Table of Contents

List of Tables

List of Figures

Chapter 1 Introduction

A wide variety of fields show consistent interest in inferring latent network structure from observed interactions, from human cognition and social infection networks, to marketing, traffic, finance, and many others. [**Inferringnetworksdiffusion_GomezRodriguez2012**] However, an increasing number of authors are noting a lack of agreement in how to approach the metrology of this problem. This includes rampant disconnects between the theoretical and methodological network analysis sub-communities[**Statisticalinferencelinks_Peel2022**], treatment of error as purely aleatory, rather than epistemic [**Measurementerrornetwork_Wang2012**], or simply ignoring measurement error in network reconstruction entirely[**ReconstructingNetworksUnkno**
This thesis builds on recent methodological recommendations for increased focus on how *dependencies* should play a central role in network analysis [**WhyHowWhen_Torres2021**], and facilitating a paradigm shift toward network analysis as *inference* of an inverse problem [**Statisticalinferencelinks_Peel2022**].

1.1 Ambiguous Metrology

Networks in the "wild" rarely exist of and by themselves. Rather, they are a model of interaction or relation *between* things that were observed. One of the most beloved examples of a network, the famed *Zachary's Karate Club*[**InformationFlowModel_Zachary1977**], is in fact reported as a list of pairwise interactions: every time a club member interacted with another (outside of the club), Zachary recorded it as two integers (the IDs of the members). The final list of pairs can be *interpreted* as an "edge list", which can be modeled with a network: a simple graph. This was famously used to show natural community structure

(Figure 1.1) that nicely matches the group separation that eventually took place when the club split into two.[**Communitystructuresocial_Girvan2002**]

Note, however, that we could have just as easily taken note of the instigating student for each interaction (*i.e.*, which student initiated conversation, or invited the other to socialize, etc.). If that relational asymmetry is available, our "edges" are now *directed*, and we might be able to ask questions about the rates that certain students are asked vs. do the asking, and what that implies about group cohesion. Additionally, the time span is assumed to be "for the duration of observation" (did the students *ever* interact), but if observation time was significantly longer, say, multiple years, we might question the credulity of treating a social interaction 2 years ago as equally important to an interaction immediately preceding the split. This is now a "dynamic" graph; or, if we only measure relative to the time of separation, at the very least a "weighted" one.

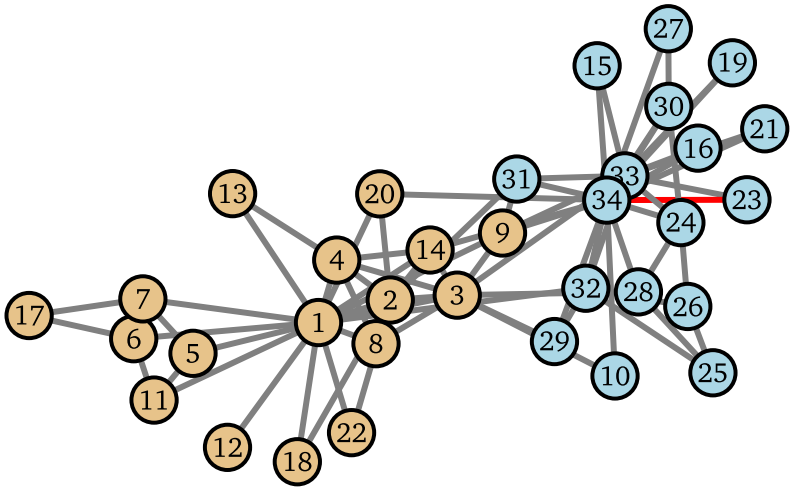


Figure 1.1: Zachary's Karate Club, with ambiguously extant edge 78 highlighted.

This observation raises an interesting metrology problem: *We do not know if any of these are true*. "Metrology" is not limited to physical units, like "meters" and "grams", but more generally is concerned with systematic quantification with uncertainty. Units provide a natural framework to describe what a metrologist is usually after: not just "how much", but

"how *accurately* and *precisely* that much", as well. When we use "metrology" in the context of network analysis, we are specifically referring to the need to:

- Quantify a network
- Consider the trueness of that quantification
- Consider the precision of that quantification.

The difference between trueness and precision is a crucial, often overlooked distinction: how close a set of measurements are to a reference value vs. how repeatable/reproducible a measurement is[**Accuracytruenessprecision_ISO1994**].

The metrological questions we posed above are ones of _trueness_we have no way to tell if Zachary's network model is specified correctly, because the reference network "type" is under-defined and we have no networks to compare it with. We simply have to take the network as a reference unto itself; it is a calibration artefact, much like a physical "meter rod". However, even with an assumed perfect "trueness", precision is often an issue as well! In fact, as illustrated in Figure 1.1, we do not know if the network being described from the original edge data even has 77 or 78 edges, due to ambiguous reporting in the original work. Lacking a precise definition of what the graph's components (*i.e.*, it's edges) are, *as measurable entities*, means we cannot estimate the accuracy of the graph, whether for trueness or precision.

1.2 Indirect Network Measurement

While the karate club graph has unquantified edge uncertainty derived from ambiguous edge measurements, we are fortunate that we *have edge measurements*. Regardless of how the data was collected, it is de facto reported as a list of pairs, which lends itself to treatment as a reference artefact. In many cases, we simply do not have such luxury.

Instead, edges are often measured *indirectly*, and instead we are given lists of node ~~co-ocurrences~~co-occurrences. Networks connecting movies as being "similar" might be

derived from data that lists sets of movies watched by each user; networks of disease spread pathways might be implied from patient infection records; famously, we might build a network of collaboration strength between academic authors by mining datasets of the papers they co-author together.

Such networks are derived from what we will call *node activation* data, *i.e.*, records of what entities happened "together", whether contemporaneously, or in some other context or artifact. For this class, *precision* might be easy to assess, having oft-repeated activations.

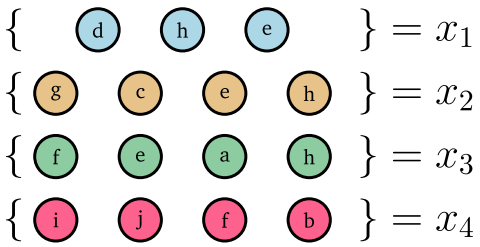
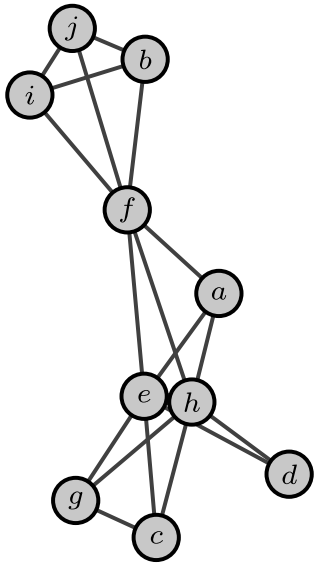


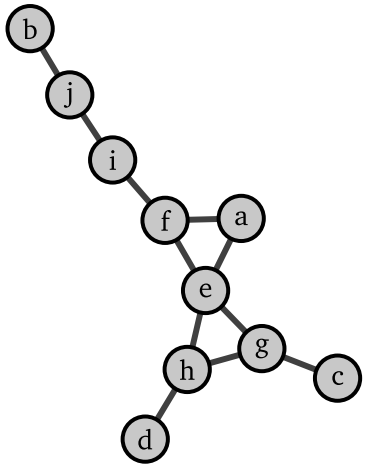
Figure 1.2: Observations as activation sets

These networkx are naturally represented as "bipartite" networks, having separate entites for, say, "papers" and "authors", and connecting them with edges (paper 1 is "connected" to its authors E,H,C, etc.). But analysts are typically seeking the collaboration network connecting authors (or papers) themselves! Networks of relationships in this situation are not directly observed, but which *if recovered* could provide estimates for community structure, importances of individual authors (e.g. as controlling flow of information), and the "distances" that separate authors from each other, in their respective domains. [**Scientificcollaborationnetworks._Newman2001**] Common practice assumes that co-authorship in any paper is sufficient evidence of at least some level of social "acquaintance", where more papers shared means more "connected".

Thus our social collaboration network in Figure 1.3a is borne out of indirect measurements: author connection is implied through "occasions when co-authorship occurred". However, authors of papers may recall times that others were added, not by their choice, but by someone else already involved. In fact, the final author list of most papers is reasonably



(a) Network based solely on co-authorship observations



(b) graph of mutual collaboration relationships i.e. the "ground truth" social network

Figure 1.3: Co-authorship vs. collaborator network

a result of individuals choosing to invite others, not a unanimous, simultaneous decision by all members. Let's imagine we wished to study the social network of collaboration more directly: if we had the luxury of being in situ as, say, a sociologist performing an academic ethnography, we might have been more strict with our definition of "connection". If the goal is a meaningful social network reflecting the strength of interraction between colleagues, perhaps we prefer that our edges represent "mutual willingness to collaborate". Edge "measurement", then, could involve records of events that show willingness to seek or participate in collaboration event, such as:

- *Author (g) asked (e), (h), and (c) to co-author a paper, all of whom agreed*
- *(i) asked (f) and (j), but (j) wanted to add (b)'s expertise before writing one of the sections*
- etc.

Each time two colleagues had an opportunity to work together *and it was seized upon* we might conclude that evidence of their relationship strengthened. With data like this,

we could be more confident in claiming our collaboration network can serve as "ground truth," as far as empirically confirmed collaborations go. However, even if the underlying "activations" are identical, our new, directly measured graph looks very different.

Fundamentally, the network in Figure 3.1b shows which relationships the authors *depend* on to accomplish their publishing activity. When causal relations between nodes are being modeled as edges, we call such a graph a *dependency network*. We will investigate this idea further later on, but ultimately, if a network of dependencies is desired (or implied, based on analysis needs), then the critical problem remaining is *how do we recover dependency networks from node activations?* What is missing, once again, is any sense of *reference value* to base our assessment of *trueness* on. This thesis is primarily concerned with a metrological need within the network analysis community to have terms and techniques for describing and dealing with this problem. What goes wrong when we use co-occurrence/activation data to estimate the dependency network? What goes wrong when we wish to use co-occurrences for metrics like centrality and assortativity, or for exploratory analyses like building relationship type inventories?

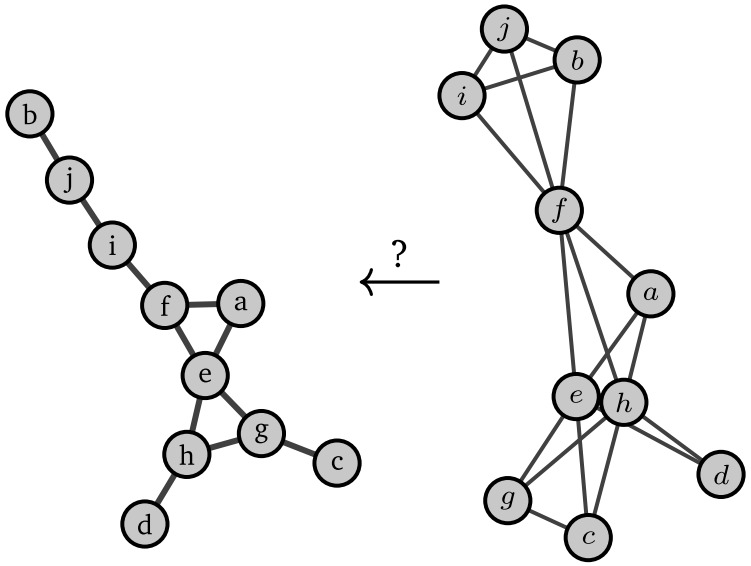


Figure 1.4: Recovering underlying dependency networks from node-cooccurrences.

Even more practically, networks created directly from bipartite-style data are notorious

for quickly becoming far too dense for useful analysis, earning them the (not-so-)loving moniker "hairballs". Network "backboning," as it has come to be called tries to find a subset of edges in this hairball that still captures its core topology in a way that's easier to visualize.[**twostagealgorithm_Slater2009**, **backbonebipartiteprojections_Neal2014**] Meanwhile, underlying networks of dependencies that *cause* node activation patterns can provide this: they are almost always more sparse than their hairballs. Accessing the dependency *backbone* in a principled way is difficult, but doing so in a rapid, scalable manner is critical for practitioners to be able to make use of it to trim their hairballs.

1.3 Scope of this work

The purpose of this thesis is to provide a solid foundation for edge metrology when our data consists of binary node activations, by framing network analysis as a problem of *inference*, as suggested by **Statisticalinferencelinks_Peel2022**. We give special focus to binary activations that occur due to spreading processes, such as random walks or cascades on an underlying carrier graph. Recovering the carrier, or, "dependency" network from node activations is of great interest to the network backboning and causal modeling communities, but often involves either unspoken sources of epistemic and aleatory error, or high computation costs (or both). To begin addressing these issues, Part I of this thesis presents a guide and review of current practices, some of their pitfalls, and how common statistical tools apply to the network recovery problem: a *Practitioner's Guide to Network Recovery*. We will cover what "measurement" means in our context, and specifically the ways we encode observations, operations, and uncertainties numerically. Clarifying what different versions of what "relation" means (whether proximity or incidence) is critical, since network structure is intended to encode such relations as mathematical objects (), despite common ambiguities and confusion around what practitioners intend on communicating through them). Then we organize a literature review to present a cohesive framework for assessing network recovery techniques, based on the assumptions and compromises being made to make the network

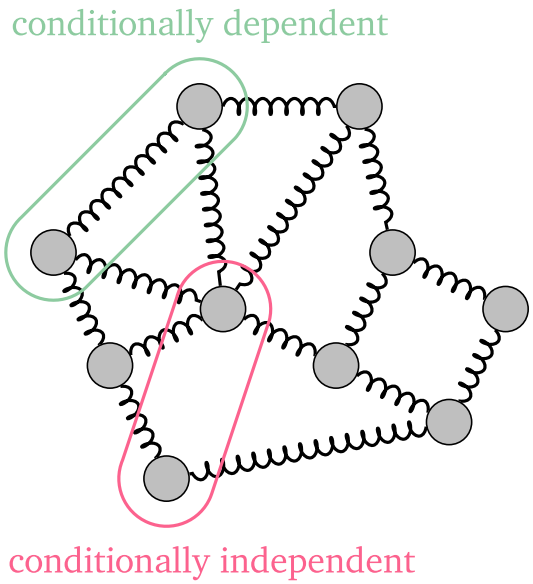


Figure 2.1: Spring system as a network of conditional dependencies

through the set **S**. Plus, it would necessitate averaging of weights over different edge ID's to arrive at a single undirected "edge weight", and many other implementation details that make keeping track of specifics difficult for practitioners.

Instead, we would like a canonical oriented distance matrix, which can be derived from the vectorized incidences in the undirected range of B (the standard basis vectors). Without loss of generality, let $u_e, v_e \in V_e$ be nodes such that $u < v$.[7] Using this, we can unambiguously define a *partition* $B(e,\cdot) = B(e,u_e) + B(e,v_e)$ between incidences on e, along with a new derived incidence, B_o, which has oriented rows like:

$$B_o(e,\cdot) = \mathbf{b}_e^o = \delta_e(u) - \delta_e(v)$$

In other words, while the unoriented incidence matrix is the "foundational" representation for graphs in our formalism, the (canonical) oriented one can be derived, even if negative incidence values are not in $\mathbb{S}$.[8]

But, now that we have removed the information on "which nodes an edge connects" from our definition of edges (since every edge is a scalar ID), how do we construct V_e without a circular dependency on B to find non-zero entries? Because of our unique identification of edges up to the combinatoric limit, we can still actually provide a unique ordering of the nodes in V_e, without searching over the entirety of B's domain. Using an identity from **ParallelEuclideandistance_Angeletti2019**, we have a closed-form equation both to retrieve the IDs of nodes u, v (given an edge e), and an edge e (given two nodes u, v), for

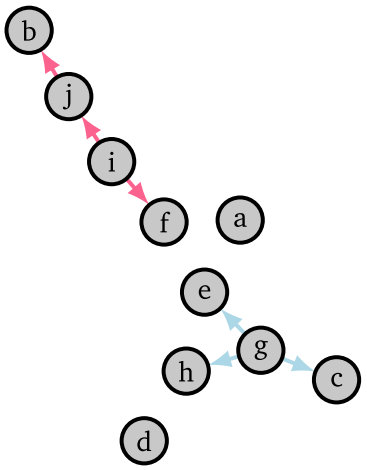
[7] the inequality is strict because self-loops are not allowed.

[8] This works as long as we are in at least a ring, since semirings in general do not need to define additive inverse operations. In this case we would limit ourselves to the oriented incidence.

This representation formalizes what practitioners call "edgelists" into a data structure that can unambiguously distinguish directed, undirected, and weighted graphs. In addition, it allows for repeated measurements of edges over the same set of nodes, while flexibly growing when new nodes arrive.[10] We are now able to encode the kinds of observations our hypothetical social scientist would be making of author collaboration interactions *as vectors*, shown in Figure 3.2.

$$R^{n\times\omega} = \begin{array}{c} \\ x_1 \\ x_2 \\ \vdots \end{array} \begin{array}{c} \begin{array}{ccccccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 & e_8 & e_9 & e_{10} & e_{11} \end{array} \\ \begin{bmatrix} . & . & . & . & 1 & . & . & . & 1 & 1 & . \\ . & . & . & 1 & . & . & 1 & 1 & . & . & . \\ & & & & \vdots & & & & & & \end{bmatrix} \end{array}$$

(a) : Observations embedded in "edge space"



(b) (hypothetical) edge-based observations

Figure 3.2: : Possible edge-based embedding of observations.

3.2 Node activation, bipartite graphs, and hypergraphs

What if an incidence structure allows for more than two incidences for the "line" set? In our binary design matrix, we might consider each observation its own "line", such that it is incident to all the activated nodes. This is no longer a graph of edges and nodes, but rather

[10]For instance, say observations are stored as sparse entries via R, and a new node arrives. First, the participating nodes can be recovered in a vectorized manner through Equation 3.6. Then, a new node id increases n, followed by reassignment of the edge IDs with $e_n(u, v)$.

a more general object.

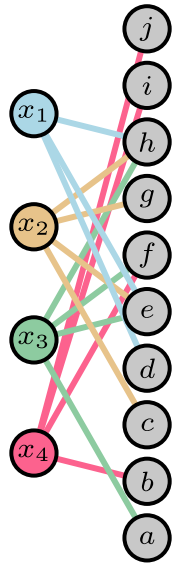
Every incidence structure can be seen as an incidence matrix, which additionally can be thought of as a *bipartite* graph. In this sense, the incidence matrix is thought of as a bi-adjacency matrix, which is a subset of a larger adjacency having two off-diagonal non-zero blocks

$$A_{BP} = \begin{pmatrix} 0_{n,n} & X^T \\ X & 0_{m,m} \end{pmatrix}$$

The graph having this adjacency structure has two sets of nodes that do not intraconnect (ergo, "bipartite"). The resulting structure for our toy example is shown next to the incidence matrix in Figure 3.3.

$$X^{m\times n} = \begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ \vdots \\ x_m \end{array} \begin{array}{c} \begin{array}{cccccccccc} a & b & c & d & e & f & g & h & i & j \end{array} \\ \left[\begin{array}{cccccccccc} . & . & 1 & . & 1 & . & 1 & 1 & . & . \\ 1 & . & . & . & 1 & 1 & . & . & . & . \\ . & 1 & . & . & . & 1 & . & . & 1 & 1 \\ . & . & . & 1 & 1 & . & . & 1 & . & . \\ & & & & \vdots & & & & & \\ & & & & & & & & & \end{array}\right] \end{array}$$

(a) : X as a (~~bipartite~~bi-adjacency) ~~incidence~~ matrix.



(b) Bipartite representation of node "activation" data

Figure 3.3: : Bipartite representation of binary design matrix

When the set of lines is a family of subsets of points, incidences on points and lines form a *hypergraph*. Hypergraphs are usually thought of as graphs where edges can connect more than two nodes, which again can be made into an incidence structure in the same way our graphs are. This isomorphism lets many of the familiar ideas on graphs (*e.g.*, walks, paths,

product operation, and have definitions in terms of *contractions* along the data/observation dimension. By relying on the (Euclidean) inner product, even with various re-weighting or normalization schemes, an analyst is making strong assumptions about their ability to reliably take measurements from linear combinations of observed activation vectors.

Essentially, if a measure relies on marginal counts or summation over the data axis (**s**), then the main assumptions are at the *local* level, about whether what we are adding together estimates our target correctly. The most basic would be to count co-occurrences, and consequently the co-occurrence probability $p_{11} = P(A, B)$. However, for very rare co-occurrences, we need to correct for rate-imbalance of the nodes in much the same way correlation normalizes covariance. This idea leads to "cosine similarity"

Note 1: Ochiai Coefficient (Cosine)

Effectively an uncentered correlation, but for binary observations the "cosine similarity" is also called the *Ochiai Coefficient* between two sets A, B, where binary "1" stands for an element belonging to the set.[**Measuresecologicalassociation_Janson1981**] In our use case, it is measured as

$$\frac{|A \cap B|}{\sqrt{|A|\,|B|}} = \sqrt{p_{1\bullet} p_{\bullet 1}} \to \frac{X^T X}{\sqrt{\mathbf{s}_i \mathbf{s}_i^T}} \quad \mathbf{s}_i = \sum_i \mathbf{x}_i$$

This interpretation of cosine similarity as the geometric mean of conditional probabilities is particularly useful when trying to approximate interaction rates. The geometric mean as a pooling operator is conserved through Bayesian updates [**ProbabilityAggregationMethods_Allard2012**], so the use of a prior with co-occurrences as base counts is possible for additive smoothing. To do this, the goemetric mean of marginal counts acts as a "psedovariable" for exposure somewhere between A and B. Empirically, this is a powerful approximation with good performance characteristics, for relatively little effort.

