Coding Hypergraphs : Theory and Tools for Complex Hypernetwork Analysis

Alessia Antelmi and Andrea Failla
alessia.antelmi@unito.it, andrea.failla@phd.unipi.it

The 16th International Conference on Advances in Social Networks Analysis and Mining
ASONAM 2024
04/09/2024

UNIVERSITÀ
DI TORINO

ICSC
Centro Nazionale di Ricerca in HPC,
Big Data and Quantum Computing

Istituto di Scienza e Tecnologie
dell'Informazione "A. Faedo"
Consiglio Nazionale delle Ricerche

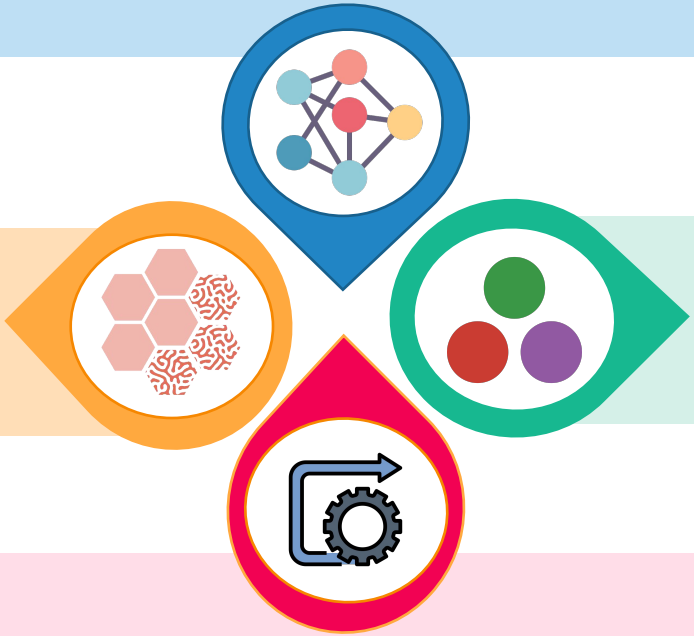
Presenters

Alessia Antelmi
Assistant Professor
University of Turin

Andrea Failla
PhD Student
University of Pisa

Outline

Hypergraph basics and tools



ASH

SimpleHypergraphs.jl

Conclusion & Future directions

Outline

Hypergraph basics and tools

Hypergraph basics & tools

What are hypergraphs?

Let's start with an example

(Enrica, Amedeo, Martina)

$Movie_1$

(Amedeo, Martina)

$Movie_2$

(Martina, Michel, Simone)

$Movie_3$

(Mary)

$Movie_4$

(Alex)

-

Let's start with an example

(Enrica, Amedeo, Martina)

Movie$_1$

(Amedeo, Martina)

Movie$_2$

(Martina, Michel, Simone)

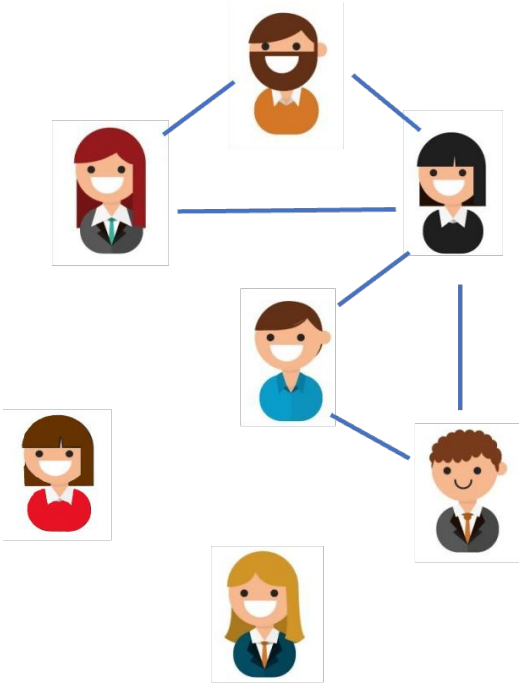
Movie$_3$

(Mary)

Movie$_4$

(Alex)

-



Hypergraphs

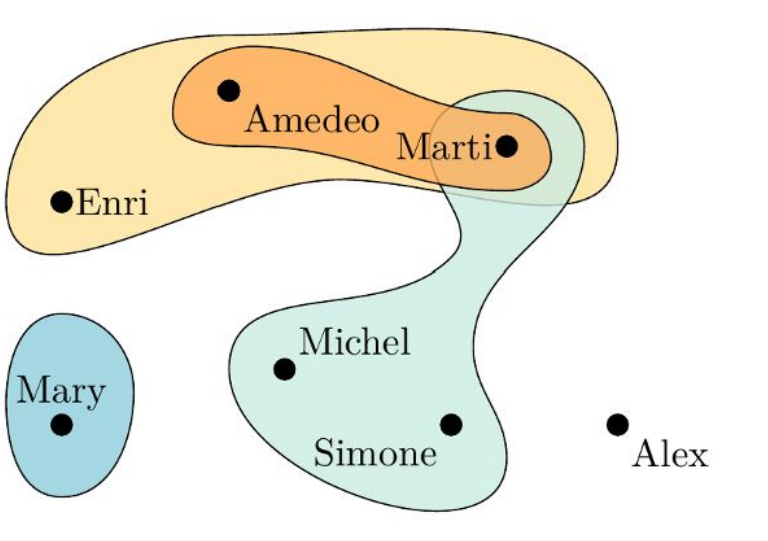
Generalization of graphs where a **hyperedge** can **connect more** than two vertices.

Hypergraphs: a formal definition

V = {Enri, Michel, Simo, Amedeo, Marti, Mary, Alex}

E = {Movie1, Movie2, Movie3, Movie4}, where

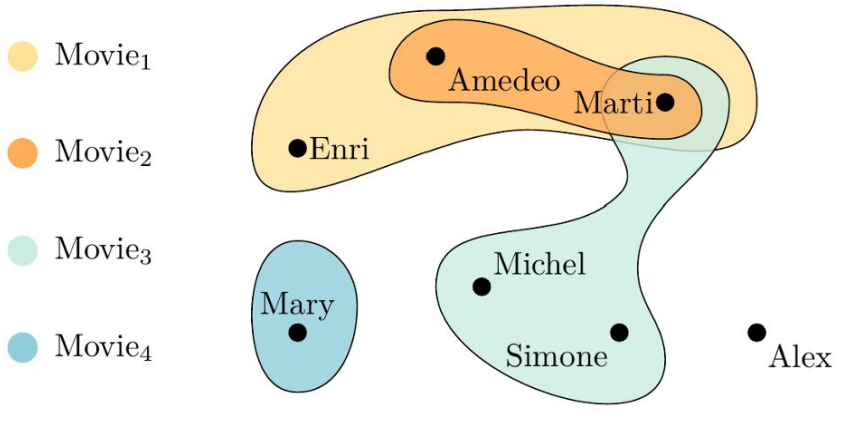
- Movie1 = {Enri, Amedeo, Marti}
- Movie2 = {Amedeo, Marti}
- Movie3 = {Marti, Michel, Simone}
- Movie4 = {Mary}



Hypergraphs: a formal definition

Definition 1.1 : Hypergraphs

A hypergraph H, denoted with $H = (\mathcal{V}, E = (e_i)_{i \in \mathcal{I}})$, on a finite set $\mathcal{V}$ and a finite set of indexes $\mathcal{I}$ is a family $(e_i)_{i \in \mathcal{I}}$ of subsets of $\mathcal{V}$ called hyperedges.



