

Inference of hyperedges and overlapping communities in hypergraphs

Received: 18 May 2022

Martina Contisciani¹✉, Federico Battiston² & Caterina De Bacco¹✉

Accepted: 2 November 2022

Published online: 24 November 2022

 Check for updates

Hypergraphs, encoding structured interactions among any number of system units, have recently proven a successful tool to describe many real-world biological and social networks. Here we propose a framework based on statistical inference to characterize the structural organization of hypergraphs. The method allows to infer missing hyperedges of any size in a principled way, and to jointly detect overlapping communities in presence of higher-order interactions. Furthermore, our model has an efficient numerical implementation, and it runs faster than dyadic algorithms on pairwise records projected from higher-order data. We apply our method to a variety of real-world systems, showing strong performance in hyperedge prediction tasks, detecting communities well aligned with the information carried by interactions, and robustness against addition of noisy hyperedges. Our approach illustrates the fundamental advantages of a hypergraph probabilistic model when modeling relational systems with higher-order interactions.

Over the past twenty years, networks have allowed to map and characterize the architecture of a wide variety of relational data, from social and technological systems to the human brain¹. Despite their success, traditional graph representation are unable to provide a faithful representation of the patterns of interactions occurring in the real-world². Collections of nodes and links—networks—can only properly encode dyadic relations. Yet, in the last few years systems as diverse as cellular networks³, structural and functional brain networks^{4,5}, social systems⁶, ecosystems⁷, social image search engines⁸, human face-to-face interactions⁹ and collaboration networks¹⁰, have shown that a large fraction of interactions occurs among three or more nodes at a time. These higher-order systems are hence best described by different mathematical frameworks such as hypergraphs¹¹, where hyperedges of arbitrary dimensions may encode structured relations among any number of system units^{12–14}. Interestingly, providing a higher-order description of the system interactions has been shown to lead to the emergence of new collective phenomena¹⁵ in diffusive^{16,17}, synchronization^{18–22}, spreading^{23–25}, and evolutionary²⁶ processes.

To properly describe the higher-order organization of real-world networks, a variety of growing^{27,28} and equilibrium models, such as generalized configuration models^{29–31} have been proposed. Tools from

topological data analysis have allowed to obtain insights into the higher-order organization of real-world networks^{32,33}, and methods to infer higher-order interactions from pairwise records have been suggested³⁴. Finally, several powerful network metrics and ideas have been extended beyond the pair, from higher-order clustering³⁵, spectral methods³⁶ and centrality^{37,38} to motifs³⁹ and network backboning⁴⁰.

Despite a few recent contributions^{41–47}, how to define and identify the mesoscale organization of real-world hypergraphs is still a largely unexplored topic. Here, we propose a new principled method to extract higher-order communities based on statistical inference. More broadly, our approach is that of generative models, which incorporate a priori community structure by means of latent variables, inferred directly from the observed interactions^{48–50}. Beyond its efficient numerical implementation, our model has several desirable features. It detects overlapping communities, an aspect that is missing in current approaches of community detection in hypergraphs and that is arguably better representative of scenarios where nodes are expected to belong to multiple groups. It also provides a natural measure to perform link prediction tasks, as it outputs the probability that a given hyperedge exists between any subset of nodes. Similarly, it allows to generate synthetic hypergraphs with given community structure, an ingredient that can be given in input or learned from data. Moreover,

¹Max Planck Institute for Intelligent Systems, Cyber Valley, 72076 Tübingen, Germany. ²Department of Network and Data Science, Central European University, 1100 Vienna, Austria. ✉e-mail: martina.contisciani@tuebingen.mpg.de; caterina.debacco@tuebingen.mpg.de

our explicit higher-order approach is not only more grounded theoretically, but also more efficient than applying graph algorithms to higher-order data projected into pairwise records.

We apply our method to a variety of real-world systems, showing that it recovers communities more robustly against noisy addition of large hyperedges than methods on projected pairwise data, it achieves high performance in predicting missing hyperedges, and it allows to determine the influence of hyperedge size in such prediction tasks. We also illustrate how our higher-order approach detects communities that are more aligned with the information carried by hyperedges than what is recorded by node attributes. Through these examples, we illustrate how a principled higher-order probabilistic approach can shed light on the role that higher-order interactions play in real-world complex systems.

Results

The Hypergraph-MT model

Here, we introduce Hypergraph-MT, a probabilistic generative model for hypergraphs with mixed-membership community structure. Based on a statistical inference framework, our model provides a principled, efficient and scalable approach to extract overlapping communities in networked systems characterized by the presence of interactions beyond the pair.

At its core, our approach assumes that nodes belong to different groups in different amounts, as specified by a set of membership vectors. These memberships then determine the probability that any subset of nodes is connected with a hyperedge. We denote a hypergraph with N nodes $\mathcal{V} = \{i_1, \dots, i_N\}$ and E hyperedges $\mathcal{E} = \{e_1, \dots, e_E\}$ as $\mathcal{H}(\mathcal{V}, \mathcal{E})$. Mathematically, this can be represented as an adjacency tensor A with entries A_{i_1, \dots, i_d} equal to the weight of a d -dimensional interaction between the nodes i_1, \dots, i_d . For instance, for contact interactions, A_{i_1, \dots, i_d} could be the number of times that nodes i_1, \dots, i_d were in close contact together.

Given these definitions, we can specify the likelihood of observing the hypergraph given a set of latent variables θ , which include the membership vectors. This relies on modeling $P(A_{i_1, \dots, i_d} | \theta)$, the probability of observing a hyperedge given θ . We model this probability as:

$$P(A_{i_1, \dots, i_d} | \theta) = \text{Pois}(A_{i_1, \dots, i_d}; \lambda_{i_1, \dots, i_d}), \tag{1}$$

where $\lambda_{i_1, \dots, i_d} = \sum_{k_1, \dots, k_d} u_{i_1 k_1} \dots u_{i_d k_d} w_{k_1, \dots, k_d}$. The set of latent variables is defined by $\theta = (\mathbf{u}, \mathbf{w})$, where \mathbf{u} is a $N \times K$ -dimensional community membership matrix and \mathbf{w} is an affinity tensor, which captures the idea that an interaction is more likely to exist between nodes of compatible communities. If only pairwise interactions exist, the affinity matrix has dimension $K \times K$. Therefore, the problem reduces to the traditional network case and can be efficiently solved⁴⁹. When higher-order interactions are present, the dimension of the affinity tensor \mathbf{w} can become arbitrarily large depending on the size d_e of a hyperedge e , i.e., the number of nodes present in it. In fact, \mathbf{w} has as many entries as all the possible d_e -way interactions between all K groups. For instance, in a hypergraph with only 2-way and 3-way interactions, we have $\mathbf{w} = [\mathbf{w}^{(2)}, \mathbf{w}^{(3)}]$ with $\mathbf{w}^{(2)}$ of dimension $K \times K$ and $\mathbf{w}^{(3)}$ of dimension $K \times K \times K$.

The question is thus how to reduce the dimension of \mathbf{w} . A relevant choice that overcomes these problems is that of assortativity⁵¹, implying that a hyperedge is more likely to exist when all nodes in it belong to the same group. This captures well situations where homophily, the tendency of nodes with similar features to be connected to each other, plays a role, as observed in social or biological networks^{49,52}. Mathematically, the only non-zero elements of \mathbf{w} are the “diagonal” ones, that is:

$$w_{k_1, \dots, k_d} = \delta_{k_1, \dots, k_d} w_{k_1, \dots, k_d}. \tag{2}$$

With this, we obtain a matrix \mathbf{w} of dimension $D \times K$, where $D = \max_{e \in \mathcal{E}} d_e$ is the maximum hyperedge size in the dataset. In principle, one could envisage other ways to restrict \mathbf{w} to control its dimension. However, we found that the choice in Eq. (2) provides a natural interpretation, results in good prediction performance on both real and synthetic datasets, and is computationally scalable. A similar problem of dimensionality reduction has been tackled in ref. 45, which investigated the more constrained case of hard-membership models.