Note 2: Odds Ratios

Along with others derived from it, the Odds ratio is based on the ratio of conditional probabilities, and takes the form

$$\mathrm{OR} = \frac{p_{11}p_{00}}{p_{01}p_{10}}$$

Yule's Q and Y [**MethodsMeasuringAssociation_Yule1912**] are mobius transforms of the (inverse) OR and $\sqrt{\mathrm{OR}}$, respectively, that map association values to $[-1, 1]$.

$$Q = \frac{\mathrm{OR}-1}{\mathrm{OR}+1} \quad Y = \frac{\sqrt{\mathrm{OR}}-1}{\sqrt{\mathrm{OR}}+1}$$

Odds ratio is important to logistic regression, where the coefficients are usually the log-odds ratios of occurrence vs. not ($\log \mathrm{OR}$).

Yule's Y, also called the "coefficient of colligation", tends to scale with association strength in an intuitive way, so that proximity to 1 or -1 paints a more useful picture than the odds-ratio alone.

Another association measure, based in information theory, asks "how much can I learn about one variable by observing another?"

Note 3: Mutual information

An estimate for the mutual information (*i.e.*, between the sample distributions) can be derived from the marginals, as above, though it is more compactly represented as a pairwise sum over the domains of each distribution being compared:

$$\mathrm{MI}(A, B) \approx \sum_{i,j \in [0,1]} p_{ij} \log\left(\frac{p_{ij}}{p_{i\bullet}p_{\bullet j}}\right)$$

It is non-negative, with 0 occurring when A and B are independent. There are many other information-theoretic measures related to MI, but we specifically bring this up as it

will be the basis for the Chow Liu method, later on.

Sometimes, especially in social networks, we might want to avoid overcounting relationships with very well-connected nodes. This was brought up with respect to the normalized Laplacian before, but we could also perform a normalization on the underlying bipartite adjacencies.

Note 4: Hyperbolic Projection

Attempts to account for the overcounting of co-occurrences on frequently occurring nodes, vs. rarer ones.[**Scientificcollaborationnetworks._Newman2001**]

$$X^T \text{diag}(1+\mathbf{s}_j)^{-1} X \quad \mathbf{s}_j = \sum_j \mathbf{x}'_j$$

This ~~reweights~~ re-weights each observation by its degree in the original ~~biartite~~ bipartite graph.

So far this is the first measure that re-weights observations *before* contraction, so that it depends on having the individual observations available (rather than only the gram matrix). In this case, each observation's entries are all equally re-weighted by the number of activations in it (each nodes "activation fraction" in that observation).

4.1.2 Resource and Information Flow

These methods are somewhere between local and global constraint scales. This is accomplished by imagining nodes as having some amount of a resource (like information or energy) and correcting for observed noise in edge activation by reinforcing the *geodesics* that most likely transmitted that resource.

First, closely related to hyperbolic re-weighting, we can imagine the bipartite connections as evenly dividing each nodes' resources, before reallocating them to the nodes they touch, in turn. For instance, we might say each author splits their time among all of the papers they

are on, and in turn every co-author "receives" an evenly divided proportion of everyone's time they are co-authoring with.

Note 5: Resource Allocation

Goes one step further than hyperbolic projection, by viewing each node as having some "amount" of a resource to spend, which gets re-allocated by observational unit. [**Bipartitenetworkprojection_Zhou2007**]

$$\text{diag}(\mathbf{s}_i)^{-1} X^T \text{diag}(\mathbf{s}_j)^{-1} X$$

Interestingly, we could see this as a two-step random-walk normalization of the bipartite adjacency matrix. First X is row-normalized, then column-normalized. The final matrix is asymmetric, so a symmetric edge strength estimate is often retrieved by mean, max, or min reduction operations.

Rather than stop after two iterations, continuing to enforce unit marginals to convergence is known as the Sinkhorn-Knopp algorithm, which converges to a doubly-stochastic matrix (both marginal directions sum to 1).

Note 6: Doubly Stochastic

If $A \in \mathbb{S}^{n \times n}$ is positive, then there exists d_1, d_2 such that

$$W = \text{diag}(d_1) A \text{diag}(d_2)$$

is doubly-stochastic, and $W(u, v)$ is the *optimal transport plan* between u and v with regularized Euclidean distance between them on a graph.[**RobustInferenceManifold_Landa2023**, **Sinkhorndistanceslightspeed_Cuturi2013**]

The doubly-stochastic filter [**twostagealgorithm_Slater2009**] removes edges from W

until just before the graph would become disconnected.

As the name implies, the optimal transport plan reflects the minimum cost to move some amount of resource from one node to another. By focusing on best-case cost, we enforce a kind of "principle of least action" to bias recovery toward edges along these geodesics.

A more direct way to do this, perhaps, is to find the shortest paths from every node to each other node, and aggregate them.

Note 7: High-Salience Skeleton

Count the number of shortest-path trees an edge participates in, out of all the shortest-path-trees (one for every node).

$$\frac{1}{n}\sum_{i=1}^{n}\mathcal{T}_{\text{sp}}(i)$$

where $\mathcal{T}_{\text{sp}}(n)$ is the shortest-path tree rooted at n [**Robustclassificationsalient_Grady2012**]

Unfortunately, HSS is forced to scale with the number of nodes, and must calculate the entire spanning tree for each one.

4.1.3 Global Structural Assumptions

Often times these constraints are as simple as "the underlying dependency graph must belong to a family $\mathcal{C}$" of graphs. Observations are thought of as emissions from a set of node distributions, where edges are representations of dependency relationship between them. To provide a foundation to formalize this notion, one framework is that of Markov Random Fields, which are undirected generalizations of bayes nets [**Markovrandomfields_Kindermann1980**] that use edges to encode *conditional dependence* between node distributions.

One of the original structures for MRFs that we could recover from observed data was a *tree*.

Note 8: Chow-Liu Spanning Tree

Enforces tree structure globally. Approximates a joint probability

$$P\left(\bigcap_{i=1}^{n} v_i\right) \approx P' = \prod_{e \in T} P(u_n(e) \mid v_n(e)) \quad T \in \mathcal{T}$$

where $\mathcal{T}$ is the set of spanning trees for the nodes. The Chow-Liu tree minimizes the Kullback-Leibler (KL) Divergence $\mathrm{KL}(P \| P')$ by finding the minimum spanning tree over pairwise mutual information weights.[**Approximatingdiscreteprobability_Chow1968**]

Recent work has made it possible to enforce spanning tree structure while efficiently performing monte-carlo-style bayesian inference, which estimates a distribution over spanning trees that explain observed behavior, and by extension the likelihood each edge is in one of these trees. [**BayesianSpanningTree_Duan2021**]

If instead we imagine our MRF as being made up of individual Gaussian emissions, then the overall network will be a multivariate gaussian with pairwise dependencies along the edges. In fact, as a consequence of the Hammersley–Clifford theorem, the conditionally independent variables are *precisely* the set of zero entries in the the precision (inverse-covariance) matrix Θ of the multivariate model. Exploiting this fact leads to a semidifinite program to minimize the frobenius-norm of Θ with the sample covariance $\|\hat{\Sigma}\Theta\|_F{}^2$

Note 9: GLASSO

Semidefinite program to find (regularized) maximum likelihood precision of graph-structured multivariate Normal distribution using ℓ_1 ("LASSO") penalty [**Sparseinversecovariance_Friedman2008**]

[2]Since the sample covariance will not give an unbiased estimate for precision, these problems often require significant regularization. This class of problems is called "covariance shrinkage", though we more specifically care about *precision shrinkage* as illustrated in Note 9.

incredible computational efficiency. [**ReconstructingNetworksUnknown_Peixoto2018**, **NetworkReconstructionCommunity_Peixoto2019**]

4.2 A Path Forward

In addition to the categories above, there is a second "axis" that practitioners should keep in mind when selecting their recovery algorithm of choice. Each of the above listed techniques can be mostly separated into two categories, based on whether hypergraphic/bipartite observations are assumed to be in *data space*, or in *model space*.

4.2.1 ~~Observation space~~ Data assumptions

Recall from Section 2.2: an operator takes our model parameters and maps them to data space. The implication for inverse problems is a need to *remove the effect* of the operator, because we cannot directly observe phenomena in a way compatible with our model (*e.g.*, which might model underlying causal effects).

This is a core point of view in **Statisticalinferencelinks_Peel2022**, where those authors describe nearly all of network analysis as *inferring hidden structure*:

> "Here we argue that the most appropriate stance to take is to frame network analysis as a problem of inference, where the actual network abstraction is hidden from view, and needs to be reconstructed given indirect data."
>
> – Peel et al., [**Statisticalinferencelinks_Peel2022**]

We note that this isn't strictly true, *assuming* that the "network" is intended to represent something measured by the direct observation. For instance, if a network is intended to represent a discretization of distances (such as a k-nearest neighbors approximation) for computational efficiency. The co-occurrence measures can be thought of as estimators of node-node distances, especially with appropriate smoothing to remove zero-valued distances from undersampling.[5] In otherwords, if an analyst wishes to discretize distances

[5] See Section 6.2.2 for an elaboration of this connection via the "forest kernel".

as incidences in a complex network, they are effectively using "high-pass" filter to remove low-similarity entries, which is an effective way to assess community structure—exactly like clustering for continuous data.[6] In fact, for an example of this exact network-as-discretization idea being used for state-of-the-art clustering performance, see HDBSCAN in **HybridApproachHierarchical_Malzer2020**.

Because it is difficult to know a priori what a domain will require of network analysts, our main recommendation is for algorithm creators to transparently describe their technique's data-space assumption:

- are observations already in *model space*, perhaps with with alleatoric noise to be removed?, or,
- are they in *data space* and require solving some form of inverse problem to recover a model specification?

Once again from the **Statisticalinferencelinks_Peel2022** review:

> Surprisingly, the development of theory and domain-specific applications often occur in isolation, risking an effective disconnect between theoretical and methodological advances and the way network science is employed in practice.
>
> – Peel et al. [**Statisticalinferencelinks_Peel2022**]

In a similar vein, we believe that a large amount of metrological inconsistency and struggle has at its heart a communication and technology transfer problem, which standardization and community toolkit support can hopefully work toward fixing.

4.2.2 ~~Filling the local+data "gap"~~

With this in mind, we show in Table 4.1 an overview of the covered approaches, and whether the method presumes operation on observations in the same space as the *model*, or if some inverse problem is needed.

[6] Or, at a slight risk of reductionism, drawing a world atlas with two colors for "above and below sea-level": useful simplification for rapid assessment of shapes.

Table 4.1: Organizing recovery methods by representation space and level

(a)

observations in...	**Data** space?	**Model** space?	**Bipartite**?
assumptions for...			
Local Structure			
Ochiai Coeff.		•	
Hyperbolic Proj.		•	•
Mutual Info.		•	
capability "gap"	•		•
Information Flow			
Resource Allocation		•	•
Doubly Stochastic	∼	•	
High-Salience Skeleton	•		
Global Structure			
Chow Liu	•		
MRF/GLASSO	•		∼
Deg. Sequence		•	•
S.B.M.	•		•

To add to the point, the last column in Table 4.1 shows whether the full bipartite representation is even needed to perform the technique, or if it is possible with the gram matrix or marginal values alone.[7]

~~In the~~

4.2.2 Model vs Estimation Approach

Because network reconstruction can be such a computationally intense problem, special consideration must be paid to not only the model, but to the *approach* taken to estimate the model. Many of the methods described conflate the two, since models are often constructed *because* of the estimation approaches they enable. It is worth distinguishing the two ideas before moving to the next chapter: a *model* is a conceptual paradigm for what a recovered network *represents*, while an *approach* is a mechanism to estimate that model's parameters.

For instance, GLASSO is often thought of as a "model", and this is somewhat true, but the model is a combination of using the Markov Random Field assumption with a definition of "goodness" based on minimizing the Frobenius norm between an estimate and our observations. The *approach* is to add a regularization term (L1), and pick one of many techniques to minimize the total loss function, *e.g.* coordinate-descent (CD). The model makes assumptions about global properties (MRF) of the graph, while the approach determines the computational complexity and can allow for certain types of information to be utilized. Coordinate descent operates node-wise, meaning our complexity will scale with network size, but can also allow data to be split along the observation dimension to allow bootstrapping techniques like Stability Selection [**StabilitySelection_Meinshausen2010**].

Recent advances have focused on generalizing *approaches* to improve performance

[7] note that any of these could make use of the bipartite "design matrix", *e.g.*, to estimate the edge support with *stability selection* [**StabilitySelection_Meinshausen2010**] by subsampling it multiple times and repeating the algorithm accordingly.

of techniques like CD. Greedy coordinate-descent (GCD) has been proposed as a way to achieve sub-quadratic convergence time (in network-size), by utilizing NN-descent to approximate a set of edge candidates at each iteration [**Scalablenetworkreconstruction_Peixoto2024**]. In a follow-up work, a generic approach to applying GCD to arbitraty network probability model used the *minimum description length* principle to overcome overfitting issues in L_1-based regularization schemes like GLASSO [**Networkreconstructionvia_Peixoto2024**]. While still limited by super-linear scaling in network size, the performance and accuracy gains using this approach are *considerable*, once an appropriate modeling framework is selected to apply it to (*e.g.* SBM or Ising).

4.2.3 Filling the "gap"

In the next sections, we focus on filling a gap for models that only make local assumptions and preserve additivity, but ~~*assumes*~~ *assume that data is not represented directly in the network model-space*~~.~~ (*i.e.* data-space assumption). First, we need a modeling framework for network recovery that retains additivity in a principled way. Much like the role that nonparametric estimators like KDE/Nadarya-Watson play in regression, or Kaplan-Meier estimators in survival analysis ~~,~~[**Topicsadvancedeconometrics_Bierens1996**, **NonparametricEstimationIncomplete_Kaplan1958**], additive models only make assumptions about local structure. But from these assumptions, they can provide critical insight into data and its structure, and push analysts to make regular "sanity checks" when results or assumptions conflict. Chapter 5 will cover *Desire Path Densities*, our modeling framework for additive, data-space model specifications.

Next, an *approach* to estimating model parameters is needed. By exploiting a common class of domain-informed constraints—namely, node activation _via spreading process—we can build an algorithm for network reconstruction that scales linearly in observation count, while remaining approximately *constant* in network size. This will be covered in Chapter 6.

A reader can analyze general problems with failing to specify a model of what "edges" actually *are* more in-depth in **Statisticalinferencelinks_Peel2022**. They also include a warning not to naively use correlational measures with a threshold, since even simple 3-node systems will easily yield false positives edges. Still, it would be helpful for practitioners to have a more explicit mental model of *why* co-occurence-based models yield systematic bias, and use that to build an alternative having some of the same benefits (speed, interpretability, uncertainty quantification, etc.)

5.1.2 Inner-Product projections and "clique bias"

Underlying correlation and co-occurrence models for edge strength is a reliance on matrices of inner products between node occurrence vectors. They all use gram matrices (or centered/scaled versions of them), which were brought up in Section 2.4. The matrix multiplication performed represents inner products between all pairs of feature vectors. For $X(i,j) \in \mathbb{B}$, these inner products sum together the times in each observation that two nodes were activated together.

However, another (equivalent) way to view matrix multiplication is as a sum of outer products

$$G(j,j') = X^T X = \sum_{i=1}^{m} X(i,j)X(i,j') = \sum_{i=1}^{m} \mathbf{x}_i \mathbf{x}_i^T$$

Those outer products of binary vectors create $m \times m$ matrices that have a 1 in every ~~i,j~~ j,j' entry where nodes ~~i,j~~ j,j' both occurred, shown in Figure 5.1. Each outer product is effectively operating as a $D_i + A_i$ with degrees normalized to 1. If the off-diagonals can be seen as adjacency matrices, they would strictly represent a clique on nodes activated in the ith observation In this sense, any method that involves transforming or re-weighting a gram matrix, is implicitly believing that the ~~kth observation was~~ ith observation is a *complete graph* for all i. This is illustrated in Figure 5.2.

If every observation of node activations leads to an implied clique, we can reframe much of the "hairball" effect as a systematic bias (i.e. measurement error in the sense of_trueness_).

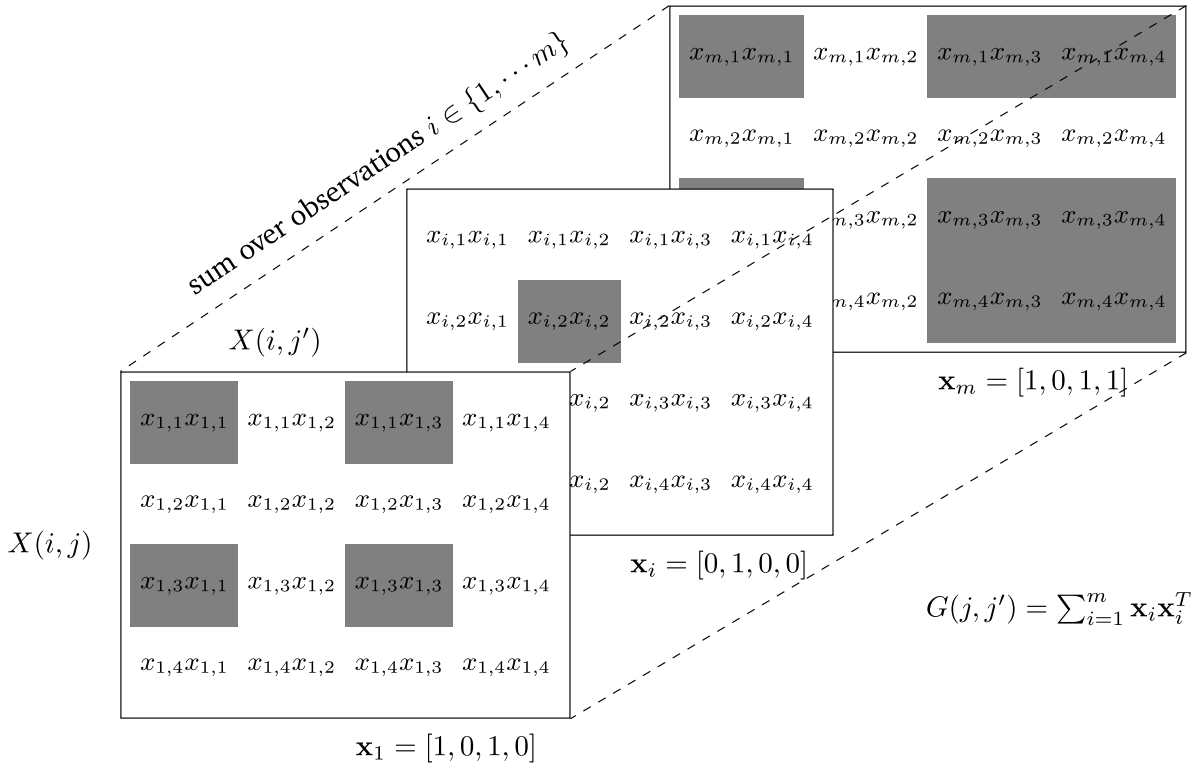


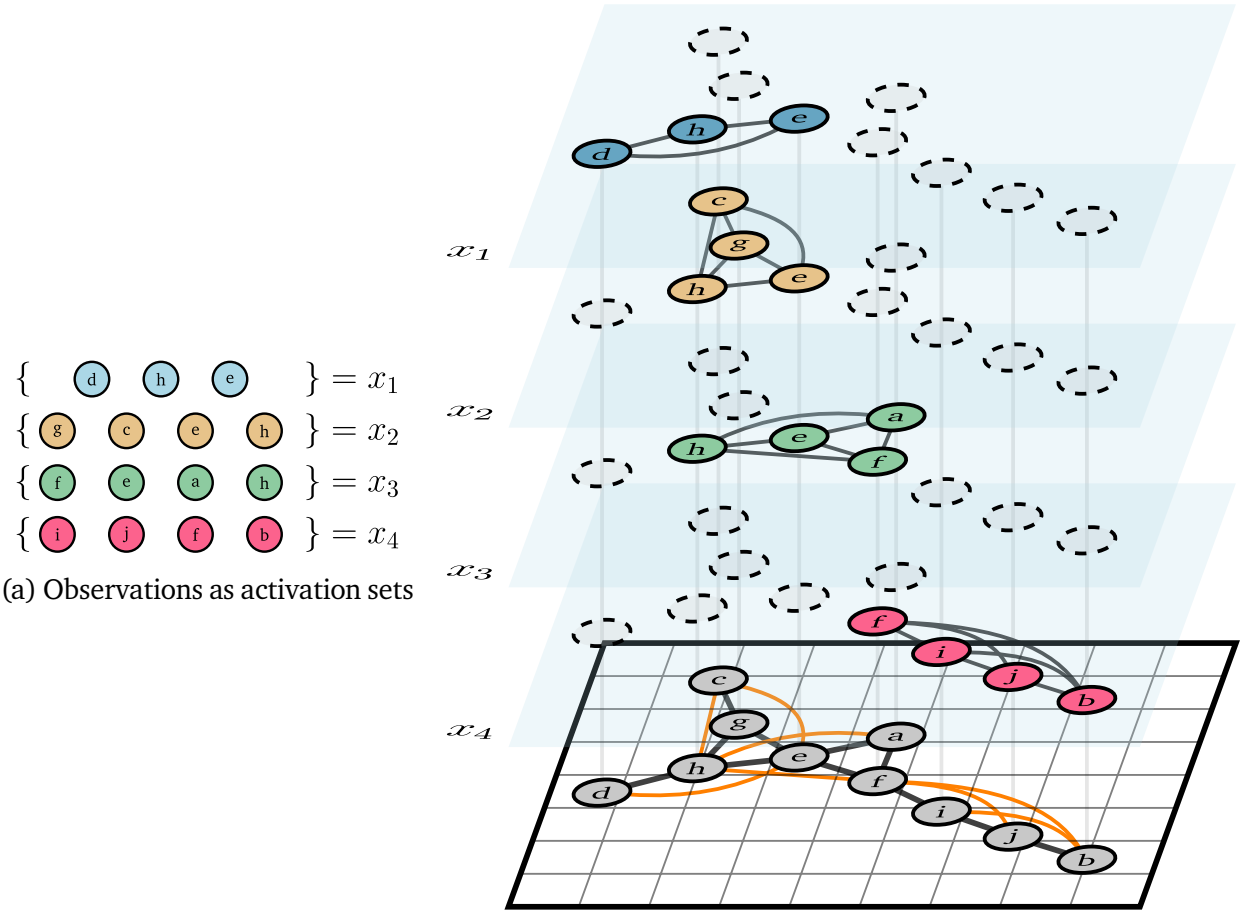
Figure 5.1: Gram matrix as sum of observation outer products

We call this *clique bias*: the inferred graph will inherently include more and more cliques of node subsets as data arrives (assumed to themselves be cliques).