A Bretto. 2013. *Hypergraph Theory: An Introduction.* Springer International Publishing.

When can hypergraphs be useful?

When one should use hypergraphs

The system to examine exhibits **group/many-to-many/high-order** interactions.

Group interactions are everywhere!

Group chats, conversations in online social media

Protein-protein interactions

Co-authors of the same publications

Co-purchase data

Group interactions are everywhere!

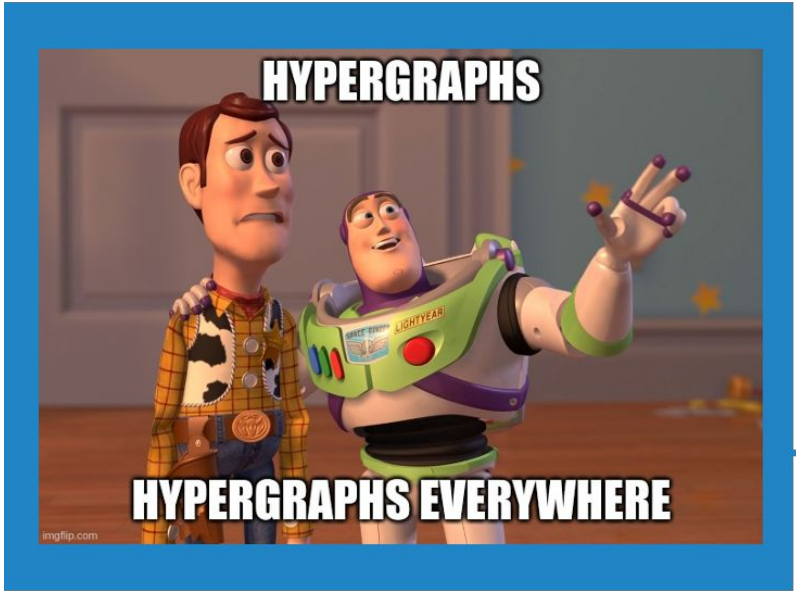
Group chats, conversations in online social media

Protein-protein interactions

Co-authors of the same publications

Co-purchase data

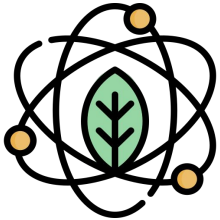
…and many more!



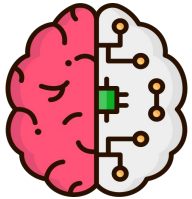
Application domains

Examples of application domains

Social Systems



Biology

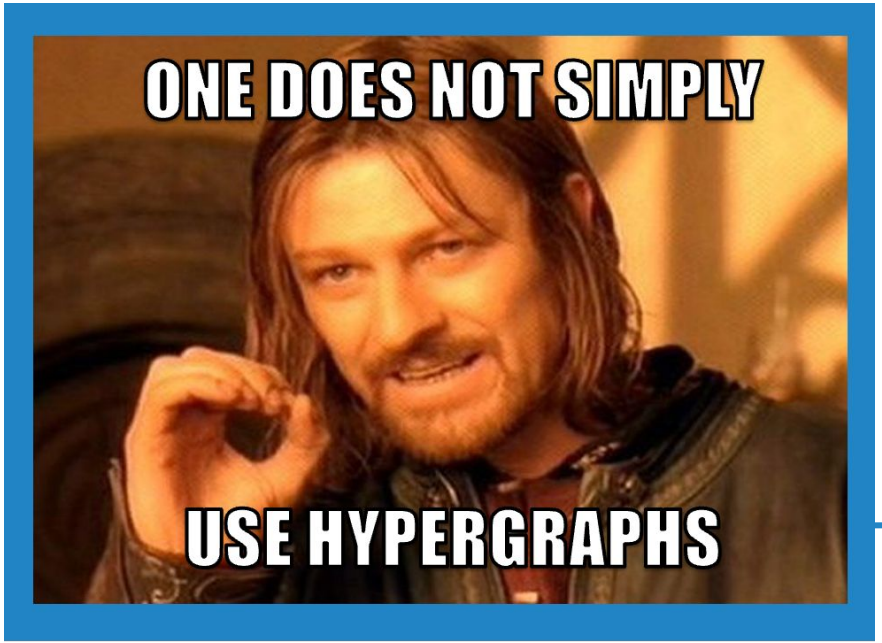


Neuroscience

Ecology

[1] Federico Battiston, Giulia Cencetti, Iacopo Iacopini, Vito Latora, Maxime Lucas, Alice Patania, Jean-Gabriel Young, Giovanni Petri, *Networks beyond pairwise interactions: Structure and dynamics*, Physics Reports, Volume 874, 2020.

[2] Sunwoo Kim, Soo Yong Lee, Yue Gao, Alessia Antelmi, Mirko Polato, Kijung Shin. *A Survey on Hypergraph Neural Networks: An In-Depth and Step-By-Step Guide.* ACM KDD 2024.



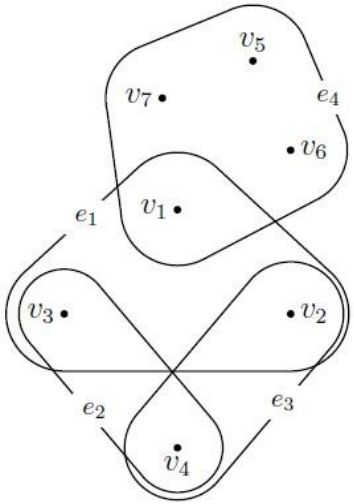
How can we leverage hypergraph representations?

A few drawbacks

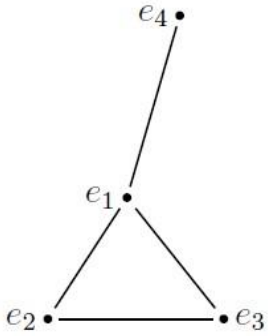
- Hypergraphs add **complexity** (e.g., exponential number of hyperedges);
- Need of **dedicate** algorithms and tools (e.g., hypergraph walks have length and width [1])

[1] Aksoy, S.G., Joslyn, C., Ortiz Marrero, C. *et al. Hypernetwork science via high-order hypergraph walks. EPJ Data Sci.* 9, 16 (2020).