Putting all together, we model the likelihood of the hypergraph as:

$$P(\mathbf{A} | \theta) = \prod_{e \in \Omega} e^{-\lambda_e} \frac{\lambda_e^{A_e}}{A_e!}, \tag{3}$$

$$\text{with } \lambda_e = \sum_k w_{d,k} \prod_{i \in e} u_{ik}, \tag{4}$$

where $\Omega = \{e | e \subseteq \mathcal{V}, d_e \geq 2\}$ is the set of all potential hyperedges. In practice, we can reduce this space by considering only the possible hyperedges of a certain size lower or equal than the maximum observed size D . In Eq. (3) we assumed conditional independence between hyperedges given the latent variables, a standard assumption in these types of models. Such a condition could in principle be relaxed following the approaches of refs. 53–55, we do not explore this here.

Having defined Eq. (3), the goal is to infer the latent variables \mathbf{u} and \mathbf{w} given the observed hypergraph \mathbf{A} . To infer the values of $\theta = (\mathbf{u}, \mathbf{w})$, we consider both maximum likelihood estimation (assuming uniform priors on the parameters) and maximum a posteriori estimation (assuming non-uniform priors). The derivations are similar and rely on an efficient expectation-maximization (EM) algorithm⁵⁶ that exploits the sparsity of the dataset, as detailed in the Methods section and in the Supplementary Note 1.

We obtain the following algorithmic updates for the membership vectors:

$$u_{ik} = \frac{\sum_{e \in \mathcal{E}} B_{ie} \rho_{ek}}{\sum_{e \in \Omega | i \in e} w_{d_e, k} \prod_{j \in e, j \neq i} u_{jk}}, \tag{5}$$

where B_{ie} is equal to the weight of the hyperedge e to which the node i belongs (it is an entry of the hypergraph incidence matrix) and ρ is a variational distribution determined in the expectation step of the EM procedure. The numerator of Eq. (5) can be computed efficiently, as we only need the non-zero entries of the incidence matrix, which is typically sparse. Instead, computing the denominator can be prohibitive depending on the value of D , the maximum hyperedge size. This is due to the summation over all possible hyperedges in Ω , which requires extracting all possible combinations $\binom{N}{d}$, for $d = 2, \dots, D$.

This problem is not present in the case of graphs, as this summation would be over N^2 terms at most. This issue clearly highlights the importance of algorithmic efficiency in handling hypergraph data, an aspect that cannot be overlooked to make a model work in practice.

We propose a solution to this problem that reduces the computational complexity to $O(NDK)$ and makes our algorithm efficient, scalable and applicable in practice. The key is to rewrite the summation over Ω such that we have an initial value that can be updated at cost $O(1)$ after each update $u_{ik}^{(t)} \rightarrow u_{ik}^{(t+1)}$, which can be done in parallel over $k = 1, \dots, K$. This formulation is explained in details in the Supplementary Note 1, where we also show how to edit the updates in Eq. (5) by imposing sparsity (with a proper prior distribution) or by constraining the membership vectors to be probability vectors such that $\sum_k u_{ik} = 1$. In both cases, we get a constant term added in the denominators of the updates.

Table 1 | Summary of higher-order datasets

	N	E	E_G	M	M_G	$\langle k \rangle$	$s(k)$	$\langle d \rangle$	$s(d)$	D	% $d=2$	% $d > 2 \in G$	K
High school	327	7818	5818	172,035	189,928	55.6	27.1	2.3	0.5	5	70.3%	88.5%	9
Primary school	242	12,704	8317	106,879	127,886	127.0	55.2	2.4	0.6	5	61.0%	87.5%	11
Workplace	92	788	755	9645	9831	17.7	8.6	2.1	0.3	4	94.2%	88.2%	5
Hospital	75	1825	1139	27,835	32,788	59.1	49.0	2.4	0.6	5	60.7%	95.1%	4
Gene-Disease	4642	2738	55,795	4131	114,444	1.7	3.6	5.8	5.2	25	32.4%	0.6%	25
Justice	38	2826	264	15,040	190,790	366.7	203.6	4.9	1.7	9	7.6%	81.8%	2
House bills	1494	41,362	360,086	47,212	2,451,751	245.8	251.6	8.9	6.6	24	18.5%	2.1%	2
Senate bills	293	19,872	22,157	27,300	732,561	482.0	396.9	7.1	5.4	24	16.5%	14.8%	2
House committees	289	106	2535	111	4312	0.7	2.0	8.6	3.6	18	0.9%	0.0%	2
Senate committees	282	275	12,761	289	41,008	16.2	12.6	16.6	6.0	25	0.0%	0.0%	2
Walmart	1025	3553	8029	5112	13,769	9.8	16.7	2.8	1.2	11	51.0%	7.0%	10
Trivago	6687	33,963	69,875	40,280	115,533	13.9	13.8	2.7	1.3	26	59.6%	16.1%	36

Shown are the number of nodes (N), number of hyperedges in the hypergraph (E) and in the graph (E_G), number of weighted hyperedges in the hypergraph (M) and in the graph (M_G), mean node degree ($\langle k \rangle$), SD of node degree ($s(k)$), mean hyperedge size ($\langle d \rangle$), SD of hyperedge size ($s(d)$), maximum hyperedge size (D), percentage of pairwise interactions (% $d=2$), percentage of pairwise interactions in the 2-combination set of hyperedges of size bigger than 2 that are already in the graph (% $d > 2 \in G$), and number of communities (K).

Finally, the updates of the affinity matrix are given by:

$$w_{dk} = \frac{\sum_{e \in \mathcal{E} | d_e = d} A_e \rho_{ek}}{\sum_{e \in \Omega | d_e = d} \prod_{j \in e} u_{jk}}. \quad (6)$$

These are also computationally efficient to implement and can be updated in parallel. Further details are in the Methods section and in the Supplementary Note 1, where we also provide a pseudocode for the whole inference routine. Additionally, in the Supplementary Note 2 we show the validation of our model on synthetic data with ground-truth community structure and the comparison against the generative method of ref. 45 and the spectral method of ref. 47. Hypergraph-MT shows a strong and increasing performance in recovering communities as the ground-truth community structure becomes stronger, similarly to the method of ref. 47. However, this method is designed to capture hard-membership communities and benefits from having an inference routine similar to the generative process of the synthetic data. In particular, Hypergraph-MT significantly outperforms the competing methods Graph-MT, Pairs-MT, and that of ref. 45 as soon as the ground-truth community structure becomes less noisy. Remarkably, this is observed in synthetic datasets that are generated with a different generating process than that of Hypergraph-MT. As a consequence, the positive performance of our method confirms the robustness and the reliability of the methodology here introduced.

Results on empirical data

We analyze hypergraphs derived from empirical data from various domains. For each one, we report a diverse range of structural properties such as number of nodes, hyperedges and their sizes, as detailed in Table 1. Moreover, the datasets provide node metadata, which we use to fix the number of communities K , aiming to compare the resulting communities with this additional information. For further details on the datasets, see the Methods section. For each hypergraph, we run Hypergraph-MT ten times with different random initialization and select the result with the highest likelihood. For comparison, we run the model on two baselines structures obtained from the same empirical data: a graph obtained from clique expansions of each hyperedge (Graph-MT), where a hyperedge of size d is decomposed in $\frac{d(d-1)}{2}$ unordered pairwise interactions; a graph obtained using only hyperedges with $d_e = 2$ (Pairs-MT). Notice that running our model on graphs reduces to MULTITENSOR—the model presented in ref. 49—with an assortative affinity matrix. As a remark, we use interchangeably the terms *graph* or *network* to refer to the data with only pairwise interactions, and the term *hypergraph* for the higher-order data.

The advantage of using hypergraphs. The goal of using the two baselines is to assess the advantage (if any) in treating a dataset with higher-order interactions as a hypergraph. Indeed, in practice higher-order data are often reduced to their projected graph, an operation, which not only generates a potentially misleading loss of information, but which is also computationally expensive⁴¹. Hence, before evaluating the performance of Hypergraph-MT on various datasets, we turn to the following fundamental question: given a dataset of high-order interactions, does a hypergraph representation bring any advantage compared to a simpler graph representation? If the answer is positive, then we should analyze the data with an algorithm that handles hypergraphs. If not, a simpler network algorithm should be enough.