For many modeling scenarios, this paradigm allows practitioners to make a more straightforward intuition-check: do clique observations *make sense* here? When a list of authors for a paper is finished, does that imply that all authors mutually interacted with all others directly to equally arrive at the decision to publish? This would be similar to assuming the authors might simultaneously enter the same room, look at a number of others (who all look exclusively at each other, as well), and *all at once* decide to publish together.

Or, from the standpoint of scaling: does each extra node activation impart an amount of information that depends on the number of activated nodes? Put another way, if we knew our observations were on a ~~"grid graph"~~planar graph, each node might require ~~two (or so)~~ around 3 new edges.[2] A tree or path adds one new edge for each new node. But a clique

[2] Triangulations are worst-case, so $|E| \leq 3|V| - 6$ due to Euler's formula



(a) Observations as activation sets

(b) Edge Measurements with Group Gambit (BoW) assumption

Figure 5.2: Inner-product projections as sums of cliques illustrating “clique bias”.

assumption means that each extra node activation adds edges quadratically in the number of already-activated nodes. Does this make sense? In our introduction, we described a more likely scenario we could expect from an observer on the ground: a researcher asks a colleague or two to collaborate, who might know a couple more with relevant expertise, and so on. From purely a logistical standpoint, it quickly becomes unfeasible for authors to mutually collaborate with all other co-authors equally: 10 coauthors already implies the 10th had to equally split interaction among 9 others to satisfy our model.

5.2 Networks as Desire Path Density Estimates

Unfortunately, methods based on inner-product thresholding are still incredibly common, in no small part due to how *easy* it is to create them from occurrence data, regardless of this "clique-bias". The ability to map an operation onto every observation, *e.g.*, in parallel, and then reduce all the observations into an aggregate edge estimate is a highly desirable algorithmic trait. This may be a reason so many projection and backboning techniques attempt to re-weight (but retain) the same basic structure, time and again.

What we need is a way to maintain the ease-of-use of inner-product network creation:

- ~~map~~ Map an operation onto each observation
- ~~reduce~~ Reduce to an aggregate edge guess over all observations

but with a more domain-appropriate operator at the observation level.

Let's start with replacements for the clique assumption. There are many non-clique classes of graphs we might believe local interactions occur on: path-graphs, trees, or any number of graphs that reflect the topolgy or mechanism of local interactions in our domain of interest. Authors have proposed classes of graphs that mirror human perception of set shapes [**Relativeneighborhoodgraphs_Jaromczyk1992**][3], or graphs whose modeled

[3] *e.g.*, for dependencies based on perception, such as human decision making tendencies, or causes based on color names.

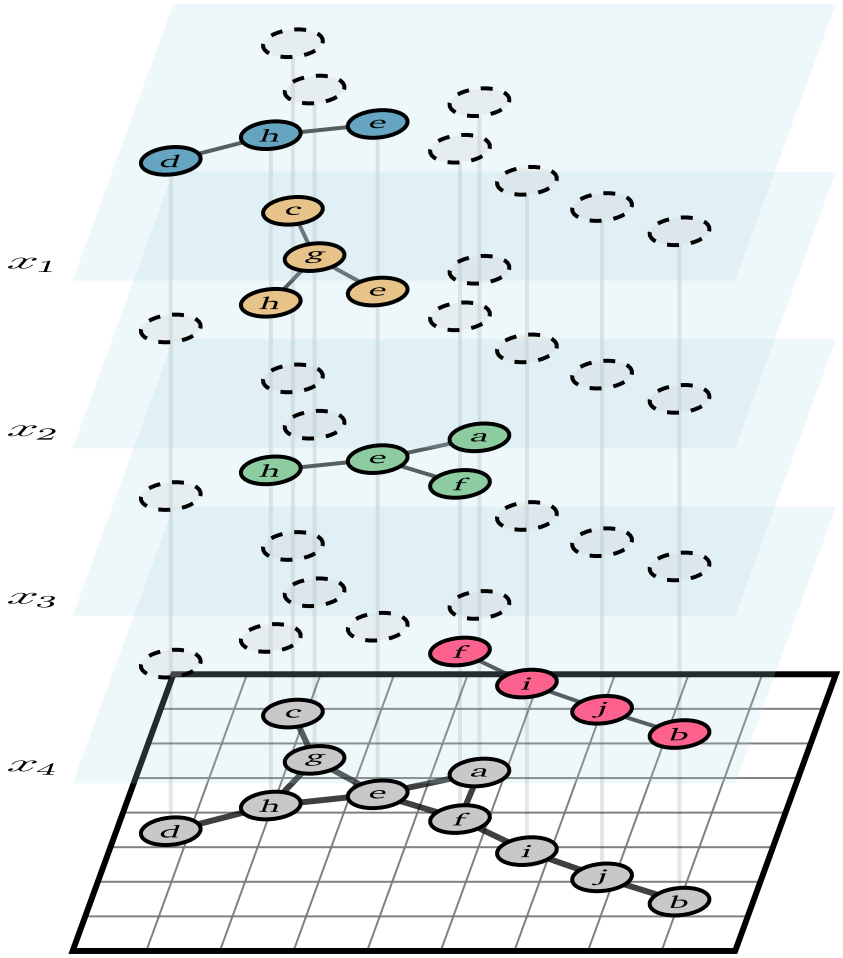


Figure 6.1: Edge Measurements with true (tree) dependencies known

Representing a vector as a sparse combination of a known set of vectors (also known as "atoms") is called *sparse approximation*.

6.2.1 Problem Specification

Sparse approximation of a vector $\mathbf{x}$ as a representation $\mathbf{r}$ using a dictionary of atoms (columns of D) is specified more concretely as [**EfficientimplementationK_Rubinstein2008**]:

$$\hat{\mathbf{r}} = \underset{\mathbf{r}}{\operatorname{argmin}} \|\mathbf{x} - D\mathbf{r}\|_2^2 \quad \text{s.t.} \|\mathbf{r}\|_0 \leq N \tag{6.2}$$

where N serves as a sparsity constraint (at most N non-zero entries). This is known to be NP-hard, though a number of efficient methods to approximate a solution are well-studies and widely used. Solving the Lagrangian form of Equation 6.2, with an ℓ_1-norm in place of ℓ_0, is known as *Basis Pursuit*[**SparseApproximateSolutions_Natarajan1995**], while greedily solving for the non-zeros of $\mathbf{r}$ one-at-a-time is called *matching pursuit*[**Matchingpursuitstime_Mallat1993**]. In that work, each iteration selects the atom with the largest inner product $\langle \mathbf{d}_{i'}, \mathbf{x} \rangle$.

We take an approach similar to this, but with the insight that the inner product will not result in desired sparsity (namely, a tree). Our dictionary in this case will be the set of edges given by B (see Section 5.2.1), while our sparsity is given by the relationship of the numbers of nodes and edges in a tree:

$$\hat{\mathbf{r}} = \underset{\mathbf{r}}{\operatorname{argmin}} \|\mathbf{x} - B^T\mathbf{r}\|_2^2 \quad \text{s.t.} \quad \|\mathbf{r}\|_0 = \|\mathbf{x}\|_0 - 1 \tag{6.3}$$

There are some oddities to take into account here. As a linear operator (see Section 2.2), B^T takes a vector of edges to node-space, counting the number of edges each node was incident to. This means that, even with a ground-truth set of interactions, B^T would take them to a new matrix $X_{\text{deg}}(i,j) : I \times J \to \mathbb{N}$, which has entries of the number of interactions each individual in observation i was involved in. While very useful for downstream analysis

- The total number of nodes is typically a *given* for a single problem setting
- In many domains, the basic *spreading* rate of diffusion model (*e.g.*, R_0, or heat conductivity), does not scale with the total size of an observation

That last point means that constant scaling with network size is generally down to the domain in question. For instance, a heat equation simulated over a small area, having a given conductivity, will not have a different conductivity over a larger area; conductivity is a material property. Similarly, a virus might have a particular basic reproduction rate, or a set of authors might have a static distribution over how many collaborators they wish to work with. The former is down to viral load generation, and the latter a sociological limit: a bigger department usually does not imply more authors-per-paper by itself.

Similar to Equation 6.5, we might reasonably assume that the expected degree of nodes is roughly constant with network size *i.e.*, an inherent property of the domain. So, the density of activation vectors (as a fraction of all possible edges) is going go scale with the inverse of n. This makes our process, which is linear in activation count, out to be *constant* $O(1)$ in network size. Then, if $\bar{s}$ is the expected non-zero count of each row of X, the final approximate complexity of FP is $O(m\bar{s})$.[7]

6.4 Simulation Study

To test the performance of FP against other backboning and recovery methods, we have developed a public repository `affinis` containing reference implementations for FP, along with many co-occurrence and backboning techniques. The library contains source code and examples for many of the presented methods, and more. ~~UPDATE w/ DOI?~~

In addition, to support the community and provide for a standard set of benchmarks for network recovery from activations, the `MENDR` reference dataset and testbench was developed. To make reproducible comparison of recovery algorithms easier, `MENDR` includes hundreds of

[7] In our reference implementation, which uses Kruskal's algorithm, the theoretical complexity is likewise $O(m\bar{s}^2 \log \bar{s})$, though in our experience the values of $\bar{s}$ are small enough to not impact the runtime significantly.

randomly generated networks in several classes, along with random walks sampled *on those networks*. It can also be extended through community contribution, using data versioning to allow consistent comparison between different reports and publications over time.

6.4.1 Experimental Method

For each algorithm shown in Table 6.1, every combination of the parameters in Table 6.2 was tested. 30 random graphs for each of nodes were tested, which was repeated again for each of three separate kinds of global graph structure. Every algorithm that could be supplied a prior via additive smoothing is shown in Table 6.1 as "α?: Yes", and a minimum-connected (tree) sparsity prior was supplied $\alpha = \frac{2}{n}$. The others, esp. GLASSO, do not have a $\frac{\text{count}}{\text{exposure}}$ form, and could not be easily interpreted in a way that allowed for additive smoothing. However, since the regularizaiton parameter for GLASSO is often critical for finding good solutions, a 5-fold cross validation was performed for each experiment to select a "best" value, with the final result run using that value. While this does have a constant-time penalty for each experiment, the reconstruction accuracy is significantly improved with this technique, and would reflect common practice in using GLASSO for this reason.

Table 6.1: Summary of algorithms compared

Algorithm	abbrev.	α?	class	source
Forest Pursuit	FP	Yes	~~hybrid~~ local	-
GLASSO	GL	No	~~MRF~~ global	[**Sparseinversecovarian** **Structureestimationdis**
Ochiai Coef.	CS	Yes	~~Counts~~ local	[**Measuresecologicalass**
Hyperbolic Projection	HYP	No	~~Counts~~ local	[**Scientificcollaboration**

Algorithm	abbrev.	α?	class	source
Doubly-Stochastic	eOT	Yes	~~Transport~~ resource	[**twostagealgorithm_Sla** **Sinkhorndistanceslights**
High-Salience Skeleton	HSS	Yes	~~Transport~~ resource	[**Robustclassificationsal**
Resource Allocation	RP	Yes	~~Transport~~ resource	[**Bipartitenetworkproje**

The three classes of random graphs represent common use cases in sparse graph recovery. In addition, the block and tree graphs are types we expect GLASSO to correctly recover in this binary setting.[**Structureestimationdiscrete_Loh2012**] The block graphs of size n were formed by taking the line-graph of randomly generated trees of size $n+1$.
Trees were randomly generated using Prüfer sequences as impelmented in NetworkX [**ExploringNetworkStructure_Hagberg2008**]. To simulate possible social networks and other complex systems that show evidence of preferential attachment, scale-free graphs were sampled through the Barabási–Albert (BA) model, which was randomly seeded with a re-attachment parameter $m \in \{1,2\}$[**EmergenceScalingRandom_Barabasi1999**].

Table 6.2: Experiment Settings (`MENDR` Dataset)

parameters	values
random graph **kind**	Tree, Block, BA($m \in \{1,2\}$)
network n-**nodes**	10,30,100,300
random **walks**	1 sample $m \sim \text{NegBinomial}(2, \frac{1}{n}) + 10$
random walk **jumps**	1 sample $j \sim \text{Geometric}(\frac{1}{n}) + 5$

parameters	values
random walk **root**	1 sample $n_0 \sim \text{Multinomial}(\mathbf{n}, 1)$
random **seed**	1, 2, … , 30

Every graph has a static ID provided by MENDR, along with generation and retrieval code for public review. New graphs kinds and sizes are simple to add for future benchmarking capability.

6.4.2 Metrics

To compare each algorithm consistently, several performance measures have been included in the MENDR testbench. They are all functions of the True Positive/Negative (TP/TN) and False Positive/Negative (FP/FN) values.

Note 10: Precision (P)

Fraction of positive predictions that are true, also called "positive predictive value" (PPV)

$$P = \frac{TP}{TP + FP}$$

Note 11: Recall (R)

Fraction of true values that were returned as positive. Also called the TP-rate (TPR), and has an inherent trade-off with precision.

$$R = \frac{TP}{TP + FN}$$

Note 12: Matthews Correlation Coefficient (MCC)

Balances all of TP,TN,FP,FN. Preferred for class-imbalanced problems (like sparse recovery) [**statisticalcomparisonMatthews_Chicco2023**]

$$\frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$$

Note 13: Fowlkes-Mallows (F-M)

~~:~~ Geometric mean of Precision and Recall, as opposed to the F-Measure that returns the harmonic mean. Also known to be the limit of the MCC as TN approaches infinity[**MCCapproachesgeometric_Crall2023**], which is useful as TN grows with n^2 but TP only with n.

$$\sqrt{P \cdot R}$$

Because this work is focused on unsupervised performance, specifically for the use of these algorithms by analysts investigating dependencies, we opt to calculate TP,TN,FP,FN at every unique edge probability/strength value returned by each algorithm. Then, because we do not know a priori which threshold level will be selected by an analyst in the unsupervised setting ~~, we~~ we need a mechanism to select from or aggregate these values to come up with an overall score. In the supervised case we would have access to the ground truth, so the "optimal" threshold can be reported. Though this is not a reliable unsupervised scenario, it can give an idea of the upper-limit for an algorithm's performance. For this purpose, we report the maximum MCC value as "MCC-max".

Another common approach was discussed in Note 6, where we find the maximum allowable edge sparsity before the graph would otherwise become disconnected. This balances the desire to achieve sparsity while also enforcing topological constraints on the graph, so that we cannot improve our precision artificially by isolating components of the overall network. This "minumum-connected" threshold was calculated for each network

estimate, and reported for MCC as "MCC-min".

Finally, a more interesting approach might be to define a distribution over likely thresholds, and find the expected value over, say, MCC, which incorporates our uncertainty in thresholding. In the future this could be informed by domain-specific thresholding tendencies, but for now we will take a conservative approach and report the expected values E[MCC] and E[F-M] over all unique threshold values (*i.e.* a flat prior over thresholds). To consistently compare the expected values, we transform the thresholds for every experiment to the range $[\epsilon, 1-\epsilon]$, to avoid division-by-0 at the extremes.

Another common approach ~~is to report~~ to score aggregation is the Average Precision Score (APS). This is not the average precision over the thresholds however, but instead the expected precision over the possible recall values achievable by the algorithm. It is approximating the integral under the parametric P-R curve, instead of the thresholds themselves.

$$\text{APS} = \sum_{e=1}^{\omega} P(e)(R(e) - R(e-1))$$

where $P(e)$ and $R(e)$ are the precision and recall at the threshold set by the edge e, in rank-order. This is more commonly done for supervised settings, however, and will report a high value as long as *any* threshold is able to return a both a high precision and a high recall, simultaneously.

6.4.3 Results - Scoring

The ~~results over every experiment are~~ median results, along with the inter-quartile-range (IQR), are summarized accross all experiments in Table 6.3.

A visualization of these results are shown in Figure 6.2, with a specific callout to compare E[MCC], MCC-min, and MCC-max in Figure 6.3. Only FP is able to report MCC and F-M values with medians over about 0.5, regularly reaching over 0.8. GLASSO is clearly the second-best at recovery in these experiments, though for scale-free networks the improve-

Table 6.3: Comparing median(IQR) scores for various metrics

(a)

metric method	E[MCC]	E[F-M]	APS	MCC-max	MCC-min
FP	**0.73 (0.19)**	**0.76 (0.15)**	0.83 (0.21)	**0.86 (0.17)**	**0.75 (0.25)**
GL	0.47 (0.15)	0.50 (0.13)	**0.90 (0.13)**	0.85 (0.12)	0.72 (0.75)
CS	0.38 (0.16)	0.45 (0.22)	0.79 (0.21)	0.74 (0.14)	0.59 (0.20)
HYP	0.34 (0.17)	0.40 (0.24)	0.59 (0.34)	0.55 (0.25)	0.35 (0.25)
eOT	0.28 (0.16)	0.34 (0.25)	0.47 (0.39)	0.49 (0.25)	0.39 (0.32)
HSS	0.23 (0.22)	0.27 (0.30)	0.33 (0.36)	0.50 (0.19)	0.25 (0.35)
RP	0.22 (0.19)	0.28 (0.30)	0.27 (0.43)	0.44 (0.31)	0.36 (0.31)

ment over simply thresholding the Ochiai coefficient is negligible. For APS, both GLASSO and Ochiai are equally able to return high scores, indicating at least one threshold for each that performed well. ~~A~~ However, the best-case MCC-max for FP is still marginally better than GLASSO, along with the MCC-min having a better median score *as well as* a much tighter uncertainty range.

To address the APS discrepancy, a simple mechanism for FP to perform equally well ~~at APS~~ is discussed in Section 7.1.