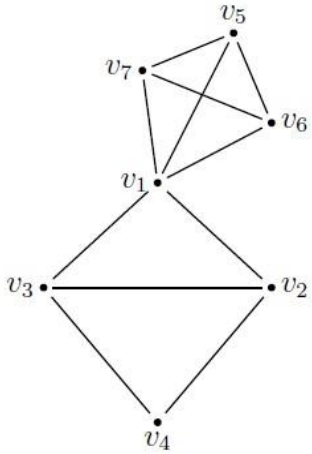
Hypergraph to graph transformations



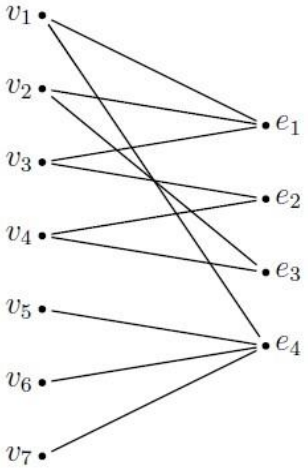
a) Hypergraph



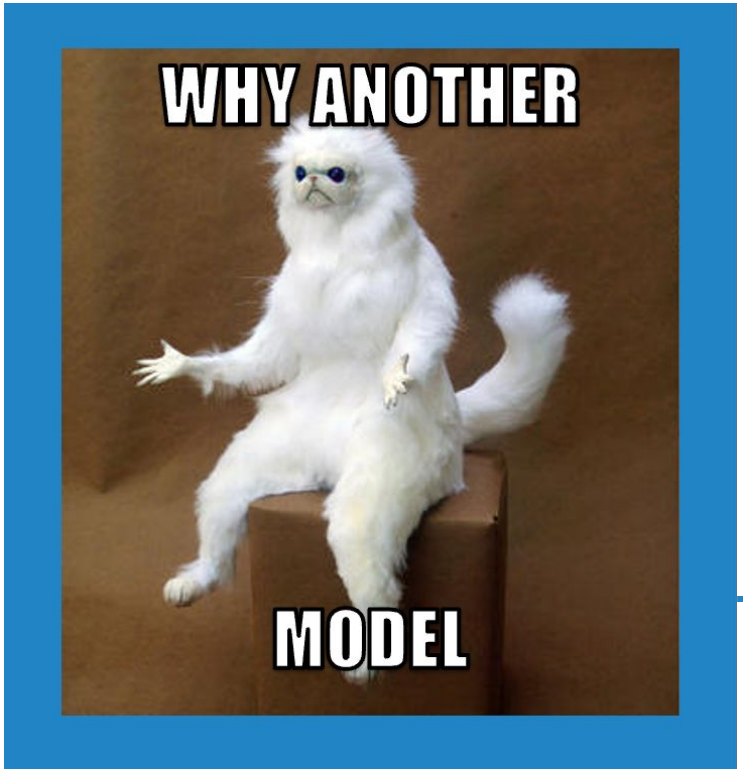
b) Line graph



c) Clique graph



b) Bipartite graph



Why should we use hypergraphs?

Limitations of transforming hypergraphs to graphs

Line graphs and clique graphs

- We lose information about group interactions
  - In practice, we cannot go back to the original hypergraph once transformed into a graph…
  - …since different hypergraphs may have the same line/clique graph.
  - Further, we may materialize relations that do not exist.

Limitations of transforming hypergraphs to graphs

Line graphs and clique graphs

- We lose information about group interactions
  - In practice, we cannot go back to the original hypergraph once transformed into a graph…
  - …since different hypergraphs may have the same line/clique graph.
  - Further, we may materialize relations that do not exist.

- Need more space
  - Line graph: each vertex of size *d* yields to *d choose 2* edges;
  - Clique graph: each hyperedge of size k yields to $k(k-1)/2$.

Limitations of transforming hypergraphs to graphs

Line graphs and clique graphs

- We lose information about group interactions
  - In practice, we cannot go back to the original hypergraph once transformed into a graph…
  - …since different hypergraphs may have the same line/clique graph.
  - Further, we may materialize relations that do not exist.

- Need more space
  - Line graph: each vertex of size *d* yields to *d choose 2* edges;
  - Clique graph: each hyperedge of size k yields to $k(k-1)/2$.

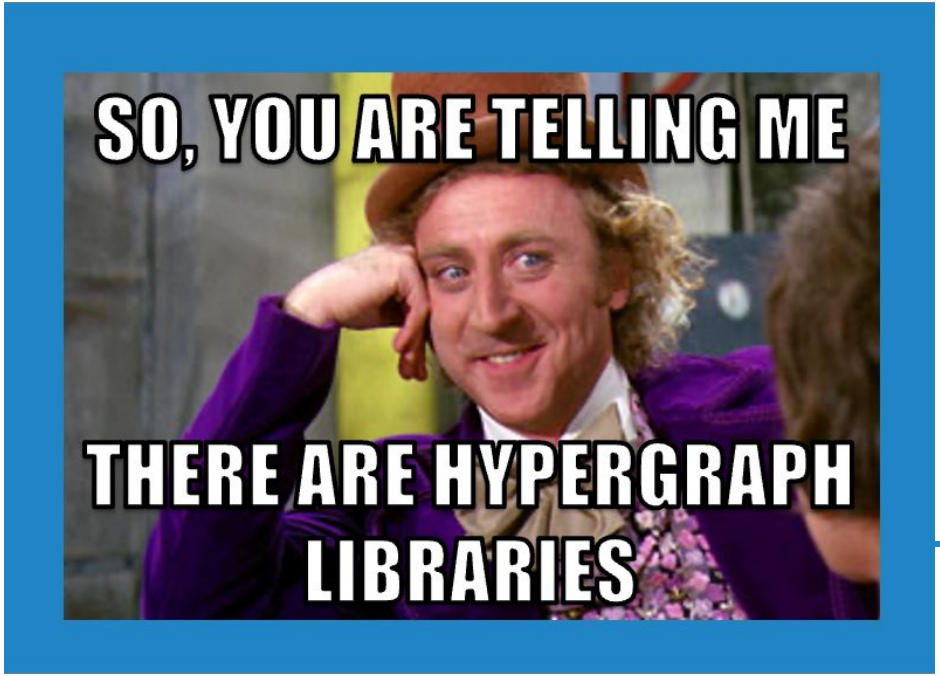
Bipartite graphs

- Vertices do not interact directly anymore.

