the the graph in Figure 1c.

less mature. Here we introduce a model for hypergraph data by considering the extension of the latent space approach for graphs as introduced in Hoff et al. (2002). In this framework the connections are modelled as a function of latent coordinates associated with the nodes, and this construction has many desirable properties which we wish to exploit when developing our model. In particular, the latent representation provides an intuitive visualisation of the graph, allows control in the joint distribution of subgraph counts, and can encourage transitive relationships.

Hypergraph data arise in a range of disciplines (see Kunegis (2013) and Leskovec and Krevl (2014)) including systems biology, neuroscience and marketing, and, depending on the context, the interactions may have different interpretations. For example, an interaction may indicate online communications, professional cooperation between individuals, or dependence between random variables. As a motivating example, consider coauthorship between academics where a connection indicates which authors contributed to an article. We let the nodes represent the population of academics and, since multiple academics may contribute to an article, it is natural to represent a publication by a hyperedge. Figure 1a shows a possible hypergraph relationship between three authors, where a shaded region indicates which authors contributed to an article. Whilst a range of inferential questions can be posed about a hypergraph, we focus on the following.

(Q1) *"Conditional on the observed relationships, how do we expect a new set of new nodes to interact with the hypergraph?"*. In the context of coauthorship, this is equivalent to asking who additional authors will collaborate with given the papers that have already been written. Once we have fitted a model to our data, this question translates into a prediction problem. The model we introduce represents the nodes of the hypergraph in a latent space, and this allows us to explore predictive distributions by simulating new nodes in the latent space and examining their connections.

(Q2) *"Which authors have a greater importance in the hypergraph?"*. By associating each node with a latent coordinate, we are able to determine a visualisation of the hypergraph from our model. In latent space models, it is typical for the nodes with greater degree to be placed more centrally in the latent representation. Hence, we can determine the importance of a node by examining its latent coordinate with respect to the full latent representation. This point will be discussed further in Section 8.

To address the inferential questions in (Q1) and (Q2) we develop a model for hypergraph data that builds upon the latent space framework for graphs introduced in Hoff et al. (2002). There

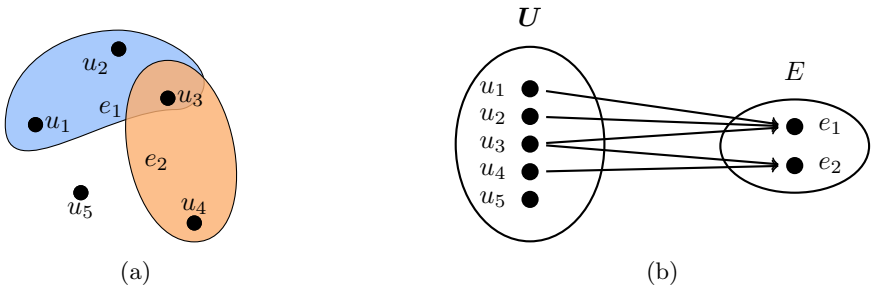


Figure 2: Alternative representations of the same hypergraph. In the left panel, the nodes represent the population and the shaded regions represent the hyperedges. If a node is contained within a shaded region it belongs to the associated hyperedge. The right panel shows a bipartite graph where the node sets U and E correspond to the population and hyperedges, respectively. An edge connecting an element of U with an element of E indicates that the member of the population belongs to the hyperedge.

exists a rich latent space network modelling literature (see Krivitsky et al. (2009), Handcock et al. (2007), Friel et al. (2016), and Kim et al. (2017)) and, although properties of these models are well understood (see Rastelli et al. (2015)), their extension to the hypergraph setting is largely unexplored. In the distance model of Hoff et al. (2002), nodes are more likely to be connected if their latent coordinates lie close together in a Euclidean sense. Since the Euclidean distance satisfies the triangle inequality, transitive relationships in which "friends of friends are likely to be friends" are likely to occur. We wish to take advantage of properties analogous to this in the hypergraph setting, and this leads us to consider (Q3). In general, it is unclear how to impose properties on a hypergraph when a bipartite representation is used (see Figure 2b). Hence, when developing our model, we rely on the representation shown in Figure 2a.

(Q3) *"How do we formulate a latent space model for hypergraphs?"*. Extending the distance model of Hoff et al. (2002) to the hypergraph setting is non-trivial and we wish to develop a model which uses constraints implied by the latent space to impose properties on the hypergraphs. For instance, recall the motivating example of coauthorship and consider the hypergraph depicted in Figure 2a. In this example, notice that u_1 and u_3 are connected by e_1 and that u_3 and u_4 are connected by e_2. It is intuitive that authors u_1 and u_4 are likely to collaborate, since both have worked with author u_3. However, since we are considering hypergraphs, there are multiple collaborations in which u_1 and u_4 work together. This leads us to consider developing a model which expresses this type of 'higher-order transitivity', where a set of authors to be more likely to collaborate if a subset of them have already written a paper together. Note that this notion of transitivity differs from other definitions of hypergraph transitivity in the literature (for example, see Mansilla and Serra (2008)).

We now review the existing literature on hypergraph analysis, and we begin with work related to community detection. Several authors have considered this problem for hypergraphs (see Chien et al. (2018), Lin et al. (2017), Kim et al. (2018), and Ghoshdastidar and Dukkipati (2014)) where the interest is in determining the community membership of each node. To facilitate this, Ghoshdastidar and Dukkipati (2014) have extended the stochastic blockmodel (SBM) of Holland et al. (1983) for graphs to the k-uniform SBM in which all hyperedges are assumed to be of the same size k. Alternatively, Zhou et al. (2006) have considered spectral clustering in the hypergraph setting to determine community membership. Related to these works is the

approach of Ng and Murphy (2018) who develop a model to capture clustering in the hyperedges by extending methodology from latent class analysis (see Lazarsfeld and Henry (1968), Goodman (1974)). Note that this differs to community detection since the focus is on clustering structures in the hyperedges, not the nodes. Furthermore, in the bipartite setting, Aksoy et al. (2016) consider a model for bipartite graphs which exhibit community structure.

Link prediction for hypergraphs has also be been explored in the literature by Benson et al. (2018) and Sharma et al. (2014). These works consider predicting future hyperedge connections given a sequence of time-indexed hyperedges. Benson et al. (2018) focus on prediction of simplicial closure events in which a set of nodes appear within the same hyperedge. Their work does not rely on a formal statistical model, but instead aims to identify which features of the hypergraph are indicative of a simplicial closure event. Alternatively, Sharma et al. (2014) consider predicting reoccurrence of previously observed hyperedges and they refer to this as 'old edge' prediction. They develop a tensor based approach to address this problem. We note here that the prediction task we consider in (Q1) differs from these works since our focus is on predicting connections for new nodes, and not future connections between the observed set of nodes.

Other authors have developed models for hypergraphs by considering the extension of existing graph models. For example, Stasi et al. (2014) and Liu et al. (2013) have developed models which allow control over the degree distribution by extending the β-model of Holland and Leinhardt (1981) and the preferential attachment model of Barabási and Albert (1999), respectively. In the model of Stasi et al. (2014) each node is assigned a parameter which controls its tendency to form connections so that a hypergraph with certain degree distribution can be expressed. Alternatively, the model of Liu et al. (2013) describes a generative process in which the hypergraph is grown from a seed hypergraph. This allows control over the degree distribution through the mechanism in which new nodes are added to the hypergraph. Additionally, whilst the latent space framework has not been considered for the representation in Figure 2a, Friel et al. (2016) have developed a latent space model for temporally evolving bipartite graphs. In this work the authors examine company directors and boards they are associated with.

There has also been a recent interest in edge-exchangeable graph models which, unlike node-exchangeable graph models, are able to express sparse graphs (see Caron and Fox (2017), Dempsey et al. (2019), Crane and Dempsey (2018), Campbell et al. (2018) and Cai et al. (2016)). Many of these models are able to express bipartite graphs, which can be used to represent a hypergraph (see Figure 2). For example, Caron and Fox (2017) present an edge-exchangeable model for network data and, in Section 3.4, they consider how their framework may be applied to bipartite networks. Related to this is the approach of Dempsey et al. (2019) who model structured interaction processes with an edge-exchangeable framework. This work considers interactions between sets of nodes, which includes hypergraphs.

Finally, in the probability literature, several authors have studied properties of random hypergraphs (see Cooley et al. (2018), Élie de Panafieu (2015) and Karoński and Łuczak (2002)) including phase transitions. A particular class of hypergraphs, known as simplicial hypergraphs, are considered in Kahle (2016). In a simplicial hypergraph the presence of a hyperedge indicates the presence of all subsets of the hyperedge. Simplicial hypergraphs can be seen as a special case of a more general construction termed a simplicial complex, which appear more broadly in the statistics literature. For their use in topological data analysis see Salnikov et al. (2018), and for their application in graphical modelling see Lunagómez et al. (2017). Additionally, they are used to determine distances between distributions in Pronzato et al. (2018) and Pronzato et al. (2019).

The main contributions of this article are as follows. First, using the representation shown in Figure 2a, we develop a latent space model for non-simplicial hypergraph data by extending the distance model of Hoff et al. (2002) to the hypergraph setting. We present a specific instance

of our model, and comment that alternative modelling choices can be explored in future work. Second, our model avoids the computationally expensive full conditional implied by the construction of Hoff et al. (2002) by relying on tools from computational geometry (see Edelsbrunner and Harer (2010)). Furthermore, by representing the nodes of the hypergraph in a low-dimensional latent space, we develop a parsimonious model that is able to express complex data structures. Third, we draw upon the previously exploited connection between latent space network models and shape theory to remove non-identifiability present in our model. More specifically, we infer the latent representation on the space of Bookstein coordinates which have so far not been explored in this context. Fourth, by simulating new nodes from the latent representation, we demonstrate how our model facilitates exploration of the predictive distributions. We also comment that the latent representation provides a convenient visualisation of the hypergraph in which more centrally placed nodes have a larger degree. Fifth, we investigate the theoretical properties of our model and, in particular, we present a framework for examining the degree distribution. Whilst this proves challenging for our model, our discussion provides an outline which can be explored for other modelling choices.

The rest of this paper is organised as follows. In Section 2 we provide the necessary background to introduce our hypergraph model in Section 3. Then, in Section 4 we discuss identifiability of our hypergraph model and, in Section 5, we describe our procedure for obtaining posterior samples. The simulation studies and real data example are presented in Section 7 and 8, respectively. Finally, we conclude with a discussion in Section 9.

2 Background

In this section we will review the latent space network modelling literature that is relevant to the model for hypergraph data we introduce in Section 3. First, in Section 2.1, we discuss the framework introduced in Hoff et al. (2002), where connection probabilities between pairs of nodes are modelled as a function of a low-dimensional latent space. Then, in Section 2.2, we discuss random geometric graphs (RGGs), where the presence of an edge between a pair of nodes is determined by the intersection of convex sets. Finally, in Section 2.3 we demonstrate how RGGs can be extended to model a restricted class of hypergraphs.

2.1 Latent Space Network Modelling

Latent space models were introduced for network data in Hoff et al. (2002) and, since their introduction, have given rise to a rich modelling literature. The key assumption of this framework is that the nodes of a network can be represented in a low-dimensional latent space, and that the probability of an edge forming between each pair of nodes can be modelled as a function of their corresponding latent coordinates. Furthermore, conditional on the iid latent coordinates, the edge between a given pair of nodes is modelled independently of all other edges. The dependence in the network is captured by the latent representation, and this can be made clear through marginalising over the latent space.

We will first describe a general latent space modelling framework for a network with N nodes. Let $\boldsymbol{Y} = \{y_{ij}\}_{i,j=1,2,\ldots,N}$ denote the observed $(N \times N)$ adjacency matrix, where y_{ij} represents the connection between nodes i and j. For a binary network, we have that $y_{ij} = 1$ if i and j share an edge and $y_{ij} = 0$ otherwise. Additionally, we let $u_i \in \mathbb{R}^d$ represent the d-dimensional latent coordinate of the i^{th} node, for $i = 1, 2, \ldots, N$. The presence of an edge is then given by

the following model.

$$\begin{aligned} Y_{ij} &\sim \text{Bernoulli}(p_{ij}), \\ p_{ij} &= P(y_{ij} = 1 | u_i, u_j, \theta) = \frac{1}{1 + \exp\{-f(u_i, u_j, \theta)\}}, \end{aligned} \tag{1}$$

where θ represents additional model parameters and p_{ij} denotes the probability of an edge forming between nodes i and j. The connection probability depends on a function f that is monotonically increasing in a measure of similarity between u_i and u_j. As an example, the distance model introduced in Hoff et al. (2002) is obtained by choosing

$$f(u_i, u_j, \theta) = \alpha - \|u_i - u_j\|, \tag{2}$$

where $\|\cdot\|$ is the Euclidean distance, and $\theta = \alpha$ represents the base-rate tendency for edges to form. The function f may also incorporate covariate information so that nodes which share certain characteristics are more likely to be connected.

We note that the choice of similarity measure will impose characteristics on graphs generated under this model. If the similarity measure is chosen to be a metric, for example the Euclidean distance, we know that it satisfies the triangle inequality. If connections exist between the pairs $\{i, j\}$ and $\{i, k\}$, we know that their latent coordinates are close in terms of the similarity measure. The triangle inequality suggests that the node pair $\{i, k\}$ is also likely to be connected, and so transitive relationships are likely.

Both asymmetric and symmetric adjacency matrices can be represented in this framework. Suppose the connections are symmetric and that there are no self ties, so that we have $y_{ij} = y_{ji}$ and $y_{ii} = 0$ for $i, j = 1, 2, \ldots, N$. In this case, the likelihood, conditional on $\boldsymbol{U}$ and θ, is given by

$$\mathcal{L}(\boldsymbol{U}, \theta; \boldsymbol{Y}) \propto \prod_{i<j} P(y_{ij} = 1 | u_i, u_j, \theta)^{y_{ij}} \left[1 - P(y_{ij} = 1 | u_i, u_j, \theta)\right]^{1 - y_{ij}}, \tag{3}$$

where $\boldsymbol{U}$ is the $(N \times d)$ matrix of latent coordinates such that the i^{th} row of $\boldsymbol{U}$ corresponds to u_i.

The model specified in (1) and (3) can be modified in many different ways. For example, we can model the connection probabilities using the probit link function instead of the logit link function. Properties of modifications of this type are discussed in Rastelli et al. (2015). We may also model non-binary connections such as integer weighted edges. For example, a Poisson likelihood allows us to model edges which represent the number of interactions between a pair of nodes. Note that this will require specification of a rate parameter which has a different interpretation to p_{ij}.

2.2 Random Geometric Graphs

In Section 2.1 we outlined the latent space modelling approach for network data. This framework specifies the probability of an edge forming between a pair of nodes as a function of their latent coordinates. In this section we will discuss random geometric graphs (RGGs) which instead model the occurrence of edges as a deterministic function of the latent coordinates. RGGs can be viewed as a special case of the latent space framework where, conditional on the latent representation, there is no uncertainty in the connections. For an in-depth discussion of RGGs see Penrose (2003).

As in the previous section, we will assume that the i^{th} node can be represented by $u_i = (u_{i1}, u_{i2}, \ldots, u_{id}) \in \mathbb{R}^d$. The presence of an edge $\{i, j\}$ is modelled through the intersection of

convex sets that are parameterised by u_i and u_j. There are many choices of convex sets, and to generate an RGG we choose the closed ball in $\mathbb{R}^d$ with centre u_i and radius r. This set is represented $B_r(u_i) = \{u \in \mathbb{R}^d | \ \|u - u_i\| \leq r\} = \left\{ u \in \mathbb{R}^d | \ \sqrt{\sum_{j=1}^d (u_j - u_{ij})^2} \leq r \right\}$, and a graph is constructed by connecting each pair of nodes $\{i, j\}$ for which $B_r(u_i) \cap B_r(u_j) \neq \emptyset$. Generating a graph in this way is equivalent to connecting pairs of nodes for which $\|u_i - u_j\| \leq 2r$. An example of this construction is given in the left and middle panel of Figure 3.

We now express the likelihood for this model as a function of the latent coordinates, keeping the notation from Section 2.1. The likelihood conditional on $\boldsymbol{U}$ and r is given by

$$\mathcal{L}(\boldsymbol{U}, r; \boldsymbol{Y}) \propto \prod_{i<j} \mathbb{1}(\|u_i - u_j\| \leq 2r)^{y_{ij}} \left[1 - \mathbb{1}(\|u_i - u_j\| \leq 2r)\right]^{1-y_{ij}}. \tag{4}$$

By comparing (3) and (4), we see that a RGG can be viewed as a latent space network model where the probability of a connection is expressed as a step function. More specifically, we have $P(y_{ij} = 1 | u_i, u_j, \theta) = \mathbb{1}(\|u_i - u_j\| \leq 2r)$ where $\theta = r$. It is clear from this that, conditional on a set of latent coordinates, a RGG is deterministic. Note that (4) is equal to 1 if there is a perfect correspondence between the observed connections $\boldsymbol{Y}$ and the connections induced by the latent coordinates $\boldsymbol{U}$ and the radius r. If there are any connections which do not correspond to each other, then (4) is equal to 0.

To specify a more general construction, we define A_i to be a convex set for which $u_i \in A_i$. In the construction above, we let $A_i = B_r(u_i)$, for $i = 1, 2, \ldots, N$. This choice of convex set is convenient since, given the radius r and coordinates $\boldsymbol{U}$, we are able to generate the graph by considering the distance between pairs of latent coordinates. For this framework to be computationally appealing, it is important that the sets A_i are easy to parameterise and their intersections are efficient to compute. An alternative choice for A_i that is common in the literature is the Voronoi cell, where $A_i = \left\{ x \in \mathbb{R}^d | \|x - u_i\| \leq \|x - u\|, u \in \mathbb{R}^d \right\}$ (see Section 3.3 of Edelsbrunner and Harer (2010)).

2.3 Random Geometric Hypergraphs

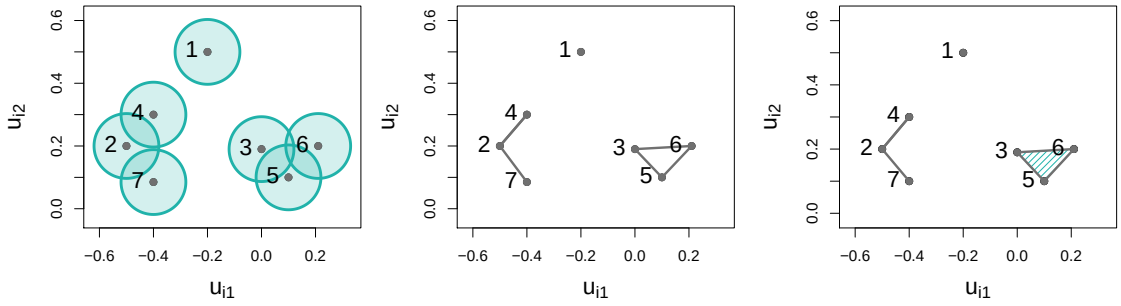
The graph generating procedure described in Section 2.2 assumed that edges occur between pairs of nodes. We can generalise this framework to model hypergraphs by considering the full intersection pattern of convex sets, and we refer to these hypergraphs as random geometric hypergraphs (RGHs). In order to do this, we introduce the concept of a nerve (Section 3.2 of Edelsbrunner and Harer (2010)). This represents the set of indices for which their corresponding convex regions have a non-empty intersection and it is given in Definition 2.1.

Definition 2.1. *(Nerve) Let $\mathcal{A} = \{A_i\}_{i=1}^N$ represent a collection of non-empty convex sets. The nerve of $\mathcal{A}$ is given by*

$$Nrv(\mathcal{A}) = \left\{ \sigma \subseteq \{1, 2, \ldots, N\} | \bigcap_{j \in \sigma} A_j \neq \emptyset \right\}. \tag{5}$$

Note that the sets $\{1\}, \{2\}, \ldots, \{N\}$ are included in $Nrv(\mathcal{A})$ and that $|\sigma| \leq N$ for $\sigma \in Nrv(\mathcal{A})$, where $|\sigma|$ is the order, or dimension, of the set.

It is clear that the nerve defines a hypergraph where $\sigma \in Nrv(\mathcal{A})$ denotes a hyperedge. Consider the sets $\sigma_1 \in Nrv(\mathcal{A})$ and $\sigma_2 \subset \sigma_1$. It follows immediately that $\sigma_2 \in Nrv(\mathcal{A})$, and all hypergraphs generated by a nerve must have this property. Hypergraphs of this type are termed simplicial. Kahle (2016) overview properties of simplicial random hypergraphs along with more general constructions.