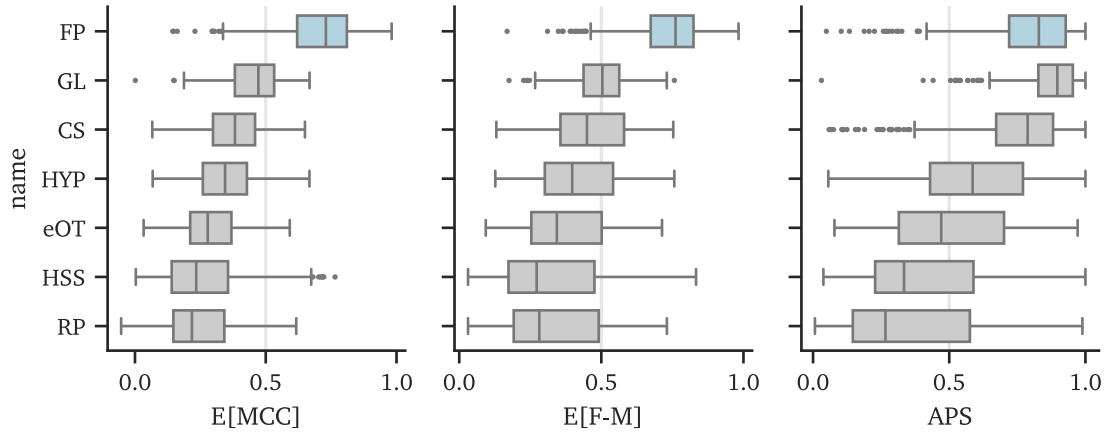


Figure 6.2: Comparison of MENDR recovery scores

Breaking down the results by graph kind in Figure 6.4, we see the remarkable ability of FP to dramatically outperform every other algorithm in MCC and F-M, showing remarkable accuracy *together with stability* over the set of threshold values. This is indicative of FP's

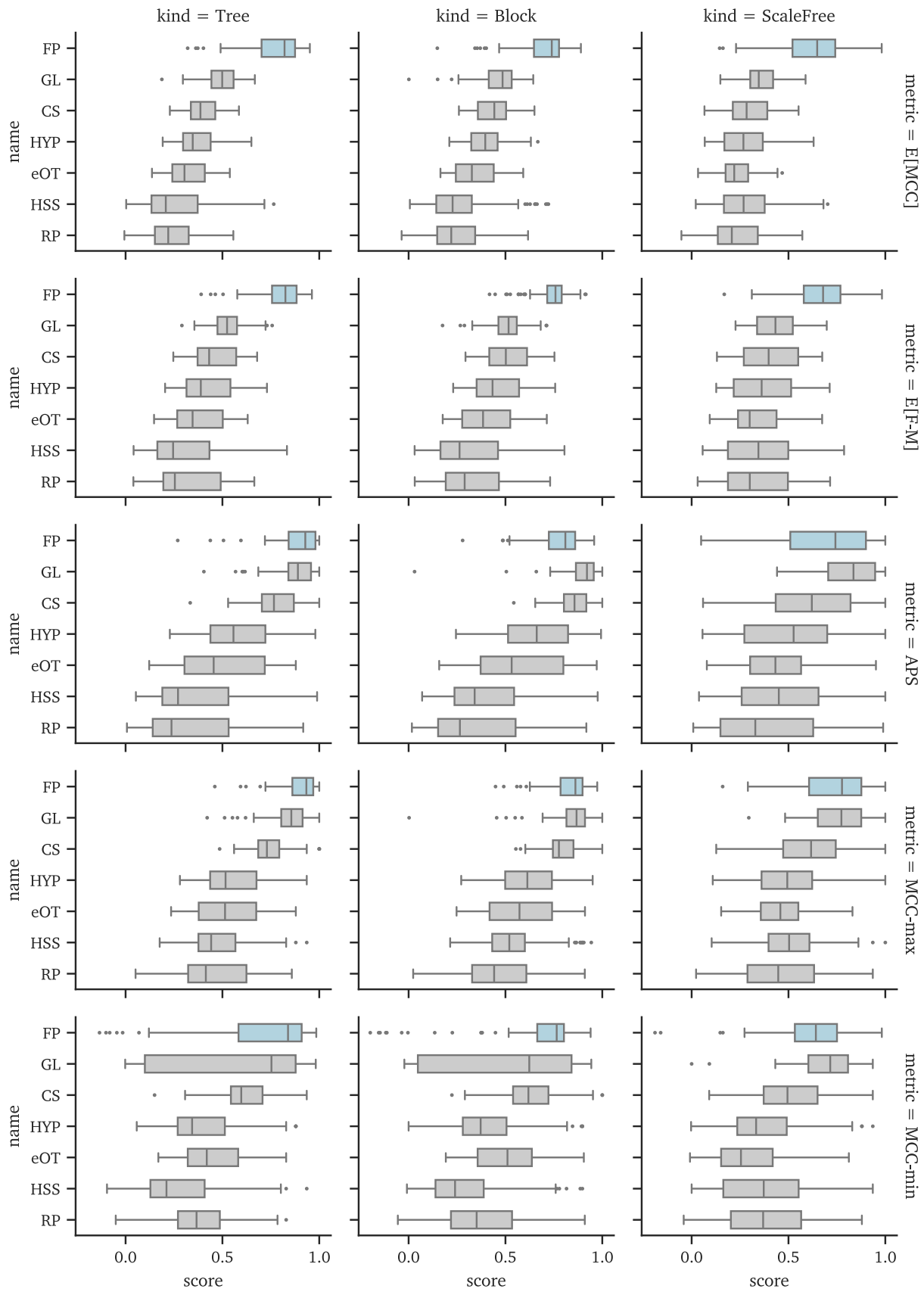
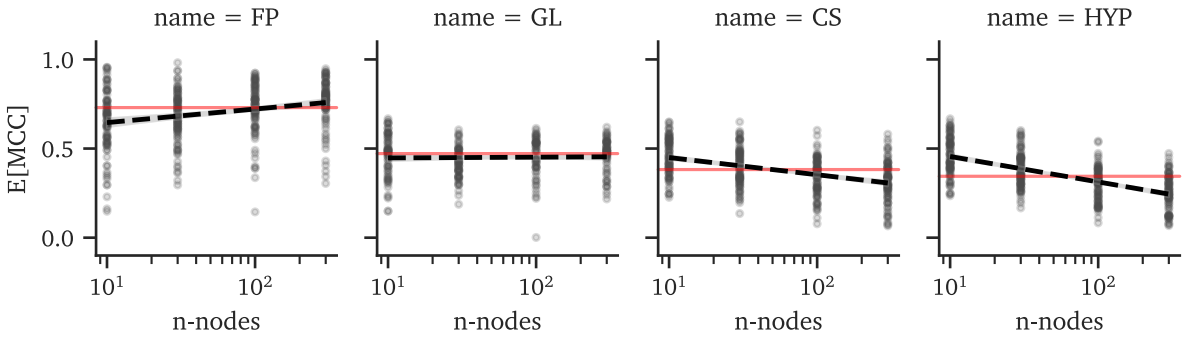
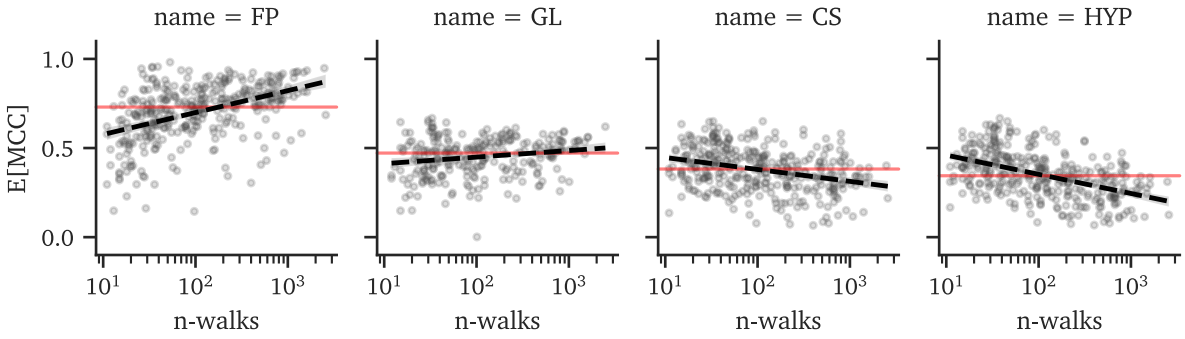


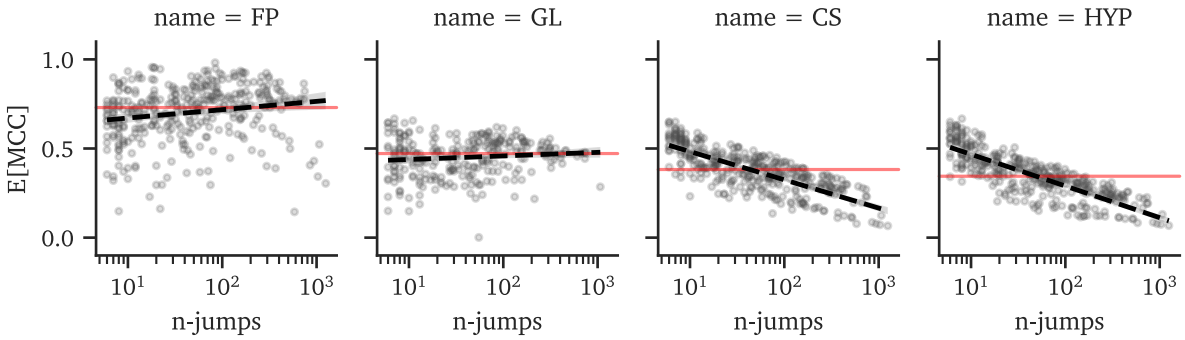
Figure 6.4: Comparison of MENDR Recovery Scores by Graph Type



(a) Trend: MCC vs network size



(b) Trend: MCC vs observation count



(c) Trend: MCC vs random-walk length

Figure 6.5: Score trends vs problem scaling

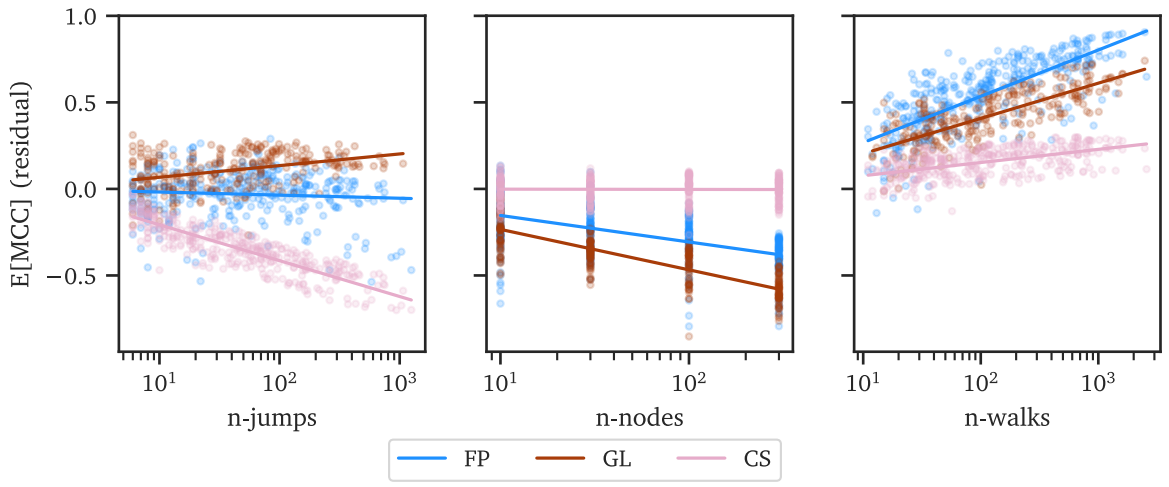


Figure 6.6: Partial Residuals (regression on E[MCC])

Interestingly, CS is largely unaffected by network size, compared to FP and GL, though GL performs the worst in this regard. However, it is in the random-walk length that we see the benefit of dependency-based algorithms. The Ochiai coefficient suffers dramatically as more nodes are activated by the spreading process, since this means the implied clique size grows by the square of the number of activations. FP remains unaffected by walk-length, while (impressively) GLASSO appears to have a marginal boost in performance when walk lengths are high.

6.4.4 Results - Runtime Performance

For both Forest Pursuit and GLASSO, runtime efficiency is critical if these algorithms are going to be adopted by analysts for backboning and recovery. Figure 6.7 shows the (log-)seconds against the same parameters from before. For similar sized networks, FP is consistently taking 10-100x less time to reach a result than GLASSO does. Additionally, many of the experiments led to ill-conditioned matrices that failed to converge for GLASSO under any of the regularization parameters tests (the "x" markers in Figure 6.7). As expected, the number of observations plot shows a clear limit in terms of controlling the lower-bound of FPs runtime, since in this serial version every observation runs one more call to MST. On

the other hand, GLASSO appears to have ~~signiicant~~ significant banding for walk length and observation counts, likely indicating dominance of network size for its runtime.

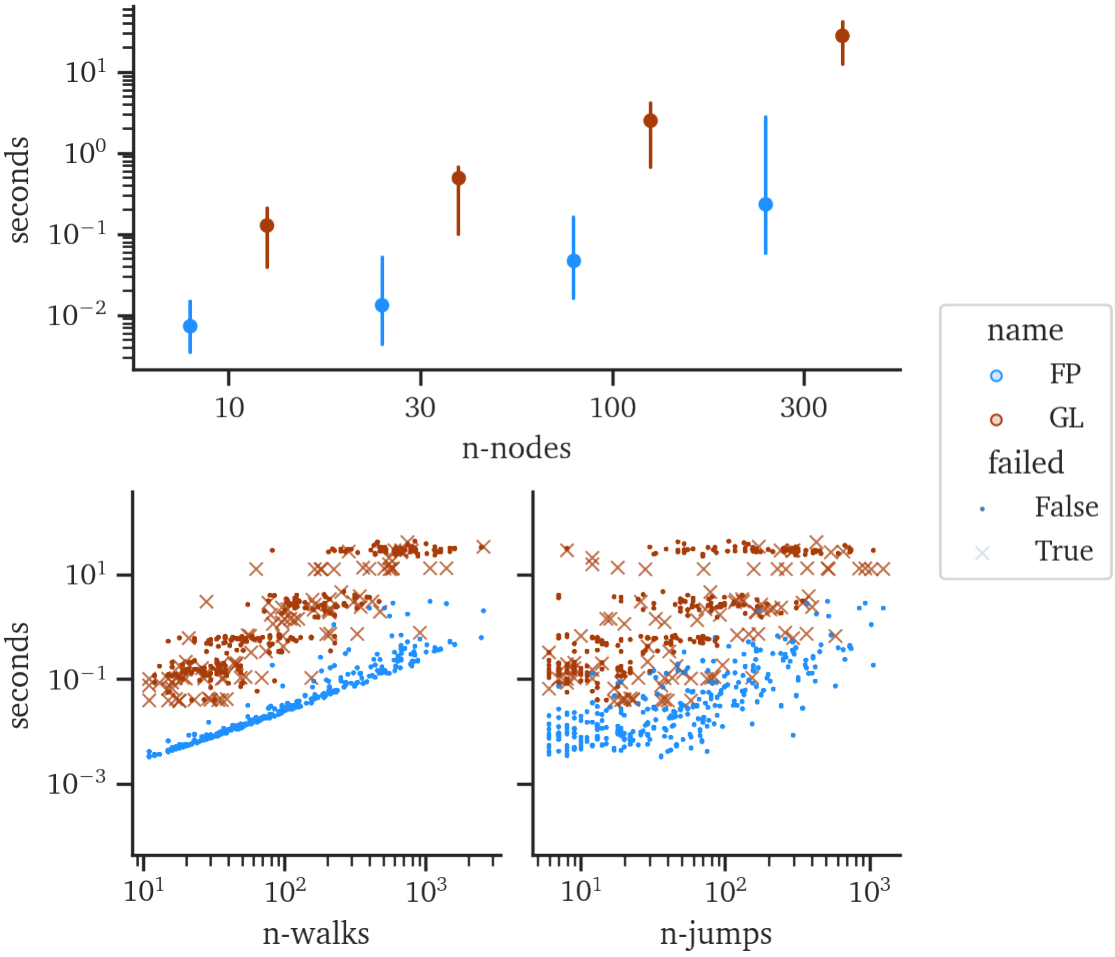


Figure 6.7: Runtime Scaling (Forest-Pursuit vs GLASSO)

To control for each of the variables, and to empirically validate the theoretical analysis in Section 6.3.2}, a regression of the same three (log-)parametwers was performed against (log-)seconds. The slopes in Figure 6.8, which are plotted on a log-log scale, correspond roughly to polynomial powers in linear scale. In regression terms, we are fitting the log of

$$y_{\text{sec}} = a x_{\text{param}}^{\gamma}$$

so that the slope in a log-log plot is γ.

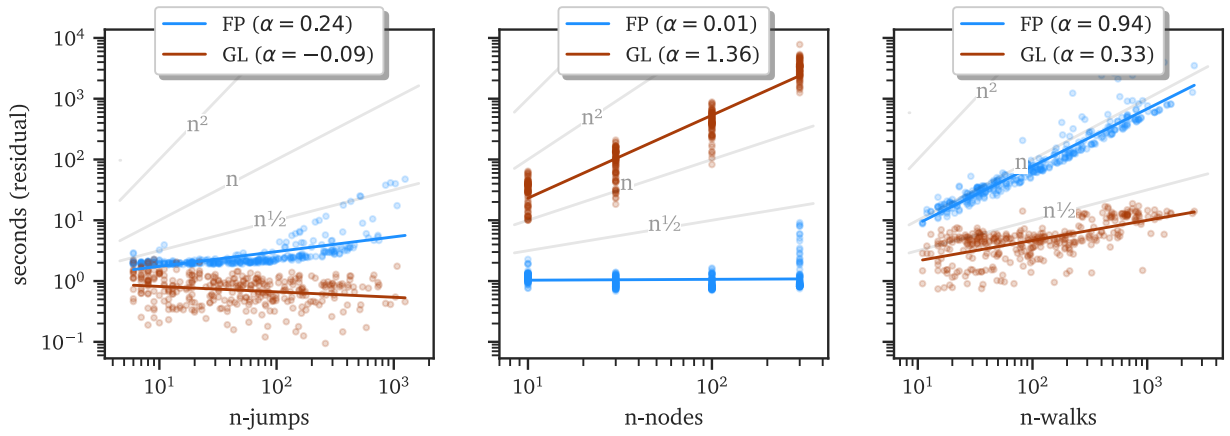


Figure 6.8: Partial Residuals (regression on computation time)

In a very close match to our previous analysis, the scaling of FP is almost entirely explained by the observation count and random-walk length, alone: the coefficient on network size shows constant-time scaling. Similarly, the scaling with observation count is very nearly linear time, as predicted. The residuals show non-linear behavior for the random-walk length parameter, which would make sense, due to the theoretical $||E|| \log ||V||$ scaling of Kruskal's algorithm. At this scale, $n \log n$ and $n^2 \log n$ complexity might appear smaller than linear time, due to the log factor. GLASSO hardly scales with random walk length, and only marginally with observation count. In typical GLASSO, the observation count has already been collapsed to calculate the empirical covariance matrix, so its effects here might be due instead to the cross-validation and the need to calculate empirical covariance for observation subsets. The big difference, however, is GLASSO scaling in significantly superlinear time—almost $O(n^2)$. This is usually the limiting factor for analyst use of such an algorithm in network analysis more generally.

7.1 Forest Pursuit Interaction Probability

Without using stability selection[**StabilitySelection_Meinshausen2010**], GLASSO is not directly estimating the "support" of the edge matrix, but the strength of each edge. to do similar with FP, we could directly estimate the frequency of edge occurrence using $R(i,e)$ marginal averages, rather than conditioning on co-occurrence. Simply multiplying each FP edge probability by the co-occurrence probability of each node-node pair gives this as well, which we call FPi: the direct "interaction probability" for each pair of nodes.

7.1.1 Simulation Study Revisited

By doing this simple re-weighting, FPi actually beats GLASSO's median APS for the dataset, but at the cost of MCC and F-M scores (which both drop to between FP and GLASSO), as Figure 7.1 demonstrates. Similarly, the individual breakdown by graph kind in Table 7.1 shows a similar pattern, with FPi coming close to GLASSO for scale-free networks, but exceeding it for trees and matching for block graphs. Still, the difference is small enough, and at such a significant penaly to MCC and F-M scores over a variety of thresholds, that it is hard to recommend the FPi re-weighting unless rate-based edge analysis is desired, e.g. if Poisson or Exponential occurrence models are desired.

Consistent with the increased APS, however, we also see an improvement in MCC-min and MCC-max. If there is a domain-driven reason to select *specific* thresholds, or if GLASSO is known to provide reasonable results for a given problem, FPi stands a good chance of improving on the optimal or min-connected MCC score.

7.1.2 Simulation Case Study

To illustrate what is going on, we have selected two specific experiments as a case study, in Figure 7.2. In the first, `BL-N030S01`, a 30-node block graph with 53 random walk samples, has FP performing worse than GLASSO and Ochiai, in terms of APS (which is reported in

Table 7.1: Comparing median(IQR) scores ~~for FP,~~ against FPi~~and GLASSO~~

(a)

metric name	APS	E[F-M]	E[MCC]	MCC-max	MCC-min
FPi	**0.91 (0.15)**	0.56 (0.21)	0.54 (0.20)	**0.89 (0.15)**	**0.80 (0.19)**
FP	0.83 (0.21)	**0.76 (0.15)**	**0.73 (0.19)**	0.86 (0.17)	0.75 (0.25)
GL	0.90 (0.13)	0.50 (0.13)	0.47 (0.15)	0.85 (0.12)	0.72 (0.75)
CS	0.79 (0.21)	0.45 (0.22)	0.38 (0.16)	0.74 (0.14)	0.59 (0.20)

~~FP FPi GL Block APS 0.78 0.9 0.9 F-M 0.74 0.57 0.51 MCC 0.7 0.55 0.46 ScaleFree APS 0.69 0.76 0.81 F-M 0.67 0.46 0.44 MCC 0.63 0.43 0.36 Tree APS 0.9 0.92 0.88 F-M 0.81 0.66 0.53 MCC 0.78 0.65 0.49~

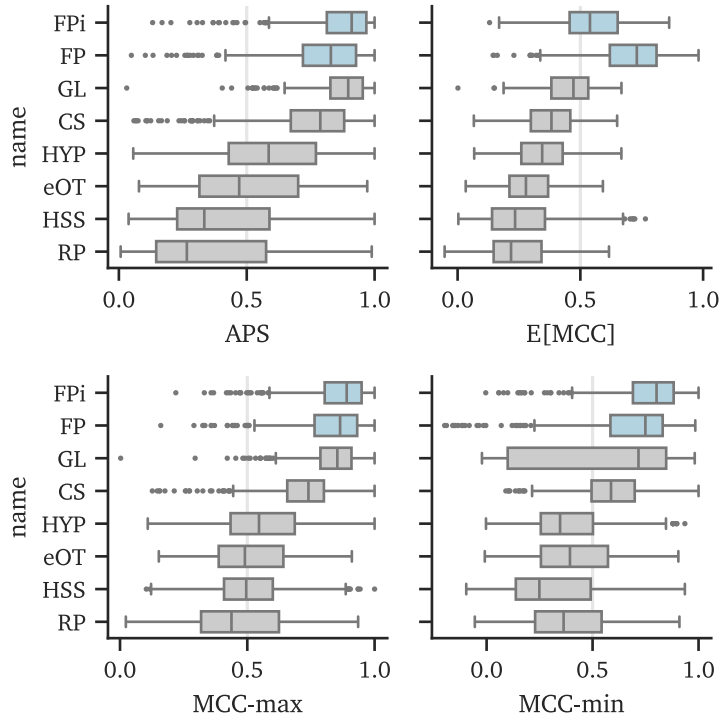


Figure 7.1: FPi shows ~~best~~ improved APS, ~~lower~~ optimal MCC, ~~F-M~~and MCC (min-connect)

the legends). We see that FP shows high precision, which drops off significantly to increase recall at all. Only a few edges had high probability (which is usually desirable for sparse approximation), and some of the true edges were missed this way. However, FPi rescaling makes rarer edges fall off earlier in the thresholding, letting the recall rise by dropping rare edges, rather than simply the low-confidence ones.