The why of hypergraph-specific tools

- Hypergraph to graph transformations represent a **trade-off** between **computability** and **accuracy**
- An increasing number of systematic studies demonstrate why one should prefer hypergraphs over graphs
  - Clearly, **in presence of high-order relationships**!

Alessia Antelmi, Gennaro Cordasco, Mirko Polato, Vittorio Scarano, Carmine Spagnuolo, and Dingqi Yang. *A Survey on Hypergraph Representation Learning*. ACM Comput. Surv. 56, 1, Article 24, 2024,



Coding hypergraphs

Existing hypergraph software libraries

- Currently, we count **13 general-purpose** hypergraph software libraries
- Specifically designed to handle hypergraphs or expansion of existing graph libraries

Existing hypergraph software libraries

- Currently, we count **13 general-purpose** hypergraph software libraries
- Specifically designed to handle hypergraphs or expansion of existing graph libraries

- Programming language
  - Python
  - Julia
  - Chapel
  - Matlab
  - C/C++
  - Rust
  - R
  - JavaScript

Existing hypergraph software libraries

1 - Chapel HyperGraph Library

2 - Gspbox

3 - Halp

4 - Hygra

5 - Hypergraph

6 - HyperGraphLib

7 - hypergraphx

8 - HyperNetX

9 - HyperX

10 - Iper

11 - NetworkR

12 - Multihypergraph

13 - SimpleHypergraphs.jl

Existing hypergraph software libraries

1 - Chapel HyperGraph Library

2 - Gspbox

3 - Halp

4 - Hygra

5 - Hypergraph

6 - HyperGraphLib

7 - **hypergraphx**

8 - **HyperNetX**

9 - HyperX

10 - Iper

11 - NetworkR

12 - Multihypergraph

13 - **SimpleHypergraphs.jl**

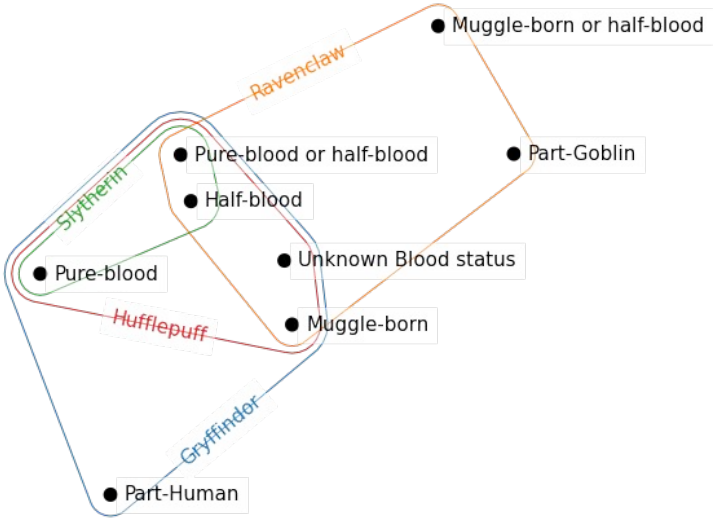
HyperNetX

- Python package to model, analyze, and visualize hypergraphs
- Developed by the Pacific Northwest National Laboratory since 2018
- Publicly available on a GitHub repository
  - https://github.com/pnnl/HyperNetX

C.A. Joslyn, S. Aksoy, D. Arendt, L. Jenkins, B. Praggastis, E. Purvine, and M. Zalewski. *Hypergraph analytics of domain name system relationships*. In Proceedings of Algorithms and Models for the Web Graph - 17th International Workshop (WAW'20), volume 12091 of Lecture Notes in Computer Science, pages 1–15. Springer, 2020.

HyperNetX

- Generalization of traditional graph metrics to hypergraphs
- Hypergraph-specific algorithms
- Visualization functionalities
- Add-on for providing optimized C++ implementations



S. G. Aksoy, C. Joslyn, C. Ortiz Marrero, B. Praggastis, and E. Purvine. *Hypernetwork science via high-order hypergraph walks.* EPJ Data Science, 9(1):16, 2020.

hypergraphx

- Python package to build, visualize, and analyze hypergraphs
- Joint project by University of Trento and Central European University
- Publicly available on a GitHub repository
  - https://github.com/HGX-Team/hypergraphx

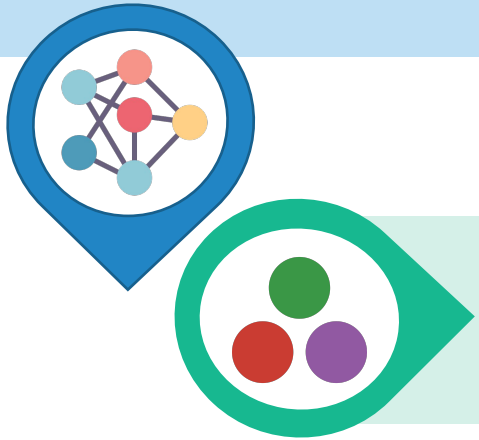
Quintino Francesco Lotito, Martina Contisciani, Caterina De Bacco, Leonardo Di Gaetano, Luca Gallo, Alberto Montresor, Federico Musciotto, Nicolò Ruggeri, Federico Battiston, *Hypergraphx: a library for higher-order network analysis*, Journal of Complex Networks, Volume 11, Issue 3, June 2023, cnad019, https://doi.org/10.1093/comnet/cnad019

hypergraphx

- Different hypergraph representations
- Basic node and hyperedge statistics
- Centrality measures
- Motifs
- Mesoscale structures (e.g., communities)
- Filters
- Generative models
- Dynamical processes
- Visualization

Outline

Hypergraph basics and tools



SimpleHypergraphs.jl

SimpleHypergraphs.jl

Simplehypergraphs.jl

- Julia package to build, visualize, and analyze hypergraphs
- Joint project by
  - Università degli Studi di Salerno (Salerno, Italy)
  - Università della Campania “Luigi Vanvitelli” (Caserta, Italy)
  - Warsaw School of Economics (Warsaw, Poland)
  - Ryerson University (Toronto, Canada)
- Publicly available on a GitHub repository
  - https://github.com/pszufe/SimpleHypergraphs.jl
  - Official Julia package registry

Antelmi, G. Cordasco, B. Kamiński, P. Prałat, V. Scarano, C. Spagnuolo, P. Szufel, *SimpleHypergraphs.jl—Novel Software Framework for Modelling and Analysis of Hypergraphs.* In: *Algorithms and Models for the Web Graph*. 2019, pp. 115–129.
Antelmi, G. Cordasco, B. Kamiński, P. Prałat, V. Scarano, C. Spagnuolo, P. Szufel, *Analyzing, Exploring, and Visualizing Complex Networks via Hypergraphs using SimpleHypergraphs.jl,* 2020, Internet Mathematics.

