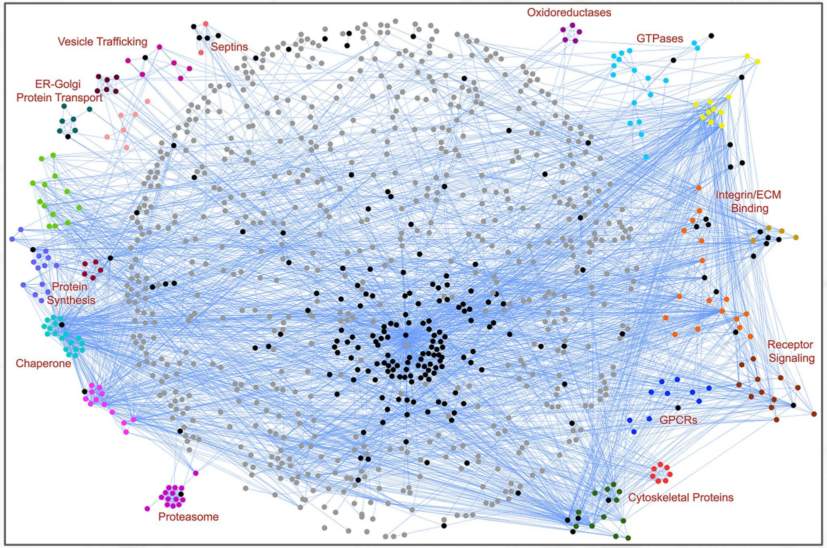
Centralities in Simplicial Complexes

Yury Biktairov Konstantin Sobolev Anton Smerdov

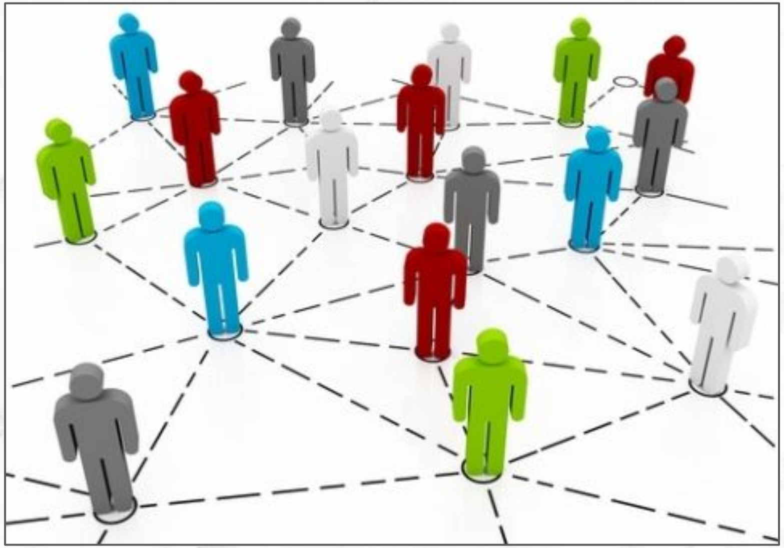
When do we need to describe interactions?

Analysis of chemical structures



PPI for some set of proteins.

Social interactions analysis



Social connections graph.

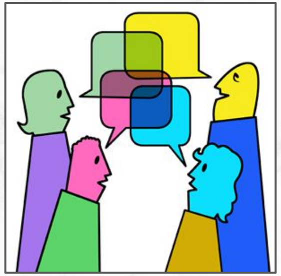
Insufficiency of conventional description

Graph representation cannot adequately describe interactions of higher order.

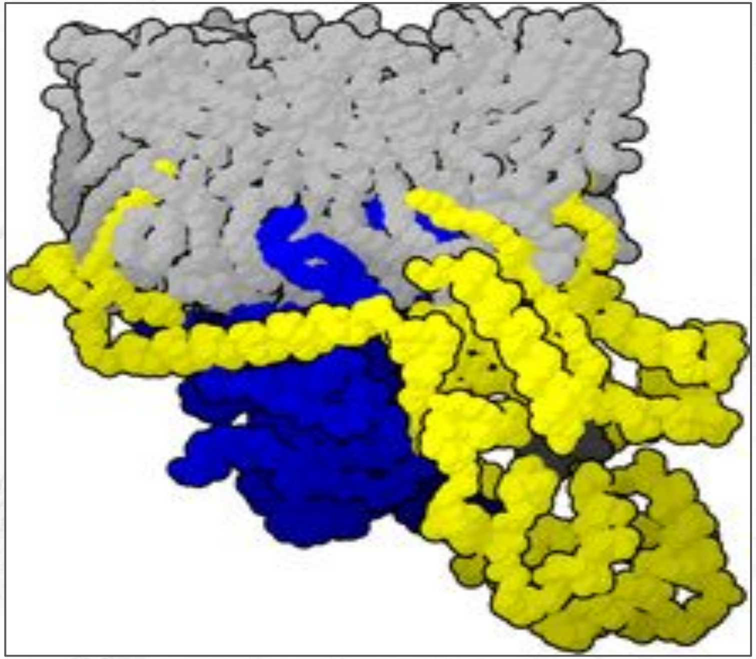
Some examples:

- Heterotrimer is a structure consisting of three jointly interacting proteins and cannot be completely described via pairwise interactions

- A conversation between four people is not a pairwise process and cannot be correctly described with a simple hypergraph edge either



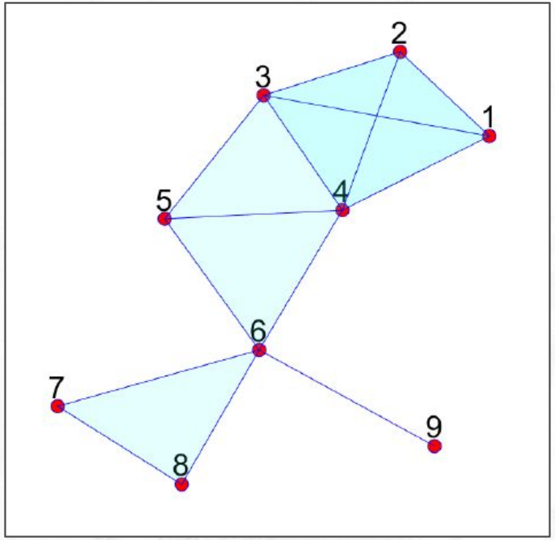
A conversation.



Simplices and Simplicial Complexes

Instead of pairwise connections we introduce simplices

- **k-simplex** is a set $\{v_0, \ldots, v_k\}$, where v_i is a set of nodes or vertices, and v_i != v_j for all i != j
- **Face of k-simplex** is a (k-1) of the form $\{v_0, \ldots, v_{i-1,} v_{i+1} \ldots, v_k\}$ for i in range(k+1)
- **Simplicial complex C** is a collection of simpleces such that if a simplex S is in C then all faces of S are also in C (closeness under taking subsets)



Example of simplicial complex.

Adjacency of simplices

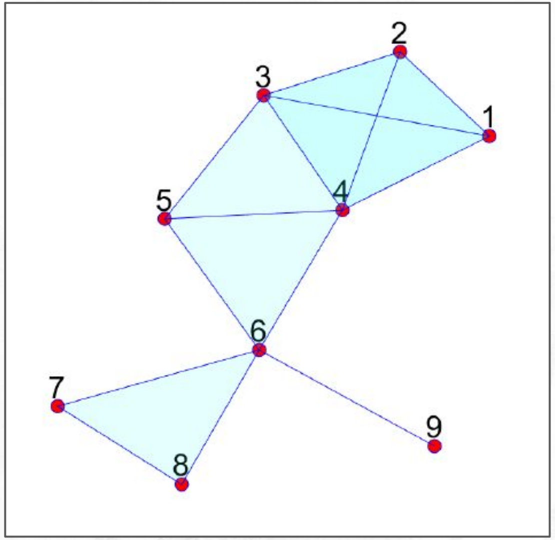
In order to calculate simplicial centralities, we should define the notion of adjacency of two simplices (of the same order)

- Let two simplices be **lower-adjacent** if they share a common face

- Let two simplices be **upper-adjacent** if they are faces of a common simplex

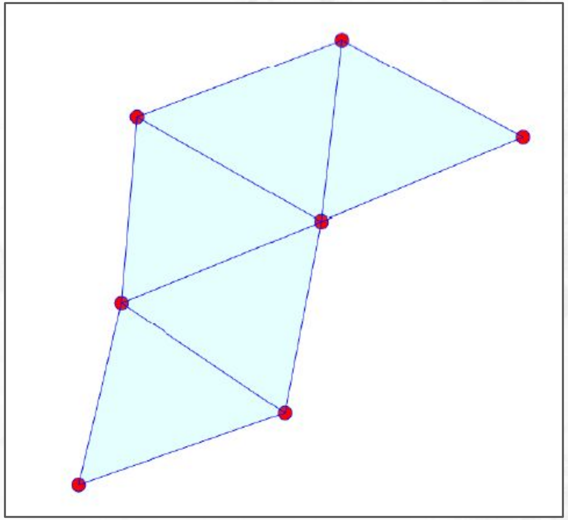
- Examples: {5,3,4} and {5,4,6} are lower-adjacent but not upped-adjacent; {3,1,4} and {3,2,1} are both lower- and upper-adjacent

- Clearly, if two simplices are upper-adjacent they are also lower-adjacent due to the closeness property



Example of simplicial complex.