In the second, `SC-N300S01` is a 300-node scale-free network with 281 walks. Both FP and FPi show significantly better recovery capability, since enough walks have visited a variety of nodes to give FP better edge coverage. In this graph, no algorithm comes within 0.25 of FP's impressive 0.88 APS~~for~~ , especially with 300 nodes and fewer than that many walks.

7.2 Generative Model for Correlated Binary Data

~~Symmetry of marked directed and undirected trees (symmetry in Q)~~ Generating multivariate, *correlated* binary data is of interest across many of the fields discussed in this work, since such models can be used as foundations structural inference (especially if they have a defined likelihood function). One of the foundational generative models for correlated binary data makes direct use of the multivariate-normal's precision matrix buy sampling directly from it and thresholding values to be binary [**generationcorrelatedartificial_Leisch1998**]. In a related sense, the multivariate probit (MVP) is used to analyse such data, treating the likelihood of observed binary outcomes as probabilities from the cumulative distribution function of the underlying multivariate normal.

Another generative model is the *Ising Model*, which has long been used as a tool for modeling particle spins in lattice structures, but in the inverse setting is seeing increased application in data science for recovering network structure [**Inversestatisticalproblems_Nguyen2017**] While techniques like Gibbs sampling might be used to sample from the Ising distribution, a common technique to estimate the structure of the lattice is to perform logistic regression on each node's activation values

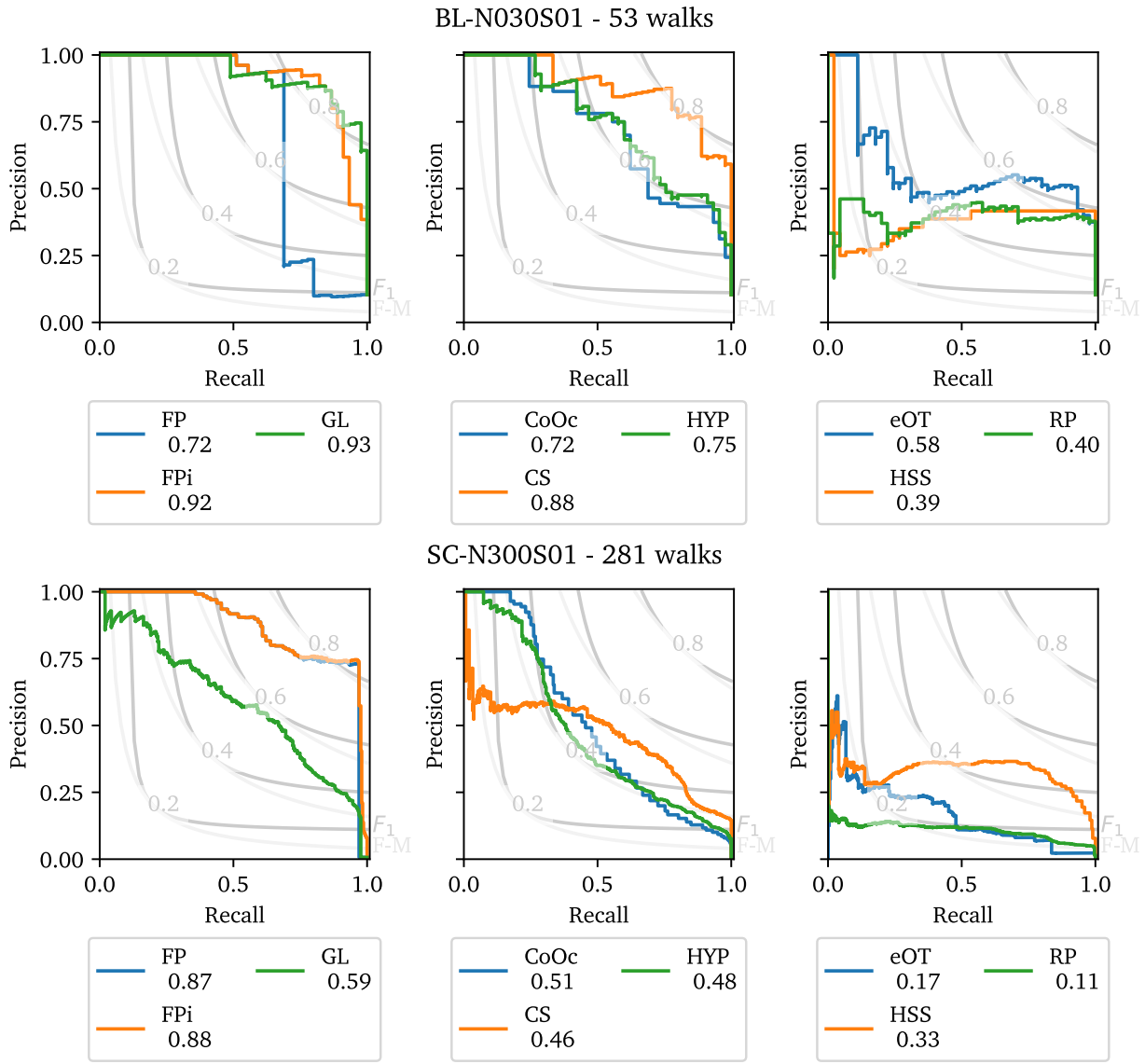


Figure 7.2: P-R curves for two experiments

using the activation of all other nodes. The equivalence of this Multivariate Logit (MVL) to the Ising model is discussed in **Assortmentoptimizationgiven_Vasilyev2025**

As was briefly mentioned in Chapter 4, another class of models makes use of the bipartite structures implied by the binary activation data. These degree sequence methods can synthesize other bipartite structures (and therefore, generate binary observations) that match a desired degree distribution [**fastballfastalgorithm_Godard2022**, **Randomlysamplingbipartite_**

If the case that the true number of nodes is unknown (or should be approximately inferred), a class of models known as the *Indian Buffet Process* [**IndianBuffetProcess_Griffiths2011**] is s distribution over all binary matrices with a finite number of rows. It can therefore sample entire datasets, like the degree sequence models, but with more flexibility in inferring the number of necessary nodes.

For all of these generative models, there is a lack of model that takes into account the underlying information available to us when we *know* that the activation structures arise from a *spreading process* (like random walk visits). Here we propose a compound distribution with a reasonable likelihood that can not only model and generate correlated binary outcomes, but also be inferred using the techniques first discussed for *Forest Pursuit*. We accomplish this by exploiting the isometry between distributions over random spanning forests and spanning tree distributions over an augmented graph. We are able to derive a simple likelihood and a rapid inference scheme using the Matrix Forest Theorem of Chebotarev and Shamis [**MatrixForestTheorem_Chebotarev2006**]

7.2.1 Marked Random Spanning Forest (RSFm) distribution

Every spanning forest on a graph described by Q can be thought of as an equivalent spanning tree over a graph augmented with an extra "source" node, which is connected to every other node with a weight $\frac{1}{\beta}$. Sampling random spanning trees on the augmented graph is equivalent to random spanning forests on the original.[**Semisupervisedlearning_Avrachenkov2017**] We can use this fact to create a distribution for node activation sets based on *co-occurrence*

on a rooted tree.

A "rooted tree" is a tree with a marked node. In Figure 7.3 we see this illustrated, where a randomly sampled tree on the graph augmented with R leads to many subtrees in the original graph. Marking one node (d) at random selects the tree that contains (d,h,e), which corresponds to record x_1 back in Figure 5.2a.

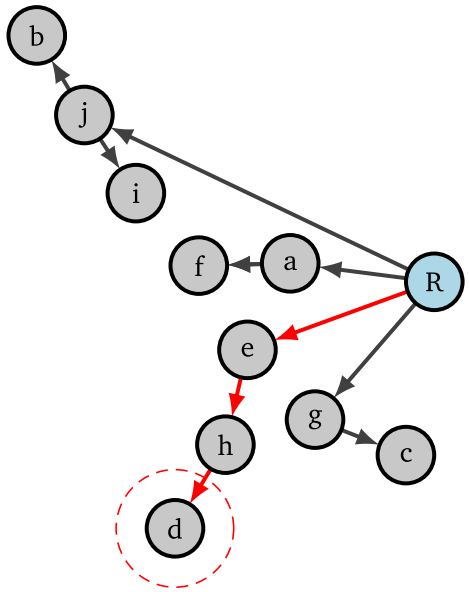


Figure 7.3: Dissemination plan as rooted RST on augmented graph

Note that the marked node does not necessarily need to be the one that the source "injected" to, since the observed activations set is equivalent for any of (d,h,e) being marked. This is an important symmetry when we will not know which node "actually" started each cascade, during inference.

It means that the node activation set and the graph structure are conditionally independent, given a sampled spanning tree on the augmented graph. Sampling efficiently from a spanning tree distribution is a well studied problem, and we can use that efficiency in combination with a node-marking (categorical) distribution to formulate an overall distribution for node activation.

Therefore, RSFm distribution models the probability of emitting node j in the i-th observation as the probability of occurring in the same tree as a marked "root" node ϕ_{ij},

Beginning with the FP point estimate, each edge in every spanning subtree can be efficiently resampled according to the *Bayesian Spanning Tree* distribution from **BayesianSpanningTree_Duan2021**. Once every edge in a tree has been resampled, the overall estimate for the desire path network can be updated, and sampling can continue. This would be very similar to the way collapsed Gibbs sampling works for Latent Dirichlet Allocation[**Latentdirichletallocation_Blei2003**], but with edges selected from a spanning forest distribution instead of "topics" from a multinomial. Derivations and implementation of such a scheme is left for future work.

7.3 Expected Forest Maximization

Another possibility is to approach the problem as a kind of matrix factorization, jointly estimating B and R in an alternating manner. Where *Forest Pursuit* was an empirical Bayes estimate for R, alternating from there between B and R leads to a simple Expectation Maximization scheme~~:~~.

~~*Alternate embedding the node activations as edge activations and combining these under a desire path model to estimate the graph structure.*~~

7.3.1 Factorization & Dictionary Learning

Jointly finding a sparse representation of data using a linear combination of basis vectors (called a "dictionary"), *and* finding an optimal dictionary with which to do the embedding, is called "sparse dictionary learning". One of the original proposed methods to solve this was the Method of Optimal Directions (MOD)[**Methodoptimaldirections_Engan1999**], which uses a sparse "pursuit"-like method to find the representation vectors, and then uses that representation to find $\hat{B} \leftarrow XR^{+}$.[2] However, our desire path estimation for independent (arcsine) Bernoulli edges gives an efficient workaround to needing the pseudo-inverse. Based

[2] R^{+} is the pseudo-inverse of R.

of inversions like this, because the global Laplacian has higher possible degrees than the subgraph, and we do not wish to bias Steiner tree estimation by node degree. Not doing so leads to rapid divergence of the loss function, as all MSTs will tend toward star-graphs around the highest-degree node.

7.3.2 EFM Simulation Study

One-shot Forest Pursuit appears to perform quite well, so it's useful to quantify the expected gain in performance by repeating it an unknown number of times. There are no generic guarantees for EM convergence, though anecdotally the number of iterations was limited in our experiments to under a thousand, and that limit was never hit while using a covergence parameter of $\epsilon = 1e-5$.

The distribution of E[MCC] score change vs. FP is shown in Figure 7.4.

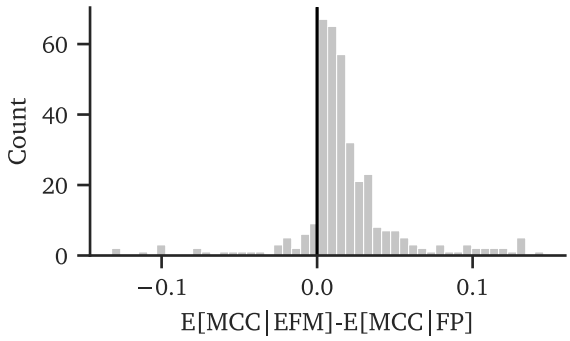


Figure 7.4: Change in Expected MCC (EFM vs FP)

While useful, it's not clear whether individual edges are more likely to be "true" edges, *given* a bigger change in EFM score. To test this, a logistic regression was performed for every experiment in MENDR against the true edge values, using the change in scores on those edges between FP and EFM as training data. To avoid overfitting, a significant amount of regularization was applied, chosen using 5-fold crossvalidation. The coefficients for all experiments are shown as a histogram in Figure 7.5.

The graph kind did not make a significant difference to EFM improvement, but overall

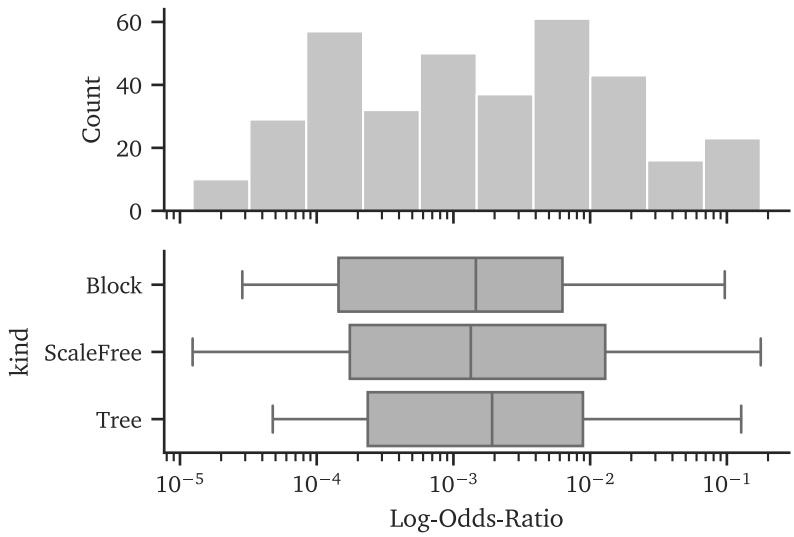


Figure 7.5: Logistic Regression Coef. (EFM - FP) vs. (Ground Truth)

log-odds improvement is very low.Still, the value is positive accross the entire dataset, so EFM does have a very small-but-nonzero impact on improving edge prediction.

The runtime graphs can also be updated, with EFM shown in Figure 7.6 and Figure 7.7 against FP and Glasso. EFM still ran significantly faster than GLASSO in this region. However, the scaling with network size is no longer constant-time, especially since convergence used above is the max-abs error, which requires that *every node* reach a minimum level of convergence and might take much longer, overall.

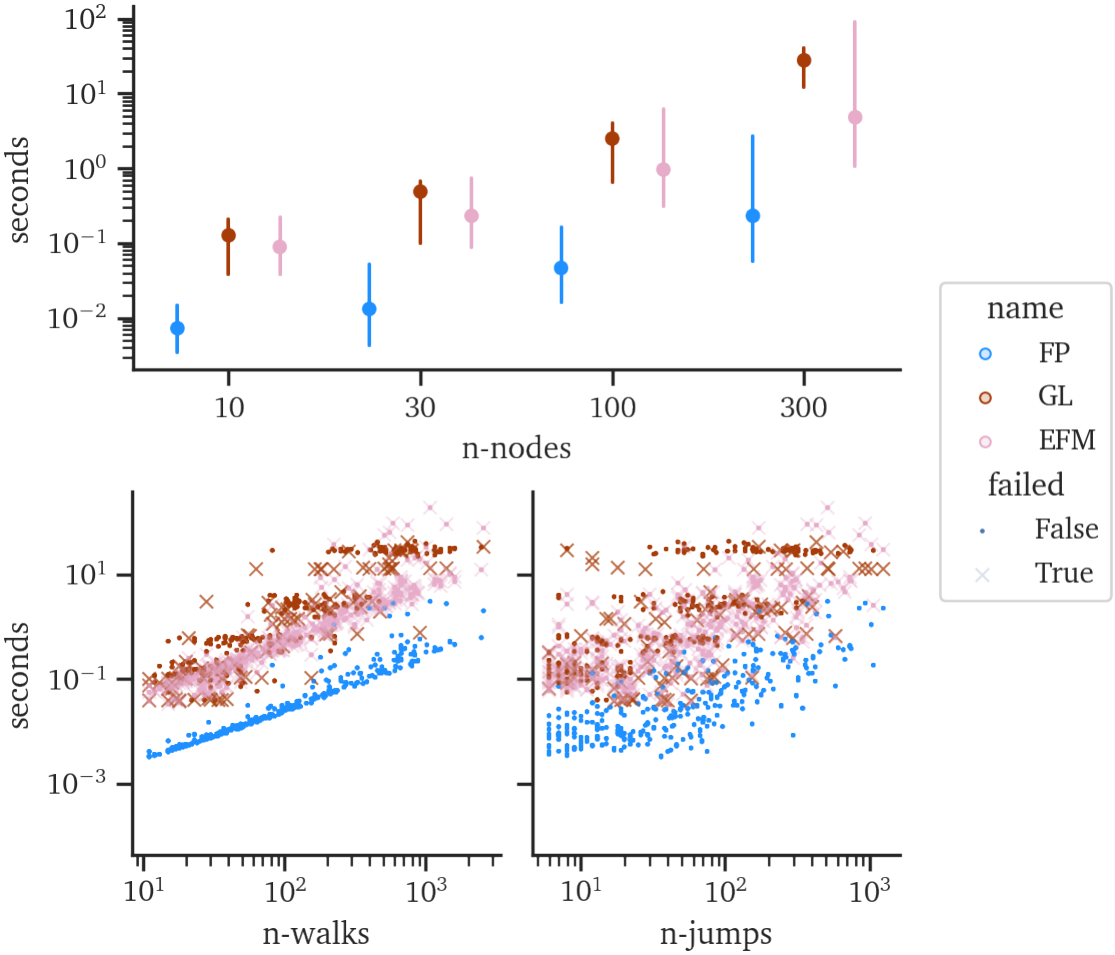


Figure 7.6: Runtime Scaling (Forest-Pursuit vs GLASSO)

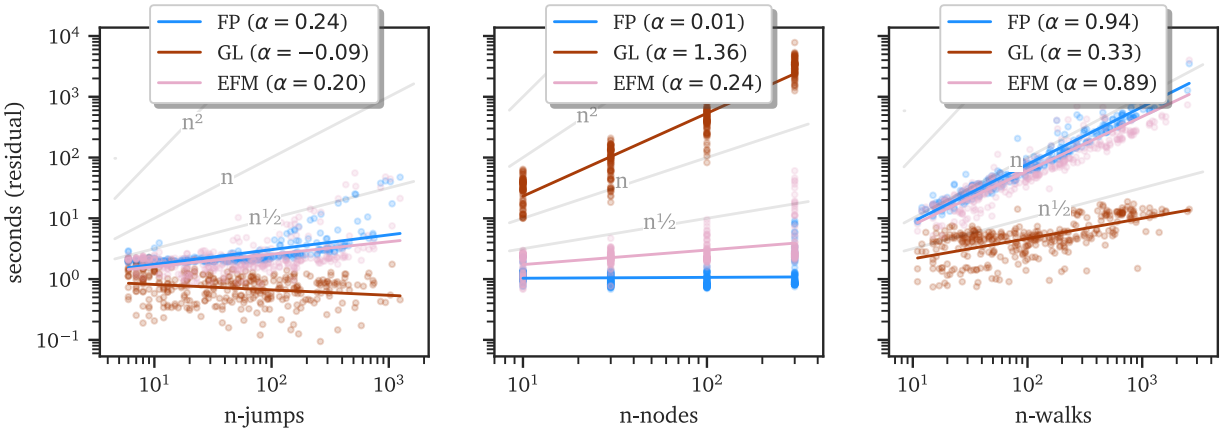


Figure 7.7: Partial Residuals (regression on computation time)