Not ‘yet another’ hypergraph library

Existing software libraries for hypergraphs

- Are a compromise between efficiency (C/C++), and the ease-of-use and expressiveness (Python and R)
- Only support a restricted set of hypergraph-related functions
- Rely on hypergraph to graph transformations and do not expose any specific methods

Not ‘yet another’ hypergraph library

Desiderata for a hypergraph library

- Software libraries specifically designed to perform operations on hypergraphs
- Flexible definition of data structures and functionalities
- An easy-to-learn and fast implementation language

Modeling hypergraphs

Simplehypergraphs.jl represents hypergraphs as:

- A collection of vertices belonging to hyperedges
- A collection of hyperedges containing vertices

Modeling hypergraphs

Simplehypergraphs.jl represents hypergraphs as:

- A collection of vertices belonging to hyperedges
- A collection of hyperedges containing vertices

and as:

- A matrix H, where the entry H[v, e] indicates the weight of the vertex v in the hyperedge e.

Modeling hypergraphs

Simplehypergraphs.jl represents hypergraphs as:

- A collection of vertices belonging to hyperedges
- A collection of hyperedges containing vertices

and as:

- A matrix H, where the entry H[v, e] indicates the weight of the vertex v in the hyperedge e.

Two-fold integration with:

Julia standard matrix type

LightGraphs.jl

Algorithms

Currently, SimpleHypergraphs.jl offers

- Methods to **generate random hypergraphs** (with or without structural constraints)
- **Hypergraph-specific procedures**, such as centrality measures, random walks, and community detection algorithms
- Two **visualization** methods
- The possibility to **attach metadata** to the hypergraph structure
- Two **serialization mechanisms**

Let's see it in practice!

Simplehypergraphs.jl basics

A Game of Thrones use case

Antelmi, G. Cordasco, B. Kamiński, P. Prałat, V. Scarano, C. Spagnuolo, P. Szufel, *Analyzing, Exploring, and Visualizing Complex Networks via Hypergraphs using SimpleHypergraphs.jl,* 2020, Internet Mathematics.

Let's see it in practice!

Simplehypergraphs.jl basics

- https://nbviewer.org/github/pszufe/SimpleHypergraphs.jl/blob/master/tutorials/basics/SimpleHypergraphs_tutorial_v4.ipynb

A Game of Thrones use case

- https://nbviewer.org/github/pszufe/SimpleHypergraphs.jl/blob/master/tutorials/basics/A%20case%20study%20-%20Game%20of%20Thrones.ipynb

Antelmi, G. Cordasco, B. Kamiński, P. Prałat, V. Scarano, C. Spagnuolo, P. Szufel, *Analyzing, Exploring, and Visualizing Complex Networks via Hypergraphs using SimpleHypergraphs.jl,* 2020, Internet Mathematics.

Analyzing social networks with SimpleHypergraphs.jl

- Analysis of business reviews from the online platform Yelp.com

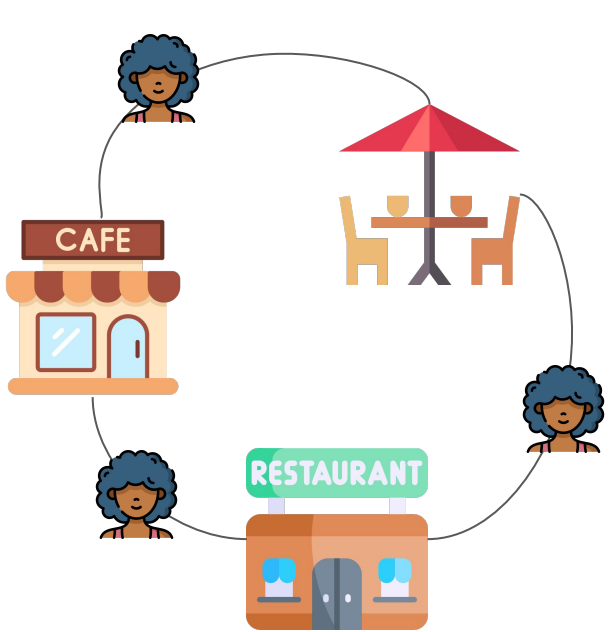
Antelmi, G. Cordasco, B. Kamiński, P. Prałat, V. Scarano, C. Spagnuolo, P. Szufel, *Analyzing, Exploring, and Visualizing Complex Networks via Hypergraphs using SimpleHypergraphs.jl,* 2020, Internet Mathematics.

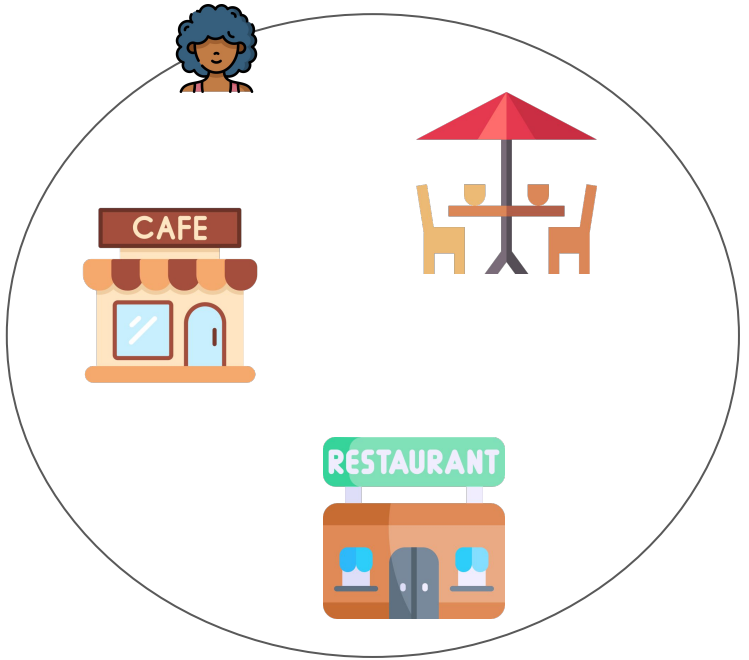
Analyzing social networks with SimpleHypergraphs.jl

- Analysis of business reviews from the online platform Yelp.com
- (Hyper)network structure
  - Vertex = business
  - Edge = user
  - Hyperedge = all businesses reviewed by the same users

Antelmi, G. Cordasco, B. Kamiński, P. Prałat, V. Scarano, C. Spagnuolo, P. Szufel, *Analyzing, Exploring, and Visualizing Complex Networks via Hypergraphs using SimpleHypergraphs.jl*, 2020, Internet Mathematics.