(a) (b) (c)

Figure 3: Example of a Čech complex. Left: $B_r(u_i)$ for $\{u_i = (u_{i1}, u_{i2})\}_{i=1}^7$ in $\mathbb{R}^2$. Middle: the graph obtained by taking pairwise intersections. Right: the hypergraph obtained by taking intersections of arbitrary order. The shaded region between nodes 3, 5 and 6 indicates a hyperedge of order 3.

In Section 2.2, we considered the choice $A_i = B_r(u_i)$ for generating a RGG. The nerve for this choice of $\mathcal{A}$ is well studied and it is referred to as the Čech complex (see Section 3.2 of Edelsbrunner and Harer (2010)), as given in Definition 2.2.

Definition 2.2. *(Čech Complex) For a set of coordinates $\boldsymbol{U} = \{u_i\}_{i=1}^N$ and a radius r, the Čech complex $\mathcal{C}$ is given by*

$$\mathcal{C} = Nrv\left(\{B_r(u_i)\}_{i=1}^N\right). \tag{6}$$

An example of a Čech complex is given in the left and right panel of Figure 3, where $Nrv(\mathcal{A}) = \{\{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{6\}, \{2,4\}, \{3,5\}, \{3,6\}, \{5,6\}, \{3,5,6\}\}$. Other common examples of complexes include the Deulaunay triangulation (Delaunay (1934)) and the Alpha complex (Edelsbrunner et al. (1983)). To obtain these complexes we choose A_i as the Voronoi cell for u_i (see Section 2.2), and A_i as the intersection of the Voronoi cell for u_i and $B_r(u_i)$, respectively.

We now introduce a subset of a complex known as the k-skeleton, which is given in Definition 2.3. This will be revisited in Section 3.2.

Definition 2.3. *(k-skeleton of the Čech complex) Let $\mathcal{C}$ denote the Čech complex, as given in Definition 2.2. The k-skeleton of $\mathcal{C}$ is given by*

$$\mathcal{C}^{(k)} = \{\sigma \in \mathcal{C} || \sigma| \leq k\}. \tag{7}$$

$\mathcal{C}^{(k)}$ represents the collection of sets in $\mathcal{C}$ which are of order that is less than or equal to k. For example, from Figure 3, we have $\mathcal{C}^{(1)} = \{\{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{6\}\}$ and $\mathcal{C}^{(2)} = \{\{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{6\}, \{2,4\}, \{3,5\}, \{3,6\}, \{5,6\}\}$. We note here that the k-skeleton can also be defined more generally for any nerve.

3 Latent space hypergraphs

In this section we will introduce a model for hypergraph data which builds upon the models discussed in Section 2. Our model will balance the computational aspects of latent space network

modelling (Section 2.1) with the approach of RGGs (Section 2.2) and RGHs (Section 2.3). The aims of our modelling framework are given in Section 3.1, notation and set-up are given in Section 3.2, and the generative model and likelihood are given in Section 3.3. Finally, in Section 3.4 we extend the model to a more flexible scenario.

3.1 Motivation

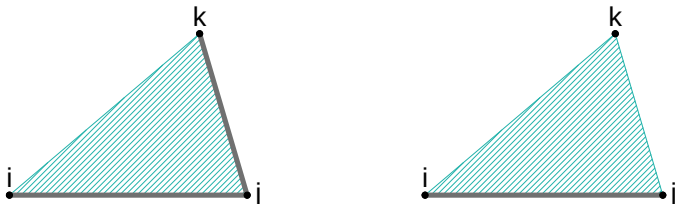
Consider a co-authorship network where nodes represent authors and edges indicate which authors contributed to a given paper. In this context, papers that have been written by more than two authors are naturally represented by a hyperedge. The hypergraph model discussed in Section 2.3 is likely not appropriate for these data since a set of authors having worked on a paper does not imply that all subsets of those authors have also written papers together. This motivates us to develop a model for non-simplicial hypergraphs.

In this section we will build upon the pairwise graph and hypergraph models introduced in Section 2. We introduce a model for hypergraph data which represents the nodes of the network in a low-dimensional space and, unlike the approach in Section 2.3, is able to express a broad class of hypergraphs. The model will be developed with the following objectives.

1. **Convenient likelihood**
   The model of Hoff et al. (2002) specifies the connection probabilities for all node pairs and so the likelihood, conditional on the latent coordinates, for this model is a function of a 2D tensor. In the hypergraph analogue of this framework, the conditional likelihood would be a function of a kD tensor, where k is the order of the hyperedges. Evaluation over a kD tensor is an order $O(N^k)$ computation, which becomes increasingly computationally expensive as the number of nodes N and the hyperedge order k grow. In contrast to this, graphs generated in the RGG (Section 2.2) and RGH (Section 2.3) framework are a deterministic function of the latent space. Hence, the conditional likelihood is equal to either 0 or 1 and this may hinder model fitting. Our aim is to develop a model which draws on the computational advantages of each of these approaches. We present a model with a likelihood that is amenable to Bayesian computation and whose evaluation does not rely on a calculation over a tensor.

2. **Simple to describe support**
   Since the edges in the model of Hoff et al. (2002) exhibit uncertainty after conditioning on the latent coordinates, it is clear that the model can express the entire space of graphs on N nodes with pairwise interactions. In the RGG framework (Section 2.2), connections are instead modelled deterministically through the intersection of convex sets. It is not clear in general how to characterise the space of graphs that represent the support for this generative model. Furthermore, when this framework is extended to model non-simplicial hypergraphs, the support for the generative model is complicated further. Based on the approach of RGHs (Section 2.3), we develop a model for non-simplicial hypergraph data whose support is straightforward to describe.

Throughout this section we will describe our model for a hypergraph on N nodes with a maximal hyperedge order K, where $2 \leq K \leq N$. We let $e \subseteq \{1, 2, \ldots, N\}$ denote a hyperedge and the presence of e is denoted by $y_e = 1$. Otherwise, we let $y_e = 0$. Where necessary, we will denote a hyperedge of order k by e_k so that $|e_k| = k$.



(a) $\{i,j\},\{j,k\},\{i,j,k\}$ (b) $\{i,j\},\{i,j,k\}$

Figure 4: Hypergraph connections that cannot be represented by a Čech complex. The shaded region represents the edge $\{i,j,k\}$.

3.2 Combining k-skeletons

In Section 2.3 we described a model for hypergraph data in which hyperedges are represented through the intersection of convex sets. When the i^{th} set is chosen as the ball of radius r and centre u_i, this model is termed the Čech complex (see Definition 2.2). Conditional on the latent coordinates, the radius r then determines the number of connections in the hypergraph. This model generates simplicial hypergraphs only and so is inappropriate when we wish to describe relations such as those shown in Figure 4.

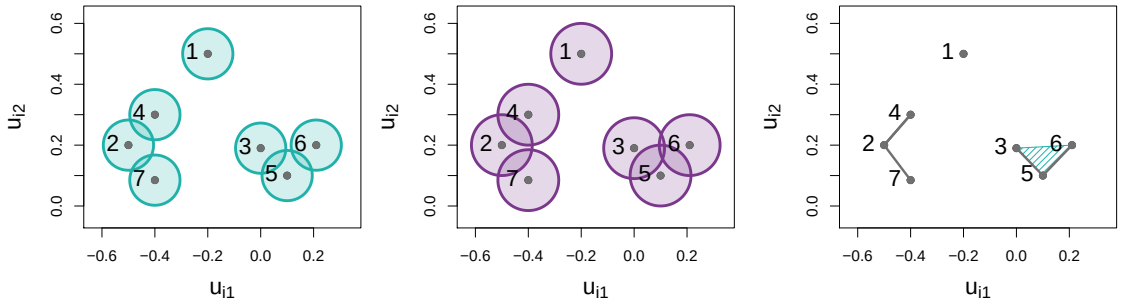
To extend this model to non-simplicial hypergraphs we consider the introduction of additional radii. As an example, consider two radii which we denote by r_2 and r_3. For the same set of latent coordinates $\boldsymbol{U}$, each of these radii will give rise to Čech complex which we denote by $\mathcal{C}_{r_2}$ and $\mathcal{C}_{r_3}$, respectively. By varying r_2 and r_3 we are able to control the edges that are present in each of $\mathcal{C}_{r_2}$ and $\mathcal{C}_{r_3}$. Now suppose for each of these complexes we only consider hyperedges of a specific order. We can, for example, construct a hypergraph by taking the union of the order 2 hyperedges in $\mathcal{C}_{r_2}$ and the order 3 hyperedges in $\mathcal{C}_{r_3}$. This construction removes the simplicial constraint on the hypergraphs, and an example of this is given in Figure 5.

We now introduce some notation to express this procedure more precisely. Firstly, recall the definition of the k-skeleton of the Čech complex given in Definition 2.3. $\mathcal{C}^{(k)}$ represents the set of hyperedges in $\mathcal{C}$ of up to order k, and we consider the relative complement of $\mathcal{C}^{(k)}$ with respect to $\mathcal{C}^{(k-1)}$ to describe the hyperedges in $\mathcal{C}$ of exactly order k. In particular, we let $\mathcal{D}^{(k)} = \mathcal{C}^{(k)} \setminus \mathcal{C}^{(k-1)}$, for $k = 2,3,\ldots,K$, denote the hyperedges of exactly order k in $\mathcal{C}$. To make explicit the dependence on the radius r, we introduce the subscript r so that $\mathcal{D}_r^{(k)}$ denotes the hyperedges of order k generated by the radius r. Now, we let r_k represent the radius which controls connections of order k, for $k = 2,3,\ldots,K$. Then we have that $\mathcal{D}_{r_k}^{(k)}$ represents the set of all order k hyperedges, and the hyperedges in the graph are given by $\cup_{k=2}^{K}\mathcal{D}_{r_k}^{(k)}$.

Example 3.1. *Consider the non-simplicial hypergraph shown in the right panel of Figure 5. For this hypergraph we have* $\mathcal{D}_{r_2}^{(2)} = \{\{2,4\},\{3,5\},\{5,6\}\}$ *and* $\mathcal{D}_{r_3}^{(3)} = \{3,5,6\}$.

To ensure the hypergraphs generated in this way are non-simplicial we must impose constraints on the radii $\boldsymbol{r} = (r_2, r_3, \ldots, r_K)$. Suppose that we have $r_3 < r_2$ and that the hyperedge $\{i,j,k\}$ is present in the hypergraph. With this constraint on the radii, it follows that the hyperedges $\{i,j\},\{i,k\}$ and $\{j,k\}$ must also be in the hypergraph, and so the resulting hypergraph is simplicial. To avoid this we impose $r_k > r_{k-1}$ for $k = 3,4,\ldots,K$.

Now we have a model for non-simplicial hypergraphs in which the nodes are represented by coordinates $\boldsymbol{U}$. This generalises the model described in Section 2.3 by introducing a different radii



(a) (b) (c)

Figure 5: Example of a Čech complex. Left: $B_{r_2}(u)$ for each of 6 points in $\mathbb{R}^2$. Middle: $B_{r_3}(u)$ for each of 6 points in $\mathbb{R}^2$. Right: $\cup_{k=2}^{3}\mathcal{D}_{r_k}^{(k)}$.

for each hyperedge order. The hypergraph is then given by the union of $\mathcal{D}_{r_k}^{(k)}$ over all hyperedge orders between 2 and K, and we refer to hypergraphs generated in this way as non-simplicial random geometric hypergraphs (nsRGH).

3.3 Generative Model and Likelihood

In Section 3.2 we described a procedure for generating non-simplicial hypergraph data. We now recall the objectives described in Section 3.1 and compare these with the behaviour of this model. If $K = 2$, the conditional likelihood for the model described in Section 3.2 is given by (4) where $r = r_2$. As noted in Section 2.2, this is equal to 1 only if there is a perfect correspondence between the edges of the observed graph and the edges induced from $\boldsymbol{U}$ and r_2. For arbitrary K, the behaviour will be similar. The support of the model is difficult to describe and model fitting may be challenging.

We now consider a modification of the model which extends and simplifies the support. By introducing a modification of the hyperedges, we broaden the support and develop a likelihood that is more amenable to Bayesian computation. To express this precisely, we first introduce some notation. Let $\mathcal{G}_{N,K}$ denote the space of hypergraphs on N nodes with maximum hyperedge order K. We write $\mathcal{G}_{N,K} = (\mathcal{V}_N, \mathcal{E}_{N,K})$, where $\mathcal{V}_N = \{1, 2, \ldots, N\}$ denotes the node labels and $\mathcal{E}_{N,K}$ denotes the set of possible hyperedges up to order K on N nodes. Let $\mathcal{E}_{N,k}$ represent the possible hyperedges of exactly order k on N nodes so that $\mathcal{E}_{N,K} = \cup_{k=2}^{K}\mathcal{E}_{N,k}$.

Let $\varphi_k \in [0, 1]$ denote the probability of switching the state of a hyperedge of order k, for $k = 2, 3, \ldots, K$. Then, for $e_k \in \mathcal{E}_{N,k}$, we sample a variable $S_k \sim \text{Bernoulli}(\varphi_k)$. If $s_k = 1$ and $y_{e_k} = 1$, we set $y_{e_k} = 0$, and if $s_k = 1$ and $y_{e_k} = 0$, we set $y_{e_k} = 1$. If $s_k = 0$ we do not alter the state of the hyperedge e_k. Hence, φ_k controls the amount of modification of the hyperedges of order k in the graph and a larger value of φ_k results in a greater number of modifications.

Conditional on a set of latent coordinates, we now have a method of generating a hypergraph whose support is simple to describe. To specify a generative model we must introduce a probability distribution from which $\boldsymbol{U}$ are sampled. We choose $u_i \overset{iid}{\sim} \mathcal{N}(\mu, \Sigma)$, for $i = 1, 2, \ldots, N$, and a hypergraph can then be generated by the procedure given in Algorithm 1.

In Algorithm 1 we must determine the presence or absence of all hyperedges given $\boldsymbol{U}$ and $\boldsymbol{r}$. This computation is not trivial, and is discussed in more detail in Appendix E.1. Here we

Algorithm 1 Sample a hypergraph $g^*_{N,K}$ given $N, K, \boldsymbol{r}, \boldsymbol{\varphi}, \mu$ and Σ.

Sample $\boldsymbol{U} = \{u_i\}_{i=1}^N$ such that $u_i \overset{iid}{\sim} \mathcal{N}(\mu, \Sigma)$, for $i = 1, 2, \ldots, N$.
For $k = 2, 3, \ldots, K$,
a) Given $\boldsymbol{U}$ and r_k, check which $e_k = \{i_1, \imath_2, \ldots, i_k\} \in \mathcal{E}_{N,k}$ satisfy $y^{(g)}_{e_k} = 1$.
To determine if $y^{(g)}_{e_k} = 1$, check that $\cap_{l=1}^k B_{r_k}(u_{i_l}) \neq \emptyset$.
b) For all $e_k \in \mathcal{E}_{N,k}$, sample $S_k \sim \text{Bernoulli}(\varphi_k)$.
Let $y^{(g^*)}_{e_k} = \left(y^{(g)}_{e_k} + s_k\right) \mod 2$

comment that this can be computed more efficiently than Algorithm 1 suggests, and we are able to avoid checking all $e_k \in \mathcal{E}_{N,k}$.

Furthermore, in Algorithm 1, the hyperedge noise is applied to all $e_k \in \mathcal{E}_{N,k}$. Since φ_k will be reasonably small in practice, we expect many values of S_k to be 0. To avoid sampling S_k for all $\binom{N}{k}$ hyperedges, we instead only sample the hyperedges which are modified. Details of this are given in Appendix B.

Given this model, we can now write down the likelihood of an observed hypergraph $h_{N,K} \in \mathcal{G}_{N,K}$, conditional on $\boldsymbol{U}, \boldsymbol{r}$ and $\boldsymbol{\varphi}$. We let $g_{N,K}(\boldsymbol{U}, \boldsymbol{r}) \in \mathcal{G}_{N,K}$ denote the hypergraph that is induced from the latent coordinates $\boldsymbol{U}$ and radii $\boldsymbol{r}$. Note that no modifications of the hyperedges have been made for this hypergraph. The likelihood of an observed hypergraph $h_{N,K}$ is then expressed by considering the discrepancy between the hyperedge configurations in $h_{N,K}$ and $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$. More specifically, we must consider how likely it is to obtain the connections in $h_{N,K}$ by altering the state of the hyperedges in $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$.

For $k = 2, 3, \ldots, K$, let

$$d_k(g_{N,K}(\boldsymbol{U}, \boldsymbol{r}), h_{N,K}) = \sum_{e_k \in \mathcal{E}_{N,k}} |y^{(g)}_{e_k} - y^{(h)}_{e_k}| \tag{8}$$

denote the distance between the order k hyperedges in $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$ and $h_{N,K}$, where $y^{(g)}_{e_k}$ and $y^{(h)}_{e_k}$ represent an order k hyperedge in $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$ and $h_{N,K}$, respectively. (8) enumerates the number of hyperedges which differ between two hypergraphs. Note that (8) is equivalent to the Hamming distance, and it is related to the l_1 norm and the exclusive or (XOR) operator.

To evaluate (8), we do not need to perform $\sum_{k=2}^K \binom{N}{k}$ computations. Instead we can evaluate each discrepancy by only considering hyperedges that are present in $y^{(g)}_{e_k}$ and or $y^{(h)}_{e_k}$. In practice, this is likely to be far less than the number of possible hyperedges and details of this calculation are discussed in Appendix E.2.

Given this notion of hypergraph distance the likelihood of observing $h_{N,K}$, conditional on $\boldsymbol{U}, \boldsymbol{r}$ and $\boldsymbol{\varphi}$, can be written as

$$\mathcal{L}(\boldsymbol{U}, \boldsymbol{r}, \boldsymbol{\varphi}; h_{N,K}) \propto \prod_{k=2}^K \varphi_k^{d_k(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})}(1-\varphi_k)^{\binom{N}{k}-d_k(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})}. \tag{9}$$

We obtain (9) by considering which hyperedges in $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$ must have their state changed to match the hyperedges in $h_{N,K}$, and which hyperedges are the same as in $h_{N,K}$. For those which differ, the probability of switching the hyperedge state is given by φ_k for an order k hyperedge.

We note here the connection between (9) and the proof of Proposition 3.1 in Lunagómez et al. (2019). Since our likelihood is of the same form, it follows that hypergraphs with a greater

number of hyperedge modification are less likely when $0 < \varphi_k < 1/2$. Hence, (9) behaves in an intuitive way.

To complete the model specification, we introduce the following prior distributions for $k = 2, 3, \ldots, K$.

$$\mu \sim \mathcal{N}(m_\mu, \Sigma_\mu), \quad \Sigma \sim \mathcal{W}^{-1}(\Phi, \nu), \quad r_k \sim \exp(\lambda_k), \quad \text{and} \quad \varphi_k \sim \text{Beta}(a_k, b_k). \tag{10}$$

The priors in (10) are chosen for conjugacy and computational convenience.

3.4 Extensions

The number of edges of order 2 can be controlled by varying the parameter r_2. The constraint $r_k > r_{k-1}$, for $k = 3, 4, \ldots, K$, implies that the value of r_2 will also impact the higher order hyperedges, which may limit the types of hypergraphs that can be expressed.

To improve the model flexibility we introduce an additional noise parameter for each hyperedge order. Previously, the noise φ_k was applied independently across all hyperedges of order k. Instead we may modify the hyperedges differently, depending on whether or not a hyperedge is present in $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$. For $k = 2, 3, \ldots, K$, let $\psi_k^{(0)} \in [0, 1]$ denote the probability of changing the state of a hyperedge in $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$ from absent to present, and let $\psi_k^{(1)} \in [0, 1]$ denote the probability of changing the state of a hyperedge in $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$ from present to absent. Suppose our observed hypergraph suggests that there are many hyperedges of order 2 and few hyperedges of order 3. By increasing the modification noise $\psi_3^{(1)}$ we can control additional hyperedges that may appear in the hypergraph due to the constraint $r_3 > r_2$. The generative model for this modification is given in Algorithm 2.

