

Motif finding

↳ a subgraph that occurs more frequently than expected (relatively)

Why would it be more frequent?

Biological network: functional reason

→ the way proteins interact

Social networks: the way the network grows → triangles

Financial networks: common transactional patterns

↳ This brings us to antimotifs or anomalies

↳ anomaly detection

Anomaly: something that is not expected to exist

Anomaly: something that is not expected to exist

Eg. an atypical financial transaction pattern might indicate

Interesting open problem:

Cortical conjecture



Consider a brain graph

Structural: how neurons connect

functional: higher-level regional connections

(connectomes)

Basically: how signals flow

Cortical Conjecture: there's repeating computational substructures of the connectome that explain "intelligence"

Generally, how we solve these:

Count occurrences of template T
in network A

Count T in network B

$C(T, A) \gg C(T, B) \rightarrow$ motif in A

$C(T, A) \ll C(T, B) \rightarrow$ anti-motif
in A



\uparrow
null model
or some relevant graph

Note: we can compare graph similarity
based on these counts

Local: between regions of graphs

Global: compare full graphs

Similar to motif finding:

\rightarrow considering subgraph templates

\rightarrow count 'em up

→ we can use these counts in explicit similarity measures

Subgraph Frequency Distance

- Given G and H
- define $N_i(G)$ as counts for template i on graph G
- define $T(G)$ as total counts over all possible $i \rightarrow T(G) = \sum_i N_i(G)$
- define $F_i(G) = \log \left(\frac{N_i(G)}{T(G)} \right)$
↑
minimize influence of any i
- define $D(G, H) = \sum_i |F_i(G) - F_i(H)|$

uses: graph classification
null model selection

Q: what templates/subgraphs?

A: graphlets and others

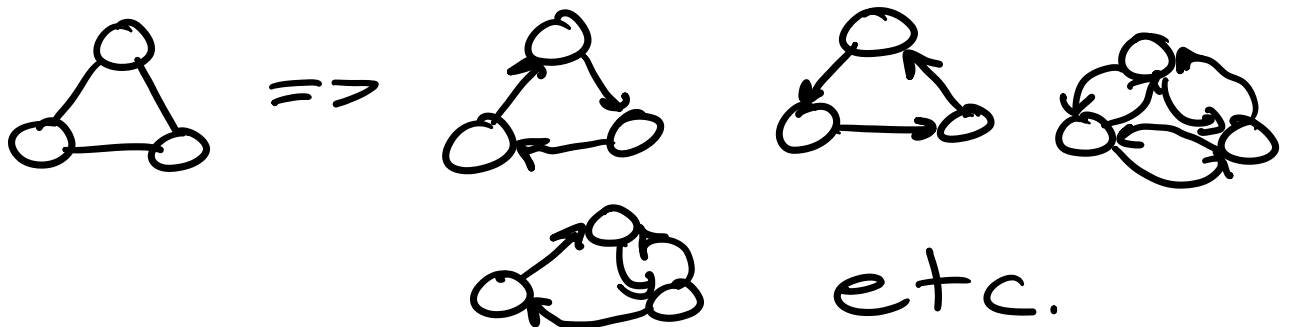
↳ Undirected 3-5 vertex
connected induced subgraphs

→ small enough to compute

→ still varied topology

→ useful enough for practical purposes

Directed: a lot more options



Another option: treelets

Another option: treelets

(Slota special)
↳ tree templates
from 3-9 vertices

Bigger → able to capture
more information
w/ fewer resources

Getting local: vertex similarity
measures

Really: just using our global measure
rooted at some v

How: we root the subgraph
and count embedding, repeat
for all possible roots (orbits)

Consider:



We count for
each auto-
morphically
unique orbit

Subgraph degree signature:

- consider vertices $u, v \in V(G)$
- consider all possible orbits across some set of templates
- For orbit i with counts u_i, v_i

$$D_i(u, v) = w_i \frac{|\log(u_i+1) - \log(v_i+1)|}{\log(\max(u_i, v_i)+2)}$$

weighting term
for automorphisms

Overall:

$$D(u, v) = \frac{\sum_i D_i(u, v)}{\sum_i w_i}$$

↳ from 0...1

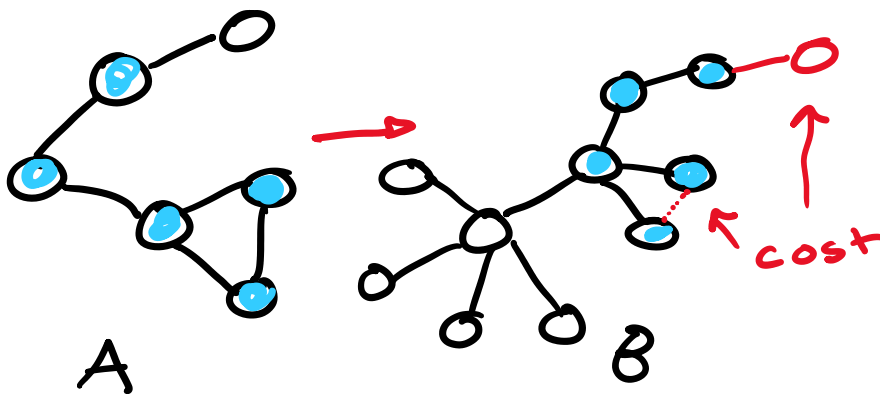
$$S(u, v) = 1 - D(u, v)$$

Uses: vertex classification (functional)
study similar/differing
regions of same network
Graph alignment across
different networks

Graph alignment across differing networks

Graph Alignment

Consider graph alignment as an approximate (sub)graph isomorphism



We generally define some cost and try to minimize it

- there are going to be differing cost metrics across fields/applications
- Functional metrics \rightarrow explicit measures of functional similarity within biological networks
- Structural metrics

Edge correctness - how many edges match

Edit distance - how many vertex/edge deletions to get an exact match

→ similar to DNA sequencing

"Chunks" ATCGTTTAC...
ATGAATGC...

Note: graph alignment is also NP-hard → computationally challenging (especially if we consider subgraph counts for similarity)

How explicitly can we use counts?

- We want to align $G \rightarrow H$

- we count all orbits for all vertices

- we count all orbits for all vertices

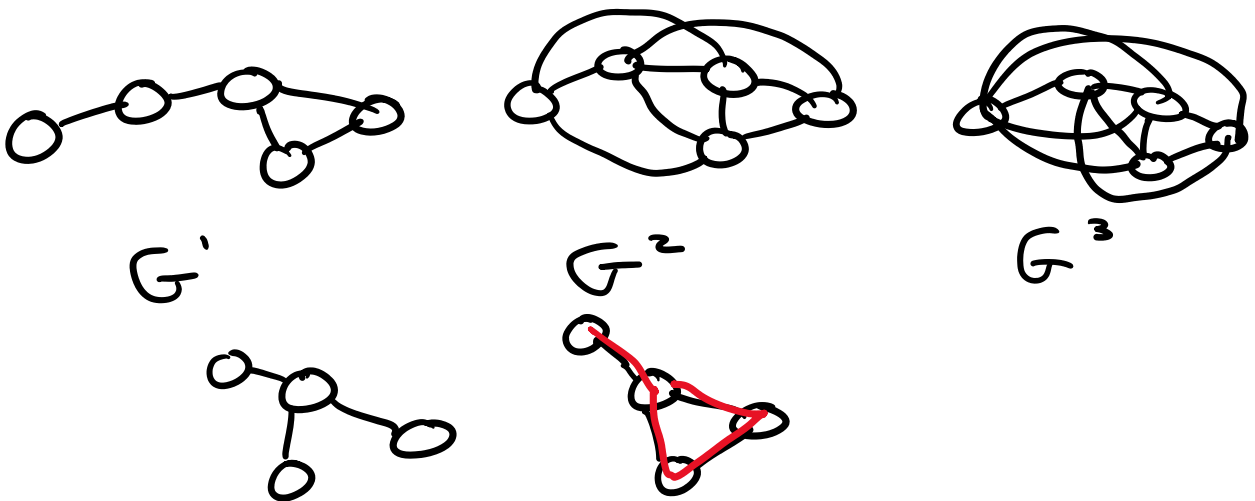
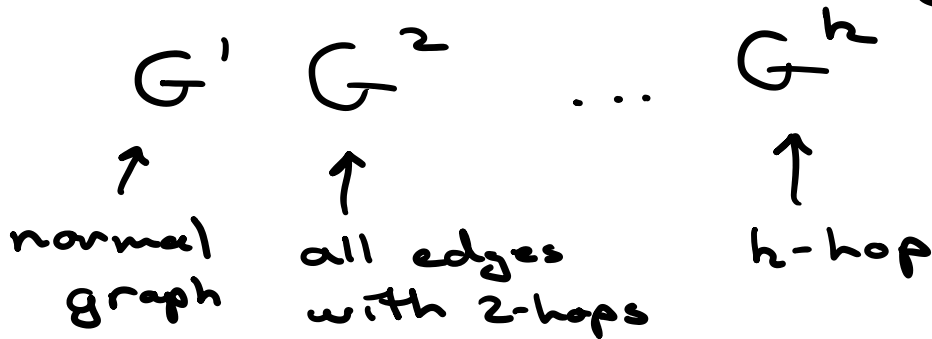
all vertices

- we can compute pairwise similarity for all vertices

- we can then greedily align

Note: we have to account for missing vertices/edges

↳ We can use power graphs



Repeat greedy matching until done