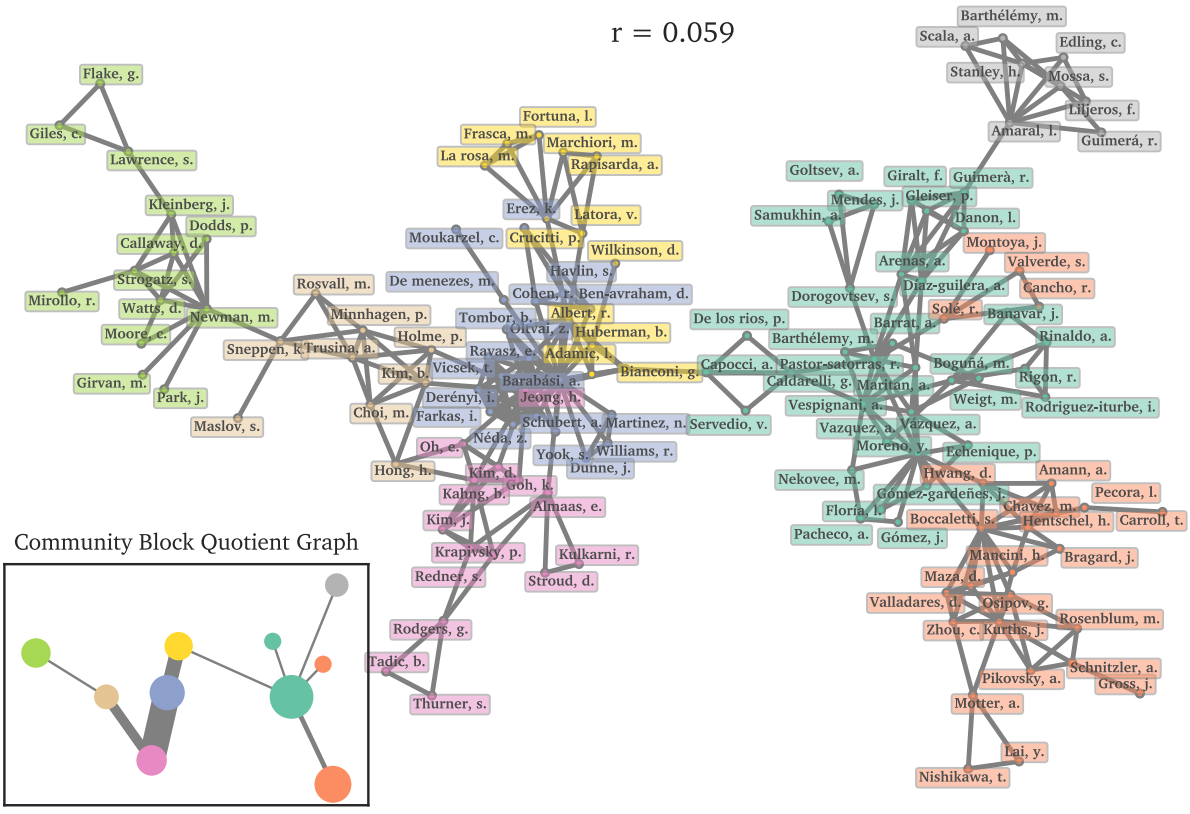


Figure 8.1: 134 Network scientists connected by co-authorship

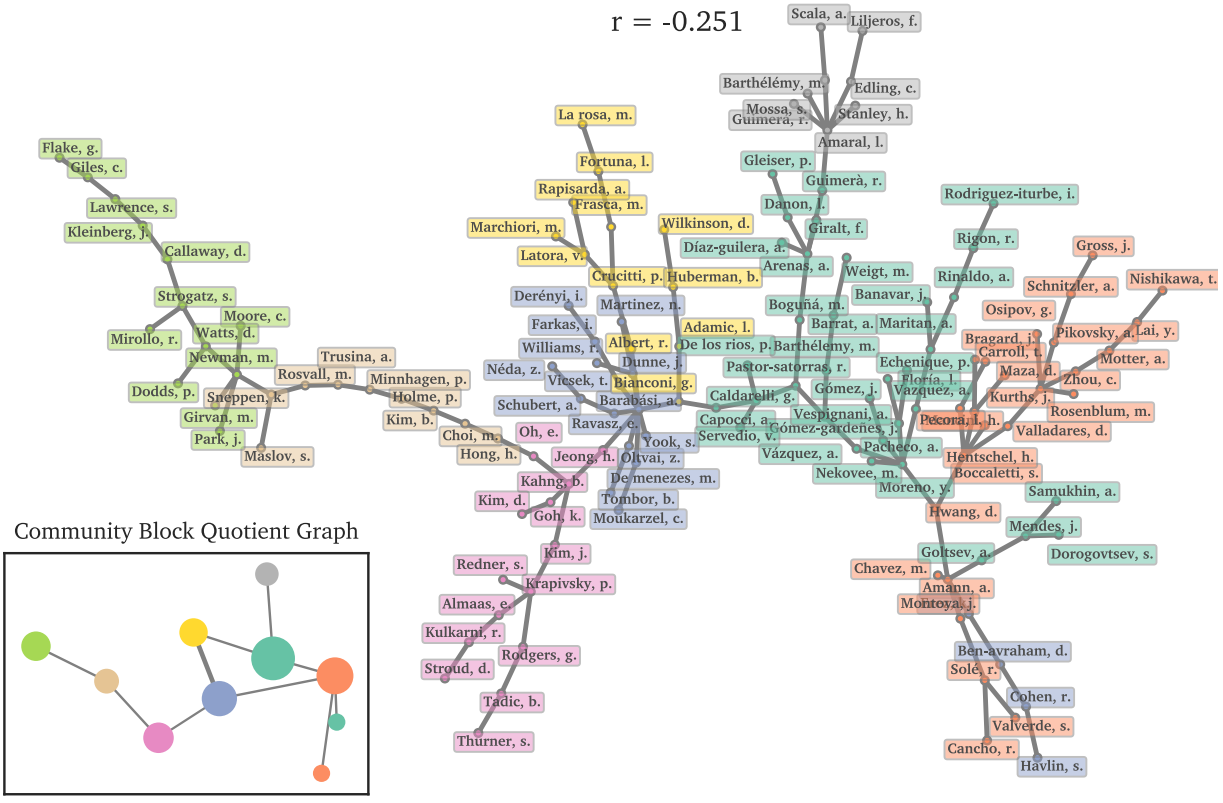


Figure 8.2: Chow-Liu tree of NetSci collaborator dependency relationships

This dependency is much more sparse, and indeed the assortativity coefficient has dropped to $r = -0.251$. In academic writing, we might even expect a lowered assortativity, since students that may not go on to be prolific in their original field would still seek out prolific advisors, initially. Unfortunately, enforcing a tree structure has some negative side effects. Overlaying the original community structure from Figure 8.1 onto the tree and calculating a quotient graph shows that the community structure is impacted by the change. Many of the communities are "unrolled" into long chains of authors, since trees cannot allow small loops or cycles. This makes for a shortest-path distance between community hubs (in the same field) to be upwards of 9-10 jumps, which goes against the scale-free/small-world nature we expect from social systems.

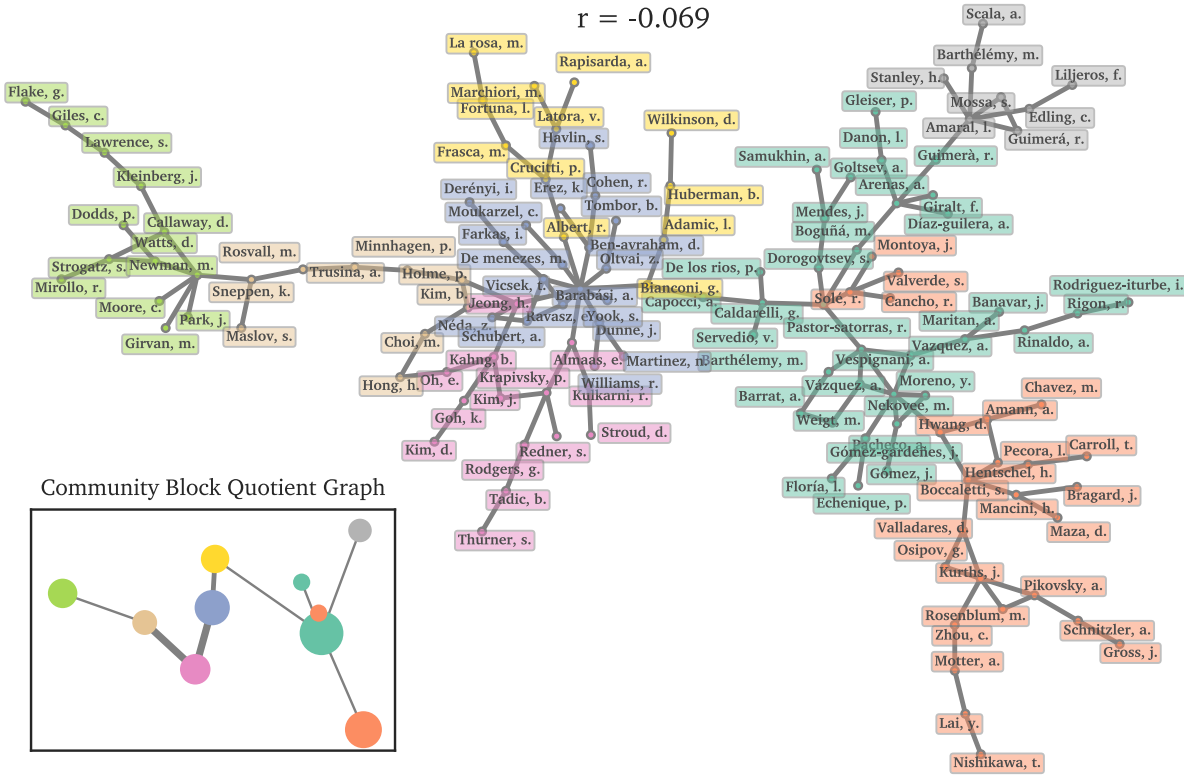


Figure 8.3: Forest Pursuit estimate of NetSci collaborator dependency relationships

With *Forest Pursuit*, we can attempt to correct for "clique bias" in the measurement of dependency relationships, while allowing for flexibility in the global network structure. The

FP recovered collaborator graph is shown in Figure 8.3.

The communities from the co-occurrence network are entirely preserved, with a nearly identical community structure in the graph quotient compared to the co-occurrence graph. But now we see an assortativity that is close to zero (slightly negative at $r = -0.069$). This brings the assortativity more in line with the World Wide Web networks, or even close to results for random preferential attachment networks that are zero in the limit[**AssortativeMixingNetworks_Newman2002**].

Relative to the tree network, there are not so many long chains, as many authors have been allowed to "loop-back" and have relationships with nearby colleagues, reducing the path distance between communities in the process. Still, like the trees, FP tends to reduce the degree of each node, which is summarized in Figure 8.4.

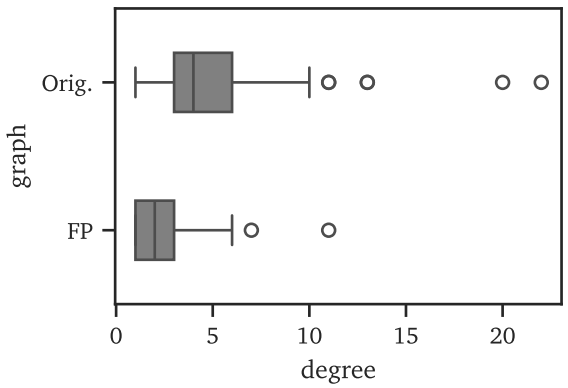


Figure 8.4: Degree distributions of FP vs co-occurrence social networks

From a domain modeling perspective, it might be reasonable to assume that any given author has 2-3 influential collaboration relationships shown, especially considering students and post-docs are likely to constitute a good number of nodes. Even from a logistics perspective (during a given time window working with an advisor) the number of times a student asks/is asked to participate in a paper has to be limited, given average publishing rates.

Only rare individuals would have upwards of 10 influential relationships, with a mean closer to the 2-3 of the FP network, rather than a full quarter having 10 and the median being 4-5

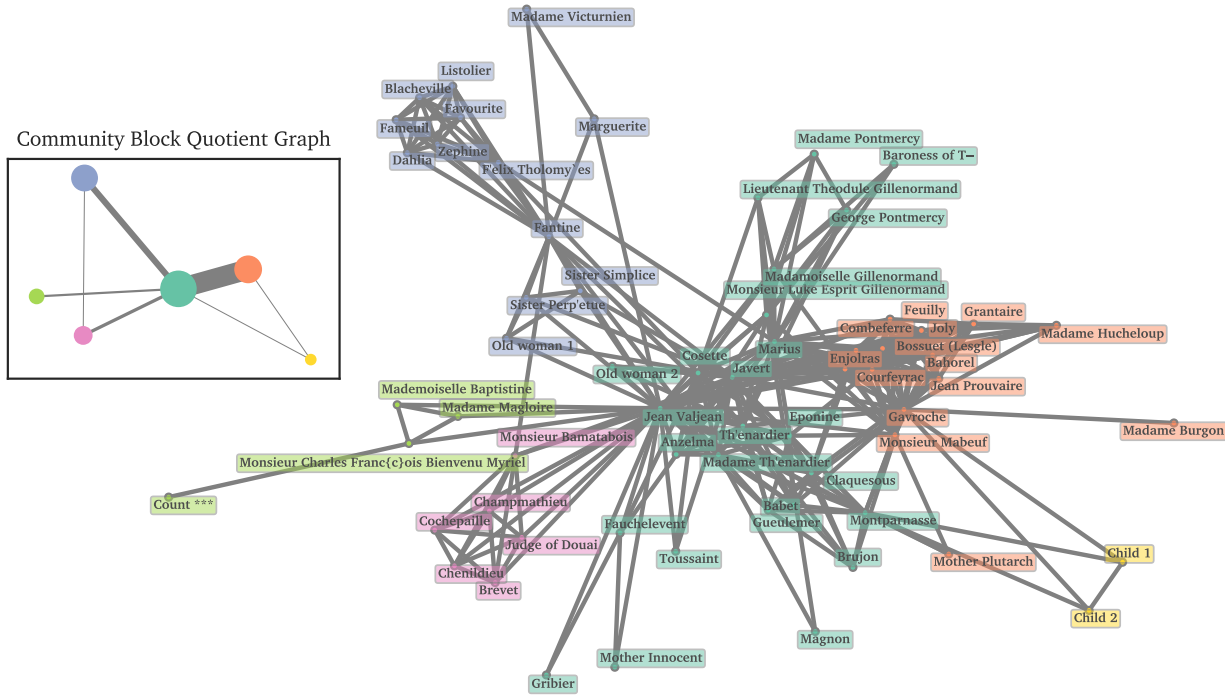


Figure 8.5: Les Miserables character co-occurrence network

Note that the communities demonstrate clear clique-like behavior. On one hand, this does make assessment of clusters easier, since characters in roughly the same scenes are densely grouped together. However, this network makes it difficult to parse relationships, such that backboning would become necessary.

We should also ask *what we want* out of this "social" network. Another way we might think about a social network of characters is "which characters *influence* the appearance (or not) of which other characters?" By extension, "which characters are *significant*, in the sense that they dictate the appearance of more characters than others?" From a domain modeling perspective, this is like asking *how* an author is deciding which characters to include in a chapter. By not correcting for clique-bias, we are implicitly assuming that Victor Hugo would be setting out to write a chapter and immediately writes down a list of every character, independently, should appear. Instead, it's likely that the desired appearance of certain characters in a chapter *leads to* the inclusion of others, as the plot requires.

Because the inclusion of certain characters *leads to* the inclusion of others (by this model), we can reasonably model the authorial "character inclusion process" as *spreading* from character to character. With this in mind, we apply Forest Pursuit to correct for clique-bias, and show the resulting dependency network (with the original community partition) in Figure 8.6.

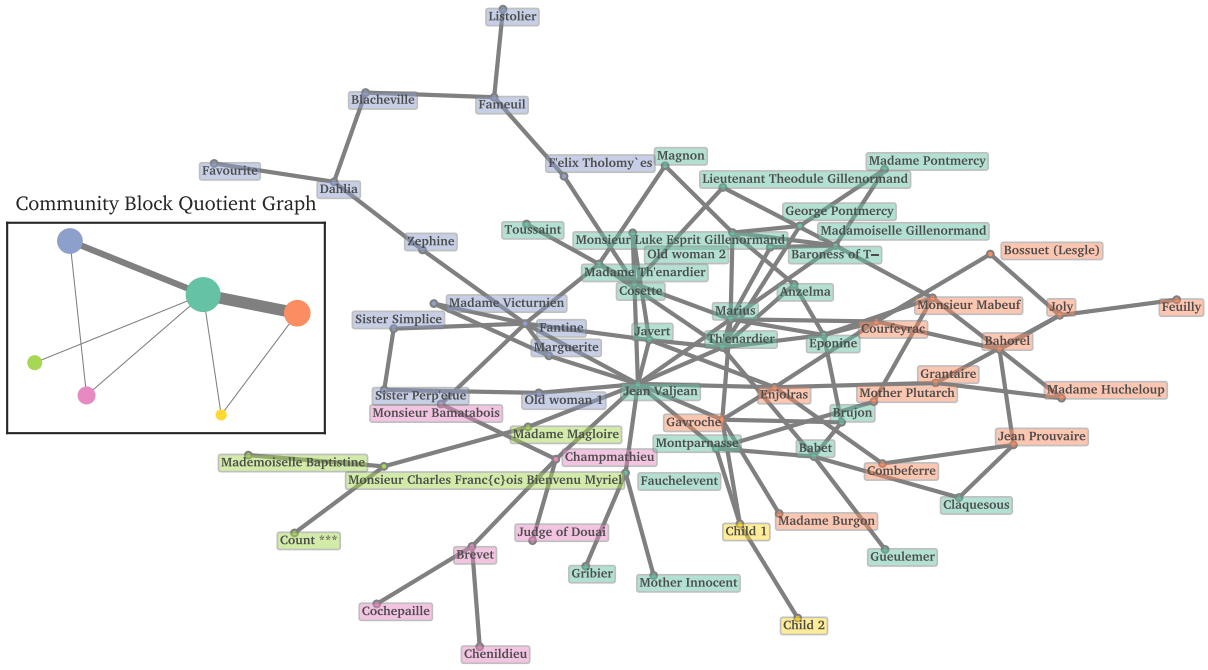


Figure 8.6: Les Miserables character dependency network (Forest Pursuit)

The network edge probabilities have been thresholded in the same manner as the DS (minimum connectivity) filter, only including necessary edges to still retain overall connectivity. As before, our community structure shown in the quotient graph has been preserved, even with the significant edge density reduction.

Applying these networks, we might wish to distinguish through them which characters are "main" and which are "supporting", though more on a gradient than in a binary/classification sense. One way to do this is through centrality measures [**Mathematicsnetworks_Newman2018**, **atlasaspiringnetwork_Coscia2021**], which estimate the "importance" of nodes in various ways. Eigenvector centrality, specifically, finds

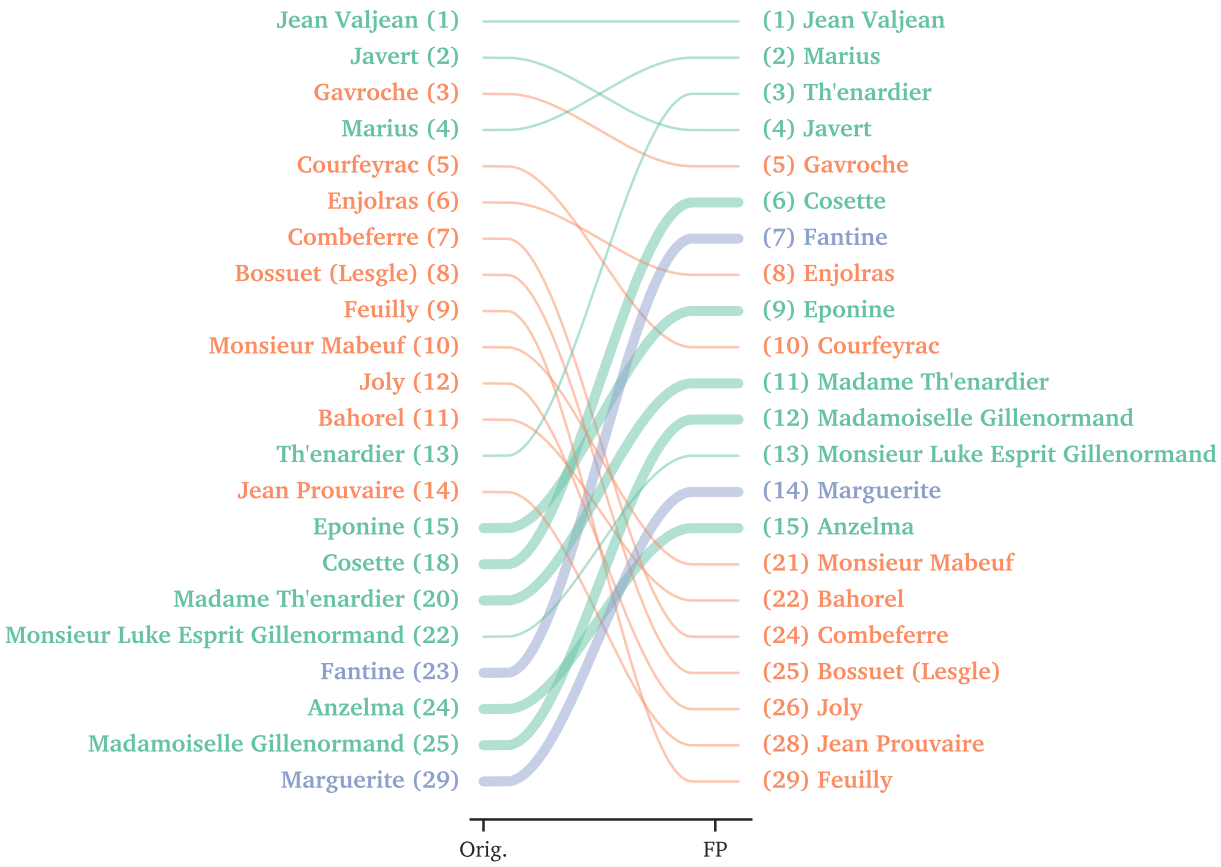


Figure 8.7: Changes in character centrality ranking for FP vs co-occurrence

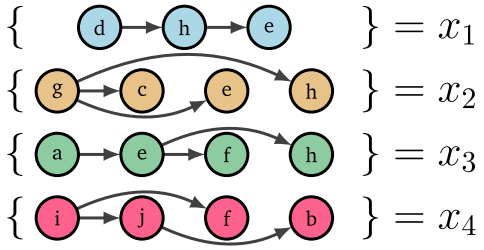


Figure 9.1: Observations as partially-ordered sets

tags or directly from a corpus of written language. What's needed is an assumption on how the observed partial order of concepts is generated. **ForagingSemanticFields_Hills2015** proposes a "foraging" mechanism, so that concepts get sequentially recalled from "semantic patches" of nearby concepts in memory. The partial order comes from our ability to maintain more than one concept in working memory, so that the next concept can be "foraged" from any of the other recently recalled ones[**magicalnumberseven_Miller1956**, **Dynamicsearchworking_Hills2012**].