Adjacency of simplices



Another example of simplicial complex.

We have four options for adjacency matrix definition now:

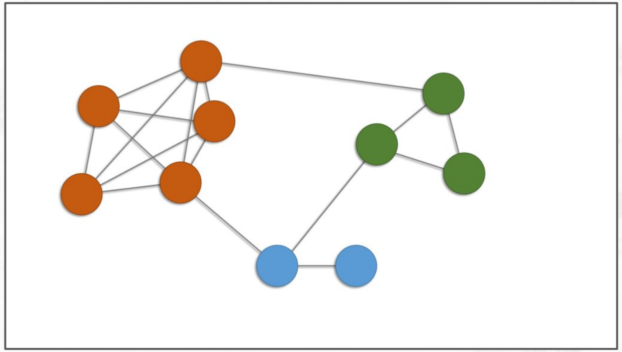
1) A(L) — does not use higher order simplices
2) A(U) — only create connections inside big simplices
3) A(L) + A(U) — exceeds one for upper-adjacent simplices
4) A(L) - A(U) — is a sensible option

So the final definition of adjacency we are going to use is

- Two simplices are **adjacent** if they are lower-adjacent but not upper-adjacent.

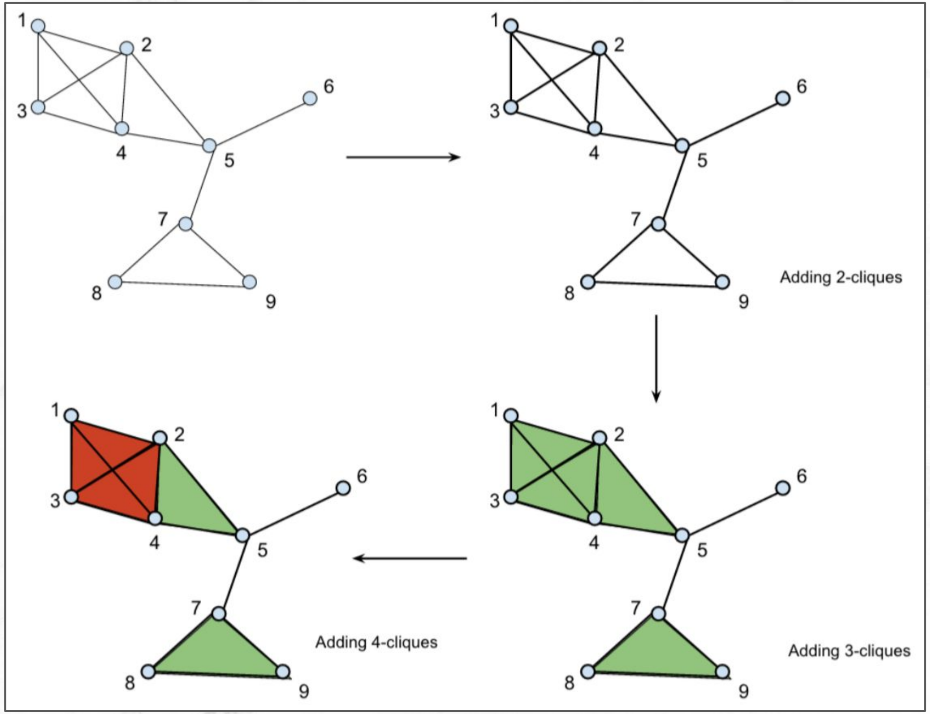
Clique complex of a graph

- A clique of some graph is it’s complete induced subgraph.
- One important property is that induced subgraph of a clique is a clique.



Cliques of a graph.

- Thus, we can use cliques to construct a meaningful simplicial complex: the closeness property of complex will be satisfied.



Example of clique complex construction.