Algorithm 2 Sample a hypergraph $g^*_{N,K}$ given $N, K, \boldsymbol{r}, \boldsymbol{\psi}^{(0)}, \boldsymbol{\psi}^{(1)}, \mu$ and Σ.

Sample $\boldsymbol{U} = \{u_i\}_{i=1}^N$ such that $u_i \overset{iid}{\sim} \mathcal{N}(\mu, \Sigma)$, for $i = 1, 2, \ldots, N$.
For $k = 2, 3, \ldots, K$,
- a) Given $\boldsymbol{U}$ and r_k, check which $e_k = \{i_1, \imath_2, \ldots, i_k\} \in \mathcal{E}_{N,k}$ satisfy $y_{e_k} = 1$.
  To determine if $y_{e_k} = 1$, that $\cap_{l=1}^k B_{r_k}(u_{i_l}) \neq \emptyset$.
- b) For all $e_k \in \mathcal{E}_{N,k}$
  If $y_{e_k}^{(g)} = 1$, set $y_{e_k}^{(g^*)} = 0$ with probability $\psi_k^{(1)}$.
  If $y_{e_k}^{(g)} = 0$, set $y_{e_k}^{(g^*)} = 1$ with probability $\psi_k^{(0)}$.

Let $\boldsymbol{\psi}^{(1)} = (\psi_2^{(1)}, \psi_3^{(1)}, \ldots, \psi_K^{(1)})$ and $\boldsymbol{\psi}^{(0)} = (\psi_2^{(0)}, \psi_3^{(0)}, \ldots, \psi_K^{(0)})$. As commented in Section 3.3, we can avoid implementing the hyperedge modification calculation for all $e_k \in \mathcal{E}_{N,k}$ by instead only considering the hyperedges whose state is switched. See Appendix B for details of this.

To express the likelihood of observing $h_{N,K}$ we let

$$d_k^{(ab)}(g_{N,K}(\boldsymbol{U}, \boldsymbol{r}), h_{N,K}) = \#\{e_k \in \mathcal{E}_{N,k} | y_{e_k}^{(g)} = a \cap y_{e_k}^{(h)} = b\} \tag{11}$$

denote number of hyperedges that have state $a \in \{0, 1\}$ in $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$ and state $b \in \{0, 1\}$ in $h_{N,K}$. For example, we have that $d_k^{(01)}(g_{N,K}(\boldsymbol{U}, \boldsymbol{r}), h_{N,K})$ represents the number of hyperedges absent in $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$ and present in $h_{N,K}$. Whilst (11) suggests a summation over all possible hyperedges, we are able to calculate this much more efficiently. Details of this are discussed in

Appendix E.2. The likelihood, conditional on $\boldsymbol{U}, \boldsymbol{r}, \boldsymbol{\psi}^{(1)}$ and $\boldsymbol{\psi}^{(0)}$, is then given by

$$\mathcal{L}\left(\boldsymbol{U}, \boldsymbol{r}, \boldsymbol{\psi}^{(1)}, \boldsymbol{\psi}^{(0)}; h_{N,K}\right) \propto \prod_{k=2}^{K} \left[\left(\psi_k^{(1)}\right)^{d_k^{(10)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})} \left(1-\psi_k^{(1)}\right)^{d_k^{(11)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})} \times \left(\psi_k^{(0)}\right)^{d_k^{(01)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})} \left(1-\psi_k^{(0)}\right)^{d_k^{(00)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})} \right]. \quad (12)$$

We obtain (12) in a similar way to (9), where we distinguish between hyperedges that are present and absent in the induced hypergraph. Note that (12) is equivalent to (9) when $\psi_k^{(1)} = \psi_k^{(0)}$, for $k = 2, 3, \ldots, K$.

The connection between (12) and proof of Proposition 3.1 in Lunagómez et al. (2019) is not as clear. It is not possible to draw a direct link between the two due to the introduction of the additional noise parameter. A similar argument will only be applicable when the density of the hyperedges remains constant.

To complete the model specification, we introduce the following prior distributions for $k = 2, 3, \ldots, K$.

$$\mu \sim \mathcal{N}(m_\mu, \Sigma_\mu), \quad \Sigma \sim \mathcal{W}^{-1}(\Phi, \nu), \quad r_k \sim \exp(\lambda_k),$$
$$\psi_k^{(0)} \sim \text{Beta}\left(a_k^{(0)}, b_k^{(0)}\right), \quad \text{and} \quad \psi_k^{(1)} \sim \text{Beta}\left(a_k^{(1)}, b_k^{(1)}\right). \quad (13)$$

The priors in (13) are chosen for computational convenience to allow sampling from the conditional posteriors as detailed in Section 5.

4 Identifiability

In Section 3 we introduced a model for non-simplicial hypergraphs which can be viewed as a modification of a nsRGH. In this section we discuss the sources of non-identifiability present in this model.

Let $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$ denote the nsRGH obtained from $\boldsymbol{U}$ and $\boldsymbol{r}$, and let $g^*_{N,K}$ be the hypergraph obtained by modifying the hyperedges in $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$ according to $\boldsymbol{\varphi}$ (see Algorithm 1). By conditioning on $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$, we can decompose the conditional distribution for $g^*_{N,K}$ in the following way.

$$p(g^*_{N,K}|\mu, \Sigma, \boldsymbol{\varphi}, \boldsymbol{r}) = p(g^*_{N,K}|g_{N,K}(\boldsymbol{U}, \boldsymbol{r}), \boldsymbol{\varphi}) p(g_{N,K}(\boldsymbol{U}, \boldsymbol{r})|\mu, \Sigma, \boldsymbol{r}). \quad (14)$$

Note that an equivalent decomposition for the model outlined in Algorithm 2 can also be expressed.

The probability of occurrence of a hyperedge in $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$ is a function of the distance between the latent coordinates. Therefore the conditional distribution $p(g_{N,K}(\boldsymbol{U}, \boldsymbol{r})|\mu, \Sigma, \boldsymbol{r})$ is invariant to distance-preserving transformations of $\boldsymbol{U}$ and will exhibit multimodality. Furthermore, we observe that scaling $\boldsymbol{U}$ and $\boldsymbol{r}$ by the same factor results in a source of model non-identifiability.

To remove these sources of non-identifiability, we define $\boldsymbol{U}$ on the Bookstein space of coordinates (see Bookstein (1986), Bookstein (1984) and Section 2.3.3 of Dryden and Mardia (1998)). Bookstein coordinates define a translation, rotation and rescaling of the points $\boldsymbol{U}$ with respect to a set of anchor points. Since these anchor points remain fixed throughout inference, the radii

$\boldsymbol{r}$ are also appropriately rescaled. For details of the Bookstein transformation, see Appendix A. We note here that, in the models introduced in Section 3, the latent coordinates were assumed to be normally distributed and the Bookstein transformation preserves this.

Bookstein coordinates were developed to transform landmark points, or significant points of interest, on shapes onto a set of coordinates which facilitates comparison. In our setting, we infer $\boldsymbol{U}$ on the space of Bookstein coordinates, which presents a simpler application since we do not need to consider the effect of the change of coordinates on the comparison between shapes. In the latent space network modelling literature, it is typical to use Procrustes analysis (Section 5 of Dryden and Mardia (1998)) as a post-processing step to remove the effect of distance preserving transformations. Due to the non-identifiability associated with scaling $\boldsymbol{U}$ and $\boldsymbol{r}$, we note that this approach is not sufficient for removing all sources of non-identifiability in our model.

From (14), we see that hyperedges can either arise from $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$ or the hyperedge modification. To maintain the properties imposed on the hypergraph from the construction of $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$, we wish to keep the parameters $\boldsymbol{\varphi}$ relatively small. However, when generating sparse hypergraphs from our model, it will become increasingly difficult to distinguish between these competing hyperedge sources. Therefore we observe model non-identifiability when the hypergraphs are sparse, although it is not clear at which level of sparsity this will occur.

5 Posterior Sampling

To sample from the model specified in Section 3.4 we implement an MCMC scheme, and an overview of this is given in Section 5.1. This section contains a high-level description of the posterior sampling procedure and we refer the reader to the relevant sections of the appendix for more detail.

5.1 MCMC scheme

To obtain posterior samples from the model given in Section 3.4 we implement a Metropolis-Hastings-within-Gibbs MCMC scheme (see Section 6.4.2 of Gamerman and Lopes (2006)). We update the latent coordinates $\boldsymbol{U}$ and radii $\boldsymbol{r}$ with a Metropolis Hastings (MH) step, and the remaining parameters are updated via Gibbs samplers. The priors for the model are specified in (13).

When updating the latent coordinates we use a random walk MH. As discussed in Section 4, we define $\boldsymbol{U}$ on the Bookstein space of coordinates and so a set of anchor points will remain fixed throughout the MCMC scheme. For $u_i \in \mathbb{R}^2$, let the anchor points be denoted by u_1 and u_2. For $i = 3, 4, \ldots, N$, we propose $u_i^* = u_i + \epsilon_u$ where $\epsilon_u \sim \mathcal{N}(0, \sigma_u I_d)$, and for $i = 1, 2$ we let $u_i^* = u_i$. We then accept $\boldsymbol{U}^* = \{u_i^*\}_{i=1}^N$ as a sample from $p(\boldsymbol{U}|\mu, \Sigma, \boldsymbol{r}, \boldsymbol{\psi}^{(0)}, \boldsymbol{\psi}^{(1)}, h_{N,K})$ with probability

$$\min\left\{1, \frac{\mathcal{L}(\boldsymbol{U}^*, \boldsymbol{r}, \boldsymbol{\psi}^{(1)}, \boldsymbol{\psi}^{(0)}; h_{N,K}) p(\boldsymbol{U}^*|\mu, \Sigma)}{\mathcal{L}(\boldsymbol{U}, \boldsymbol{r}, \boldsymbol{\psi}^{(1)}, \boldsymbol{\psi}^{(0)}; h_{N,K}) p(\boldsymbol{U}|\mu, \Sigma)}\right\}, \tag{15}$$

where $p(\boldsymbol{U}|\mu, \Sigma) = \prod_{i=1}^N p(u_i|\mu, \Sigma)$. Since the proposal mechanism is symmetric in terms of $\boldsymbol{U}$ and $\boldsymbol{U}^*$, the term associated with this does not appear in (15).

The acceptance ratio (15) is for the entire $(N \times d)$ latent representation $\boldsymbol{U}$. As N grows, jointly proposing all latent coordinates will become increasingly inefficient due to the dimension of $\boldsymbol{U}$. We instead partition the latent coordinates into disjoint sets $\{U_l\}_{l=1}^L$ such that $\cup_{l=1}^L U_l = \{1, 2, \ldots, N\}$, and perform the MH update for each of these L sets separately. This approach will be used in the examples in Sections 7 and 8, and details of this are given in Algorithm 3.

To update $\boldsymbol{r}$, we let $\boldsymbol{r}^* = (r_2^*, r_3^*, \ldots, r_K^*)$ where $r_k^* = r_k + \epsilon_r$ and $\epsilon_r \sim \mathcal{N}(0, \sigma_r)$. Then we accept $\boldsymbol{r}^*$ as a sample from $p(r|\boldsymbol{U}, \boldsymbol{\psi}^{(0)}, \boldsymbol{\psi}^{(1)}, \mu, \Sigma)$ with probability

$$\min\left\{1, \frac{\mathcal{L}(\boldsymbol{U}, \boldsymbol{r}^*, \boldsymbol{\psi}^{(1)}, \boldsymbol{\psi}^{(0)}; h_{N,K}) p(\boldsymbol{r}^*|\boldsymbol{\lambda})}{\mathcal{L}(\boldsymbol{U}, \boldsymbol{r}, \boldsymbol{\psi}^{(1)}, \boldsymbol{\psi}^{(0)}; h_{N,K}) p(\boldsymbol{r}|\boldsymbol{\lambda})}\right\}, \tag{16}$$

where $p(\boldsymbol{r}|\boldsymbol{\lambda}) = \prod_{k=2}^{K} p(r_k|\lambda_k)$.

All other parameters can be sampled directly from their full conditionals, and the details of the MCMC scheme for i_{max} iterations are given in Algorithm 3. Initialisation for the MCMC is non-trivial, and a discussion of this can be found in Appendix D.

Algorithm 3 MCMC scheme to obtain posterior samples of $\boldsymbol{U}, \boldsymbol{r}, \boldsymbol{\psi}^{(0)}, \boldsymbol{\psi}^{(1)}, \Sigma$ and μ.

Let $N, K, L, i_{max} \in \mathbb{N}$.
Initialise
Determine initial values for $\boldsymbol{U}, \boldsymbol{r}, \boldsymbol{\psi}^{(0)}, \boldsymbol{\psi}^{(1)}, \Sigma$ and μ using Algorithm 6 (see Appendix D).
For i **in** $1, 2, \ldots, i_{max}$
1) Sample $\mu^{(i)}$ from $p(\mu|\boldsymbol{U}, \Sigma, m_\mu, \Sigma_\mu)$ (see Appendix C.1).
2) Sample $\Sigma^{(i)}$ from $p(\Sigma|\boldsymbol{U}, \mu, \Phi, \nu)$ (see Appendix C.2).
3) Partition $\{u_3, u_4, \ldots, u_N\}$ into L sets U_l.
For $l = 1, 2, \ldots, L$
For $i \in U_l$, propose $u_i^* = u_i + \epsilon_u$, where $\epsilon_u \sim \mathcal{N}(0, \sigma_u I_d)$
Accept proposal with probability (15).
4) **For** $k = 2, 3, \ldots, K$, propose $\boldsymbol{r}^*$, where ${r_k}^* = r_k + \epsilon_r$ and $\epsilon_r \sim \mathcal{N}(0, \sigma_r)$
Accept proposal $\boldsymbol{r}^*$ with probability (16).
5) **For** $k = 2, 3, \ldots, K$
Sample $\psi^{(0)}k$ from $p\left(\psi_k^{(0)}|\boldsymbol{U}, \boldsymbol{r}, h_{N,K}, a_k^{(0)}, b_k^{(0)}\right)$ (see Appendix C.3).
Sample $\psi_k^{(1)}$ from $p\left(\psi_k^{(1)}|\boldsymbol{U}, \boldsymbol{r}, h_{N,K}, a_k^{(1)}, b_k^{(1)}\right)$ (see Appendix C.4).

To implement the MCMC scheme, there are a number of computational considerations we must address (see Appendix E). Firstly, to evaluate the likelihood we need to determine the hyperedges in the hypergraph generated from $\boldsymbol{U}$ and $\boldsymbol{r}$, $g(\boldsymbol{U}, \boldsymbol{r})$. We rely on methodology from the computational topology literature to do this, and details of the approach used are given in Appendix E.1. Secondly, to calculate the likelihood we note that we can avoid the summation over the entire set of hyperedges as suggested by (11) and, in Appendix E.2, we discuss this in more detail.

6 Theoretical Results

In this section we study the behaviour of the node degree in the hypergraph model detailed in Algorithm 1. We begin by making observations on the structure of our model in Section 6.1, and then consider the probability of a hyperedge occurring in a nsRGH in Section 6.2. Finally, we present our results for the expected node degree in Section 6.3 and the degree distribution in Section 6.4.

6.1 Observations

To study the properties of the node degree we first comment that the nodes in our hypergraph model are exchangeable, and so we focus on obtaining results for the degree of the i^{th} node. To guide the structure of our proofs we make the following observations.

(O1) **A hypergraph generated from our model is a modification of a nsRGH**
To generate a hypergraph from our model, we first determine hyperedges through the intersection of sets $B_r(u_i)$ for $i = 1, 2, \ldots, N$. The indicators for the presence and absence of hyperedges are then modified according to the noise parameters $\boldsymbol{\varphi}$. Since the modifications are applied independently, we can view our hypergraph model as an Erdős-Rényi modification of a nsRGH generated from $\boldsymbol{U}$ and $\boldsymbol{r}$.

(O2) **Hyperedges of different orders occur independently in our model**
Conditional on the latent coordinates $\boldsymbol{U}$ and radii $\boldsymbol{r}$, the hyperedges of order k occur independently of hyperedges of order $k' \neq k$. Therefore, we can consider the degree distribution as a sum over the degree distributions for hyperedges of exactly order k.

Throughout this section, we will assume that the number of nodes N and maximum hyperedge order K are fixed. We let $g(\boldsymbol{U}, \boldsymbol{r}) = g \in \mathcal{G}_{N,K}$ denote the nsRGH generated from the coordinates $\boldsymbol{U}$ and radii $\boldsymbol{r}$. A hypergraph is generated from our model by modifying g with noise $\boldsymbol{\varphi}$, and we denote this hypergraph by $g^* \in \mathcal{G}_{N,K}$. Additionally, we denote the degree of order k hyperedges and the overall degree of the i^{th} node in g by $\text{Deg}^g_{(i,k)} = \sum_{\{e_k \in \mathcal{E}_{N,k} | i \in e_k\}} y^{(g)}_{e_k}$ and $\text{Deg}^g_{(i)} = \sum_{k=2}^{K} \text{Deg}_{(i,k)}$, respectively.

Using this notation, we recall the distribution decomposition for $h_{N,K}$ in (14) and comment that this follows from (O1). Finally, we will assume that the covariance matrix for the latent coordinates Σ is diagonal so that $\Sigma_{ll} = \sigma_l^2$ for $l = 1, 2, \ldots, d$ and $\Sigma_{lm} = 0$ for $l \neq m$. Note that this assumption is not restrictive since, for any normally distributed set of points in $\mathbb{R}^d$, we can apply a distance-preserving transformation which maps the covariance matrix onto a diagonal matrix.

6.2 Properties of a nsRGH

In this section we consider the probability an order k hyperedge occurring in a nsRGH generated from $\boldsymbol{U}$ and $\boldsymbol{r}$, and we denote this by $p_{e_k} = P(y^{(g)}_{e_k} = 1 | \mu, \Sigma, r_k)$. We present results for $k = 2$ in Section 6.2.1 and discuss the connection probability for $k \geq 3$ in Section 6.2.2.

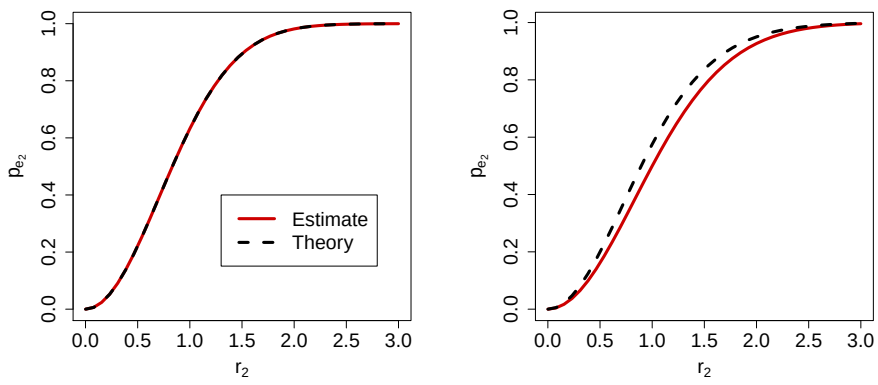
6.2.1 Connection Probabilities for $k = 2$

Recall that an edge $e_2 = \{i, j\}$ is present in $g(\boldsymbol{U}, \boldsymbol{r})$ if $B_{r_2}(u_i) \cap B_{r_2}(u_j) \neq \emptyset$. Hence, to obtain an expression for the occurrence probability p_{e_2}, we consider the probability of the coordinates u_i and u_j lying within distance $2r_2$ of each other. This probability is given in Proposition 6.1 and follows by considering the distribution of a squared Normal random variable.

Proposition 6.1. *Let $U_i \sim \mathcal{N}(\mu, \Sigma)$, for $i = 1, 2, \ldots, N$, and $\Sigma = diag(\sigma_1^2, \sigma_2^2, \ldots, \sigma_d^2)$. The probability of an edge $e_2 = \{i, j\}$ occurring in $g(\boldsymbol{U}, \boldsymbol{r})$ is given by*

$$p_{e_2} = P(\|U_i - U_j\| \leq 2r_2 | \Sigma) = \int_0^{(2r_2)^2} \sum_{l=1}^{d} f\left(z; \frac{1}{2}, 4\sigma_l^2\right) dz, \tag{17}$$

where $f(z; a, b) = \dfrac{b^a}{\Gamma(a)} z^{a-1} e^{-bz}$ is the pdf of a $\Gamma(a, b)$ random variable.