In this section, we briefly cover a method for network inference by **Humanmemorysearch_Jun2015** that utilizes partial order information from ordered lists of concepts, called INVITE. We use it to demonstrate improvement over bag-of-words and markov assumptions for downstream technical language processing [**Technicallanguageprocessing_Brundage2021**] tasks, as originally demonstrated in [**UsingSemanticFluency_Sexton2019**, **OrganizingTaggedKnowledge_Sexto**

Finally, we show that using *Forest Pursuit* for partially ordered data can still be quite useful for network backboning, and for a fraction of the computational cost. We investigate a network recovery task from verbal/semantic fluency data [**Estimatingsemanticnetworks_Zemla2018**], which involves recovery of a network of animal relationships from memory and recall experiments. Even without directly using partial order information, proper data preparation along with previously-discussed (un-ordered) recovery methods can lead to significantly improved network backboning and analysis capability

Table 9.1: Maintenance Work Order as categorized vs. tagged data

(a)

"Hydraulic Leak at cutoff unit; Missing fitting replaced"	
Categorization:	
Subsystem	142_HYD_SYSTEM
Error Code	ERR_142A
Action Taken	PART_ORDERED
Tags:	
objects	cutoff_unit, hydraulic, fitting
problems/actions	leak, replace

9.1 Technical Language Processing with INVITE

Maintenance work orders are often represented as categorized data, though increasingly quantitative use of a technician's descriptions is being pursued by maintenance management operations [**BenchmarkingKeywordExtraction_Sexton2018**, **CategorizationErrorsData_Sexton2019**]. Tags are another way to flexibly structure this otherwise "unstructured" data, which Table 9.1 shows in comparison to more traditional categorization.

Whether entered directly or generated from text by keyword extraction, the tags will tend to have ordering information readily available. A traditional way to model this kind of text is through either bag-of-words (the co-occurrence node activation data already discussed) or as a sequence of order-n markov model emissions. An order-n markov model MCn estimates the probability of observing the ith item t_i in a sequence T as

$$P(t_i \mid T) \approx P(t_i \mid t_{i-1}, \cdots, t_{i-n})$$

Unlike the clique bias from before, assuming markov jumps for each observation leads to a different kind of bias, with higher precision but reduced recall as shown in Figure 9.2.

Without knowing the underlying dependency relationships, it's difficult to estimate which edges were used by a random-walker, since subsequent visits in memory to a "tag" are not

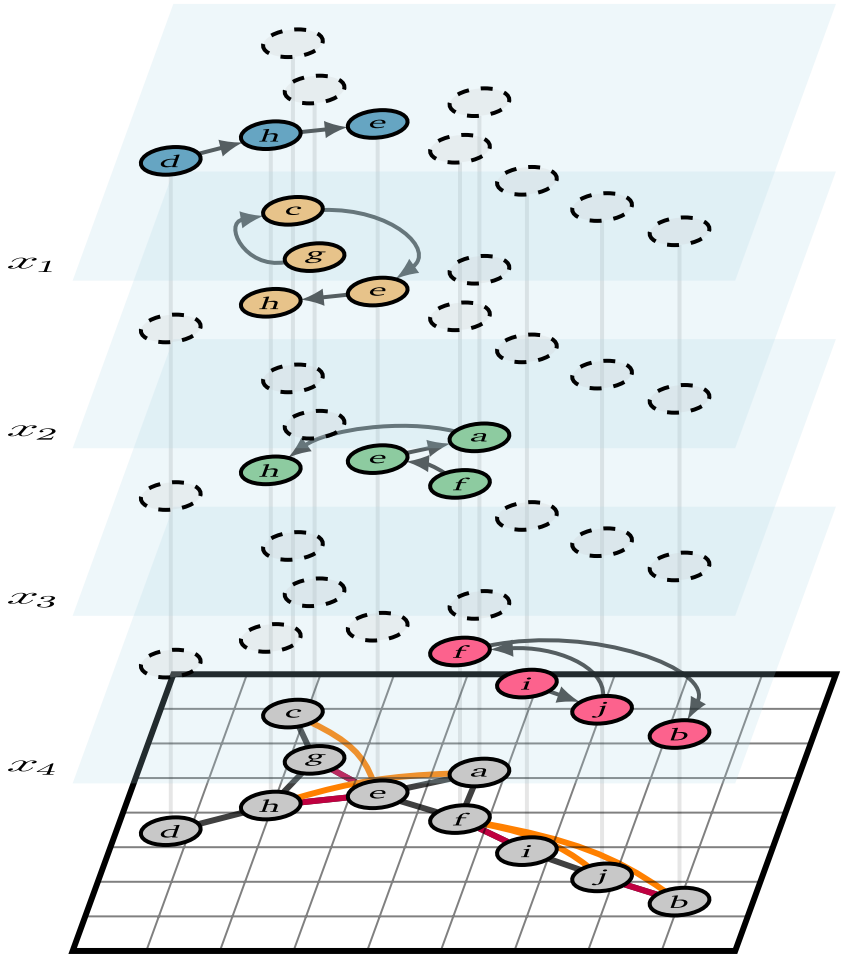


Figure 9.2: Partial-order edge measurements with Markov assumption

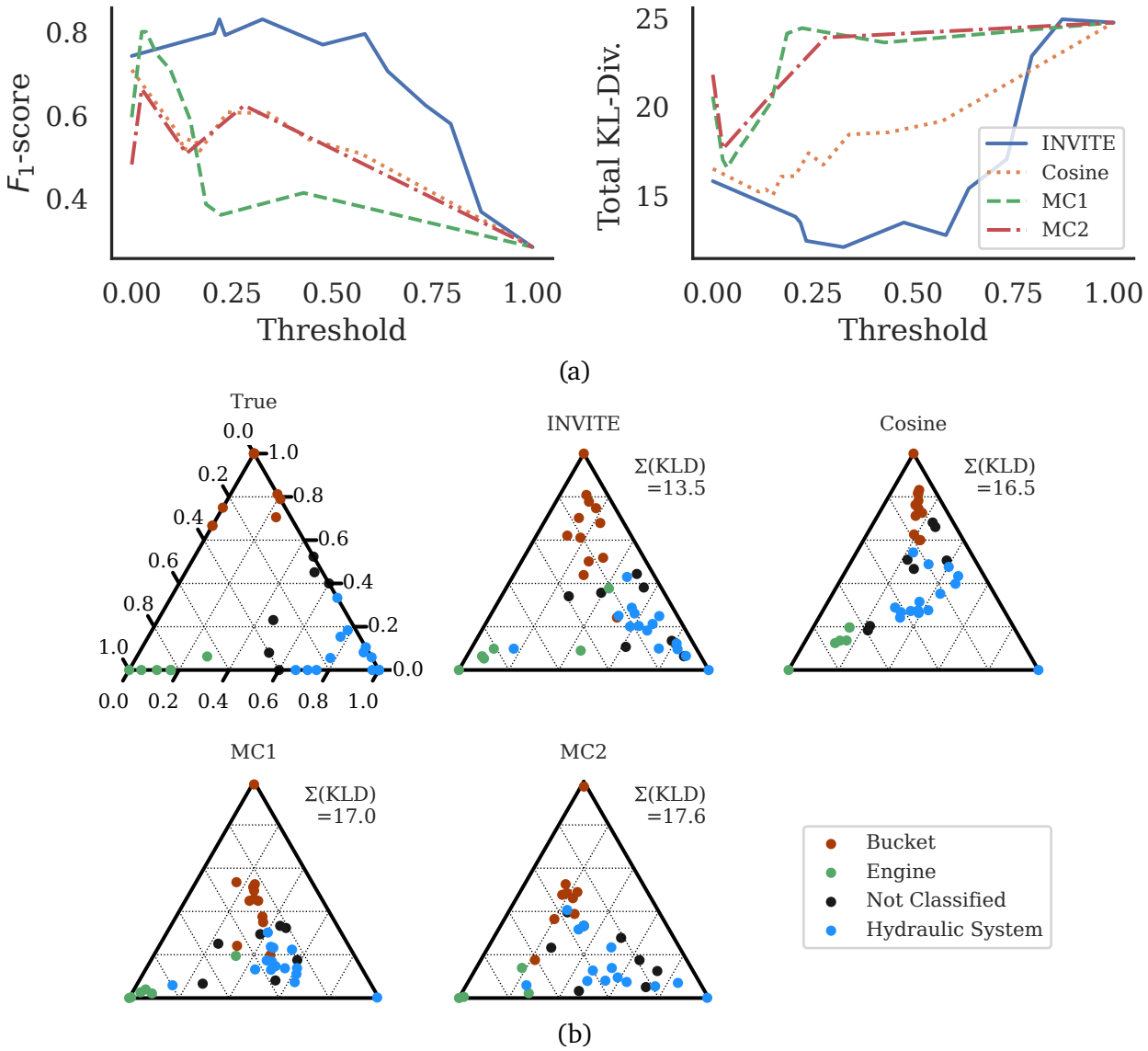


Figure 9.3: Semisupervised MWO Tag Classification with Network Label Propagation

good sample size, as well as lists with at least two animals. This resulted in 100 animals over 293 fluency lists. However, we ignore this filtering when creating the 10-animal rolling windows, to avoid inclusion of unrelated animals into prematurely filtered windows. After re-applying the filter, 100 animals appear across 8020 working-memory windows. Figure 9.4 shows the effects of this preprocessing on marginal distributions.

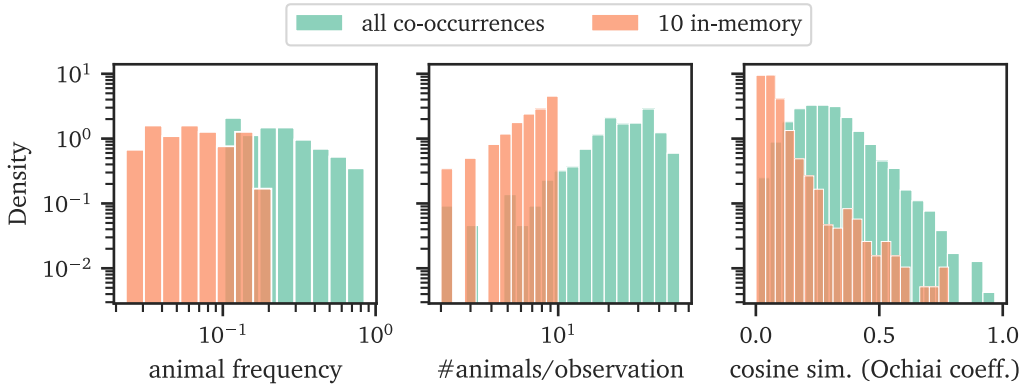


Figure 9.4: Effects of rolling-window activations on observation data

Doing this preprocessing (for rolling-windows of 10) shifts relative animal frequencies downward (since there are many more "observations" from the rolling window), while also shifting the number of animals-per-observation to be strictly less than 10. As desired, the pairwise cosine similarity of all vectors $\mathbf{x}'_{j'}, \mathbf{x}'_{j}$ is significantly reduced. While many participants might cover similar animals *overall*, we want to investigate animal dependencies locally, and we don't expect individuals to always recall animals in the same memory "area" the whole time.

9.2.2 Edge Connective Efficiency and Diversity

To compare the results of different backboning techniques, we introduce a new simple measure to quantify a network's sparsity, in terms of how many edges more than $n-1$ (the

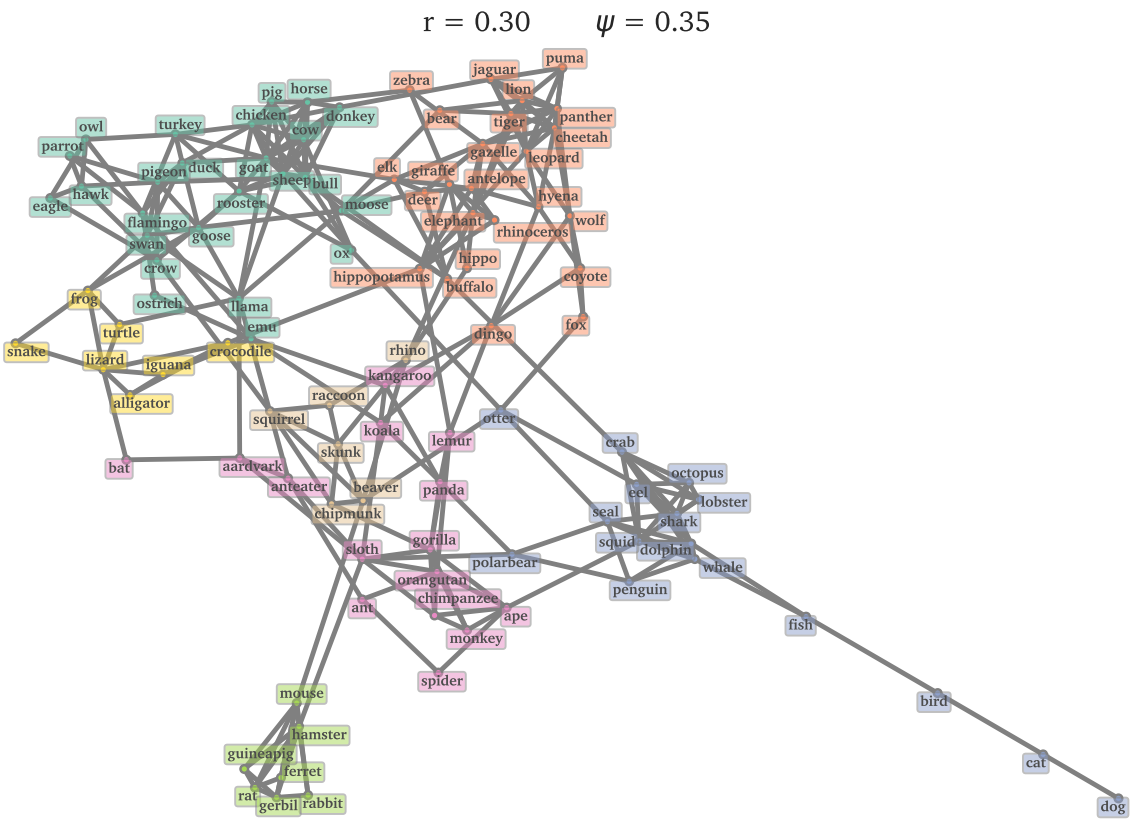


Figure 9.5: Verbal Fluency (animals) Network Backbone (Doubly-Stochastic)

these issues. Clusters are largely intact, instead represented by large branches/subtrees off the main group. However, some community relationships have been sacrificed to maintain strong individual edges, such as *monkey+giraffe* for location similarity at the expense of separating two halves of the pink cluster across a wide distance. More alternate paths between creatures (i.e. loops) are needed to better represent our perception of animal relationships.

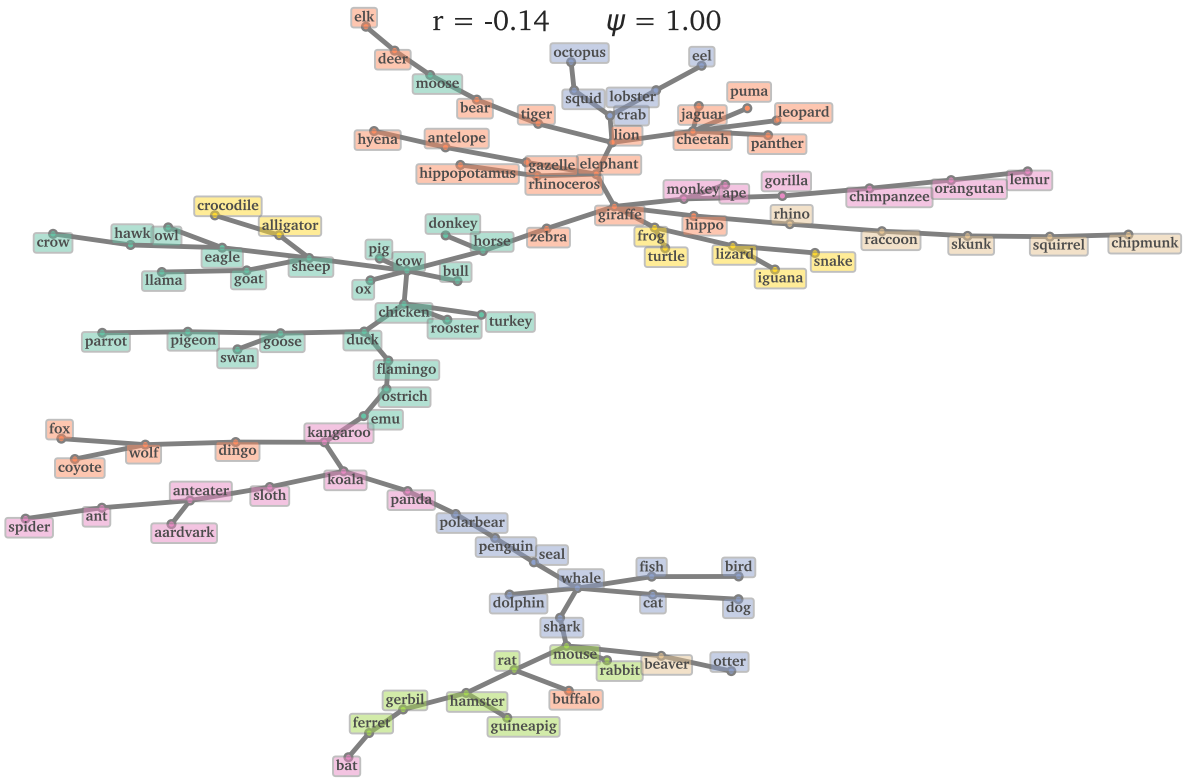


Figure 9.6: Verbal Fluency (animals) Dependency Network (Chow-Liu Tree)

The other dependency network recovery method is GLASSO, which we have similarly thresholded at the minimum-connected point. It only slightly improves on connective efficiency ($\psi = 0.44$), though the cliques are replaced with much more dispersed connections throughout the graph. We also see that reasonable inter-group connections are better represented, such as *rabbit+squirrel*, though *cat* and *dog* are still isolated due to overrepresentation throughout the dataset.

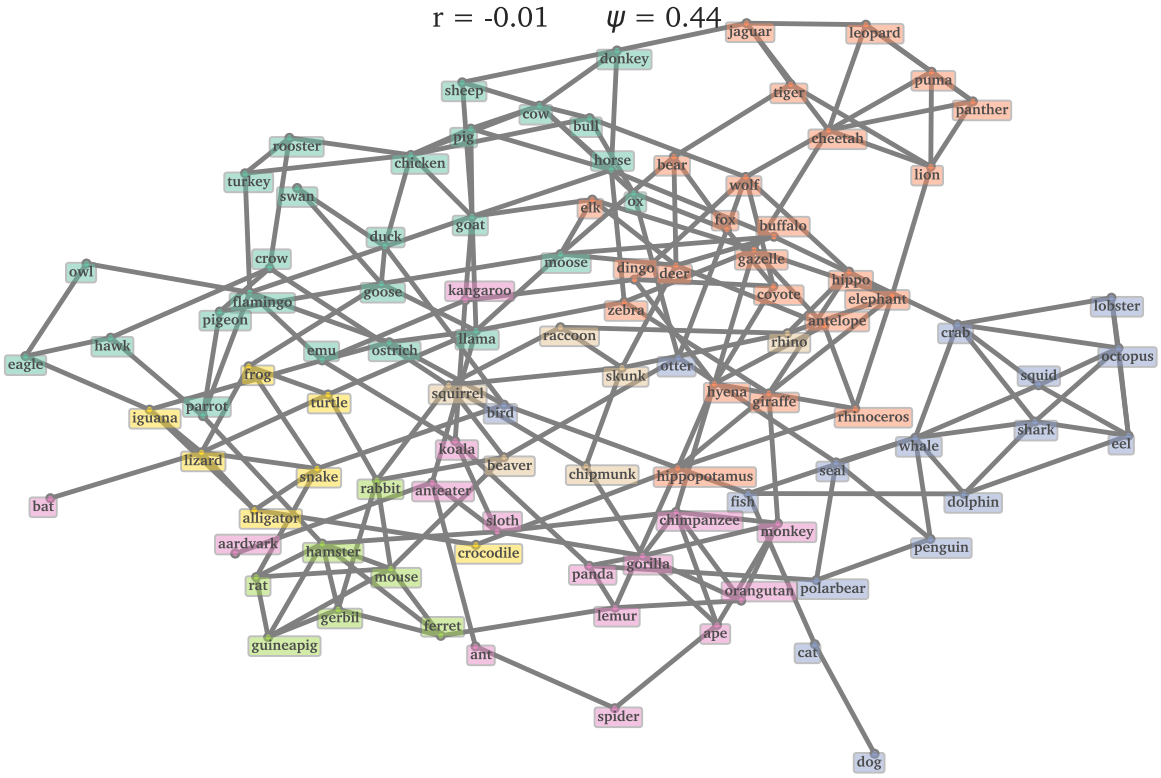


Figure 9.7: Verbal Fluency (animals) Dependency Network (GLASSO)

Subjectively, the GLASSO network is still difficult for an analyst to synthesize into useful knowledge, with so many edges, while the DS network only really managed to communicate one "kind" of knowledge (the context clusters). We would ideally prefer a backbone that provides a wider diversity of important edge "types", for an analyst to better understand the kinds of animal relationships humans perceive.