To this end, we analyze four datasets describing human close-proximity contact interactions obtained from wearable sensor data at a high school (High school), a primary school (Primary school), a workplace (Workplace) and a hospital (Hospital). For the analysis, we run the model on the three different structures (hypergraph, clique expansions, and pairwise edges) described above. For each dataset, we compare the inferred partitions with the node metadata that describe either the classes, the departments, or the roles the nodes belong to. We measure closeness to the metadata with the F1-score, a measure for hard-membership classification. It ranges between 0 and 1, where 1 indicates perfect matching between inferred and given partitions. Table 2 shows the performance with the different structures, and both hypergraphs and graphs perform similarly. Notice that the average size of hyperedges in these datasets is around 2.2; thus interactions are mainly pairwise to start with. Moreover, interactions with $d_e > 2$ include people who already interact pairwise (see column % $d > 2 \in G$ in Table 1). Hence, a clique expansion of these is not expected to provide much distinct information from that already present in the pairwise subset of the dataset. Overall, these results suggest that hypergraphs do not bring any additional advantage for these types of datasets, and running a network algorithm would be enough.

To understand how this assessment may change, we present a toy example built from the High school dataset. We select the subset of nodes belonging to two classes (2BIO1 and MP*2 in our example), and we manipulate it by artificially adding a large hyperedge. It simulates an event where ten external people (guests) and a random subset of ten existing nodes are participating. This is represented by the gray hyperedge of dimension 20 in Fig. 1 (left). Here, the green nodes are the external guests, while the blue and orange nodes are the randomly-selected students from the two classes, respectively. While we only add one hyperedge, its size significantly differs from that of all the other existing hyperedges. In

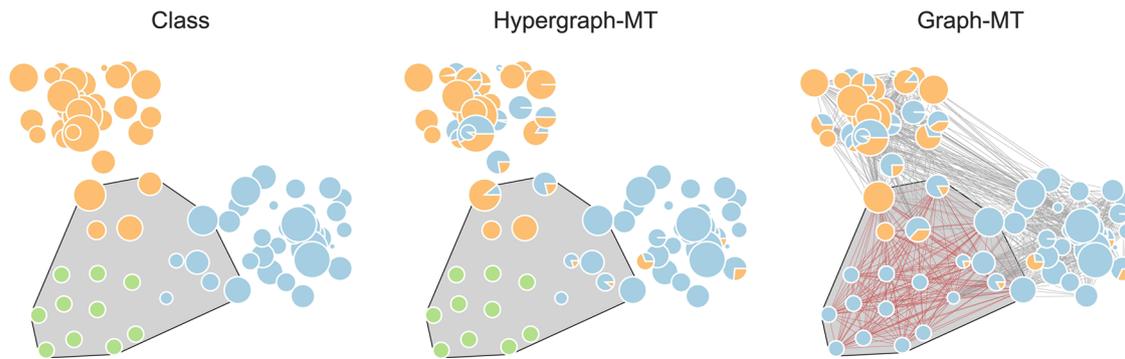


Fig. 1 | The advantage of hypergraph representation: an illustrative example. The left plot shows a subset of the High school dataset, with nodes belonging to the classes 2BIO1 (light blue) and MP*2 (orange), and ten external guests (green). Node size is proportional to the degree. The gray hyperedge simulates an event, and we omit the other hyperedges for visualization clarity. The central plot displays the partition extracted by Hypergraph-MT and on the right we find the partition

extracted by Graph-MT. In the latter, the gray edges denote the interactions in the graph (obtained by clique expansions) before the event, and the red edges are the interactions added because of the simulated event. This example shows the advantage of using hypergraphs as this representation is more resilient to the addition of a noisy hyperedge and is more robust in detecting communities.

particular, a clique expansion resulting from this additional hyperedge brings in $\binom{20}{2}$ new edges of size 2 (red in the figure). Hence, we expect this additional information to impact the structure of Graph-MT much more than the hypergraph. Figure 1 shows that Hypergraph-MT is not biased by the presence of this individual large hyperedge, and it well recovers the external guests by assigning zero memberships to them for both classes. Conversely, Graph-MT assigns the guests to the blue class. With this toy example, we show a possible scenario where hypergraphs have an advantage, as this representation is more resilient to the addition of a noisy hyperedge and is more robust in detecting communities.

Hyperedge prediction: analysis of a Gene-Disease dataset. We now turn our attention to the analysis of a higher-order Gene-Disease dataset, where nodes are genes, and a hyperedge connects genes that are associated with a disease. Here, we focus on the ability of our model to predict missing hyperedges. We measure prediction performance using a cross-validation protocol where hyperedges are divided into train and test sets. The train set is used for parameter estimation, while performance is evaluated on the test set. We compute the area under the receiver-operator curve (AUC), and use the probability assigned by our model of a hyperedge to exist as input scores for this metric. For Graph-MT, the probability of a hyperedge to exist is computed as the product of the probabilities that each single edge exists. For details, see the Methods section. When evaluating Pairs-MT, we measure the AUC on the subset of test hyperedges of size 2. To perform a balanced comparison in this case, we also measure the AUC for both Hypergraph-MT and Graph-MT on this set (pairs), while still training on the whole train set. This provides information on the utility of large hyperedges to predict pairwise interactions.

Table 2 | Comparison of community detection algorithms in human close-proximity contact interactions datasets

	Hypergraph-MT	Graph-MT	Pairs-MT
High school	0.757	0.776	0.755
Primary school	0.907	0.916	0.928
Workplace	0.829	0.820	0.830
Hospital	0.580	0.491	0.554

For each dataset, we show the F1-score obtained by comparing a node metadata against the inferred partitions from the hypergraphs (Hypergraph-MT), the graphs obtained by clique expansions (Graph-MT), and the graphs given only by the registered pairwise interactions (Pairs-MT).

We vary the maximum hyperedge size D to show how each method responds to the incorporation of progressively larger edges in terms of prediction tasks. Interestingly, we observe a strong shift in performance around $D=15,16$, where Hypergraph-MT significantly outperforms Graph-MT and Pairs-MT (see Fig. 2). This highlights that hyperedges with larger size carry useful information that cannot be fully captured via clique expansions. This is true regardless of the type of missing edges being predicted (hyperedges or pairs-only). In addition, predictive performance is improved homogeneously across hyperedge sizes in the held-out set. Namely, we are not improving just in predicting the pairs-only, as shown by Hypergraph-MT (pairs), but also those of bigger sizes, see Supplementary Fig. 3. This is where Graph-MT fails because the additional information introduced by the clique expansions produces a much denser graph than the input data that may not be correlated with the true existing hyperedges, thus blurring the observations given in the input. These results not only highlight the ability of our model to predict missing data, but also how the knowledge of large hyperedges helps the prediction of hyperedges of smaller sizes.

Overlapping communities and interpretability: analysis of a Justice dataset. Together with hyperedge prediction, Hypergraph-MT allows to extract relevant information also on the mesoscale organization of real-world hypergraphs. As a case study, we analyze a dataset recording all the votes expressed by the Justices of the Supreme Court in the U.S. from 1946 to 2019 case by case. Justices are nodes, and hyperedges connect Justices that expressed the same vote in a given case. The structure of this hypergraph is different from the others analyzed above: it has fewer nodes ($N=38$) but it is denser ($E=2826$), on average a Justice votes 367 times. Similarly, the graph obtained with clique expansion has substantially fewer edges ($E_G=264$) but with higher weights than the hypergraph. See Table 1 for details. Examining the communities inferred in these two markedly distinct structures can provide direct insights into the particular aspects captured by a hypergraph formulation. To this end, we compare the inferred partitions with the political parties of the Justices, i.e., Democrat or Republican, information provided as node metadata. We use the cosine similarity (CS), a metric that measures the distance between vectors, and thus it is better suited to capture mixed-membership communities. The CS varies between 0 and 1, where 1 means that the inferred partition matches perfectly the one shown by political affiliation. For each node, we compute the CS between its political party and the partitions inferred by Hypergraph-MT and Graph-MT. Figure 3a shows the point-by-point comparison between the resulted cosine similarities of the two methods. Here, each marker is a Justice