(a) (b)

Figure 6: Estimate of probability of a hyperedge occurring for $k = 2$ and increasing r_k compared to predicted probability. Points were simulated from a Normal distribution with $\mu = (0, 0)$, and $\Sigma = I_2$ in (a) and $\Sigma = \left(\begin{smallmatrix} 2 & 0 \\ 0 & 1 \end{smallmatrix}\right)$ in (b).

Proof. See Appendix F.1. □

To check the validity of this result, Figure 6 shows an empirical estimate of p_{e_2} compared to the result in Proposition 6.1 for two choices of Σ. Next, we will consider the connection probability p_{e_k} for $k \geq 3$.

6.2.2 Connection Probabilities for $k \geq 3$

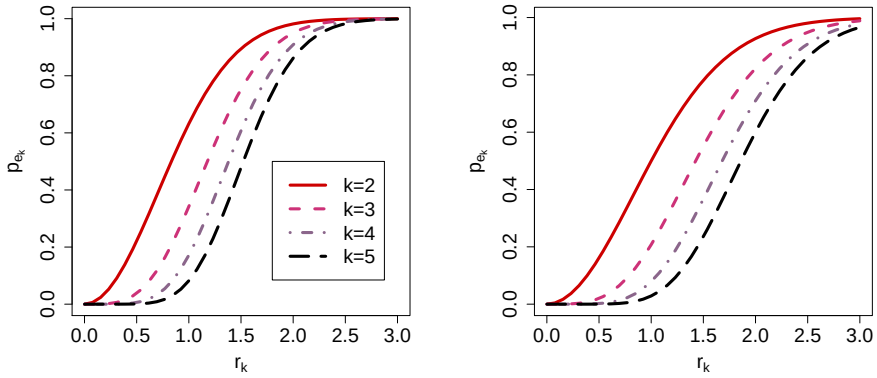
To determine p_{e_k} for $k \geq 3$, we first note that the edge $e_k = \{i_1, i_2, \ldots, i_k\}$ is present in $g(\boldsymbol{U}, \boldsymbol{r})$ if $\cap_{l=1}^{k} B_{r_k}(u_{i_l}) \neq \emptyset$. This condition is equivalent to the coordinates $\{u_{i_l}\}_{l=1}^{k}$ being contained within a ball of radius r_k (see section 3.2 of Edelsbrunner and Harer (2010)). This is depicted in Figure 16, and for more details of this see Appendix E.1. Given this observation, we can determine p_{e_k} by finding the probability of exactly k points falling within a ball of radius r_k.

For normally distributed coordinates, the probability of a point falling within a ball of radius r_k and centre c is given by

$$P(u \in B_{r_k}(c) | \mu, \Sigma) = \int_{B_{r_k}(c)} p(u|\mu, \Sigma)\, du. \tag{18}$$

(18) presents a challenging computation and results for this integral are provided in Gilliland (1962) when $d = 2$. It is therefore not possible to obtain an exact expression for p_{e_k} when $k \geq 3$ and we instead rely on empirical approximations.

Figure 7 shows Monte Carlo estimates of p_{e_k} for increasing r_k, where the case $k = 2$ is provided for reference. Points were sampled from $\mathcal{N}(0, \Sigma)$ and the left and right panel show the estimated connection probabilities for $\Sigma = I_2$ and $\Sigma = \left(\begin{smallmatrix} 2 & 0 \\ 0 & 1 \end{smallmatrix}\right)$, respectively. From the figure, we see that a larger radius is required to obtain the same probability of connection as k grows. Additionally, by comparing the left and right panels of Figure 7, we see that, as the elements of Σ increase, the radii r_k must also increase to obtain the same probability of a connection.



(a) (b)

Figure 7: Empirical probability of a hyperedge occurring for $k = 2, 3, 4, 5$ and increasing r_k. Points were simulated from a Normal distribution with $\mu = (0, 0)$, and $\Sigma = I_2$ in (a) and $\Sigma = \left(\begin{smallmatrix} 2 & 0 \\ 0 & 1 \end{smallmatrix}\right)$ in (b).

6.3 Expected Degree for the i^{th} Node

In this section we consider the expected degree of the i^{th} node. This is presented in Proposition 6.2 and follows from (O1) and (O2). From this result we observe how the parameters e_k and p_{e_k} affect to connectivity of our model.

Proposition 6.2. *Let p_{e_k} denote the probability of the hyperedge e_k occurring in the nsRGH g, and let g^* be the hypergraph obtained by modifying g with noise $\boldsymbol{\varphi}$. The expected degree of the i^{th} node is given by*

$$\mathbb{E}\left[Deg^{g^*}_{(i)} | \boldsymbol{\varphi}, \Sigma, \boldsymbol{r}\right] = \sum_{k=2}^{K} \binom{N-1}{k-1} \left[(1-\varphi_k)p_{e_k} + \varphi_k(1-p_{e_k})\right]. \tag{19}$$

Proof. See Appendix F.2. □

6.4 Degree Distribution for the i^{th} Node

In this section we build upon Section 6.3 and consider the degree distribution for the i^{th} node. Using the observations from Section 6.1 and the results in Section 6.2, we present the degree distribution for edges with $k = 2$ in Proposition 6.3. This result follows by considering the probability of a hyperedge being present in g^* given its state in g, and by observing that $k = 2$ hyperedges contributing to the degree of the i^{th} node occur independently of each other.

Proposition 6.3. *Let g represent the nsRGH generated from $\boldsymbol{U}$ and $\boldsymbol{r}$, and let g^* be the hypergraph obtained by modifying g with noise $\boldsymbol{\varphi}$. It follows that*

$$Deg^{g^*}_{(i,2)} | \boldsymbol{\varphi}, \Sigma, \boldsymbol{r} \sim Binomial\left(N-1, (1-\varphi_2)p_{e_2} + \varphi_2(1-p_{e_2})\right), \tag{20}$$

where $p_{e_2} = P(y^{(g)}_{e_2} = 1 | \Sigma, r_2)$ is the probability of e_2 being present in g.

Proof. See Appendix F.3. □

Next, we consider the degree distribution for the i^{th} node for order $k = 3$ hyperedges. Consider the hyperedges $\{i, j, k\}$ and $\{i, j, l\}$ which contribute to $\text{Deg}^{g}_{(i,3)}$. Since these hyperedges both depend on i and j, it follows that the hyperedges present in the degree summation are no longer independent and so we rely on an approximation. This is presented in Proposition 6.4 and a similar result can be obtained for hyperedges with order $k > 3$.

Proposition 6.4. *Let g represent the nsRGH generated from $\boldsymbol{U}$ and $\boldsymbol{r}$, and let g^* be the hypergraph obtained by modifying g with noise $\boldsymbol{\varphi}$. It follows that*

$$Deg^{g^*}_{(i,3)} | \boldsymbol{\varphi}, \Sigma, \boldsymbol{r} \dot{\sim} Poisson\left(\sum_{\{e_3 \in \mathcal{E}_{N,3} | i \in e_3\}} \binom{N-1}{2} \left[(1-\varphi_3) p_{e_3} + \varphi_3 (1 - p_{e_3}) \right] \right), \tag{21}$$

where $p_{e_3} = P(y^{(g)}_{e_3} = 1 | \Sigma, r_3)$ is the probability of e_3 being present in g and $X \dot{\sim} f(x)$ indicates that X is approximately distributed according to $f(x)$.

Proof. See Appendix F.4. □

Propositions 6.3 and 6.4 tell us how the model parameters affect the degree distribution of the i^{th} node. From these results we may also obtain summaries about the degree, such as the expectation and variance. It is also interesting to examine the predictive degree by integrating over the prior distributions for $\boldsymbol{\varphi}, \mu$ and Σ. However, as commented in Section 6.2.2, obtaining an analytic expression for $P(y^{(g)}_{e_k} = 1 | r_k)$ is a challenging problem.

Throughout this section, we have presented results for the degree distribution of the i^{th} node. These results are intuitive given the observations detailed in (O1) and (O2) and demonstrate the effect of the connection probability in the graph $g(\boldsymbol{U}, \boldsymbol{r})$ combined with the strength of the noise parameters $\boldsymbol{\varphi}$. We note here that the results in this section can be extended to the model detailed in Algorithm 2.

7 Simulations

In this section we describe two different simulation studies. We begin Section 7.1 with an investigation of the flexibility of our modelling approach in comparison with two other hypergraph models from the literature. Then, in Section 7.2, we examine the predictive degree distribution conditional on an observed hypergraph.

7.1 Model depth comparisons

In this study we explore the range of hypergraphs that can be expressed in our modelling framework. We further compare this with two other statistical models from the literature, which have been designed according to different modelling aims. For each model, we specify several cases which are designed to highlight particular aspects of the model. Then, we simulate hypergraphs for each case and record summary statistics to characterise the simulated hypergraphs. We will begin by outlining the models of Stasi et al. (2014) and Ng and Murphy (2018) for a hypergraph with N nodes, and then describe and justify the choice of cases and summary statistics. Finally, we discuss the results of the simulations.

We first describe an extension of the β-model for random graphs (see Holland and Leinhardt (1981)), introduced by Stasi et al. (2014). In this model each node in the hypergraph is assigned

a parameter which controls its tendency to form edges, and we denote this parameter by β_i, for $i = 1, 2, \ldots, N$. Let $y_{e_k} = 1$ denote the presence of the hyperedge $e_k = \{i_1, i_2, \ldots, i_k\} \subseteq \{1, 2, \ldots, N\}$ for $k \geq 2$. The probability of the hyperedge e_k occurring is then given by

$$p(y_{i_1 i_2 \ldots i_k} = 1) = \frac{\exp\{\beta_{i_1} + \beta_{i_2} + \ldots \beta_{i_k}\}}{1 + \exp\{\beta_{i_1} + \beta_{i_2} + \ldots \beta_{i_k}\}}. \tag{22}$$

Since the hyperedges are assumed to occur independently conditional on $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_N)$, the likelihood is obtained by taking the product of Bernoulli likelihoods over all possible hyperedges $\mathcal{E}_{N,K}$. This likelihood can be shown to belong to the exponential family. Stasi et al. (2014) introduce several variants of this model, however we only rely on the above for our study.

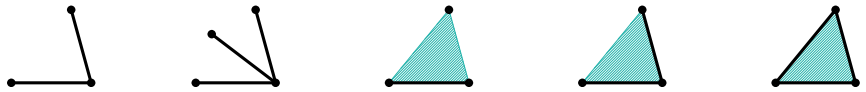
Next, we overview the model introduced in Ng and Murphy (2018) which assumes that hyperedges can be clustered according to their topic and size. In this context, the topic clustering implies that the hyperedges can be partitioned into latent classes and the probability of a node belonging to a hyperedge depends on its latent class. As an example, consider a coauthorship network where papers are represented as hyperedges. We may classify papers according to their academic discipline and impose that certain authors are more likely to contribute to papers within different disciplines. The size clustering is with respect to the hyperedge order, and this allows the model to capture variation in the size of hyperedges. To specify this model, we assume T topic clusters and S size clusters. It is assumed that the i^{th} node belongs to an edge with size label s and topic label t with probability $\alpha_s \phi_{it}$, so that $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_S)$ controls the size clusters and $\boldsymbol{\phi} = \{\phi_{it}\}_{i=1,2,\ldots,N, t=1,2,\ldots,T}$ controls the topic clusters. Additionally, we let $\pi = (\pi_1, \pi_2, \ldots, \pi_T)$ and $\tau = (\tau_1, \tau_2, \ldots, \tau_S)$ denote the prior topic and size assignment probabilities, respectively. To write down the likelihood, we let $x_{ij} = 1$ indicate that the i^{th} node belongs to the j^{th} edge, $z_{jt}^{(1)} = 1$ indicate that the j^{th} edge has topic label t, and $z_{js}^{(2)} = 1$ indicate that the j^{th} edge has size label s. The likelihood is then given by

$$\mathcal{L}(x, z^{(1)}, z^{(2)}; \theta) = \prod_{j=1}^{M} \prod_{t=1}^{T} \prod_{s=1}^{S} \left[\pi_t \tau_s \prod_{i=1}^{N} (\alpha_s \phi_{it})^{x_{ij}} (1 - \alpha_s \phi_{it})^{1 - x_{ij}} \right]^{z_{jt}^{(1)} z_{js}^{(2)}}. \tag{23}$$

Finally, to ensure the model is identifiable, we set $\alpha_S = 1$. Ng and Murphy (2018) also introduce a version of this model which only assumes a topic clustering, but we do not use this for our study.

In this simulation study we consider the models detailed above and the hypergraph model described in Algorithm 2. Before describing the set up, we note that each of these models has been designed for a different purpose. The β-model of Stasi et al. (2014) allows fine control over the degree distribution through the parameters $\boldsymbol{\beta}$, and the model of Ng and Murphy (2018) is designed to describe hyperedges which exhibit a clustering structure. The modelling choices in our approach impose differing characteristics on the hypergraphs simulated from this model. By determining the hyperedges from the Čech complex, we expect that the resulting hypergraphs will exhibit transitivity since, if $\{i, j\}$ and $\{i, k\}$ lie sufficiently close to be connected, then $\{j, k\}$ is likely to also be present in the hypergraph. Additionally, the presence of hyperedges $\{i, j\}, \{i, k\}$ and $\{j, k\}$ suggests that the hyperedge $\{i, j, k\}$ is more likely to be present. To model hypergraphs with different characteristics within our framework, we can investigate the effect of changing the assumed distribution of the latent coordinates and using an alternative simplicial complex. This point will be discussed in more detail in Section 9.

We now describe the set of metrics used to capture the above model behaviours. To capture the connectivity patterns we expect from our model, we record counts for the subgraphs depicted in Figure 8. The degree distribution and spread of hyperedge orders are measured by recording



(a) 3 star (b) 4 star (c) h_1 (d) h_2 (e) h_3

Figure 8: Motifs counted in model simulations.

the percentiles of the node degrees and edge sizes, respectively. Additionally, we record the density of the hyperedges of order $k = 2$ and $k = 3$. Note that, since the number of possible hyperedges of order k is $\binom{N}{k}$, we expect that the density of order k edges will decrease as k grows. Finally, to determine the clustering in the hypergraph, we project the hypergraph onto a pairwise graph such that the edge $\{i, j\}$ exists if i and j are present in the same hyperedge. Then, given this pairwise graph, we determine the community structure using the leading eigenvector with the function `cluster_leading_eigen()` in the igraph package in R (Csardi and Nepusz (2006)). We report the modularity of this clustering and the number of clusters. The modularity measures the strength of the clustering and lies within $[-1, 1]$, where a high value indicates that the network can be divided clearly into clusters. For each model, we have specified several cases which showcase the features of the model and a summary of these cases can be found in Table 1.

Figures 9, 10 and 11 show the results of simulation for 10,000 repetitions. Clockwise from the top left, these plots show subgraph counts for the motifs depicted in Figure 8, percentiles of the hyperedge order, the number of clusters and modularity, the density of hyperedges of order 2 and 3, and percentiles of the node degree. We are able to demonstrate the strengths of each model by comparing these metrics. For each model, the cases correspond to those described in Table 1.

In Figure 9 we see the summary measures for the β hypergraph model. From this bottom-left plot of this figure, we observe that each case demonstrates a very different behaviour in the degree distribution as we would expect. In case 1, all nodes have a similar degree and, in case 2, a portion of nodes have either a very large or small degree. It is also apparent that, since case 2 generates denser hypergraphs, larger motif counts are observed. The maximum hyperedge order was set to 4, and so no hyperedges for $k \geq 5$ are generated.

Figure 10 shows the equivalent plot for the model of Ng and Murphy (2018). Firstly, we note that the topic clustering is consistently captured for all simulated hypergraphs. We see that the degree distributions are largely the same, however by comparing case 1 and 4 we observe that we can express different levels of connectivity. When simulating from this model, we are unable to explicitly control the order of the hyperedges and we note that this is controlled by the probabilities $\boldsymbol{\alpha}$ and $\boldsymbol{\phi}$. We also observe reasonably little variation in the motif counts and, for most cases, find that triangles are more prevalent than the hypergraph motifs.

Finally, Figure 11 shows our results for the latent space hypergraph model. Overall, we observe that there are a greater number of motifs observed than in the previous models. Additionally, we see that we are able to demonstrate more control over the motif counts. For example, consider case 3 where the $k = 2$ hyperedges are denser and in case 4 where the $k = 3$ hyperedges are denser. The counts for triangles and h_1 subgraphs clearly reflect the number of hyperedges of each order. In this model we also observe some control over the degree distribution and density. As expected, when we increase the latent dimension to $d = 3$ and fix all other parameters we obtain a sparser hypergraph. To see this, compare cases 1 and 5. Finally, since our graph is not designed to capture clustering, we do not observe consistent estimates for the number of

Model	Case	Parameters
Stasi et al. (2014)	1) All nodes equally likely to form connections	$\beta_i = -1.4$ for $i = 1, 2, \ldots, N$
	2) Some nodes more likely to form connections	$\boldsymbol{\beta} = (-0.5, -0.53, \ldots, -1.97, -2)$
Ng and Murphy (2018)	1) Hyperedges in a single cluster	$G = K = 1$, $a = 1, \phi_{i1} = 0.075$, $\pi = b1, \tau = 1$
	2) Distinct topic clusters only	$G = 3, K = 1$, $a = 1, \phi_{i1} = 0.25$ for $i \in \mathcal{A}$, $\phi_{i2} = 0.25$ for $i \in \mathcal{B}$, $\phi_{i3} = 0.25$ for $i \in \mathcal{C}$, $\pi = (1/3, 1/3, 1/3), \tau = 1$
	3) Distinct size clusters only	$G = 1, K = 3$, $a = (0.2, 0.5, 1)$, $\phi_{i1} = 0.15, \pi = 1, \tau = (1/3, 1/3, 1/3)$
	4) Fuzzy topic clusters	$G = 2, K = 3$, $a = (0.4, 1), \phi_{i1} = 0.3$ for $i \in \mathcal{A}$, $\phi_{i2} = 0.3$ for $i \in \mathcal{B}$, $\phi_{i1} = \phi_{i2} = 0.2$ for $i \in \mathcal{C}$, $\pi = (1/2, 1/2), \tau = (1/3, 1/3, 1/3)$
LSH	1) Strongly correlated Σ	$\boldsymbol{r} = (0.18, 0.3, 0.35)$, $\mu = (0, 0)$, $\Sigma = 0.25 \begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix}$, $\boldsymbol{\psi}_0 = (0.01, 0.01, 0.01)$, $\boldsymbol{\psi}_1 = (0.01, 0.01, 0.01)$
	2) No correlation in Σ	$\boldsymbol{r} = (0.18, 0.3, 0.35), \mu = (0, 0)$, $\Sigma = 0.25 \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, $\boldsymbol{\psi}_0 = (0.01, 0.01, 0.01)$, $\boldsymbol{\psi}_1 = (0.01, 0.01, 0.01)$
	3) Dense in e_2, sparse in e_3, e_4	$\boldsymbol{r} = (0.2, 0.3, 0.35), \mu = (0, 0)$, $\Sigma = 0.25 \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, $\boldsymbol{\psi}_0 = (0.01, 0.01, 0.01)$, $\boldsymbol{\psi}_1 = (0.01, 0.5, 0.01)$
	4) Sparse e_2, e_4, dense in e_3	$\boldsymbol{r} = (0.1, 0.35, 0.4), \mu = (0, 0)$, $\Sigma = 0.25 \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, $\boldsymbol{\psi}_0 = (0.01, 0.01, 0.01)$, $\boldsymbol{\psi}_1 = (0.01, 0.01, 0.01)$
	5) Increase latent dimension from $d = 2$ to $d = 3$	$\boldsymbol{r} = (0.18, 0.3, 0.35), \mu = (0, 0)$, $\Sigma = 0.25 \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$, $\boldsymbol{\psi}_0 = (0.01, 0.01, 0.01)$, $\boldsymbol{\psi}_1 = (0.01, 0.01, 0.01)$

Table 1: Cases for each hypergraph model considered in the model depth comparison study. The case numbers correspond to the labels in Figures 9, 10 and 11. For all cases set $N = 50$ and, where appropriate, $K = 4$.

clusters. As commented previously, we may alter aspects of our model to incorporate community structure or to vary the degree distribution. For example, we may model the latent coordinates as a mixture of Gaussians.