Analyzing social networks with SimpleHypergraphs.jl





Antelmi, G. Cordasco, B. Kamiński, P. Prałat, V. Scarano, C. Spagnuolo, P. Szufel, *Analyzing, Exploring, and Visualizing Complex Networks via Hypergraphs using SimpleHypergraphs.jl*, 2020, Internet Mathematics.

Analyzing social networks with SimpleHypergraphs.jl

Which network model conveys more information about the ground-truth properties of the dataset?

Antelmi, G. Cordasco, B. Kamiński, P. Prałat, V. Scarano, C. Spagnuolo, P. Szufel, *Analyzing, Exploring, and Visualizing Complex Networks via Hypergraphs using SimpleHypergraphs.jl,* 2020, Internet Mathematics.

Analyzing social networks with SimpleHypergraphs.jl

- Task: **clustering**
- Algorithm: **label propagation**
- Ground-truth partition: **type of cuisine**
- Evaluation metric: **normalized mutual information**
- Data:
  - Five different sub-(hyper)networks, each one containing only reviews with the same number of stars, from 1 to 5.
  - Restaurants

Raghavan, U. N., Albert, R., and Kumara, S. *Near linear time algorithm to detect community structures in large-scale networks.* Physical review. E, Statistical, nonlinear, and soft matter physics 76 (2007).

Antelmi, G. Cordasco, B. Kamiński, P. Prałat, V. Scarano, C. Spagnuolo, P. Szufel, *Analyzing, Exploring, and Visualizing Complex Networks via Hypergraphs using SimpleHypergraphs.jl*, 2020, Internet Mathematics.

Analyzing social networks with SimpleHypergraphs.jl

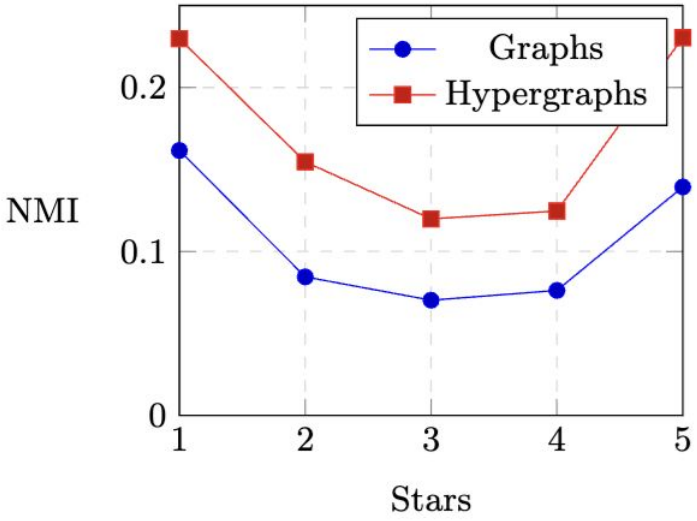
- Task: **clustering**
- Algorithm: **label propagation**
- Ground-truth partition: **type of cuisine**
- Evaluation metric: **normalized mutual information**
- Data:
  - Five different sub-(hyper)networks, each one containing only reviews with the same number of stars, from 1 to 5.
  - Restaurants

Stars	H_i $(\|V\|;\|E\|)$	G_i $(\|V\|;\|E\|)$
1	(29479; 244671)	(29479; 240412)
2	(28055; 173140)	(28055; 484527)
3	(30369; 177792)	(30369; 2636712)
4	(32987; 301578)	(32987; 4384044)
5	(32558; 590320)	(32558; 2187473)

Raghavan, U. N., Albert, R., and Kumara, S. *Near linear time algorithm to detect community structures in large-scale networks.* Physical review. E, Statistical, nonlinear, and soft matter physics 76 (2007).

Antelmi, G. Cordasco, B. Kamiński, P. Prałat, V. Scarano, C. Spagnuolo, P. Szufel, *Analyzing, Exploring, and Visualizing Complex Networks via Hypergraphs using SimpleHypergraphs.jl*, 2020, Internet Mathematics.

Analyzing social networks with SimpleHypergraphs.jl

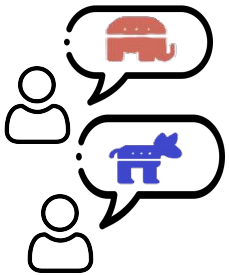


ASH: Attributed Stream-Hypergraphs

Overview

A

Node Attributes



Correlation between structure and metadata

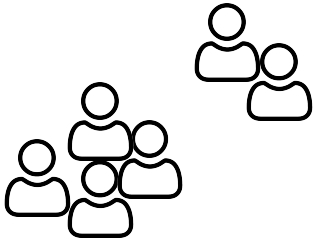
S

Evolving Topologies

Structures (and metadata) vary over time

H

High-order Interactions



Beyond pairwise/dyadic connectivity patterns

Failla A. ,Citraro S,, Rossetti G., Attributed Stream Hypergraphs: temporal modeling of node-attributed high-order interactions, Applied Network Science, 2023

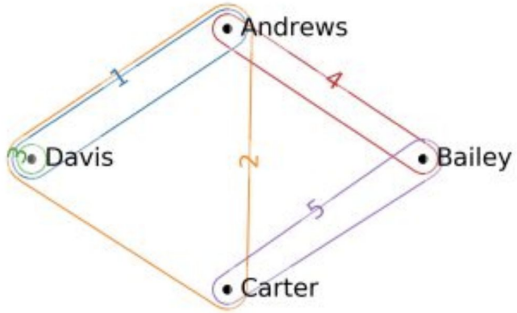
Properties of Stream-Hypergraphs

- Nodes and (hyper)edges are represented as **whole quantities** only when present at **all** instants in the stream (node/edge contribution)