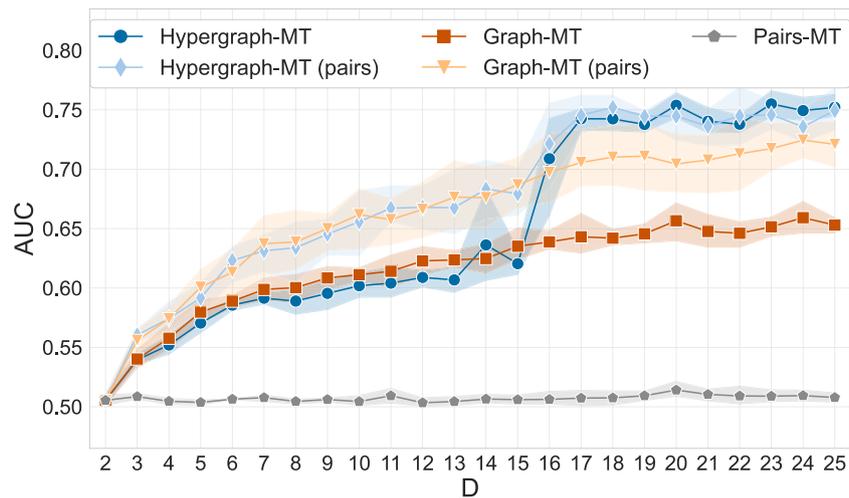


Fig. 2 | Critical size for hyperedge prediction in a Gene-Disease dataset. We measure the AUC by varying the maximum hyperedge size D . The results are averages and standard deviations over 5-fold cross-validation test sets, and the baseline for AUC is the random value 0.5. We run the model on the hypergraphs (Hypergraph-MT), the graphs obtained by clique expansions (Graph-MT), and the graphs given only by the registered pairwise interactions (Pairs-MT). To perform a

balanced comparison against Pairs-MT, for Hypergraph-MT and Graph-MT we additionally measure the AUC on the subset of test hyperedges of size 2 (pairs), while still training on the whole train set. The plot shows the existence of a critical hyperedge size beyond which the higher-order algorithm significantly outperforms alternative methods.

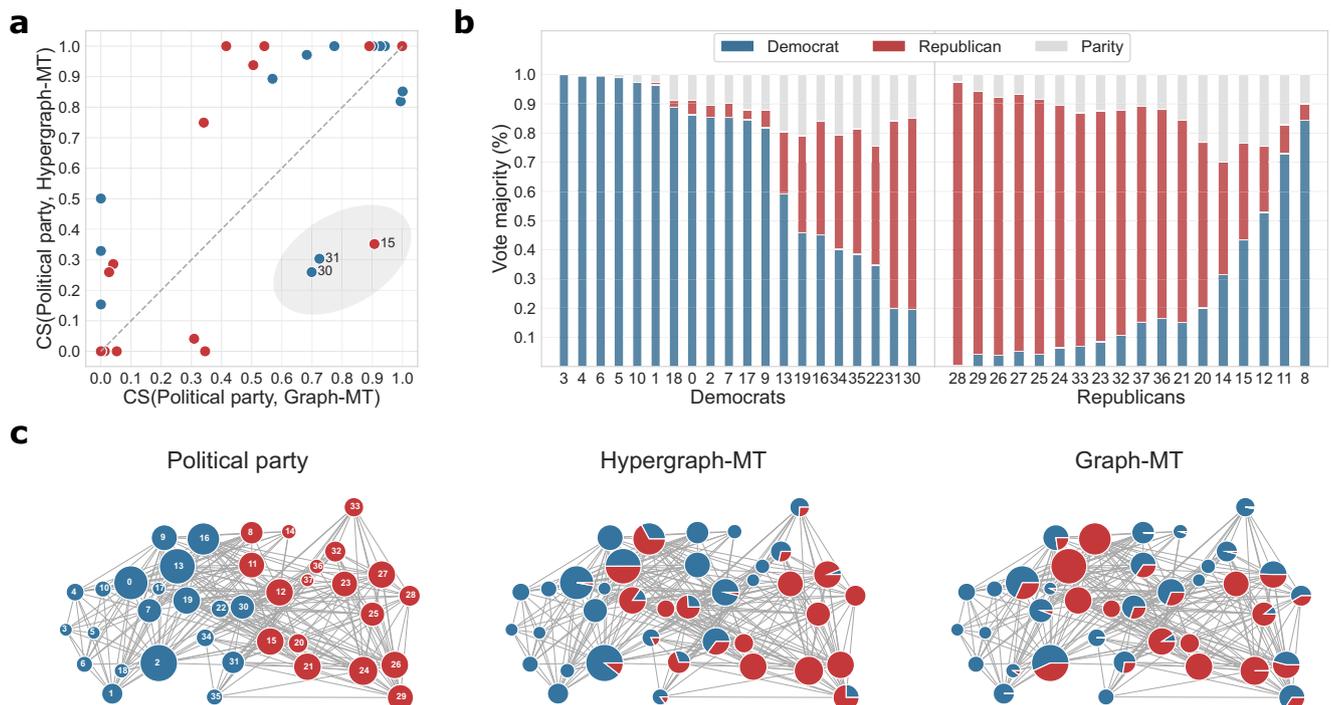


Fig. 3 | Inference of overlapping communities in a co-voting higher-order dataset of the U.S. Justices. **a** Point-by-point comparison between the cosine similarities (CS) obtained by Hypergraph-MT and Graph-MT. For each Justice (marker in the plot), we compute the CS between the partitions inferred by the methods and the political party of the Justices, i.e., Democrat (blue) and Republican (red). **b** Vote majority proportion of the hyperedges of each Justice. Every hyperedge is colored based on the majority political party of the Justices involved in it,

i.e., either Democratic, Republican, or equally distributed (gray). Then, for every Justice, we extract the percentage of times that they participate in hyperedges of a given majority. **c** Data partition according to the political party (left), and the mixed-membership communities inferred by Hypergraph-MT (center) and Graph-MT (right). Node size is proportional to the degree, node labels are Justice IDs, and the interactions are the edges of the projected graph.

and colors represent their political parties. Points above (below) the diagonal represent Justices for which the communities inferred by Hypergraph-MT (Graph-MT) align better with the political party. In several cases the two models infer memberships that align similarly with political affiliation: upper-right corner, where both models are

aligned well, and lower-left corner, where they are both not aligned well. The interesting behavior is shown in the bottom-right area highlighted in gray, containing three Justices whose political affiliations are more closely associated with the communities inferred by Graph-MT than those of Hypergraph-MT. To investigate these cases,

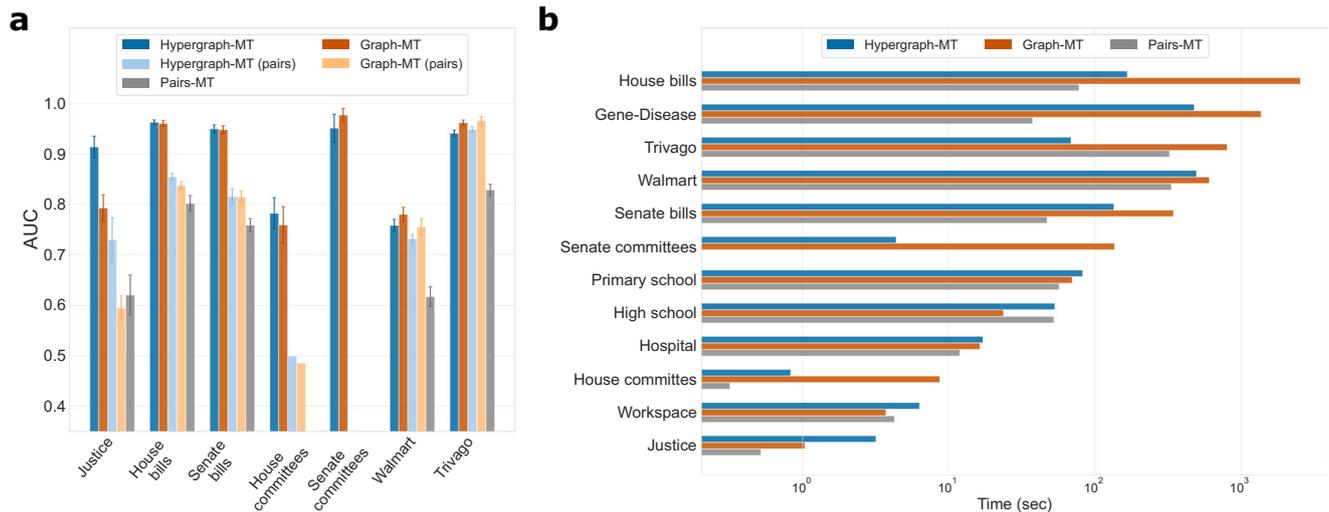


Fig. 4 | Hyperedge prediction performance and computational complexity in higher-order datasets. **a** The performance of hyperedge prediction is measured with the AUC, whose baseline is the random value 0.5. The results are averages and standard deviations over 5-fold cross-validation test sets. For each dataset, we run the model on the hypergraphs (Hypergraph-MT), the graphs obtained by clique expansions (Graph-MT), and the graphs given only by the registered pairwise