In this study, we have clearly demonstrated the advantages of each modelling approach. It is clear that, for the construction we have chosen, our model presents a flexible framework that is particularly appropriate for hypergraph data which exhibit large motif counts. However, we may make alternative choices for the distribution of the latent coordinates and the Čech complex to

adapt our framework to express hypergraphs with different characteristics. We note that, whilst simulating these graphs may be straightforward, fitting them may be much more challenging. This point will be discussed further in Section 9.

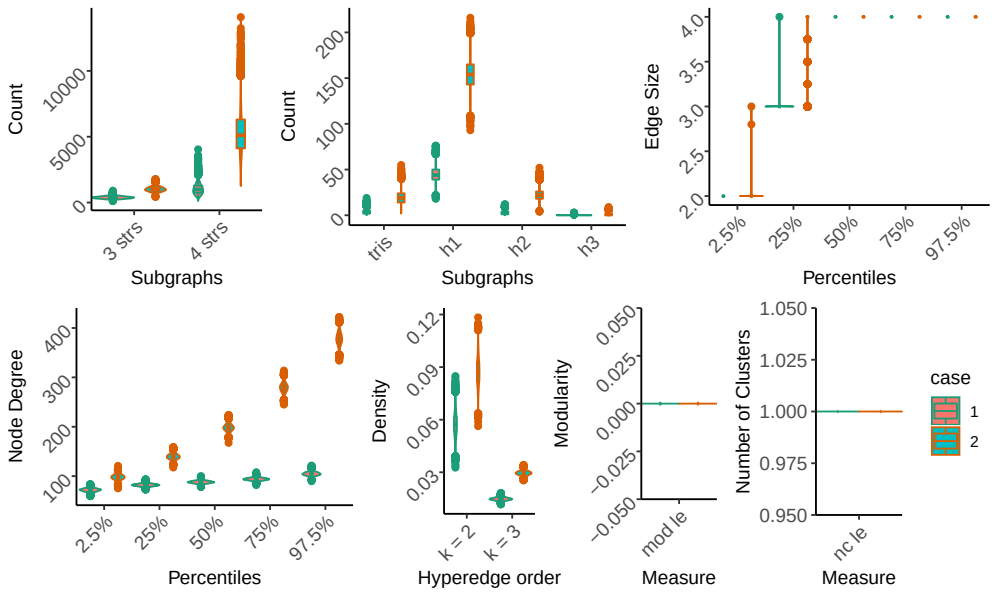


Figure 9: Summary of hypergraphs simulated from the model of Stasi et al. (2014). The cases are detailed in Table 1.

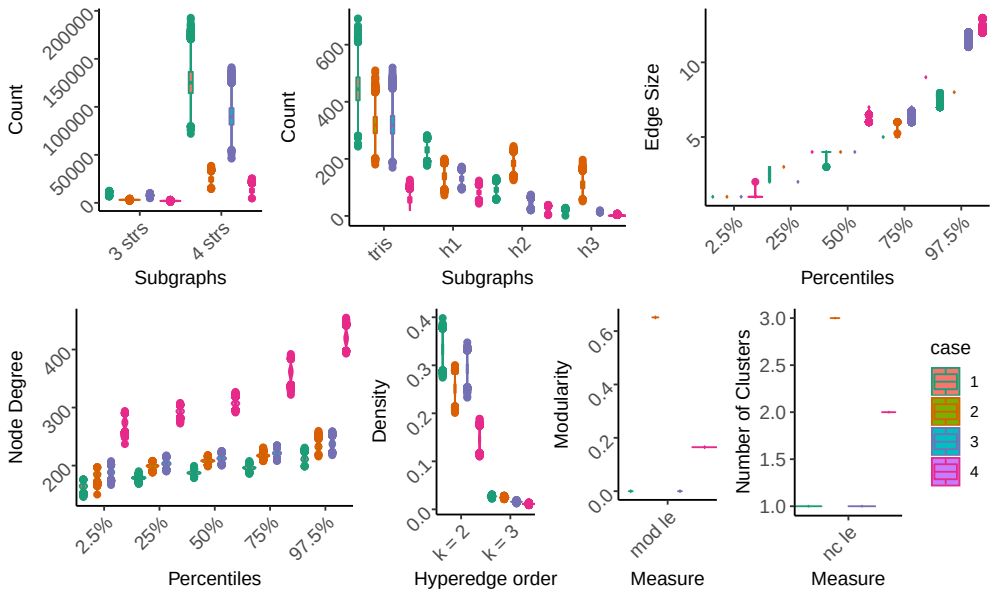


Figure 10: Summary of hypergraphs simulated from the model of Ng and Murphy (2018). The cases are detailed in Table 1.

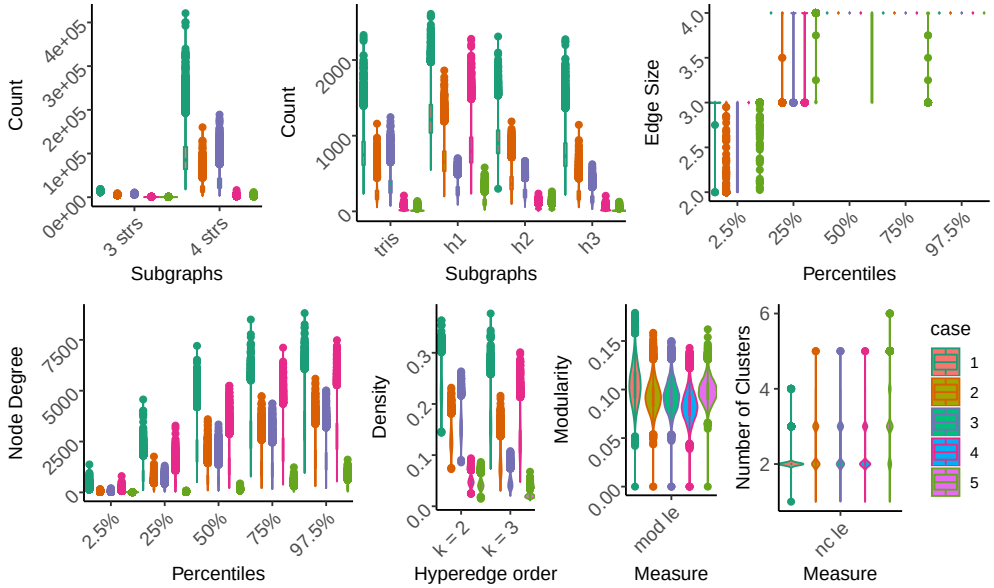


Figure 11: Summary of hypergraphs simulated from the model detailed in Algorithm 2. The cases are detailed in Table 1.

7.2 Prior Predictive vs Posterior Predictive

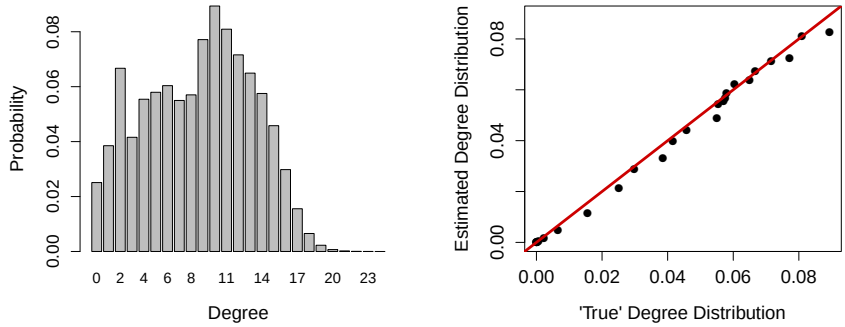
In this section we examine the predictive degree distribution conditional on an observed hypergraph. To explore the predictive distribution, we rely on the latent representation to simulate new nodes and their associated connections given estimated model parameters. Since the models of Stasi et al. (2014) and Ng and Murphy (2018) contain node specific parameters, we comment that it is not immediately obvious how to implement an analogue of this in either of their frameworks. We begin by describing the study and set up, and then present our findings.

Using the latent space representation, we are able to examine how newly simulated nodes connect to an observed hypergraph. Suppose that we have fitted the hypergraph model detailed in Algorithm 2 to a hypergraph h_{obs} and we obtain the parameter estimates $\hat{\mu}, \hat{\Sigma}, \hat{\boldsymbol{r}}, \hat{\boldsymbol{\varphi}}^{(0)}$ and $\hat{\boldsymbol{\varphi}}^{(1)}$. Conditional on these estimates, we may simulate new nodes and determine the hyperedges induced from these additional nodes. Through repeated simulation we can then empirically estimate the predictive degree distribution of the newly simulated nodes and h_{obs}.

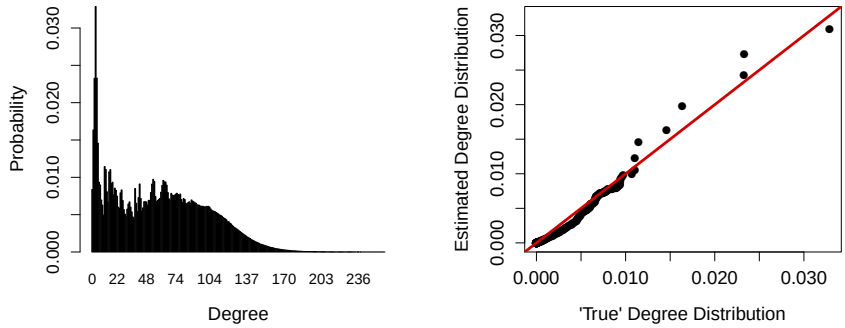
To implement this procedure, we being by simulating a hypergraph h_{sim} according to Algorithm 2 with $N = 50, K = 3, \boldsymbol{r} = (0.32, 0.4), \psi_k^{(0)} = \psi_k^{(1)} = 0.001, \mu = (0.16, 1.24)$ and $\Sigma = \begin{pmatrix} 0.58 & 0 \\ 0 & 0.58 \end{pmatrix}$. We then estimate the model parameters for this hypergraph and, after 10,000 post burn-in iterations, we obtain $\hat{\boldsymbol{r}} = (0.13, 0.16), \hat{\boldsymbol{\psi}}^{(0)} = (0.0058, 0.0014), \hat{\boldsymbol{\psi}}^{(1)} = (0.0057, 0.0035), \hat{\mu} = (-0.13, 0.44)$ and $\hat{\Sigma} = \begin{pmatrix} 0.14 & -0.0039 \\ -0.0039 & 0.078 \end{pmatrix}$.

To estimate the posterior predictive degree distribution we apply the following procedure n_{rep} times.

1. Simulate coordinates $u_i^* \sim \mathcal{N}(\hat{\mu}, \hat{\Sigma})$, for $i = 1, 2, \ldots, N^*$.
2. Determine the hypergraph obtained by including the hyperedges induced from $\boldsymbol{U}^* = \{u_i^*\}_{i=1}^{N^*}$, $\hat{\boldsymbol{r}}, \hat{\boldsymbol{\psi}}^{(0)}$ and $\hat{\boldsymbol{\psi}}^{(1)}$. We refer to this hypergraph as h_{sim}^*.



(a) Predictive degree distributions for hyperedges of order $k = 2$.



(b) Predictive degree distributions for hyperedges of order $k = 3$.

Figure 12: Comparison of prior and posterior predictive degree distributions for $N^* = 10$ newly simulated nodes. In each figure, the left panel shows the prior predictive degree distribution, and the right panel shows a qq-plot of the prior and posterior predictive degree distributions. Figures 12a and 12b show the degree distributions for hyperedges of order 2 and 3, respectively.

3. Calculate the degree distribution of h^*_{sim}.

By averaging over the $n_{rep} = 100,000$ simulated degree distributions, we then obtain an estimate of the degree distribution. Since we know the true model parameters, we also estimate the true predictive degree distribution. We refer to this distribution as the prior predictive below.

Recall that, for $u_i \in \mathbb{R}^2$, two coordinates are specified as anchor points throughout posterior sampling. The fixing of these points determines the scaling and hence affects the magnitude of Σ and $\boldsymbol{r}$. It is therefore not appropriate to compare parameter estimates with the truth directly. However, by comparing the prior and posterior predictive degree distributions, we consider a fair comparison between the true and estimated model parameters. We expect that these two distributions are similar if the model has been fitted well.

The full estimated prior and posterior predictive degree distributions are presented in Figure

12. For each subplot, the left panel shows the estimated prior predictive and the right panel shows a qq-plot of the prior and posterior predictive degree distributions. We see that there is a strong correspondence between the two distributions, particularly for the $k = 2$ edges. However, there is a slight difference in the upper tail for hyperedges of order $k = 3$. This may be due to the complexity of the space.

We may also examine the distributions for hyperedges occurring between newly simulated nodes only, and the hyperedges occurring between the nodes of h_{sim} and the newly simulated nodes. Plots for these cases are presented in Appendix G and, for both cases, we see a close correspondence between the prior and posterior predictive degree distributions.

8 Star Wars: A New Hope

In this section we consider a dataset constructed from the script of 'Star Wars: A New Hope' which describes co-occcurrence between the eight main characters. We represent this as a hypergraph where the nodes represent characters and hyperedges indicate which characters appeared in a scene together. In this dataset we have $N = 8$ and $K = 4$, and we remove repeated hyperedges and hyperedges of order one to ensure the data is amenable to analysis under our model. This dataset was considered in Ng and Murphy (2018), and we compare and contrast our methodology to this approach.

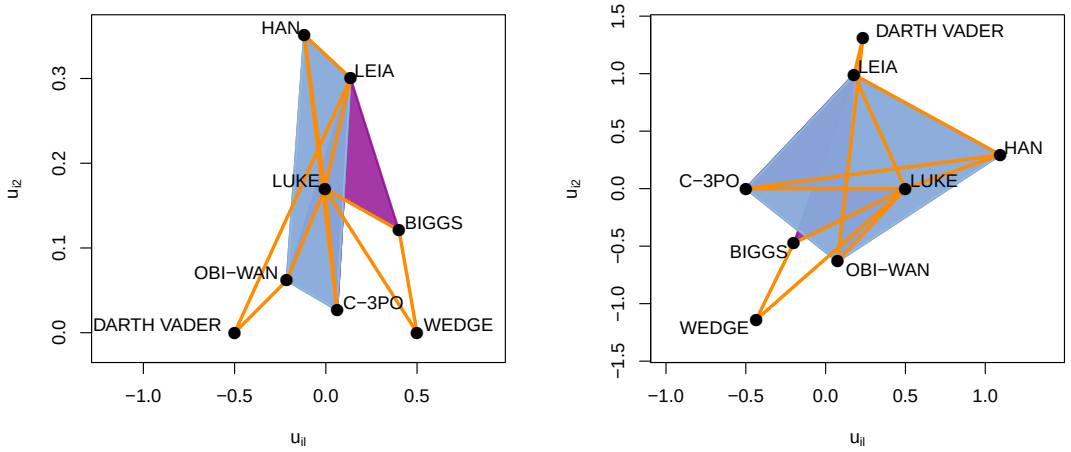
Recall that, in our model, observed hyperedges can be explained by the latent geometry or the hyperedge modification. To ensure most hyperedges are explained by the latent representation, we fix an upper limit for the parameters $\boldsymbol{\varphi}$. In doing so, we encourage interpretable latent coordinates and improve the quality of predictive inference. To begin, we fit the model detailed in Algorithm 1 and set the upper limit for φ_k to be $0.75 \times$ the density of order k hyperedges, for $k = 3, 4$.

The posterior mean of the latent coordinates after 37500 post burn-in iterations is given in Figure 13a. In this figure, orange lines indicate a pairwise connection, and blue and purple regions correspond to order 3 and 4 hyperedges, respectively. We observe a group of well-connected nodes which contains the character "Luke" in its centre. In Ng and Murphy (2018), this character was highlighted to be likely to occur in the two largest topic clusters, and we comment that the latent representation reflects this characters importance. Note that a similar observation is made in the network visualisation literature, where nodes with a greater number of connections are placed more centrally. The main group of nodes in Figure 13a is largely determined by the order 3 and 4 hyperedges between "Leia", "C-3PO", "Luke", "Obi-Wan" and "Han". We note that the characters "Wedge" and "Darth Vader" are less connected, and so we see them located on the periphery of the latent representation.

We also consider setting the upper limit for φ_k to be $1.5 \times$ the density of order k hyperedges, for $k = 3, 4$, and the posterior mean of the latent coordinates for this case is given in Figure 13b. Here we also see the importance of the character "Luke" reflected in the latent coordinates, however the increased noise parameter means that the latent coordinates are less constrained. To make this point further, we now consider the variability in the observed connections.

As mentioned in Section 7, our modelling framework allows us to explore predictive distributions. Given the fitted model, we can simulate new connections to examine how variable the degree of specific nodes are expected to be. More specifically, we consider the degree distributions for the characters "Luke" and "Darth Vader" by repeatedly simulating their connections given $\hat{\boldsymbol{U}}$, $\hat{\boldsymbol{r}}$ and $\hat{\boldsymbol{\varphi}}$. We do this for both choices of upper limits on $\boldsymbol{\varphi}$.

The results of this for upper limits of 0.75 and 1.5 times the hyperedge densities are shown in Figures 14a and 14b, respectively. In both Figures, we see a clear difference between the levels



(a) Upper limit of φ_k is set to $0.75 \times$ the observed order k hyperedge density.

(b) Upper limit of φ_k is set to $1.5 \times$ the observed order k hyperedge density.

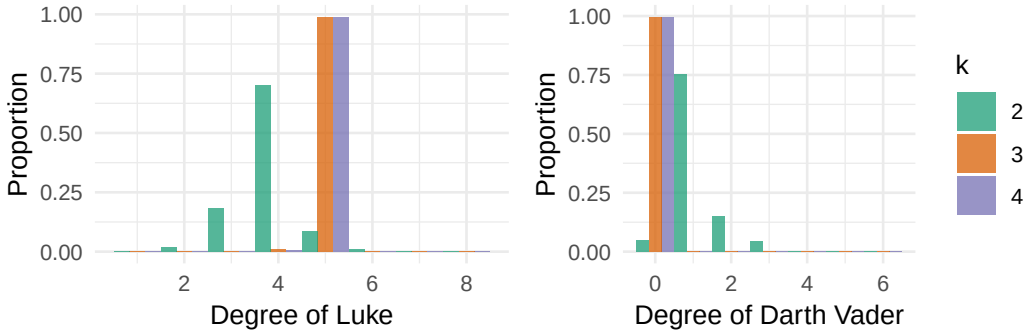
Figure 13: Posterior mean of latent coordinates for the Star Wars dataset for different upper limits on $\boldsymbol{\varphi}$. Connections in orange, blue and purple correspond to hyperedges of order $k = 2, 3$ and 4, respectively.

of connectivity for each of these characters. Since "Luke" is more centrally located with respect to the latent coordinates, we observe that this characters is expected to be more connected than "Darth Vader" who is located on the periphery. This tells us that nodes which are more centrally located are expected to be more connected in the hypergraph. Comparing Figures 14a and 14b, we more variability and a higher level of connectivity in Figure 14b. This is due to the larger upper limit on $\boldsymbol{\varphi}$. Since the noise is estimated to be larger, the latent representation explains fewer of the observed hyperedges. Therefore, when we fix the coordinates to $\hat{\boldsymbol{U}}$, our estimates may not reflect the observed hypergraph well.

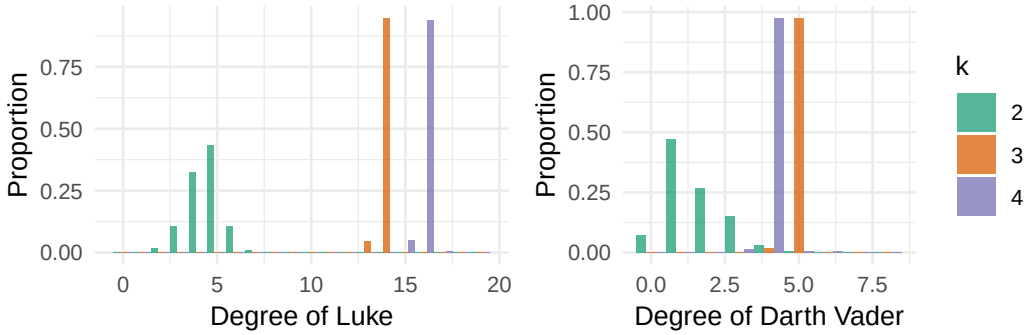
Although we are able to draw parallels between the our observations and the observations of Ng and Murphy (2018), our approach differs considerably. We now comment on the advantages of each approach. Firstly, in Ng and Murphy (2018) the authors incorporate multiple occurrences of a hyperedge and hyperedges containing a single node into their analysis. Our methodology does not facilitate this, and so the dataset was reduced accordingly. Secondly, our model provides a visualisation of the hypergraph and the approach of Ng and Murphy (2018) does not. When the parameters $\boldsymbol{\varphi}$ are appropriately constrained, this visualisation reflects many observations of the analysis in Ng and Murphy (2018). Finally, our framework allows us to investigate the predictive distributions, and this has allowed us to comment on the expected variability of the degree of certain nodes.