$$u = \frac{|T_u|}{|T|} \qquad (u, v) = \frac{|T_{u,v}|}{|T|}$$

Properties of Stream-Hypergraphs

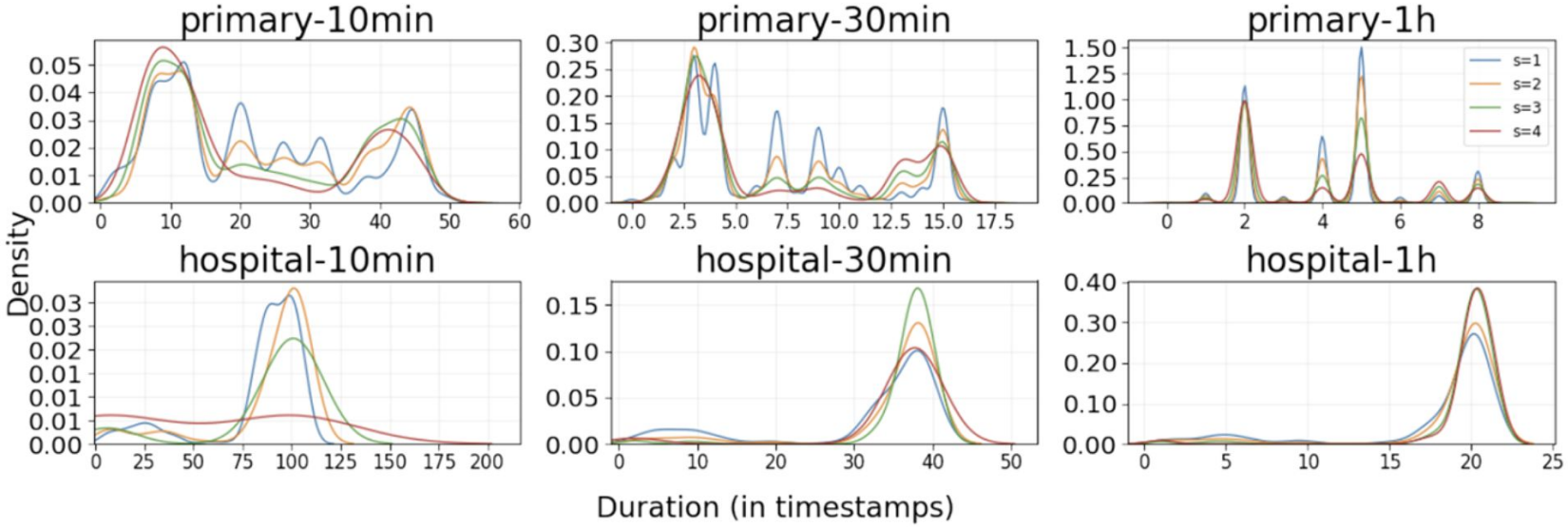
- Nodes and (hyper)edges are represented as **whole quantities** only when present at **all** instants in the stream
- Walks and paths have a **length**, a **width**, and a **duration**



Example:
- 1 and 2 are 2-incident (they share {Davis, Andrews})
- 5 and 4 are 1-incident (they share {Bailey})

Properties of Stream-Hypergraphs

- Nodes and (hyper)edges are represented as **whole quantities** only when present at **all** instants in the stream
- Walks and paths have a **length**, a **width**, and a **duration**



Enriching Topology: Node Profiles

- Entities in complex systems exhibit **a wide range of features** that affect/relate to the topology
- ASHs allows to study **higher-order temporal trends of homophily**

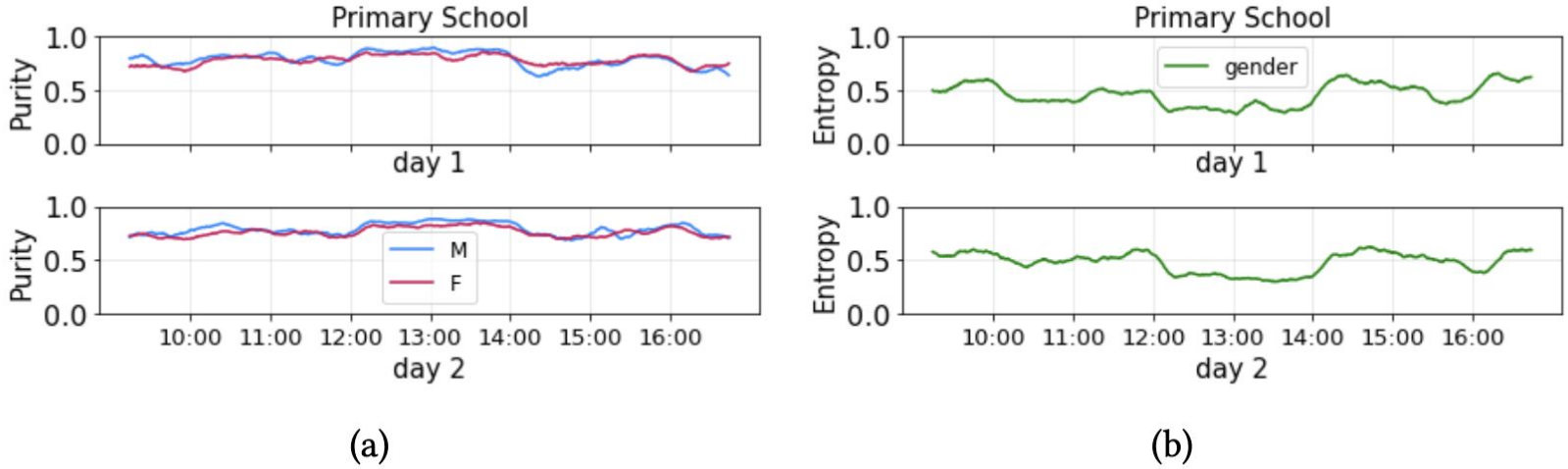
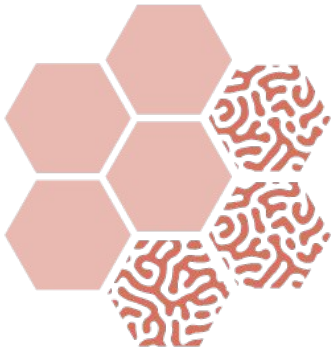


Figure 2: (a) Purity and (b) Entropy trends in the Primary School contact dataset.



ASH

CONDA

GitHub

python™ Package Index

Installation

if you want to use conda:

```
conda install ash_model
```

Or, if you have pip installed:

```
pip install ash_model
```

Alternatively, you can install the latest version directly from GitHub:

```
pip3 install git+https://github.com/GiulioRossetti/ASH.git
```

From the ground up…

Create an empty ASH with no nodes and no edges.

```
from ash_model import ASH
h = ASH(hedge_removal=True)
```

```
h.add_hyperedge({1,2,3,4}, start=0, end=1)
```

```
h.add_hyperedges([{1,2,3}, {2, 3}, {3, 1, 4}], start=1, end=2)
```

```
h.get_star(1) # set of ids of the hyperedges that contain node 1
h.get_star(1, hyperedge_size=4, tid=2) # set of ids of the hyperedges that contain node 1,
                                       # have size 4
                                       # and are active at time 2
```

Let's add node semantics…

```
# one node at a time
h.add_node(0, start=0)
h.add_node(1, start=0, end=1)

# multiple nodes, same time window
h.add_nodes([1,2,3], start=0)
h.add_nodes([4,5,6], start=0, end=1)
```