interactions (Pairs-MT). To perform a balanced comparison against Pairs-MT, for Hypergraph-MT and Graph-MT we additionally measure the AUC on the subset of test hyperedges of degree 2 (pairs), while still training on the whole train set. **b** Computational complexity of Hypergraph-MT, Graph-MT, and Pairs-MT for the different higher-order datasets. We show the running time for one realization.

we inspect the information carried by the hyperedges. Specifically, for each hyperedge we measure the majority political party based on the affiliation of the Justices involved in it. For instance, a hyperedge of size 5 made of 4 democrats and 1 republican has a Democratic majority. We also account for ties, when equal numbers of Justices are in both parties. Then, for each Justice, we extract the percentage of times that they participate in hyperedges of a given majority. This measure indicates the tendency of Justices to vote more often aligned with democrats or republicans, an information summarized in Fig. 3b. We observe Justices that consistently vote with their own party majority (e.g., Justice 3 votes mainly with other democrats, Justice 28 mainly with other republicans), but also cases in which the political party of the Justice is not aligned with the voting behavior expressed by their hyperedges. For example, node 30 (Justice Ruth Bader Ginsburg) is associated with the Democratic Party, but most of her votes align with those of republican Justices. This behavior is captured by Hypergraph-MT, which assigns her a membership more peaked in the community made of republicans and only partially to the one of democrats, as shown in Fig. 3c. Instead, Graph-MT assigns her mostly to the community of democrats. This mismatch between hypergraph information and political affiliation explains the lower value of cosine similarity in Fig. 3a. Similar conclusions can be drawn for node 31 and 15. More generally, the overlapping memberships inferred by Hypergraph-MT match more closely the voting behavior of Justices than those inferred by Graph-MT, as shown in the pie markers in Fig. 3c.

In addition to community structure, Hypergraph-MT outperforms Graph-MT also in the hyperedge prediction task. Figure 4a shows how Hypergraph-MT achieves higher AUC than Graph-MT, in both predicting pairwise and higher-order interactions. This further corroborates the hypothesis that information is lost when decoupling higher-order interactions via clique expansion. This example illustrates why it is critical to consider hypergraphs when hyperedges contain information that can be lost by clique expansion. It also shows the advantage of considering overlapping communities when nodes' behaviors are nuanced and no clear affiliation to one group is expected. As Supreme Court cases span a wide range of topics, we may expect Justices to exhibit a diversity of preferences (and thus voting

behaviors) that cannot be fully captured by a binary political affiliation. Hence, models that consider overlapping communities can provide a variety of patterns that better represents this diversity. Finally, this example also confirms that metadata should be carefully used as “ground-truth” communities, thus encouraging a careful exploration of the relationship between node metadata, information contained in the hyperedges and community structure⁵⁷.

The computational efficiency of Hypergraph-MT. Beyond accuracy, algorithmic efficiency is necessary for a widespread applicability of statistical inference models to large-scale datasets. Hence, we now assess the performance of our model on a variety of systems from different domains, focusing on the analysis of the computational efficiency of Hypergraph-MT as compared to alternative approaches. The higher-order datasets include co-sponsorship and committee memberships data of the U.S. Congress, co-purchasing behavior of customers on Walmart, and clicking activity of users on Trivago (Table 1). Hypergraph-MT and Graph-MT perform similarly in terms of predicting missing hyperedges on most of these datasets, as shown in Fig. 4a. This suggests that in such cases, the information learned from the clique expansion is similar to that contained in a hypergraph representation. While one may be tempted to conclude that using a dyadic method should be favored in these cases, we argue that predictive performance may not be the only metric to use to make this decision. Indeed, time complexity also plays a role here, as many of these datasets have large hyperedges. While we have extensively discussed the efficiency of Hypergraph-MT, one should also consider the cost of running dyadic methods on clique expansion of large data. In fact, this depends on the number of pairs generated in the expansion, a quantity related to both the amount and size of hyperedges. As a result, the size of a graph obtained by clique expansion can become arbitrarily large. For instance, the House bills data results in almost 4×10^5 edges, as opposed to the 4×10^4 hyperedges given by the hypergraph representation. This difference of an order of magnitude has a significant impact in terms of computational complexity. In fact, we observe a difference of an order of magnitude also in the running time of the algorithms, as shown in Fig. 4b, where we plot the time to run the three methods on

each dataset. While for datasets with small hyperedges (e.g., the close-proximity data discussed above) running time is similar for Hypergraph-MT and Graph-MT, we observe significant differences for datasets with larger maximum size D , with Hypergraph-MT being much faster to run. Hypergraph-MT may therefore be the algorithm of choice for large system sizes. See Supplementary Note 2 for further results about the computational complexity of the methods on synthetic data with variable size.

Discussion

Here, we have introduced Hypergraph-MT, a mixed-membership probabilistic generative model for hypergraphs, which proposes a first way to extract the overlapping community organization of nodes in networked systems with higher-order interactions. In addition to detecting communities, our model provides a principled tool to predict missing hyperedges, thus serving as a quantitative evaluation framework for assessing goodness of fit. This feature is particularly useful in the absence of metadata when evaluating community detection schemes. In practice, our model considers an assortative affinity matrix, which makes its algorithmic implementation highly scalable. The computational complexity is also significantly reduced by an efficient routine to compute expensive quantities at low cost in each update, a problem not present in the case of graphs. We have applied our model to a wide variety of social and biological hypergraphs, discussing accuracy in the hyperedge and community structure inference tasks. Moreover, we have showed that Hypergraph-MT outperforms clique expansion methods with respect to running time, making it a suitable solution also for higher-order datasets with large hyperedges.

Our method has a substantial advantage in systems where hyperedges contain important information that can be lost by considering non-higher-order methods on projected dyadic graphs. For instance, it allows quantifying how maximum hyperedge size impacts performance and unveils the presence of critical sizes beyond which higher-order algorithms may significantly outperform dyadic methods, as shown in a Gene-Disease dataset. Hypergraph-MT also has the benefits of being more resilient to the addition of large noisy hyperedges and of being more robust in detecting communities that are more closely aligned with the information carried by hyperedges, as shown in the analysis of the U.S. Justices.

There are natural methodological extensions to further expand the range of applications covered by our model. Here, we have considered an assortative affinity matrix, but alternative formulations could be considered to target different types of structures. The challenge would be to increase flexibility while keeping the dimensionality of the problem under control. Moreover, our model takes in input hyperedges of one type, but there could be multiple types of ways to connect a subset of nodes. Expanding our approach to these cases would be analogous to extend single-layer networks to multi-layer ones. This may be done by suitably defining different types of affinity matrices for each type of high-order interaction, as in ref. 49. Similarly, our model might be extended to extract temporal higher-order communities in the presence of time-varying interactions with memory^{58,59}. Finally, hypergraphs may carry additional information beyond the one contained in hyperedges. This calls for further developments to rigorously incorporate information such as node attributes into the model formulation^{60,61}. While here we have focused on analyzing real-world data, our generative model can also be used to sample synthetic data with hypergraph structure. In particular, our model could prove useful for practitioners interested in utilizing synthetic benchmarks of hypergraphs, allowing a better characterization of higher-order topological properties, including simplicial closure³⁵ and higher-order motifs³⁹. Taken together, Hypergraph-MT provides a fast and scalable tool for inferring the structure of large-scale hypergraphs, contributing to a better

understanding of the networked organization of real-world higher-order systems.