9 Discussion

In this paper we have introduced a latent space model for hypergraph data in which the nodes are represented by coordinates in $\mathbb{R}^d$. To extend the framework introduced in Hoff et al. (2002),



(a) Upper limit of φ_k is set to 0.75 $\times$ the observed order k hyperedge density.



(b) Upper limit of φ_k is set to 1.5 $\times$ the observed order k hyperedge density.

Figure 14: Predicted degree distributions conditional on the fitted model. Given $\hat{\boldsymbol{U}}$, $\hat{\boldsymbol{r}}$ and $\hat{\boldsymbol{\varphi}}$ we simulate the connections in the hypergraph to estimate the degree distribution, and the upper limit for $\boldsymbol{\varphi}$ is 0.75 the hyperedge density in Figure 14a and 1.5 the hyperegde density in Figure 14b. The left plots show the degree distribution for "Luke" and the right plots show the degree distribution for "Darth Vader". The order 2, 3, and 4 hyperedges are shown in green, orange, and purple, respectively.

we have relied on a modification of a nerve construction which allows us to express non-simplicial hypergraphs. This application of a nerve draws a connection between stochastic geometry and latent space network models, and allows us to develop a parsimonious hypergraph model. The latent representation imposes properties on the hypergraphs generated from our model, including a type of 'higher-order transitivity'. This property, in which a presence of an order k hyperedge is more likely given the presence of subsets of the hyperedge, is highlighted in the model depth simulations in Section 7.1. In particular, we see a greater presence of certain subgraph counts in comparison to the two other models considered. It is important to note that, depending on the modelling choices, particular hypergraph relationships may be challenging to represent using our nerve construction. For example, the maximum number of possible leaves in a star will be limited by the dimension of the latent space. However this may be mediated by either choosing a different convex set to generate the nerve, increasing the probability of hyperedge modification or adopting a different specification for the latent positions.

The modification of the indicators for the hyperedges has two main motivations. Firstly, without this modification, the conditional distribution $\mathcal{L}(\boldsymbol{U}, \boldsymbol{r}; h_{N,K})$ would be equal to one only

when there is a perfect correspondence between the observed and estimated hyperedges. Hence model fitting may be difficult. Secondly, the modification extends the support of the model to the space of all hypergraphs and, without the modification, it is unclear whether an observed hypergraph is expressible within our framework which greatly limits the applicability of our model. We note here that techniques such as tempering can be used to aid model fitting, but the challenge of characterising the support of the model still remains. From (14) we observed that a hypergraph generated from our model is a modification of a nsRGH and, as the probability of modification grows, the generated hypergraphs will behave more like the hypergraph analogue to an Erdős-Rényi random graph. Therefore, to maintain the hyperedge properties inherited from the latent space, we impose that the probability of modification is small. For a hypergraph on N nodes, the density of order k hyperedges is given by the fraction of possible order k hyperedges that are present in the hypergraph. It is clear that, to obtain a similar density of hyperedges for different k, the probability of modification must scale according to number of possible hyperedges of each order. This means that the magnitude of the modification probability will decrease as k grows. The hyperedge modification has a connection with measurement error models in which the observations are assumed to be measured with some noise. In the context of networks, Le et al. (2018) investigate the recovery of an underlying true network given a set of noisy observations. Whilst this differs from our setting, we can view our model in a similar way in which the nsRGH represents the truth. This helps motivate our observation on the magnitude of the probability of hyperedge modification.

To obtain posterior samples we rely on a Metropolis-Hastings-within-Gibbs MCMC scheme in which each parameter is sampled conditionally on the remaining parameters. In Section 4, we observed that the conditional distribution $p(g_{N,K}(\boldsymbol{U}, \boldsymbol{r})|\mu, \Sigma, \boldsymbol{r})$ is invariant to rotations, translations and reflections of the latent coordinates $\boldsymbol{U}$. However, since samples are obtained from the conditionals, these sources of non-identifiability can be removed in a post-processing step using a Procrustes transform. This approach is typically used for latent space network models and ensures that the samples have a clear interpretation. We instead infer the latent representation on the Bookstein space of coordinates, which avoids the need for post-processing and further removes the source of non-identifiability from joint rescaling of $\boldsymbol{U}$ and $\boldsymbol{r}$. To initialise the MCMC, we rely on techniques commonly used in the latent space network literature. Since random initialisation of $\boldsymbol{U}$ and $\boldsymbol{r}$ performs poorly, we use generalised multidimensional scaling (GMDS) to determine initial values of $\boldsymbol{U}$ and the radii are then scaled accordingly. GMDS was used in this context by Sarkar and Moore (2006), and details of the MCMC initialisation are given in Appendix D. We also exploit the connection with computational topology in our MCMC scheme. By relying on existing tools we are able to sample from our model and, in particular, to calculate the Čech complex we use the GUDHI C++ library (see The GUDHI Project (2015)).

Although the model we present can in theory be extended to hypergraphs with arbitrary maximum edge size K, this proves computationally challenging in practice. In our examples, we have predominantly restricted to cases in which K is equal to 3 or 4. This allows insight on the hypergraph relationships, but this restriction of K is currently a large limitation of our methodology. In the next paragraph we comment on a possible direction for a more scalable model. As previously mentioned, there are some motifs, for example stars, which may be difficult to express in our framework. Whilst the hyperedge modification can aid fitting here, increasing the probability of modification may have undesirable effects on the other connections. The number of leaves that can be expressed in a star is also related with the choice of latent dimension (see Helly's Theorem, Section 3.2 of Edelsbrunner and Harer (2010)). For interpretation, we have assumed that d is small and choosing d in a more principled manner requires careful consideration. In this paper we have presented two different models (detailed in Algorithms 1 and 2) for hypergraph. The model given in Algorithm 1 has a single modification parameter for each

order hyperedge, and this model has a clear interpretation. From Proposition 3.1 in Lunagómez et al. (2019), it follows that hypergraphs with a greater number of modifications are less likely to occur. However, the interpretation for the model in Algorithm 2 is less straightforward. Since there are competing sources of modification, the above argument only applies when the density of the hypergraphs remain fixed. A related framework is considered in Le et al. (2018), where the authors consider recovery of a true network given a set of noisy realisations. In this work, the sources of edge noise are divided into false positives and false negatives. By viewing the underlying nsRGH $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$ as the truth, we see that the probabilities of hyperedge modification in Algorithm 2 have an analogous interpretation.

There are a number of extensions to the model we have presented that can be explored. For instance, the choice of underlying distribution on $\boldsymbol{U}$ will affect the characteristics of the hypergraphs expressed by the model. This can be examined from a theoretical perspective by adapting the arguments discussed in Section 6, or practically by simulating from the generative model. Exploring this would further explain which aspects of the model depth simulations in Section 7.1 are an artefact of our modelling choices. In Spencer and Rohilla Shalizi (2017) the authors assume that the latent coordinates in a latent position are generated according to a Poisson process, and show that this is able express graphs with various levels of sparsity. It would be interesting to examine this within the hypergraph setting, and we leave this to future work. Another extension would be to consider modelling hypergraphs which exhibit community structure. One approach would be to assume that the latent coordinates are distributed according to a mixture of Gaussians, an idea which has been explored in the latent space network literature (for example, see Handcock et al. (2007)). Alternatively, a generalisation of the latent position framework in which connections are determined via a dot product (see Rubin-Delanchy et al. (2017)) is able to express graphs with community structure. This idea may also be considered in the hypergraph setting, though this extension is likely to be more involved. There also exist several other interesting extensions for which the adaptation of our model is less clear. In many real world examples, the hyperedges may occur multiple times. This motivates developing a model for non-binary hypergraphs, though it is not clear how to extend our model accordingly. Another line of future work stems from considering alternatives to the Čech complex. One choice to consider is the Vietoris-Rips complex (Vietoris (1927), Gromov (1987)) in which an order k hyperedge exists if each of the $\binom{k}{2}$ balls of radius r intersect. Evaluating this complex only requires considering the pairwise intersections, and so is likely more scalable in terms of K. Finally, in our paper we have not explored the addition of covariate information. Typically covariate information is included through an autoregressive term and a similar approach may be explored here.

References

Aksoy, S., Kolda, T. G., and Pinar, A. (2016). Measuring and Modeling Bipartite Graphs with Community Structure. *arXiv e-prints*, page arXiv:1607.08673.

Barabási, A.-L. and Albert, R. (1999). Emergence of scaling in random networks. *Science*, 286(5439):509–512.

Barabási, A.-L. and Pósfai, M. (2016). *Network science*. Cambridge University Press, Cambridge.

Benson, A. R., Abebe, R., Schaub, M. T., Jadbabaie, A., and Kleinber, J. (2018). Simplicial closure and higher-order link prediction. *Proceedings of the National Academy of Sciences*, 115(48):E11221–E11230.

Berg, M. d., Cheong, O., Kreveld, M. v., and Overmars, M. (2008). *Computational Geometry: Algorithms and Applications*. Springer-Verlag TELOS, Santa Clara, CA, USA, 3rd ed. edition.

Bookstein, F. L. (1984). A statistical method for biological shape comparisons. *Journal of Theoretical Biology*, 107(3):475 – 520.

Bookstein, F. L. (1986). Size and shape spaces for landmark data in two dimensions. *Statist. Sci.*, 1(2):181–222.

Cai, D., Campbell, T., and Broderick, T. (2016). Edge-exchangeable graphs and sparsity. In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems 29*, pages 4249–4257. Curran Associates, Inc.

Campbell, T., Cai, D., and Broderick, T. (2018). Exchangeable trait allocations. *Electron. J. Statist.*, 12(2):2290–2322.

Caron, F. and Fox, E. B. (2017). Sparse graphs using exchangeable random measures. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(5):1295–1366.

Chien, I., Lin, C.-Y., and Wang, I.-H. (2018). Community detection in hypergraphs: Optimal statistical limit and efficient algorithms. In Storkey, A. and Perez-Cruz, F., editors, *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pages 871–879, Playa Blanca, Lanzarote, Canary Islands. PMLR.

Cooley, O., Fang, W., Del Giudice, N., and Kang, M. (2018). Subcritical random hypergraphs, high-order components, and hypertrees. *arXiv e-prints*, page arXiv:1810.08107.

Cox, T. F. and Cox, M. A. A. (2000). *Multidimensional Scaling*. Chapman & Hall, 2 edition.

Crane, H. and Dempsey, W. (2018). Edge exchangeable models for interaction networks. *Journal of the American Statistical Association*, 113(523):1311–1326.

Csardi, G. and Nepusz, T. (2006). The igraph software package for complex network research. *InterJournal*, Complex Systems:1695.

Delaunay, B. (1934). Sur la sphére vide. *Bulletin de l'Académie des Sciences de l'URSS, Classe des Sciences Mathématiques et Naturelles*, pages 793–800.

Dempsey, W., Oselio, B., and Hero, A. (2019). Hierarchical network models for structured exchangeable interaction processes. *arXiv e-prints*, page arXiv:1901.09982.

Dryden, I. L. and Mardia, K. V. (1998). *Statistical Shape Analysis*. Wiley, Chichester.

Duchesne, P. and Micheaux, P. L. D. (2010). Computing the distribution of quadratic forms: Further comparisons between the liutangzhang approximation and exact methods. *Computational Statistics and Data Analysis*, 54(4):858 – 862.

Edelsbrunner, H. and Harer, J. (2010). *Computational Topology - an Introduction*. American Mathematical Society.

Edelsbrunner, H., Kirkpatrick, D., and Seidel, R. (1983). On the shape of a set of points in the plane. *IEEE Trans. Inf. Theor.*, 29(4):551–559.

Élie de Panafieu (2015). Phase transition of random non-uniform hypergraphs. *Journal of Discrete Algorithms*, 31:26 – 39. 24th International Workshop on Combinatorial Algorithms (IWOCA 2013).

Erdös, P. and Rényi, A. (1959). On random graphs, I. *Publicationes Mathematicae (Debrecen)*, 6:290–297.

Friel, N., Rastelli, R., Wyse, J., and Raftery, A. E. (2016). Interlocking directorates in irish companies using a latent space model for bipartite networks. *Proceedings of the National Academy of Sciences*, 113(24):6629–6634.

Gamerman, D. and Lopes, H. F. (2006). *Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference*. Chapman and Hall/CRC, 2 edition.

Ghoshdastidar, D. and Dukkipati, A. (2014). Consistency of spectral partitioning of uniform hypergraphs under planted partition model. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 27*, pages 397–405. Curran Associates, Inc.

Gilliland, D. C. (1962). Integral of the bivariate normal distribution over an offset circle. *Journal of the American Statistical Association*, 57(300):758–768.

Goodman, L. A. (1974). Exploratory latent structure analysis using both identifiable and unidentifiable models. *Biometrika*, 61(2):215–231.

Gromov, M. (1987). Hyperbolic groups. *Essays in Group Theory, Mathematical Sciences Research Institute Publications*.

Handcock, M. S., Raftery, A. E., and Tantrum, J. M. (2007). Model-based clustering for social networks. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 170(2):301–354.

Hoff, P. D., Raftery, A. E., and Handcock, M. S. (2002). Latent space approaches to social network analysis. *Journal of the American Statistical Association*, 97(460):1090–1098.

Holland, P. W., Laskey, K. B., and Leinhardt, S. (1983). Stochastic blockmodels: First steps. *Social Networks*, 5(2):109 – 137.

Holland, P. W. and Leinhardt, S. (1981). An exponential family of probability distributions for directed graphs. *Journal of the American Statistical Association*, 76(373):33–50.

Kahle, M. (2016). Random simplicial complexes. *arXiv e-prints*, page arXiv:1607.07069.

Karoński, M. and Łuczak, T. (2002). The phase transition in a random hypergraph. *Journal of Computational and Applied Mathematics*, 142(1):125 – 135. Probabilistic Methods in Combinatorics and Combinatorial Optimization.

Kim, B., Lee, K., Xue, L., and Niu, X. (2017). A Review of Dynamic Network Models with Latent Variables. *arXiv e-prints*, page arXiv:1711.10421.

Kim, C., Bandeira, A. S., and Goemans, M. X. (2018). Stochastic Block Model for Hypergraphs: Statistical limits and a semidefinite programming approach. *arXiv e-prints*, page arXiv:1807.02884.

Kolaczyk, E. D. (2009). *Statistical Analysis of Network Data: Methods and Models*. Springer Publishing Company, Incorporated, 1st edition.

Krivitsky, P. N., Handcock, M. S., Raftery, A. E., and Hoff, P. D. (2009). Representing degree distributions, clustering, and homophily in social networks with latent cluster random effects models. *Social Networks*, 31(3):204 – 213.

Kunegis, J. (2013). Konect: The koblenz network collection. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13 Companion, pages 1343–1350, New York, NY, USA. ACM.

Lazarsfeld, P. and Henry, N. (1968). *Latent structure analysis*. Houghton, Mifflin.

Le, C. M., Levin, K., and Levina, E. (2018). Estimating a network from multiple noisy realizations. *Electron. J. Statist.*, 12(2):4697–4740.

Leskovec, J. and Krevl, A. (2014). SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data.

Lin, C., Chien, I. E., and Wang, I. (2017). On the fundamental statistical limit of community detection in random hypergraphs. In *2017 IEEE International Symposium on Information Theory (ISIT)*, pages 2178–2182.

Liu, D., Blenn, N., and Mieghem, P. V. (2013). Characterising and modelling social networks with overlapping communities. *Int. J. Web Based Communities*, 9(3):371–391.

Lunagómez, S., Mukherjee, S., Wolpert, R. L., and Airoldi, E. M. (2017). Geometric representations of random hypergraphs. *Journal of the American Statistical Association*, 112(517):363–383.

Lunagómez, S., Olhede, S. C., and Wolfe, P. J. (2019). Modeling Network Populations via Graph Distances. *arXiv e-prints*, page arXiv:1904.07367.

Mansilla, S. P. and Serra, O. (2008). On s-arc transitive hypergraphs. *European Journal of Combinatorics*, 29(4):1003 – 1011. Homomorphisms: Structure and Highlights.

Marin, J.-M., Pudlo, P., Robert, C. P., and Ryder, R. J. (2012). Approximate bayesian computational methods. *Statistics and Computing*, 22(6):1167–1180.

Ng, T. L. J. and Murphy, T. B. (2018). Model-based clustering for random hypergraphs. *ArXiv e-prints*.

Penrose, M. (2003). *Random Geometric Graphs*. Oxford Studies in Probability.

Petersen, K. B. and Pedersen, M. S. (2012). The matrix cookbook. Version 20121115.

Pronzato, L., Wynn, H. P., and Zhigljavsky, A. (2019). Bregman divergences based on optimal design criteria and simplicial measures of dispersion. *Statistical Papers*, 60(2):195–214.

Pronzato, L., Wynn, H. P., and Zhigljavsky, A. A. (2018). Simplicial variances, potentials and mahalanobis distances. *Journal of Multivariate Analysis*, 168:276 – 289.

Rastelli, R., Friel, N., and Raftery, A. E. (2015). Properties of Latent Variable Network Models. *arXiv e-prints*, page arXiv:1506.07806.

Rubin-Delanchy, P., Priebe, C. E., Tang, M., and Cape, J. (2017). A statistical interpretation of spectral embedding: the generalised random dot product graph. *arXiv e-prints*, page arXiv:1709.05506.

Salnikov, V., Cassese, D., and Lambiotte, R. (2018). Simplicial complexes and complex systems. *European Journal of Physics*, 40(1):014001.

Salter-Townshend, M., White, A., Gollini, I., and Murphy, T. B. (2012). Review of statistical network analysis: models, algorithms, and software. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 5(4):243–264.

Sarkar, P. and Moore, A. W. (2006). Dynamic social network analysis using latent space models. In Weiss, Y., Schölkopf, B., and Platt, J. C., editors, *Advances in Neural Information Processing Systems 18*, pages 1145–1152. MIT Press.

Sharma, A., Srivastava, J., and Chandra, A. (2014). Predicting Multi-actor collaborations using Hypergraphs. *arXiv e-prints*, page arXiv:1401.6404.

Spencer, N. A. and Rohilla Shalizi, C. (2017). Projective, Sparse, and Learnable Latent Position Network Models. *arXiv e-prints*, page arXiv:1709.09702.

Stasi, D., Sadeghi, K., Rinaldo, A., Petrović, S., and Fienberg, S. E. (2014). β models for random hypergraphs with a given degree sequence. *ArXiv e-prints*.

Teerapabolarn, K. (2014). An improvement of poisson approximation for sums of dependent bernoulli random variables. *Communications in Statistics - Theory and Methods*, 43(8):1758–1777.

The GUDHI Project (2015). *GUDHI User and Reference Manual*. GUDHI Editorial Board.

Vietoris, L. (1927). Über den hheren zusammenhang kompakter räume und eine klasse von zusammenhangstreuen abbildungen. *Mathematische Annalen*, pages 454–472.

Zhou, D., Huang, J., and Schölkopf, B. (2006). Learning with hypergraphs: Clustering, classification, and embedding. In *Proceedings of the 19th International Conference on Neural Information Processing Systems*, NIPS'06, pages 1601–1608, Cambridge, MA, USA. MIT Press.

A Bookstein coordinates

In Bookstein coordinates, a set of points are chosen as the anchor points. They are fixed in the space and all other points are translated, rotated and scaled according to these points. In Appendix A.1 we describe the Bookstein coordinates in $\mathbb{R}^2$, and in Appendix A.2 we describe the Bookstein coordinates in $\mathbb{R}^3$.