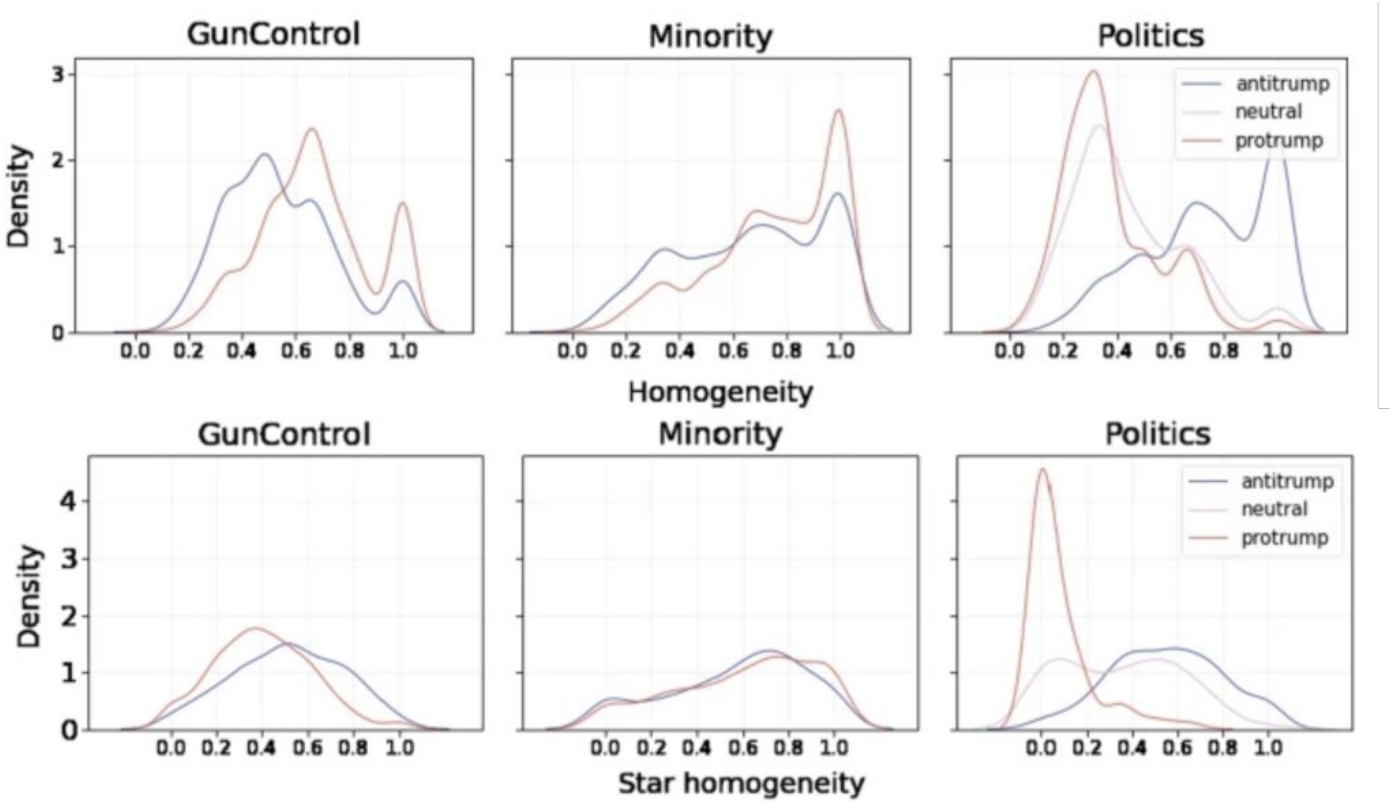
```
from ash_model import NProfile
profile = NProfile(node_id=1, name='Alice', party='L') # add attributes at creation
profile.add_attribute('age', 24) # add attribute with dedicated method
```

```
profile = NProfile(node_id=1, name='Alice', party='L', age=24)
h.add_node(1, start=0, attr_dict=profile)
profile2 = NProfile(node_id=1, name='Alice', party='R', age=25)
h.add_node(1, start=1, attr_dict=profile2)
```

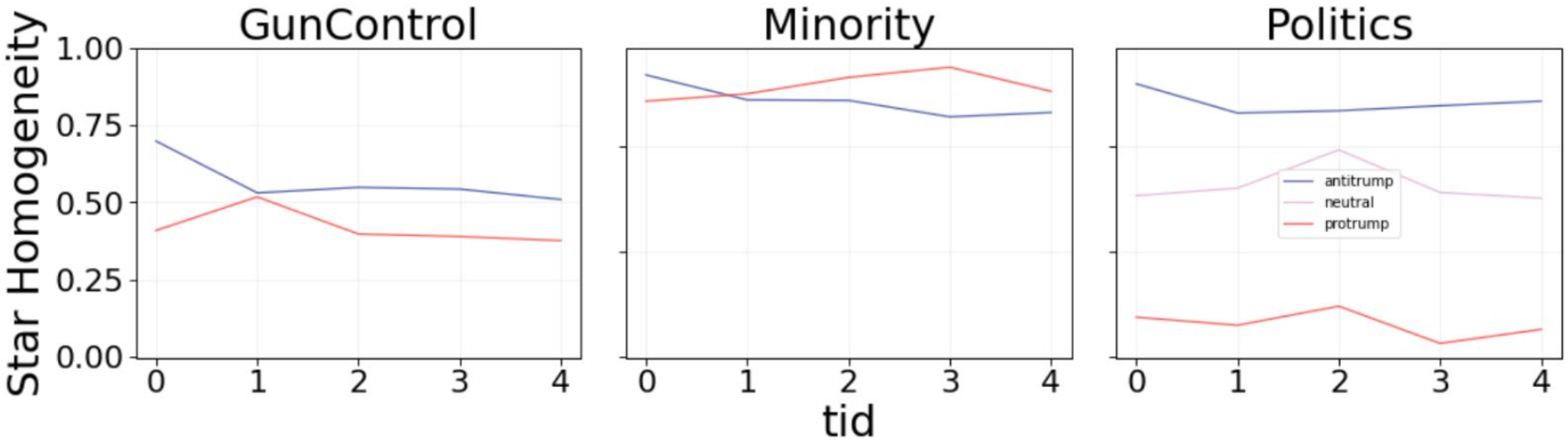
Case Study: Reddit Political Discussions

Fraction of neighbors that share the same opinion as the ego

Fraction of discussions where ego is involved and where her opinion is predominant



Case Study: Reddit Political Discussions



Wrapping up…

Summary

- Hypergraphs are **effective tools** to analyze and mine **group** interactions.
- Many hypergraph libraries exist, few are **efficient** and/or **actively maintained**
- **SimpleHypergraphs.jl** is an efficient all-round software for handling hypergraph data & tasks
- **ASH** allows modeling temporal hypergraphs and node attribute dynamics

Future Directions?

- Higher-order methods have high complexity
  - Design efficient/parallelizable high-order methods

- Hypergraph extensions of graph concepts may be limiting
  - Hypergraph native algorithmic design

Thanks!

alessia.antelmi@unito.it
andrea.failla@phd.unipi.it