Methods

Inference of Hypergraph-MT

Hypergraph-MT models the likelihood of the hypergraph $\mathbf{A} = \{A_e\}_{e \in \mathcal{E}}$ as:

$$P(\mathbf{A}|\theta) = \prod_{e \in \Omega} e^{-\lambda_e} \frac{\lambda_e^{A_e}}{A_e!}, \tag{7}$$

where $\lambda_e = \sum_k w_{d_e,k} \prod_{i \in e} u_{ik}$. The set of latent variables is defined by $\theta = (\mathbf{u}, \mathbf{w})$, where \mathbf{u} is a $N \times K$ -dimensional community membership matrix and \mathbf{w} is a $D \times K$ -dimensional affinity matrix, where $D = \max_{e \in \mathcal{E}} d_e$ is the maximum hyperedge size in the dataset. Each entry w_{dk} represents the density of hyperedges of size d in the community k . Notice, we only consider the assortative regime, to reduce the dimensionality of the affinity tensor \mathbf{w} . The product runs over $\Omega = \{e|e \subseteq \mathcal{V}, d_e \geq 2\}$, that is, the set of all potential hyperedges. In practice, we can reduce this space by considering only the possible hyperedges of a certain size lower or equal than the maximum observed size D . For instance, if the maximum size of interactions in a hypergraph is $D = 4$, then we should not expect to see hyperedges of size 5, and we can define $\Omega = \{e|e \subseteq \mathcal{V}, 2 \leq d_e \leq D\}$.

With this formulation, Hypergraph-MT is a mixed-membership probabilistic generative model for hypergraphs. The main intuition behind it is that a hyperedge is more likely to exist between nodes with the same community membership. In fact, hyperedges in which even a single value $u_{ik} = 0$ appears, are assigned a null probability. The goal is thus to infer the latent variables \mathbf{u} and \mathbf{w} given the observed hypergraph \mathbf{A} .

We infer the parameters using a maximum likelihood approach. Specifically, we maximize the log-likelihood

$$L = - \sum_{e \in \Omega} \sum_k w_{d_e,k} \prod_{i \in e} u_{ik} + \sum_{e \in \mathcal{E}} A_e \log \sum_k w_{d_e,k} \prod_{i \in e} u_{ik} \tag{8}$$

with respect to $\theta = (\mathbf{u}, \mathbf{w})$, where we neglect the factorial term, which is independent of the parameters. As the summation in the logarithm renders the calculations difficult, we employ a variational approximation using Jensen's inequality, that gives

$$\mathcal{L}(\boldsymbol{\rho}, \theta) = - \sum_{e \in \Omega} \sum_k w_{d_e,k} \prod_{i \in e} u_{ik} + \sum_{e \in \mathcal{E}} A_e \sum_k \rho_{ek} \log \left(\frac{w_{d_e,k} \prod_{i \in e} u_{ik}}{\rho_{ek}} \right). \tag{9}$$

For each $e \in \mathcal{E}$, we consider a variational distribution ρ_{ek} over the communities k : this is our estimate of the probability that the hyperedge e exists due to the contribution of the community k . The equality holds when

$$\rho_{ek} = \frac{w_{d_e,k} \prod_{i \in e} u_{ik}}{\sum_k w_{d_e,k} \prod_{i \in e} u_{ik}}. \tag{10}$$

Maximize Eq. (8), is then equivalent to maximize Eq. (9) with respect to both θ and $\boldsymbol{\rho}$. We estimate the parameters by using an expectation-maximization (EM) algorithm, where at each step one updates $\boldsymbol{\rho}$ using Eq. (10) (E-step) and then maximizes $\mathcal{L}(\boldsymbol{\rho}, \theta)$ regarding $\theta = (\mathbf{u}, \mathbf{w})$ by setting partial derivatives to zero (M-step). This procedure is repeated until the log-likelihood converges. The fixed point is a local maximum, but it is not guaranteed to be the global maximum. Therefore, we perform ten runs of the algorithm with different random initialization for θ , taking the fixed point with the largest value of the log-likelihood. For further details, see Supplementary Note 1.

Hyperedge prediction and cross-validation

We assess the performance of our model by measuring the goodness in predicting missing hyperedges. In these experiments, we use a 5-fold cross-validation routine: we divide the dataset into five equal-size groups (folds), selected uniformly at random, and give the models access to four groups (training data) to learn the parameters; this contains 80% of the hyperedges. One then predicts the hyperedges in the held-out group (test set). By varying which group we use as the test set, we get five trials per realization. When we use the baseline Pairs-MT, the training and the test sets are the subsets extracted from the initial ones, containing only the hyperedges with $d_e = 2$. Instead, when we use the baseline Graph-MT, we train the model on the graph obtained from clique expansions of the hyperedges in the training set.

As a performance metric, we measure the area under the receiver-operator characteristic curve (AUC) on the test data, and the final results are averages over the five folds. The AUC is the probability that a random true positive is ranked above a random true negative; thus the AUC is 1 for perfect prediction, and 0.5 for chance. Since the set of all possible hyperedges is large, it is not possible to compute the AUC on the whole training and test sets; hence we proceed with samples. In detail, we fix the number of comparisons we want to evaluate, here 10^3 . We then sample 10^3 values from the non-zero entries (where exist a hyperedge) of the sets, and we save the inferred hyperedge probabilities in a vector R_1 . We sample the same number of values from the zero entries (where do not exist a hyperedge), keeping this set balanced with R_1 in terms of hyperedge size distribution. We save the inferred hyperedge probabilities of this set of entries in a vector R_0 . We then make element-wise comparisons and compute the AUC as

$$\text{AUC} = \frac{\sum(R_1 > R_0) + 0.5 \sum(R_1 = R_0)}{|R_1|}, \quad (11)$$

where $\sum(R_1 > R_0)$ stands for the number of times R_1 has a higher value than R_0 in the element-wise comparisons; and $|R_1| = |R_0|$ is the length of the vector, which is equal to the number of comparisons we fix.

To predict the existence of a hyperedge, we use different approaches according to the structure under analysis. For Hypergraph-MT, the probability of a hyperedge is given by Eq. (1). For Graph-MT, instead, we compute the probability of a hyperedge as the product of the probabilities of each edge of its clique expansion to exist. That is, $P(A_e) = \prod_{(ij) \in e_2} P(A_{ij} > 0)$, where e_2 is the 2-combination set of the hyperedge e . Notice, all the single pairwise interactions have to exist, to have a probability of the hyperedge greater than zero. When evaluating Pairs-MT, we measure the AUC only on the subset of the test set containing edges, i.e., hyperedges with $d_e = 2$. To perform a balanced comparison in this case, we also measure the AUC for both Hypergraph-MT and Graph-MT on this set (pairs), while still training on the whole train set. This provides information on the utility of large hyperedges to predict pairwise interactions.

Description of the datasets

In the main text, we analyze hypergraphs derived from empirical data from various domains, and we provide a summary of study datasets in Table 1. To perform the inference in these datasets, we need to choose the number of communities K . In general, K can be selected using model selection criteria. For instance, one could evaluate the model's predictive performance—for example in the link prediction task—for varying numbers of communities, and then choose the best performing K . Here, for simplicity, we fix the number of communities K equal to the number of classes of a node metadata, aiming to compare the resulting communities with this additional information.

We first analyze four datasets collected by the SocioPatterns collaboration (<http://www.sociopatterns.org>), which describe human

close-proximity contact interactions obtained from wearable sensor data. The High-school dataset describes the interactions between students of nine different classrooms⁶². In the Primary school, nodes are students and teachers and a hyperedge connects groups of people that were all jointly in proximity to one another^{63,64}. Also here, the number of communities reflects the classrooms to which each student belongs, and it includes an additional class for the teachers. The Workplace dataset contains the contacts of individuals of five different departments, measured in an office building in France⁶⁵. Lastly, the Hospital hypergraph collects the interactions between patients, patients and health-care workers (HCWs) and among HCWs in a hospital ward in France⁶⁶. The number of communities corresponds then to the number of roles in the ward.