A.1 Bookstein coordinates in $\mathbb{R}^2$

In $\mathbb{R}^2$, we set the anchor points $u_1^B = (u_{11}^B, u_{12}^B)$ and $u_2^B = (u_{21}^B, u_{22}^B)$ to be $(-1/2, 0)$ and $(1/2, 0)$, respectively. Let $\boldsymbol{U}^B$ denote the Bookstein coordinates and $\boldsymbol{U}$ denote the untransformed coordinates. Then $\boldsymbol{U}^B$ is given by

$$\begin{aligned}\boldsymbol{U}^B &= cR(\boldsymbol{U} - \boldsymbol{b}) \\ &= \frac{1}{\sqrt{(u_{21}^B - u_{11}^B)^2 + (u_{22}^B - u_{12}^B)^2}} \begin{bmatrix} \cos(a) & \sin(a) \\ -\sin(a) & \cos(a) \end{bmatrix} \left(\boldsymbol{U} - \frac{1}{2} \begin{bmatrix} u_{11}^B + u_{21}^B \\ u_{12}^B + u_{22}^B \end{bmatrix} \right), \end{aligned} \tag{24}$$

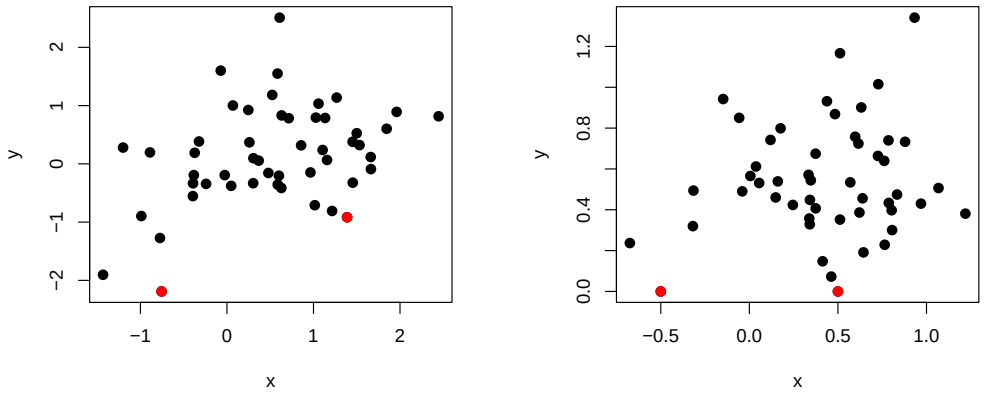


Figure 15: Bookstein transformation in $\mathbb{R}^2$. Left: original coordinates. Right: transformed Bookstein coordinates. The points highlighted in red are mapped to $(-1/2, 0)$ and $(1/2, 0)$.

where $a = \arctan\{(u^B_{22} - u^B_{12})/(u^B_{21} - u^B_{11})\}$. The Bookstein transformation can hence be seen as a translation, rotation and rescaling of the coordinates $\boldsymbol{U}$. Figure 15 shows an example of the Bookstein transformation in $\mathbb{R}^2$.

Furthermore, if $\boldsymbol{U} \sim \mathcal{N}(\mu, \Sigma)$, then we know that $\boldsymbol{U}^B \sim \mathcal{N}(\mu^B, \Sigma^B)$ where

$$\mu^B = cR(\mu - b), \tag{25}$$

$$\Sigma^B = c^2 R \Sigma R^T. \tag{26}$$

A.2 Bookstein coordinates in $\mathbb{R}^3$

Section 4.3.3 of Dryden and Mardia (1998) gives the Bookstein transformation for $\boldsymbol{U}$ where $u_i \in \mathbb{R}^3$. Following from (24) we first set $u^B_1 = (-1/2, 0, 0), u^B_2 = (1/2, 0, 0)$ and $u^B_3 = (u^B_{31}, u^B_{32}, 0)$. Then for $i = 4, 5, \ldots, N$ and $l = 1, 2, 3$ we calculate

$$w_{il} = u_{il} - \frac{1}{2}(u^B_{1l} + u^B_{2l}). \tag{27}$$

The Bookstein coordinate u^B_i for $i = 4, 5, \ldots, N$ is then given by

$$u^B_i = R_1 R_2 R_3 (w_{i1}, w_{i2}, w_{i3})/D_{12} \tag{28}$$

where

$$R_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix}, \qquad R_2 = \begin{bmatrix} \cos(\omega) & 0 & \sin(\omega) \\ 0 & 1 & 0 \\ -\sin(\omega) & 0 & \cos(\omega) \end{bmatrix} \tag{29}$$

$$R_3 = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad D_{12} = 2(w^2_{21} + w^2_{22} + w^2_{23})^{1/2}. \tag{30}$$

Furthermore, we have

$$\theta = \arctan(w_{22}/w_{21}) \tag{31}$$

$$\omega = \arctan(w_{23}/(w_{21}^2 + w_{22}^2)^{1/2}) \tag{32}$$

$$\phi = \arctan\left(\frac{(w_{21}^2 + w_{22}^2)w_{33} - (w_{21}w_{31} + w_{22}w_{32})w_{23}}{(w_{21}^2 + w_{22}^2 + w_{23}^2)^{1/2}(w_{21}w_{32} - w_{31}w_{22})}\right). \tag{33}$$

We see that the transformation in $\mathbb{R}^3$ is more involved than in $\mathbb{R}^2$ since we need to consider the effect of rotations over three different axes. Note that R_1, R_2 and R_3 correspond to a rotation around the x, y and z axes, respectively.

B Modifying the Hyperedge Indicators

In the generative model detailed in Algorithm 1, the indicators for all order k hyperedges are modified with probability φ_k. Since the probability φ_k is small, we expect a small number of the $\binom{N}{k}$ possible order k hyperedges to be modified and so naively simulating a Bernoulli(φ_k) random variable for each hyperedge is wasteful. Here we discuss an alternative approach for this step which avoids considering all possible hyperedges, and we comment that this can easily be adapted for the model detailed in Algorithm 2.

Instead of sampling $\binom{N}{k}$ Bernoulli random variables, we instead begin by sampling the number of order k hyperedges whose indicator we modify, n_k, from a Binomial $\left(\binom{N}{k}, \varphi_k\right)$. Then, we randomly sample n_k hyperedges from $\mathcal{E}_{N,k}$. When sampling a hyperedge, we want to sample an index from $\{i_1 < i_2 < \cdots < i_k | i_1, i_2, \ldots, i_k \in \{1, 2, \ldots, N\}\}$ and we do this by sampling i_1 to i_k in incrementally. This avoids sampling from the $\binom{N}{k}$ possible combinations directly and so is more efficient.

We will now discuss this in more detail for hyperedges of order $k = 2$. To sample indiced i_1 and i_2 such that $i_1 < i_2$ we

1. Sample i_1 with probability $p(i_1) = \dfrac{N - i_1}{\sum_{i=1}^{N-1}(N-i)}$, for $i_1 = 1, \ldots, (N-1)$.
2. Sample $i_2|i_1$ with probability $p(i_2|i_1) = \dfrac{1}{N - i_1}$, for $i_2 = (i_1 + 1), \ldots, N$.

A similar procedure can be used for arbitrary k.

Note that this procedure ignores the dependence between samples since, once a hyperedge is sampled, the remaining hyperedges are sampled from a subset of hyperedges of size $\binom{N}{k} - 1$. However, we expect this effect to be negligible since the majority of hyperedges are not modified.

C Conditional Posterior Distributions

C.1 Conditional posterior for μ

The conditional posterior for μ is given by

$$p(\mu|\boldsymbol{U}, \Sigma, m_\mu, \Sigma_\mu) \propto p(\boldsymbol{U}|\mu, \Sigma)p(\mu|m_\mu, \Sigma_\mu) \tag{34}$$

where $p(\mu|m_\mu, \Sigma_\mu) = \mathcal{N}(m_\mu, \Sigma_\mu)$ and $p(\boldsymbol{U}|\mu, \Sigma) = \prod_{i=1}^{N} \mathcal{N}(u_i|\mu, \Sigma)$.

We have

$$p(\mu|\boldsymbol{U},\Sigma,m_\mu,\Sigma_\mu) \propto \exp\left\{-\frac{1}{2}\sum_{i=1}^{N}(u_i-\mu)^T\Sigma^{-1}(u_i-\mu)-\frac{1}{2}(\mu-m_\mu)^T\Sigma_\mu^{-1}(\mu-m_\mu)\right\} \quad (35)$$

By recursively applying the result in Section of 8.1.7 Petersen and Pedersen (2012) we obtain

$$p(\mu|\boldsymbol{U},\Sigma,m_\mu,\Sigma_\mu) = \mathcal{N}\left((N\Sigma^{-1}+\Sigma_\mu^{-1})^{-1}\left(\Sigma^{-1}\sum_{i=1}^{N}u_i+\Sigma_\mu^{-1}m_\mu\right),(N\Sigma^{-1}+\Sigma_\mu^{-1})^{-1}\right). \quad (36)$$

C.2 Conditional posterior for Σ

The conditional posterior for Σ is given by

$$p(\Sigma|\boldsymbol{U},\mu,\Phi,\nu) \propto p(\boldsymbol{U}|\mu,\Sigma)p(\Sigma|\Phi,\nu) \quad (37)$$

where $p(\Sigma|\Phi,\nu)=\mathcal{W}^{-1}(\Phi,\nu)$ and $p(\boldsymbol{U}|\mu,\Sigma)=\prod_{i=1}^{N}\mathcal{N}(u_i|\mu,\Sigma)$.

We have

$$p(\Sigma|\boldsymbol{U},\mu,\Phi,\nu) \propto \prod_{i=1}^{N}\frac{1}{|\Sigma|^{1/2}}\exp\left\{-\frac{1}{2}(u_i-\mu)^t\Sigma^{-1}(u_i-\mu)\right\}|\Sigma|^{-(\nu+d+1)/2}\exp\left\{-\frac{1}{2}tr\left(\Phi\Sigma^{-1}\right)\right\} \quad (38)$$

$$\propto |\Sigma|^{-(\nu+d+N+1)/2}\exp\left\{-\frac{1}{2}\left[tr\left(\Sigma^{-1}\sum_{i=1}^{N}(u_i-\mu)(u_i-\mu)^T\right)+tr(\Phi\Sigma^{-1})\right]\right\} \quad (39)$$

$$\propto |\Sigma|^{-(\nu+d+N+1)/2}\exp\left\{-\frac{1}{2}tr\left(\left[\sum_{i=1}^{N}(u_i-\mu)(u_i-\mu)^T+\Phi\right]\Sigma^{-1}\right)\right\}. \quad (40)$$

Line (40) follows from the symmetry of Σ and $\sum_{i=1}^{N}(v_i-\mu)(v_i-\mu)^T$, and properties of the trace operator.

Hence, we obtain

$$p(\Sigma|\boldsymbol{U},\mu,\Phi,\nu) = \mathcal{W}^{-1}\left(\Phi+\sum_{i=1}^{N}(u_i-\mu)(u_i-\mu)^T,\nu+N\right). \quad (41)$$

C.3 Conditional posterior for $\psi_k^{(0)}$

The conditional posterior for $\psi_k^{(0)}$ is given by

$$p\left(\psi_k^{(0)}|\boldsymbol{U},\boldsymbol{r},h_{N,K},a_k^{(0)},b_k^{(0)}\right) \propto \mathcal{L}\left(\boldsymbol{U},\boldsymbol{r},\boldsymbol{\psi}^{(1)},\boldsymbol{\psi}^{(0)};h_{N,K}\right)p\left(\psi_k^{(0)}|a_k^{(0)},b_k^{(0)}\right) \quad (42)$$

where $\mathcal{L}\left(\boldsymbol{U},\boldsymbol{r},\boldsymbol{\psi}^{(1)},\boldsymbol{\psi}^{(0)};h_{N,K}\right)$ is as in (12) and $p\left(\psi_k^{(0)}|a_k^{(0)},b_k^{(0)}\right)=\text{Beta}\left(a_k^{(0)},b_k^{(0)}\right)$.

We have

$$p(\psi_k^{(0)}|\boldsymbol{U},\boldsymbol{r},h_{N,K},a_k^{(0)},b_k^{(0)})$$
$$\propto \left(\psi_k^{(0)}\right)^{d_k^{(01)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})} \left(1-\psi_k^{(0)}\right)^{d_k^{(00)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})} \left(\psi_k^{(0)}\right)^{a_k^{(0)}-1} \left(1-\psi_k^{(0)}\right)^{b_k^{(0)}-1} \tag{43}$$
$$\propto \left(\psi_k^{(0)}\right)^{d_k^{(01)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})+a_k^{(0)}-1} \left(1-\psi_k^{(0)}\right)^{d_k^{(00)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})+b_k^{(0)}-1}. \tag{44}$$

Hence, we obtain

$$p(\psi_k^{(0)}|\boldsymbol{U},\boldsymbol{r},h_{N,K},a_k^{(0)},b_k^{(0)})$$
$$= \text{Beta}\left(d_k^{(01)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})+a_k^{(0)}, d_k^{(00)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})+b_k^{(0)}\right). \tag{45}$$

C.4 Conditional posterior for $\psi_k^{(1)}$

The conditional posterior for $\psi_k^{(1)}$ is given by

$$p\left(\psi_k^{(1)}|\boldsymbol{U},\boldsymbol{r},h_{N,K},a_k^{(1)},b_k^{(1)}\right) \propto \mathcal{L}\left(\boldsymbol{U},\boldsymbol{r},\boldsymbol{\psi}^{(1)},\boldsymbol{\psi}^{(0)};h_{N,K}\right) p\left(\psi_k^{(1)}|a_k^{(1)},b_k^{(1)}\right) \tag{46}$$

where $\mathcal{L}\left(\boldsymbol{U},\boldsymbol{r},\boldsymbol{\psi}^{(1)},\boldsymbol{\psi}^{(0)};h_{N,K}\right)$ is as in (12) and $p\left(\psi_k^{(1)}|a_k^{(1)},b_k^{(1)}\right) = \text{Beta}\left(a_k^{(1)},b_k^{(1)}\right)$.

We have

$$p(\psi_k^{(1)}|\boldsymbol{U},\boldsymbol{r},h_{N,K},a_k^{(1)},b_k^{(1)})$$
$$\propto \left(\psi_k^{(1)}\right)^{d_k^{(10)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})} \left(1-\psi_k^{(1)}\right)^{d_k^{(11)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})} \left(\psi_k^{(1)}\right)^{a_k^{(1)}-1} \left(1-\psi_k^{(1)}\right)^{b_k^{(1)}-1} \tag{47}$$
$$\propto \left(\psi_k^{(1)}\right)^{d_k^{(10)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})+a_k^{(1)}-1} \left(1-\psi_k^{(1)}\right)^{d_k^{(11)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})+b_k^{(1)}-1}. \tag{48}$$

Hence, we obtain

$$p(\psi_k^{(1)}|\boldsymbol{U},\boldsymbol{r},h_{N,K},a_k^{(1)},b_k^{(1)})$$
$$= \text{Beta}\left(d_k^{(10)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})+a_k^{(1)}, d_k^{(11)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})+b_k^{(1)}\right). \tag{49}$$

D MCMC initialisation

For the MCMC scheme in Algorithm 3, a random initialisation is likely to perform poorly. Here we discuss our approach for initialising the MCMC scheme, and we begin with the the latent coordinates $\boldsymbol{U}$.

In Sarkar and Moore (2006), the authors present a procedure for inferring the latent coordinates in the scenario where the network is temporally evolving. Their scheme begins with an initialisation which uses generalised multidimensional scaling (GMDS). Traditional MDS (see Cox and Cox (2000)) finds a set of coordinates in $\mathbb{R}^d$ whose pairwise distances correspond to a distance matrix specified as an input. In GMDS, the distance is extended beyond the Euclidean distance and, in our context, we use the shortest path between nodes i and j as the distance

measure. To calculate the shortest paths we introduce a weighted adjacency matrix which incorporates the intuition than nodes which appear in a hyperedge are likely closer than nodes which are only connected by a pairwise edge. Once we have an initial value of $\boldsymbol{U}$ we then transform these coordinates onto the Bookstein space of coordinates. Our initialisation procedure for $\boldsymbol{U}$ is given in Algorithm 4.

Algorithm 4 Initialise $\boldsymbol{U}$.

Input: Observed hypergraph $h_{N,K}$
1) Let $A \in \mathbb{R}^{N\times N}$ denote a weighted adjacency matrix.
For $i, j \in \{1, 2, \ldots, N\}$, if $\{i, j\}$ are connected by a hyperedge
- let $A_{(i,j)} = 1$ if $\{i, j\}$ are only connected by a hyperedge of order $k = 2$,
- let $A_{(i,j)} = \lambda$ if $\{i, j\}$ are connected by a hyperedge of order $k > 2$.

2) Find the distance matrix $D \in \mathbb{R}^{N\times N}$, where $D_{(i,j)}$ is the shortest path between nodes $\{i, j\}$ in the weighted graph determined by A. For $i = j$, let $D_{(i,j)} = 0$.
3) Apply MDS to D to obtain coordinates $\boldsymbol{U}_0 \in \mathbb{R}^{N\times d}$.
4) Specify the index of the anchor points, and transform $\boldsymbol{U}_0$ onto Bookstein coordinates (see Appendix A).

Algorithm 5 Initialise μ and Σ.

Input: Hypergraph $h_{N,K}$, $\boldsymbol{r}_0$, $\boldsymbol{\psi}_0^{(0)}$, $\boldsymbol{\psi}_0^{(1)}$, $\mu \sim \mathcal{N}(m_\mu, \Sigma_\mu)$, $\Sigma \sim \mathcal{W}^{-1}(\Phi, \nu)$, N_{smp} and ϵ.
1) Calculate $T(h_{N,K})$, where $T(\cdot)$ is a vector of hypergraph summary statistics.
Let $n = 0$.
2) While $n < N_{smp}$
-Sample $\mu^* \sim \mathcal{N}(m_\mu, \Sigma_\mu)$ and $\Sigma^* \sim \mathcal{W}^{-1}(\Phi, \nu)$.
-Sample $u_i^* \sim \mathcal{N}(\mu^*, \Sigma^*)$ for $i = 1, 2, \ldots, N$.
Let $\boldsymbol{U}^*$ be the $N \times d$ matrix whose i^{th} row is u_i^*.
-Given initial $\boldsymbol{r}_0$, determine the hypergraph $g_{N,K}(\boldsymbol{U}^*, \boldsymbol{r}_0)$.
-Let $g_{N,K}^*$ by the hypergraph obtained by modifying $g_{N,K}(\boldsymbol{U}^*, \boldsymbol{r}_0)$ with noise $\boldsymbol{\psi}_0^{(0)}$ and $\boldsymbol{\psi}_0^{(1)}$
-Calculate $T\left(g_{N,K}^*\right)$.
-If $|T(h_{N,K}) - T(g_{N,K}^*)| < \epsilon$
Accept samples μ^* and Σ^*.
3) Let μ_0 and Σ_0 be the average of N_{smp} samples.

The radii $\boldsymbol{r}$ depend on the scale of $\boldsymbol{U}$, and so they are initialised in terms of $\boldsymbol{U}_0$. Given the initial latent coordinates, $\boldsymbol{r}_0$ is chosen to be the minimum radius which induces all edges that are present in $h_{N,K}$. The noise parameters $\boldsymbol{\psi}^{(0)}$ and $\boldsymbol{\psi}^{(1)}$ are initialised by sampling from their prior, where the prior values suggest that the perturbations are small.

To initialise the parameters μ and Σ we use an ABC scheme (see Marin et al. (2012) for an overview). In this scheme we first sample μ and Σ from their priors. Conditional on these samples, we then sample a hypergraph. By comparing summary statistics of the sampled hypergraph with the observed hypergraph, we determine whether or not our sampled hypergraph is similar enough to the observed hypergraph. If so, we accept the sampled μ and Σ. We choose the number of hyperedges of order $k = 2$, the number of hyperedges of order $k = 3$ and the number of triangles as our summary statistics. This initialisation scheme is detailed in Algorithm 5 and the full initialisation scheme is given in Algorithm 6.

Algorithm 6 Procedure for initialising MCMC scheme in Algorithm 3.

Input Observed hypergraph $h_{N,K}$
1) Determine $\boldsymbol{U}_0$ by applying Algorithm 4.
2) Let initial radii $\boldsymbol{r}_0$ be the smallest radii which induce all hyperedges observed in $h_{N,K}$, conditional on $\boldsymbol{U}_0$.
3) Sample $\boldsymbol{\psi}_0^{(0)}$ and $\boldsymbol{\psi}_0^{(1)}$ from their prior distributions.
4) Sample μ_0 and Σ_0 by applying Algorithm 5.