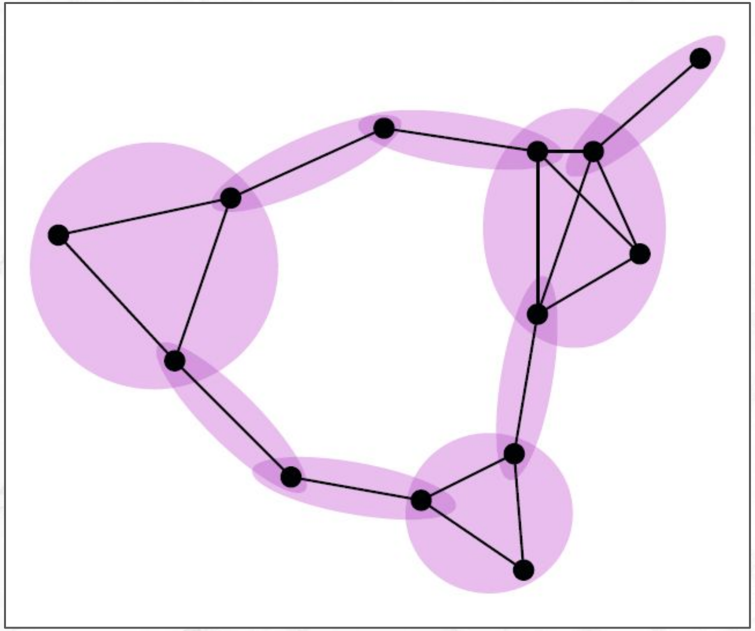
Clique complex construction

Possible algorithms for clique complex construction include:

- Bruteforce method
- Incremental algorithm
- Maximal algorithm

Advanced methods are only beneficial in sparse graphs.

We use bruteforce since our calculations have bottleneck elsewhere — adjacency matrix is quadratic in number of simplices.



Maximal cliques of a graph.

Data

- EEG data for 100 participants
- Binary target: depression, 50% positives
- Correlation matrix for 117 points in the brain
- Preprocessing:

Correlation matrix

1	0.83	0.72
0.83	1	0.23
0.72	0.23	1

thresholding, h = 0.8

remove self-connections

Adjacency matrix

0	1	1
1	0	0
1	0	0

Features: median of

Characteristics name	Formula	Description
Simplicial Closeness	$C(F) = \frac{1}{\sum_{Y \neq F} d(Y,F)}$	Reciprocal of simplicial farness, $d(Y,F)$ - shortest path from Y to F
Simplicial Betweenness	$g(F) = \sum_{S \neq F \neq T} \frac{\sigma_{ST}(F))}{\sigma_{ST}}$	σ_{ST} # of shortest paths between S and T, $\sigma_{ST}(F)$ # --//-- frough F
Simplicial Degree	$\delta_k = \sum_i A_{i\bullet} = \sum_j A_{\bullet j}$	number of other k-simplices to which s is adjacent
Clustering Coefficient	$c_u = \frac{2T(u)}{deg(u)(deg(u)-1)}$	where T(u) is the number of triangles through node u and deg(u) is the degree of u
Average Degree for Neighbours	$k_{nn,i} = \frac{1}{\lvert N(i) \rvert} \sum_{j \in N(i)} k_j$	N(i) are neighbours of node i and kj is the degree of node j which belongs to N(i)

Features

Characteristics name	Description
Number of edges	Self-explanatory
Average global efficiency	Average efficiency of all pairs of nodes
Average local efficiency	The average of the local efficiencies of each node

Experimental results

- AUC ROC
- Repeated K-Fold (5 splits, 10 repeats)
- 8/16/24 features

Model Name	**1st level features score**	**2nd level features score**	**3rd level features score**
Logistic regression	0.543	**0.638**	0.572
Random Forest	0.511	0.591	**0.597**
SVC	0.552	0.540	0.570
KNN	**0.555**	0.465	0.505

Feature selection

- Lasso with C=0.03
- 2/4/6 features

Model Name	**1st level features score**	**2nd level features score**	**3rd level features score**
Logistic regression	0.598	0.637	0.636
Random Forest	0.526	**0.659**	**0.703**
SVC	**0.599**	0.578	0.599
KNN	0.447	0.523	0.548

The most important features

1st level features	2nd level features	3rd level features
Degrees 1 lvl Global efficiency 1 lvl	Betweenness 2 lvl Degrees 1 lvl Local efficiency 2 lvl Local efficiency 1 lvl	AvgNodeDegree 3.1 lvl Closeness 3.2 lvl Clustering 3.1 lvl Clustering 3.2 lvl Global efficiency 3.1 lvl Local efficiency 3.1 lvl

Summary

- Implemented methods for extracting high-level features from graphs
- Applied to depression detection
- New features give reasonable score improvement
- The most important features are figured out

Thank you for your attention