We then analyze the Gene-Disease dataset, that describes the gene-disease associations provided by expert curated resources (e.g., UNIPROT, CTI)⁶⁷. Nodes correspond to genes, and each hyperedge is the set of genes associated with a disease. We keep only the genes with a non-nan value of the Disease Pleiotropy Index (DPI), a quantity that considers if the diseases associated with the gene are similar among them and belong to the same disease class or belong to different disease classes. We use this attribute to fix the number of communities because it may indicate the different behaviors of the genes in the datasets. Moreover, we keep hyperedges with size $2 \leq d_e \leq 25$.

The second case study in the main text presents the analysis of the Justice hypergraph constructed from the data in <http://scdb.wustl.edu/about.php>. This dataset records all the votes expressed by the justices of the Supreme Court in the U.S. from 1946 to 2019 case by case. Nodes correspond to justices, and each hyperedge is the set of justices that expressed the same vote in a case. The number of communities corresponds to the number of political parties, i.e., Democrat and Republican.

The following datasets have been downloaded from <https://www.cs.cornell.edu/~arb/data/>. We analyze hypergraphs created from U.S. congressional bill co-sponsorship data, where nodes correspond to congresspersons and hyperedges correspond to the sponsor and all cosponsors of a bill in either the House of Representatives (House bills) or the Senate (Senate bills)^{45,68,69}. We also use two datasets from the U.S. Congress in the form of committee memberships^{45,70}. Each hyperedge is a committee in a meeting of Congress, and each node again corresponds to a member of the House (House committees) or a senator (Senate committees). A node is contained in a hyperedge if the corresponding legislator was a member of the committee during the specified meeting of Congress. In all these congressional datasets, the node labels give the political parties of the members, thus all of them have $K = 2$. For these datasets, we run the model with different values of $D = 2, \dots, 25$ and choose the best value among them.

In addition to the congressional datasets, we analyze the Walmart hypergraph⁷¹. Here, each node is a product, and a hyperedge connects a set of products that were co-purchased by a customer in a single shopping trip. We fix the number of communities equal to the product category labels. Lastly, we analyze the Trivago dataset⁴⁵. Nodes correspond to hotels listed at trivago.com, and each hyperedge corresponds to a set of hotels whose website was clicked on by a user of Trivago within a browsing session. For each hotel, the node label gives the country in which it is located, and we fix K based on this information. For Walmart and Trivago, we consider a subset of the hypergraph to reduce the sparsity, as done in ref. 45. The c -core of a hypergraph \mathcal{H} is defined as the largest subhypergraph \mathcal{H}_c such that all nodes in \mathcal{H}_c have size at least c . For Walmart, we use the 3-core hypergraph, and for Trivago, we work with the 5-core hypergraph.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

The datasets used in the paper are publicly available from their sources listed in the Methods section.

Code availability

An open-source algorithmic implementation of the model is publicly available and can be found at <https://github.com/mcontisc/Hypergraph-MT>.

References

- Boccaletti, S., Latora, V., Moreno, Y., Chavez, M. & Hwang, D.-U. Complex networks: Structure and dynamics. *Phys. Rep.* **424**, 175–308 (2006).
- Lambiotte, R., Rosvall, M. & Scholtes, I. From networks to optimal higher-order models of complex systems. *Nat. Phys.* **15**, 313–320 (2019).
- Klamt, S., Haus, U.-U. & Theis, F. Hypergraphs and cellular networks. *PLOS Computat. Biol.* **5**, e1000385 (2009).
- Petri, G. et al. Homological scaffolds of brain functional networks. *J. R. Soc. Interface* **11**, 20140873 (2014).
- Giusti, C., Ghrist, R. & Bassett, D. S. Two's company, three (or more) is a simplex. *J. Comput. Neurosci.* **41**, 1–14 (2016).
- Benson, A. R., Gleich, D. F. & Leskovec, J. Higher-order organization of complex networks. *Science* **353**, 163–166 (2016).
- Grilli, J., Barabás, G., Michalska-Smith, M. J. & Allesina, S. Higher-order interactions stabilize dynamics in competitive network models. *Nature* **548**, 210–213 (2017).
- Gao, Y. et al. Visual-textual joint relevance learning for tag-based social image search. *IEEE Trans. Image Process.* **22**, 363–376 (2012).
- Cencetti, G., Battiston, F., Lepri, B. & Karsai, M. Temporal properties of higher-order interactions in social networks. *Sci. Rep.* **11**, 1–10 (2021).
- Patania, A., Petri, G. & Vaccarino, F. The shape of collaborations. *EPJ Data Sci.* **6**, 1–16 (2017).
- Berge, C. *Graphs and Hypergraphs* (North-Holland Pub. Co., 1973).
- Battiston, F. et al. Networks beyond pairwise interactions: structure and dynamics. *Phys. Rep.* **874**, 1–92 (2020).
- Torres, L., Blevins, A. S., Bassett, D. & Eliassi-Rad, T. The why, how, and when of representations for complex systems. *SIAM Rev.* **63**, 435–485 (2021).
- Battiston, F. & Petri, G. *Higher-Order Systems* (Springer, 2022).
- Battiston, F. et al. The physics of higher-order interactions in complex systems. *Nat. Phys.* **17**, 1093–1098 (2021).
- Schaub, M. T., Benson, A. R., Horn, P., Lippner, G. & Jadbabaie, A. Random walks on simplicial complexes and the normalized hodge 1-laplacian. *SIAM Rev.* **62**, 353–391 (2020).
- Carletti, T., Battiston, F., Cencetti, G. & Fanelli, D. Random walks on hypergraphs. *Phys. Rev. E* **101**, 022308 (2020).
- Bick, C., Ashwin, P. & Rodrigues, A. Chaos in generically coupled phase oscillator networks with nonpairwise interactions. *Chaos* **26**, 094814 (2016).
- Skardal, P. S. & Arenas, A. Higher order interactions in complex networks of phase oscillators promote abrupt synchronization switching. *Commun. Phys.* **3**, 1–6 (2020).
- Millán, A. P., Torres, J. J. & Bianconi, G. Explosive higher-order kuramoto dynamics on simplicial complexes. *Phys. Rev. Lett.* **124**, 218301 (2020).
- Lucas, M., Cencetti, G. & Battiston, F. Multiorder laplacian for synchronization in higher-order networks. *Phys. Rev. Res.* **2**, 033410 (2020).
- Gambuzza, L. V. et al. Stability of synchronization in simplicial complexes. *Nat. Commun.* **12**, 1–13 (2021).
- Iacopini, I., Petri, G., Barrat, A. & Latora, V. Simplicial models of social contagion. *Nat. Commun.* **10**, 1–9 (2019).
- Chowdhary, S., Kumar, A., Cencetti, G., Iacopini, I. & Battiston, F. Simplicial contagion in temporal higher-order networks. *J. Phys.* **2**, 035019 (2021).
- Neuhäuser, L., Mellor, A. & Lambiotte, R. Multibody interactions and nonlinear consensus dynamics on networked systems. *Phys. Rev. E* **101**, 032310 (2020).
- Alvarez-Rodriguez, U. et al. Evolutionary dynamics of higher-order interactions in social networks. *Nat. Human Behav.* **5**, 586–595 (2021).
- Kovalenko, K. et al. Growing scale-free simplices. *Commun. Phys.* **4**, 1–9 (2021).
- Millán, A. P., Ghorbanchian, R., Defenu, N., Battiston, F. & Bianconi, G. Local topological moves determine global diffusion properties of hyperbolic higher-order networks. *Phys. Rev. E* **104**, 054302 (2021).
- Courtney, O. T. & Bianconi, G. Generalized network structures: The configuration model and the canonical ensemble of simplicial complexes. *Phys. Rev. E* **93**, 062311 (2016).
- Young, J.-G., Petri, G., Vaccarino, F. & Patania, A. Construction of and efficient sampling from the simplicial configuration model. *Phys. Rev. E* **96**, 032312 (2017).
- Chodrow, P. S. Configuration models of random hypergraphs. *J. Complex Netw.* **8**, cnaa018 (2020).
- Patania, A., Vaccarino, F. & Petri, G. Topological analysis of data. *EPJ Data Sci.* **6**, 1–6 (2017).
- Sizemore, A. E., Phillips-Cremins, J. E., Ghrist, R. & Bassett, D. S. The importance of the whole: topological data analysis for the network neuroscientist. *Netw. Neurosci.* **3**, 656–673 (2019).
- Young, J.-G., Petri, G. & Peixoto, T. P. Hypergraph reconstruction from network data. *Commun. Phys.* **4**, 1–11 (2021).
- Benson, A. R., Abebe, R., Schaub, M. T., Jadbabaie, A. & Kleinberg, J. Simplicial closure and higher-order link prediction. *Proc. Natl Acad. Sci. USA* **115**, E11221–E11230 (2018).
- Krishnagopal, S. & Bianconi, G. Spectral detection of simplicial communities via hodge laplacians. *Phys. Rev. E* **104**, 064303 (2021).
- Benson, A. R. Three hypergraph eigenvector centralities. *SIAM J. Math. Data Sci.* **1**, 293–312 (2019).
- Tudisco, F. & Higham, D. J. Node and edge nonlinear eigenvector centrality for hypergraphs. *Commun. Phys.* **4**, 1–10 (2021).
- Lotito, Q. F., Musciotto, F., Montresor, A. & Battiston, F. Higher-order motif analysis in hypergraphs. *Commun. Phys.* **5**, 79 (2022).
- Musciotto, F., Battiston, F. & Mantegna, R. N. Detecting informative higher-order interactions in statistically validated hypergraphs. *Commun. Phys.* **4**, 1–9 (2021).
- Wolf, M. M., Klinvex, A. M. & Dunlavy, D. M. 2016 *IEEE High Performance Extreme Computing Conference (HPEC)* 1–7 (IEEE, 2016).
- Vazquez, A. Finding hypergraph communities: a bayesian approach and variational solution. *J. Stat. Mech.* **2009**, P07006 (2009).
- Carletti, T., Fanelli, D. & Lambiotte, R. Random walks and community detection in hypergraphs. *J. Phys.* **2**, 015011 (2021).
- Eriksson, A., Edler, D., Rojas, A., de Domenico, M. & Rosvall, M. How choosing random-walk model and network representation matters for flow-based community detection in hypergraphs. *Commun. Phys.* **4**, 1–12 (2021).
- Chodrow, P. S., Veldt, N. & Benson, A. R. Generative hypergraph clustering: From blockmodels to modularity. *Sci. Adv.* **7**, eabh1303 (2021).
- Chodrow, P., Eikmeier, N. & Haddock, J. Nonbacktracking spectral clustering of nonuniform hypergraphs. Preprint at <https://arxiv.org/abs/2204.13586> (2022).
- Zhou, D., Huang, J. & Schölkopf, B. Learning with hypergraphs: Clustering, classification, and embedding. *Adv. Neural Inf. Process. Syst.* **19**, 1601–1608 (2006).
- Ball, B., Karrer, B. & Newman, M. E. Efficient and principled method for detecting communities in networks. *Phys. Rev. E* **84**, 036103 (2011).