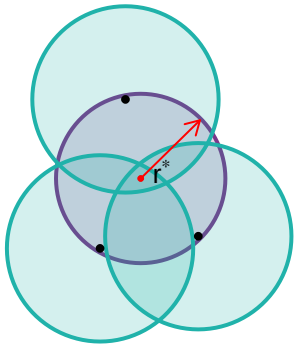


Figure 16: The blue shaded regions correspond to $B_r(u_i)$, for $i = 1, 2, 3$, and the purple shaded region is the smallest enclosing ball of the points. The statements $r^* < r$ and $B_r(u_1) \cap B_r(u_2) \cap B_r(u_3) \neq \emptyset$ are equivalent.

E Practicalities

To implement the MCMC scheme given in Algorithm 3 there are a number of practical considerations we must address. In this section we comment on these where, in E.1 we discuss an approach for determining the presence of a hyperedge of arbitrary order and in E.2 we discuss efficient evaluation of the likelihood.

E.1 Smallest Enclosing Ball

Here we discuss an approach for determining the presence of a hyperedge conditional on $\boldsymbol{U}$ and $\boldsymbol{r}$. Recall that a hyperedge $e_k = \{i_1, i_2, \ldots, i_k\}$ is present if $B_{r_k}(u_{i_1}) \cap B_{r_k}(u_{i_2}) \cap \cdots \cap B_{r_k}(u_{i_k}) \neq \emptyset$. Hence, in order to determine whether $y_{e_k} = 1$, we must find whether or not the sets corresponding to the nodes in the hyperedge have a non-empty intersection.

Alternatively, note that it is equivalent to determine whether the coordinates $\{u_{i_1}, u_{i_2}, \ldots, u_{i_k}\}$ are contained within a ball of radius r_k (see section 3.2 of Edelsbrunner and Harer (2010)). Figure 16 shows this for an example with $k = 3$. This means that we can rephrase the above into the following procedure.

1. Determine the smallest enclosing ball B for the coordinates $\{u_{i_1}, u_{i_2}, \ldots, u_{i_k}\}$.
2. If the radius of B is less than r_k, the hyperedge $e_k = \{i_1, i_2, \ldots, i_k\}$ is present in the hypergraph.

To compute the smallest enclosing ball we can rely on the the miniball algorithm (see Section 3.2 of Edelsbrunner and Harer (2010)), which may be also be referred to as the minidisk algorithm

(see Section 4.7 of Berg et al. (2008)). Before we will providing the algorithmic details, we will first discuss the intuition behind this algorithm. In the discussion below, we will follow the explanation of Edelsbrunner and Harer (2010).

For a set of points, it is clear that a given point is either contained within B or it lies on the boundary of B. The set of points on the boundary entirely determine B and, when the number of points is much larger than the dimension, the chance of a point lying on the boundary is small. Miniball exploits these facts to partition the set of points into those which are contained within B and those which lie on the boundary. The algorithm begins by sampling a point u from the full set of points $u_{i_1 \dots i_k} = \{u_{i_1}, u_{i_2}, \dots, u_{i_k}\}$. If u lies within the smallest enclosing ball of $u_{i_1 \dots i_k} \setminus u$, then we know it lies within B and so it does not influence B. Alternatively, if u lies on the boundary then we must include it in the set which determines B. Miniball iterates over all points in this way to determine the set of points on the boundary. Then, once we have the minimum closing ball B with radius r^*, we check whether $r^* < r_k$ to determine the presence of a hyperedge.

To determine the set of order k hyperedges present in the graph, we rely on the simplicial property of the Čech complex (see Section 2.3). By observing that all subsets of an order k hyperedge must also be connected, we reduce the search space from all possible hyperedges to those whose subsets are present.

We now present the algorithmic details of the miniball algorithm (see Section 3.2 of Edelsbrunner and Harer (2010)). For the edge $e_k = \{i_1, i_2, \dots, i_k\}$, we let $\sigma_1 \subseteq e_k$ and $\sigma_2 \subseteq e_k$ denote subsets which partition e_k so that $\sigma_1 \cap \sigma_2 = e_k$. After a pass of the algorithm, the set σ_2 will contain the index of nodes which lie on the boundary of the smallest enclosing ball B. Hence, σ_2 represents the nodes which determine B. We initialise the miniball algorithm with $\sigma_1 = e_k$ and σ_2, and iteratively identify which nodes from σ_1 belong in σ_2. The procedure is given in Algorithm 7.

Algorithm 7 Miniball

1) Set $\sigma_1 = e_k$ and $\sigma_2 = \emptyset$
2) **if** $\sigma_1 = \emptyset$, compute the miniball B of σ_2
   else choose $u \in \sigma_1$
   -Calculate the miniball B which contains the points $\sigma_1 \setminus u$ in its interior and the points σ_2 on its boundary
   -**if** $u \notin B$, then set B to be the miniball B which contains the points $\sigma_1 \setminus u$ in its interior and the points $\sigma_2 \cup u$ on its boundary

An alternative description of this algorithm can be found in Section 4.7 of Berg et al. (2008), and for efficient implementation of the Čech complex we rely on the GUDHI C++ library (The GUDHI Project (2015)).

E.2 Evaluating $\mathcal{L}(\boldsymbol{U}, \boldsymbol{r}, \boldsymbol{\psi}^{(1)}, \boldsymbol{\psi}^{(0)}; h_{N,K})$

The likelihood given in (12) requires the enumeration of hyperedge discrepancies between the observed hypergraph $h_{N,K}$ and the induced hypergraph $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$. In this section we note that this does not require a summation over all possible hyperedges, and so can be computed far more efficiently than (11) suggests. We first discuss evaluation of (12), and then observe that (9) can be evaluated in a similar way.

To evaluate the likelihood we have the hyperedges present in $h_{N,K}$ and the hyperedges present in $g_{N,K}(\boldsymbol{U}, \boldsymbol{r})$. In practice, as the data example from Section 8 suggests, the number of hyperedges in each of these hypergraphs is much smaller than the number of possible hyperedges

$\sum_{k=2}^{K}\binom{N}{k}$. Let $n_k^{(h)}$ and $n_k^{(g)}$ denote the number of order k hyperedges in $h_{N,K}$ and $g_{N,K}(\boldsymbol{U},\boldsymbol{r})$, respectively. To evaluate the likelihood, we begin by enumerating the number of order k hyperedges which are present in both hypergraphs to obtain $d_k^{(11)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K})$. This can easily be computed by evaluating the number of intersection between the hyperedges in $h_{N,K}$ and $g_{N,K}(\boldsymbol{U},\boldsymbol{r})$. Then, for $k=2,3,\ldots,K$, it follows that

$$d_k^{(10)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K}) = n_k^{(g)} - d_k^{(11)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K}), \tag{50}$$

$$d_k^{(01)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K}) = n_k^{(h)} - d_k^{(11)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K}), \tag{51}$$

$$\begin{aligned} d_k^{(00)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K}) = \binom{N}{k} - \Big[& d_k^{(11)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K}) \\ & + d_k^{(10)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K}) + d_k^{(01)}(g_{N,K}(\boldsymbol{U},\boldsymbol{r}),h_{N,K}) \Big]. \end{aligned} \tag{52}$$

Hence, we are able to avoid summation over all possible hyperedges. We can easily calculate the distance specified in (8) from the above, by observing that it is given by the sum of (50) and (51).

F Proofs for Propositions in Section 6

F.1 Proof of Proposition 6.1

We have $U_i \sim \mathcal{N}(\mu,\Sigma)$ for $i=1,2,\ldots,N$ and $\Sigma = \mathrm{diag}(\sigma_1^2,\sigma_2^2,\ldots,\sigma_d^2)$. Our goal is to obtain an expression for p_{e_2}, and we begin by noting

$$p_{e_2} = P\left(\|U_i - U_j\| \le 2r_2 | \mu,\Sigma\right) = P\left((U_i-U_j)^T(U_i-U_j) \le 4(r_2)^2 | \mu,\Sigma\right). \tag{53}$$

Hence, we consider the distribution of $X_{ij}^T X_{ij}$ where $X_{ij} = U_i - U_j$.

From properties of the Normal distribution, we have that $X_{ij} = U_i - U_j \sim \mathcal{N}(0, 2\Sigma)$. From Section 1 of Duchesne and Micheaux (2010), we find

$$X_{ij}^T X_{ij} | \Sigma = \sum_{l=1}^{d} \lambda_l \chi_1^2 \tag{54}$$

where λ_l is the l^{th} eigenvalue of 2Σ. Since Σ is diagonal, the eigenvalues of 2Σ are given by $\lambda_l = 2\sigma_l^2$. Furthermore, since a χ_1^2 distribution is equivalent to a $\Gamma(1/2,2)$ distribution, we have

$$Z_{ij}|\Sigma = X_{ij}^T X_{ij}|\Sigma \sim \sum_{l=1}^{d} \Gamma\left(\frac{1}{2}, 2(2\sigma_l^2)\right). \tag{55}$$

Hence, we have the result.

F.2 Proof of Proposition 6.2

$y_{e_k}^{(g)} = 1$ and $y_{e_k}^{(g^*)} = 1$ indicate that the hyperedge e_k is present in g and g^*, respectively. To begin, we consider the probability of e_k being present in g^*. We may observe $y_{e_k}^{(g^*)} = 1$ from either $y_{e_k}^{(g)} = 1$ or $y_{e_k}^{(g)} = 0$. In the first case, we want to keep the state of e_k unaltered and, in the second case, we want to modify the state of the edge. Hence, we have

$$P(y_{e_k} = 1^{(g^*)}|\Sigma) = (1-\varphi_k)P(y_{e_k}^{(g)} = 1|\Sigma) + \varphi_k(1 - P(y_{e_k}^{(g)} = 1|\Sigma)). \tag{56}$$

The degree of the i^{th} node with respect to order k hyperedges is obtained by summing over all possible hyperedges e_k that are incident to i. Hence, we have

$$\text{Deg}^{g}_{(i,k)} = \sum_{\{e_k \in \mathcal{E}_{N,k} | i \in e_k\}} y^{(g)}_{e_k} \tag{57}$$

and, in total, there are $\binom{N-1}{k-1}$ possible order k hyperedges that are incident to i. Therefore, it follows that

$$\mathbb{E}\left[\text{Deg}^{g}_{(i,k)} | \Sigma\right] = \binom{N-1}{k-1} P(y_{e_k} = 1^{(g)} | \Sigma). \tag{58}$$

Note that (58) is valid for dependent probabilities and we rely on this for hyperedges of order $k \geq 3$. As an example, consider the hyperedges $\{i, j, k\}$ and $\{i, j, l\}$ when $k = 3$. It is clear that both of these hyperedges depend on both i and j, and so there is dependence between the hyperedges and the summation in the calculation of $\text{Deg}^{g}_{(i,k)}$.

Now we consider $\mathbb{E}\left[\text{Deg}^{g^*}_{(i,k)}\right]$. By the law of total expectation, we have

$$\mathbb{E}\left[\text{Deg}^{g^*}_{(i,k)} | \varphi_k, \Sigma\right] = \mathbb{E}\left[\mathbb{E}\left[\text{Deg}^{g^*}_{(i,k)} | \text{Deg}^{g}_{(i,k)}\right] | \varphi_k, \Sigma\right] \tag{59}$$

From (56), it follows that

$$\mathbb{E}\left[\text{Deg}^{g^*}_{(i,k)} | \varphi_k, \Sigma\right] = \mathbb{E}\left[(1-\varphi_k)\text{Deg}^{g}_{(i,k)} + \varphi_k \left(\binom{N-1}{k-1} - \text{Deg}^{g}_{(i,k)}\right) | \varphi_k, \Sigma\right] \tag{60}$$

$$= (1-\varphi_k)\mathbb{E}\left[\text{Deg}^{g}_{(i,k)} | \varphi_k, \Sigma\right] + \varphi_k \left(\binom{N-1}{k-1} - \mathbb{E}\left[\text{Deg}^{g}_{(i,k)} | \varphi_k, \Sigma\right]\right). \tag{61}$$

Using (57), we obtain

$$\mathbb{E}\left[\text{Deg}^{g^*}_{(i,k)} | \varphi_k, \mu, \Sigma\right] = \binom{N-1}{k-1} \left[(1-\varphi_k) P(y_{e_k} = 1^{(g)} | \mu, \Sigma) + \varphi_k (1 - P(y^{(g)}_{e_k} = 1 | \mu, \Sigma))\right]. \tag{62}$$

The final result then follows from Observation (O2).

F.3 Proof of Proposition 6.3

The probability of an edge e_2 being present in g^* is given by

$$P\left(y^{(g^*)}_{e_2} = 1 | \Sigma, r_2, \varphi_2\right) = (1-\varphi_2)p_{e_2} + \varphi_2(1-p_{e_2}), \tag{63}$$

where p_{e_2} is the probability that e_2 is present in g.

The degree distribution of the i^{th} node is a sum of independent Bernoulli trials since the edges $\{i, j\}$ and $\{i, k\}$ occur independently given conditioning on i. There are $N-1$ possible order 2 edges which contain i. Hence it follows that

$$\text{Deg}^{g^*}_{(i,2)} | r_2, \varphi_2, \Sigma \sim \text{Binomial}\left(N-1, (1-\varphi_2)p_{e_2} + \varphi_2(1-p_{e_2})\right). \tag{64}$$

F.4 Proof of Proposition 6.4

The degree of the i^{th} node for order k hyperedges in g^* is given by

$$\text{Deg}^{g^*}_{(i,k)} = \sum_{\{e_k \in \mathcal{E}_{N,k} | i \in e_k\}} y^{g^*}_{e_k}. \tag{65}$$

In the proof of Proposition 6.3 (see Appendix F.3), we relied on the observation that hyperedges can occur from the nsRGH $g(\boldsymbol{U}, \boldsymbol{r})$ or the hyperedge modification. Since the modification is applied independently over all hyperedges, we now need to consider the dependence in the nsRGH hyperedges of order $k > 2$.

For $k = 3$, consider the hyperedges $\{i, j, k\}$ and $\{i, j, l\}$. Note that these hyperedges both depend on i and j, and so it is clear that $P(y^{(g)}_{ijk} = 1)$ and $P(y^{(g)}_{ijl} = 1)$ are not independent. Now the argument used in Appendix F.3 is no longer appropriate. In this case, we need to consider the sum of $\binom{N-1}{2}$ dependent Bernoulli trials. For p_{ijk} small, we can approximate this by a Poisson distribution with rate parameter $\lambda = \sum_{\{e_k \in \mathcal{E}_{N,3} | i \in e_k\}} y^{(g^*)}_{e_k}$ (see Teerapabolarn (2014)). Hence, we have the result.

We now comment in this approximation by making the following observation

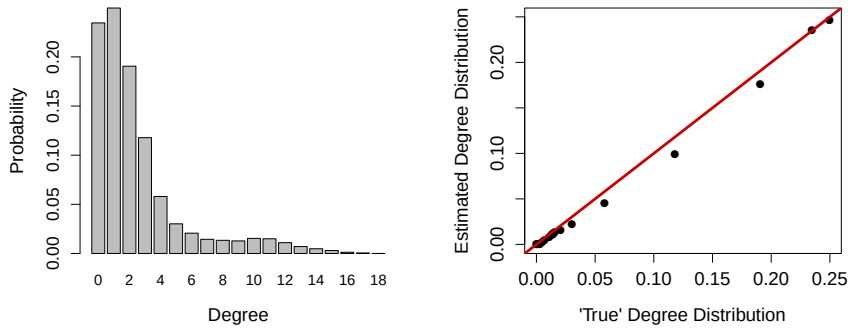
$$P(y^{(g)}_{ijk} = 1 \cap y^{(g)}_{ijl} = 1 | \mathcal{A}) = P(y^{(g)}_{ijk} = 1 | \mathcal{A}) P(y^{(g)}_{ijl} = 1 | \mathcal{A}) \tag{66}$$

where $\mathcal{A} = \{y^{(g)}_{ij}, y^{(g)}_{ik}, y^{(g)}_{jk}, y^{(g)}_{il}, y^{(g)}_{jl}\}$.

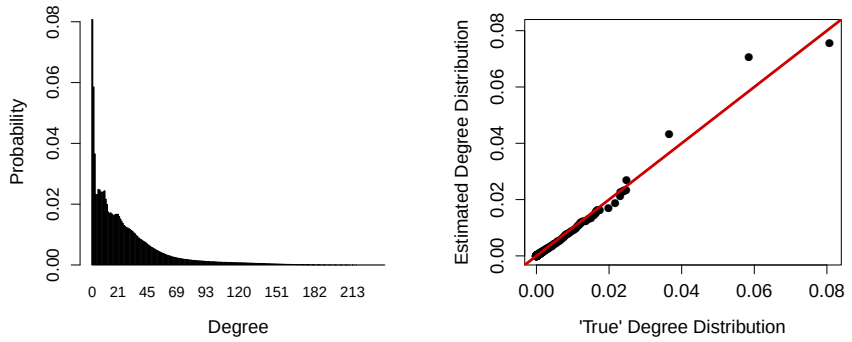
In the simplicial case, the presence of an order 3 hyperedge implies the presence of all order 2 subedges. Hence, in the simplicial, or near simplicial setting, the dependence will not be as significant and so a Binomial argument may be used.

G Prior and Posterior Predictive Degree Distributions

Here we present additional plots for the simulation study detailed in Section 7.2. Figure 17 compares the predictives for the connections between the observed nodes and the newly simulated nodes, and Figure 18 compares the predictives for connections occurring among the newly simulated nodes only. In both figures, we see a close correspondence between the prior and predictive distributions.

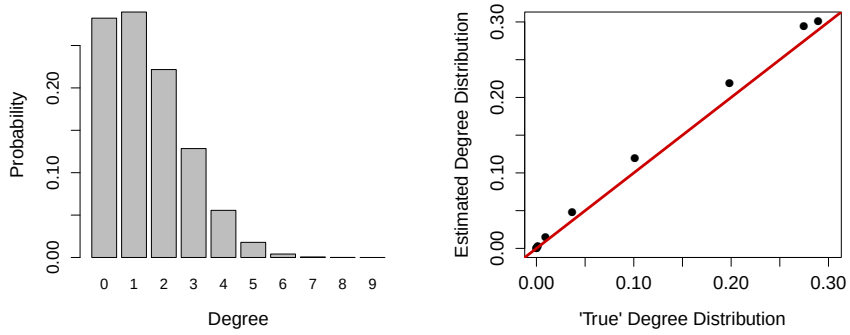


(a) Predictive degree distributions for hyperedges of order $k = 2$.

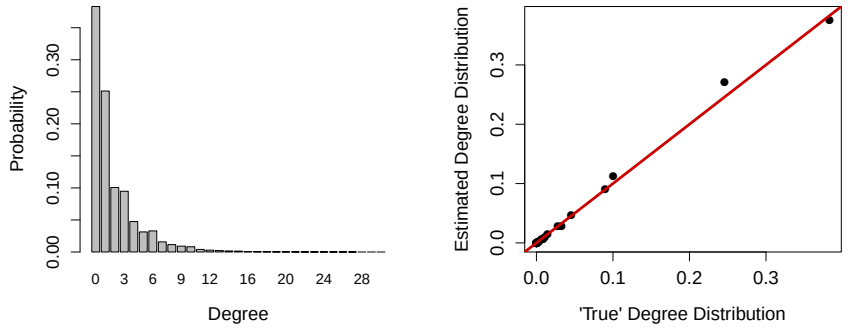


(b) Predictive degree distributions for hyperedges of order $k = 3$.

Figure 17: Comparison of prior and posterior predictive degree distributions for hyperedges occurring between the $N^* = 10$ newly simulated nodes and the nodes of the observed hypergraph. In each figure, the left panel shows the prior predictive degree distribution, and the right panel shows a qq-plot of the prior and posterior predictive degree distributions. Figures 17a and 17b show the degree distributions for hyperedges of order 2 and 3, respectively.



(a) Predictive degree distributions for hyperedges of order $k = 2$.



(b) Predictive degree distributions for hyperedges of order $k = 3$.

Figure 18: Comparison of prior and posterior predictive degree distributions for hyperedges occurring between the $N^* = 10$ newly simulated nodes only. In each figure, the left panel shows the prior predictive degree distribution, and the right panel shows a qq-plot of the prior and posterior predictive degree distributions. Figures 18a and 18b show the degree distributions for hyperedges of order 2 and 3, respectively.