49. De Bacco, C., Power, E. A., Larremore, D. B. & Moore, C. Community detection, link prediction, and layer interdependence in multilayer networks. *Phys. Rev. E* **95**, 042317 (2017).
50. Goldenberg, A., Zheng, A. X., Fienberg, S. E. & Airoldi, E. M. A survey of statistical network models. *Found. Trends Mach. Learn.* **2**, 129–233 (2010).
51. Fortunato, S. & Hric, D. Community detection in networks: a user guide. *Phys. Rep.* **659**, 1–44 (2016).
52. Asikainen, A., Iñiguez, G., Ureña-Carrión, J., Kaski, K. & Kivelä, M. Cumulative effects of triadic closure and homophily in social networks. *Sci. Adv.* **6**, eaax7310 (2020).
53. Safdari, H., Contisciani, M. & De Bacco, C. Generative model for reciprocity and community detection in networks. *Phys. Rev. Res.* **3**, 023209 (2021).
54. Contisciani, M., Safdari, H. & De Bacco, C. Community detection and reciprocity in networks by jointly modelling pairs of edges. *J. Complex Netw.* **10**, cnac034 (2022).
55. Safdari, H., Contisciani, M. & De Bacco, C. Reciprocity, community detection, and link prediction in dynamic networks. *J. Phys.* **3**, 015010 (2022).
56. Dempster, A. P., Laird, N. M. & Rubin, D. B. Maximum likelihood from incomplete data via the em algorithm. *J. R. Stat. Soc. Ser B (Methodological)* **39**, 1–22 (1977).
57. Peel, L., Larremore, D. B. & Clauset, A. The ground truth about metadata and community detection in networks. *Sci. Adv.* **3**, e1602548 (2017).
58. Scholtes, I. et al. Causality-driven slow-down and speed-up of diffusion in non-markovian temporal networks. *Nat. Commun.* **5**, 1–9 (2014).
59. Rosvall, M., Esquivel, A. V., Lancichinetti, A., West, J. D. & Lambiotte, R. Memory in network flows and its effects on spreading dynamics and community detection. *Nat. Commun.* **5**, 1–13 (2014).
60. Contisciani, M., Power, E. A. & De Bacco, C. Community detection with node attributes in multilayer networks. *Sci. Rep.* **10**, 1–16 (2020).
61. Newman, M. E. & Clauset, A. Structure and inference in annotated networks. *Nat. Commun.* **7**, 1–11 (2016).
62. Mastrandrea, R., Fournet, J. & Barrat, A. Contact patterns in a high school: a comparison between data collected using wearable sensors, contact diaries and friendship surveys. *PLoS ONE* **10**, e0136497 (2015).
63. Gemmetto, V., Barrat, A. & Cattuto, C. Mitigation of infectious disease at school: targeted class closure vs school closure. *BMC Infect. Dis.* **14**, 1–10 (2014).
64. Stehlé, J. et al. High-resolution measurements of face-to-face contact patterns in a primary school. *PLoS ONE* **6**, e23176 (2011).
65. Génois, M. et al. Data on face-to-face contacts in an office building suggest a low-cost vaccination strategy based on community linkers. *Netw. Sci.* **3**, 326–347 (2015).
66. Vanhems, P. et al. Estimating potential infection transmission routes in hospital wards using wearable proximity sensors. *PLoS ONE* **8**, e73970 (2013).
67. Piñero, J. et al. The disgenet knowledge platform for disease genomics: 2019 update. *Nucl. Acids Res.* **48**, D845–D855 (2020).
68. Fowler, J. H. Connecting the congress: A study of cosponsorship networks. *Polit. Anal.* **14**, 456–487 (2006).
69. Fowler, J. H. Legislative cosponsorship networks in the us house and senate. *Soc. Netw.* **28**, 454–465 (2006).
70. Stewart, C. III & Woon, J. Congressional Committee assignments, 103rd to 114th Congresses, 1993–2017: House, *Technical Report*, MIT mimeo (2008).
71. Amburg, I., Veldt, N. & Benson, A. *Clustering in Graphs and Hypergraphs with Categorical Edge Labels*. 706–717 (Association for Computing Machinery, 2020).

Acknowledgements

The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting M.C.; C.D.B. and M.C. were supported by the Cyber Valley Research Fund. F.B. acknowledges support from the Air Force Office of Scientific Research under award number FA8655-22-1-7025. The authors thank Philip S. Chodrow and Nate Veldt for useful discussions.

Author contributions

M.C. and C.D.B. developed the algorithm and performed the experiments. M.C., F.B., and C.D.B. all conceived the research, analyzed the results and wrote the manuscript.

Funding

Open Access funding enabled and organized by Projekt DEAL.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41467-022-34714-7>.

Correspondence and requests for materials should be addressed to Martina Contisciani or Caterina De Bacco.

Peer review information *Nature Communications* thanks Filippo Radicchi, and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Reprints and permissions information is available at <http://www.nature.com/reprints>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2022