To illustrate this, we show the *Forest Pursuit* (FP) network in Figure 9.8. It has also been filtered to minimum-connected, like DS and GLASSO, though in this case the connective efficiency to reach that threshold is a staggering $\psi = 0.88$.

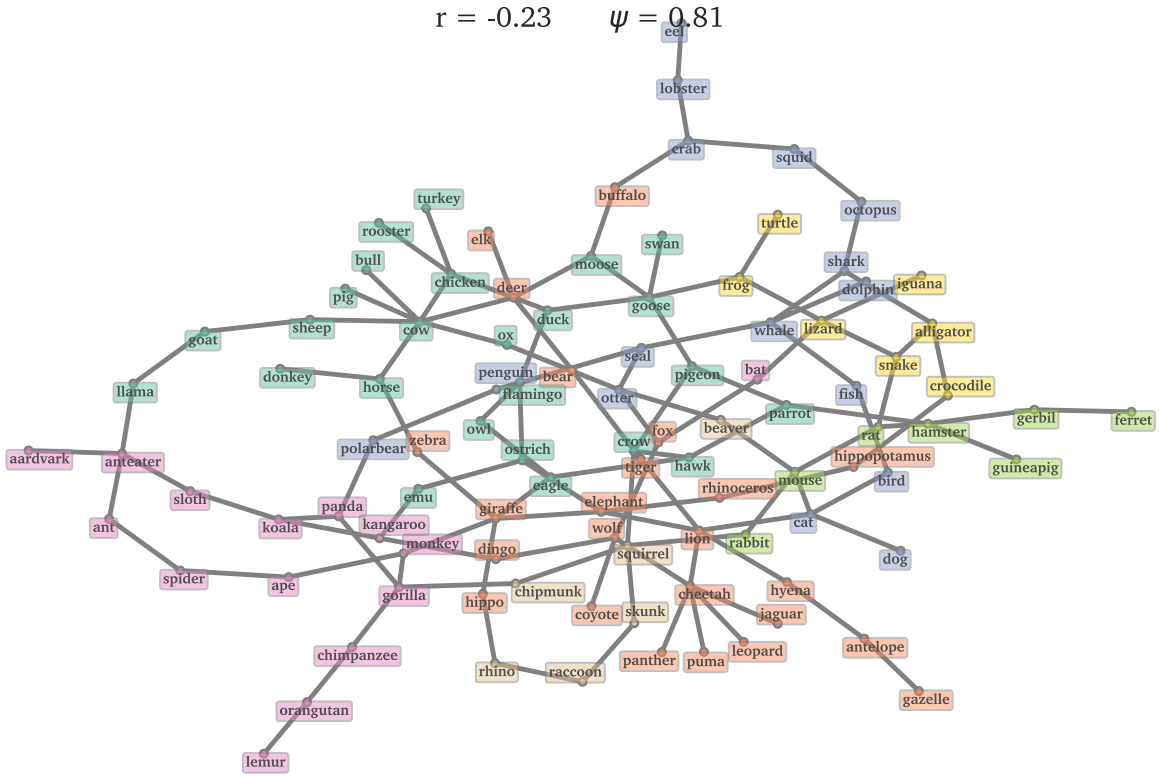


Figure 9.8: Verbal Fluency (animals) Dependency Network (Forest Pursuit)

Unlike the other networks that push "generalist" nodes like *cat* and *dog* onto long, distant chains, those chains are used in the FP network to hold rare subgroups of clusters, treating them as "gated" by the prominent "hubs" of those groups. For example, *cat* is correctly linked to *mouse* and *lion* (in addition to *bird*), while *lemur* is pushed down a longer chain of

primates, "gated" by *gorilla*. Similarly with *eel* through *lobster* and *crab*.

A much broader edge-type diversity is also made apprarent with many non-context-based relationships made obvious with the improved sparsity. An analyst has an easier job of creating "edge-type inventories", making the FP backbone an excellent exploratory assistant: animals can be related because they are:

- Co-located
- Taxonomically similar (*cheetah*+*leopard*)
- Famous predator/prey pairs (*cat*+*mouse*)
- Pop-culture references (*lion*→*tiger*→*bear*)[7]
- Similar in ecological niche/role (*koala*+*sloth*)
- lexically similar/rhyming (*moose*+*goose*)
- Related through conservation or public awareness (*panda*+*gorilla*)
- etc.

This is further reflected in how FP alters the way *centrality* measures behave. Replicating Figure 8.7 for these graphs, Figure 9.9 shows the change in rank across the top 15 animals for the DS, GLASSO, and FP networks.

The DS centrality finds the most densely connected clique and gives all of its members incredibly high values. Meanwhile, the none of the top 5 most common animals (*dog,cat,lion,tiger,bear*)[8] have high centrality *at all*. Both GLASSO and DS have farm animals (*chicken,goat,cow*) as the most important, despite the idea that goat likely could be reached from e.g. *cow* quite often. FP adds more variety, giving hub-animals from different communities high centrality scores, each of which could lead to a variety of different paths. While

[7]Note that the dependency-based methods correctly interpred these three as *not* being mutually connected in a triad, but specifically with this ordering (*tiger* in the middle).

[8]Interestingly, *dog* never appears in centrality measures, and *none* of the networks connect dog to any other animal than *cat*. Meanwhile, *wolf* is associated more with *fox*, *coyote*, *dingo*, etc., which are notably all predators of farm animals.
With the primary community structures being what they are (context/location-based), it seems that humans tend to put dogs in a category all their own.

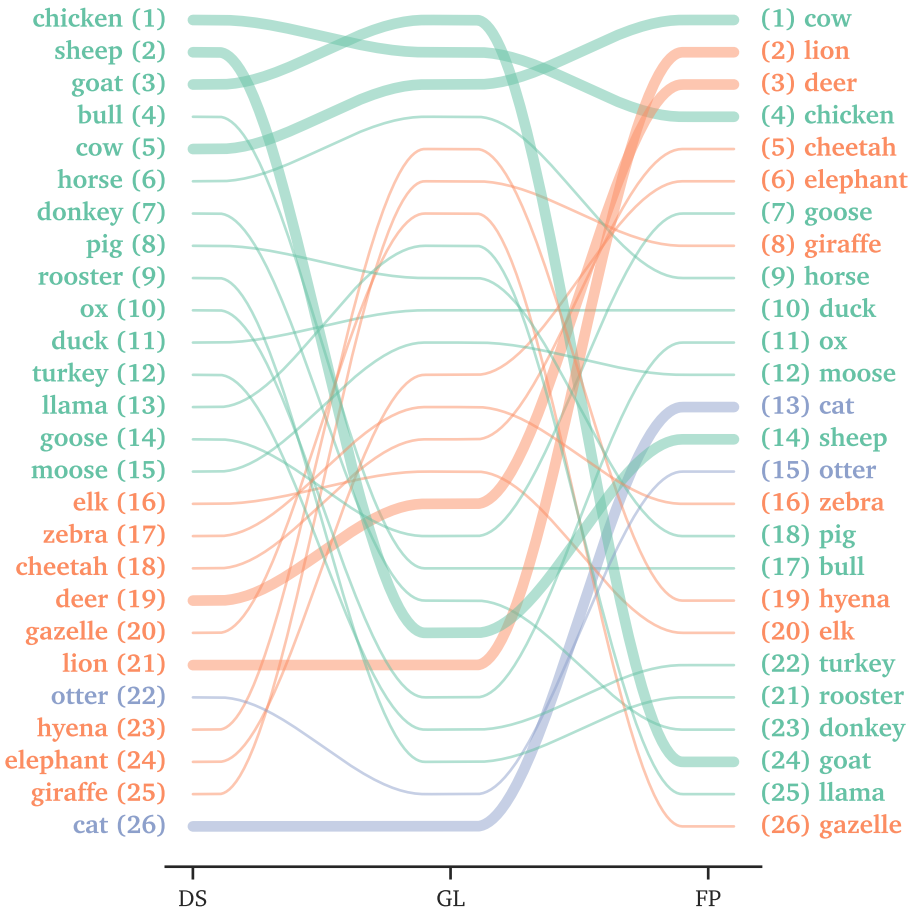
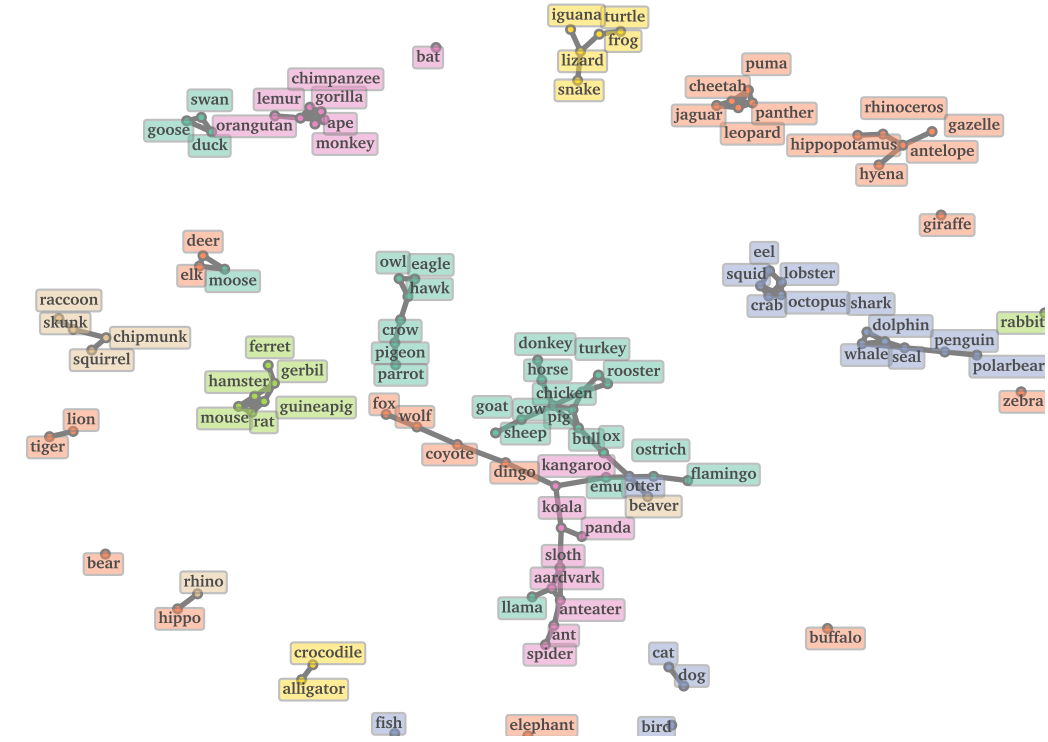
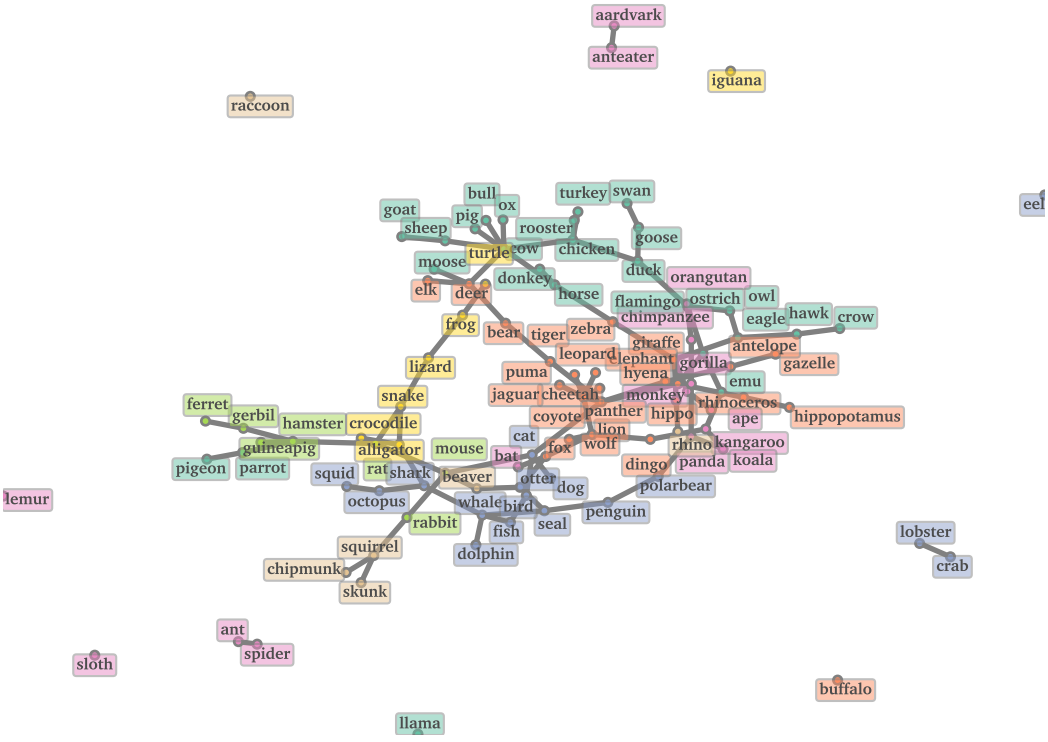


Figure 9.9: Changes in animal centrality ranking for FP vs co-occurrence,GLASSO

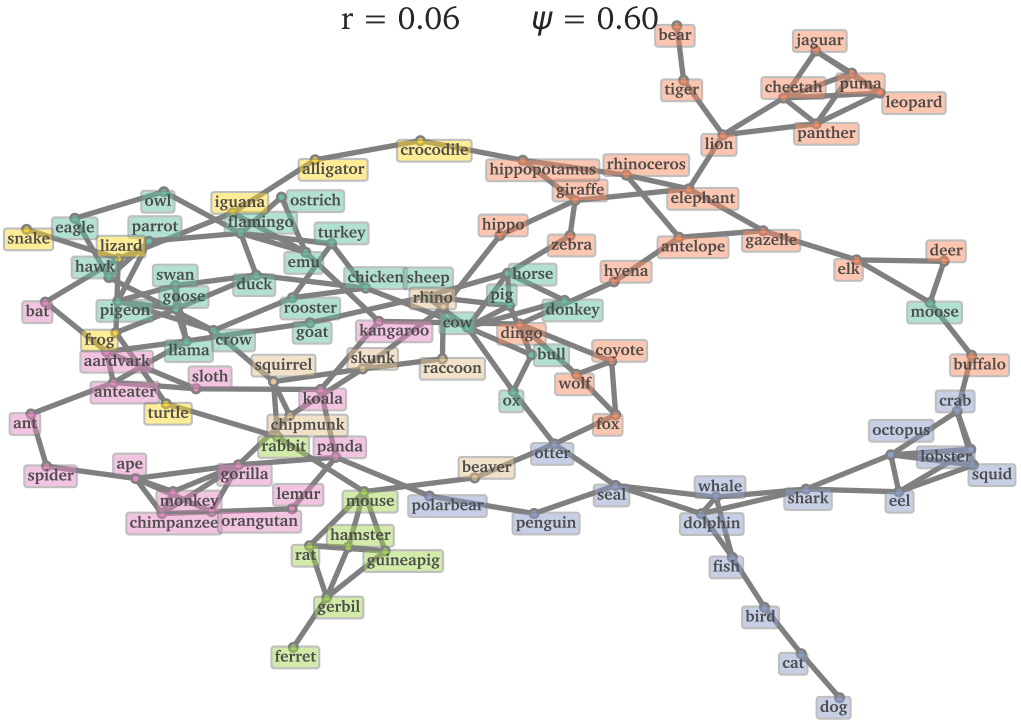


(a) co-occurrence retains local communities at the cost of global structure

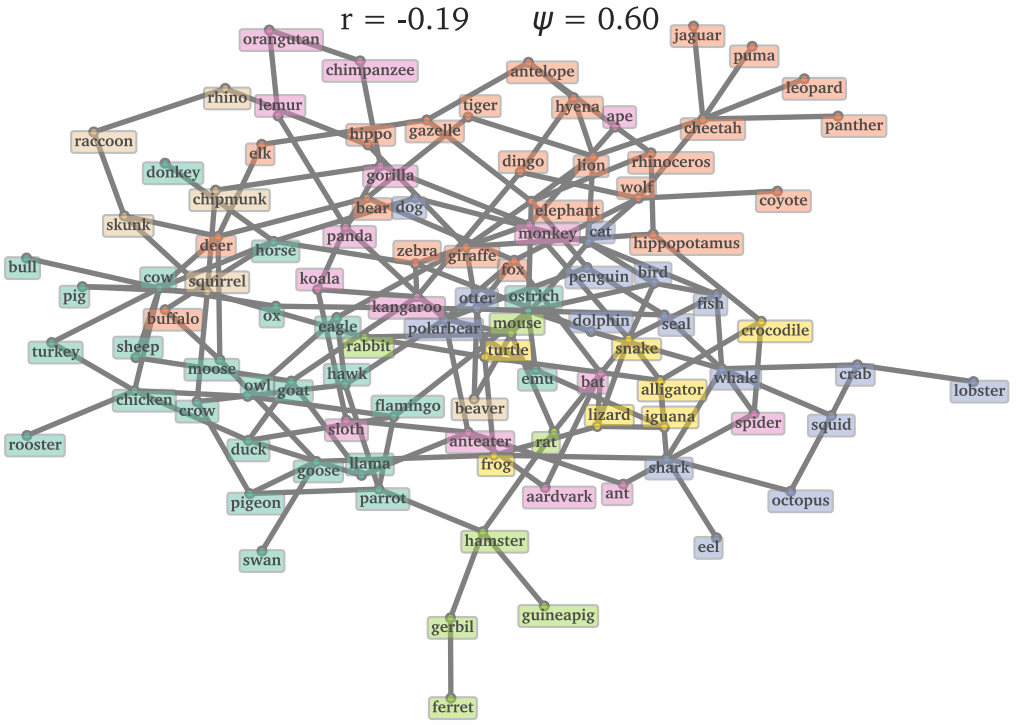


(b) clique-bias correction preserves central structure by disconnecting rare nodes

Figure 9.10: Differences in structural preservation with over-thresholding.



(a) Doubly Stochastic filter with FP (node degree) preprocessing



(b) GLASSO Precision estimate with FP (node-degree) preprocessing

Figure 9.11: Forest Pursuit preprocessing for Doubly-Stochastic and GLASSO recovered networks

Chapter 10 Conclusion & Future Work

Practitioners have long struggled with a lack of techniques for metrological quantification and measurement error handling in network science. There is an ongoing need to specify valid network recovery models—ones that are not only assessed for precision, but designed for *trueness*. **Measurementerrornetwork_Wang2012** provides a "reclassification" of measurement error in networks, focusing on the true/false positive/negative dichotomies for both edge and node reporting. This implicitly assumes that, like the karate-club graph [**InformationFlowModel_Zachary1977**], our observations are of network components (edges/nodes). When the object to be measured is not directly observable (as is the case for recovery from node activations), measurement error can also arise both from model noise sensitivity (lack of precision) and misspecification (lack of trueness). Because of this:

> The practice of ignoring measurement error is still mainstream, and robust methods to take it into account are underdeveloped.
>
> – Tiago Peixoto [**ReconstructingNetworksUnknown_Peixoto2018**]

10.1 ~~Summary of contributions~~Discussion and limitations

~~To develop the practice of taking measurement error into account, we have proposed a combination of problem framing, measurement aggregation techniques, and methods for bias correction when recovering network structure from observed random walk visits~~Much of our discussion on *Forest Pursuit* and its extensions has revolved around assumptions about data availability and generation. Practitioners often face situations where networks are recovered in essentially "unsupervised" settings, and their data could reasonably be

modeled as arising from a spreading process on the graph they wish to recover. However, these assumptions do not hold in general, and it's worth discussion how they impact the application of Forest Pursuit and where future research could fill in theoretical and practical gaps.

This thesis provides:

10.1.1 Validation and Network Dynamics

An interpretation of network recovery as an *inverse problem* comparable to sparse approximation, with concise definitions of data, edge, and node vector spaces from an underlying incidence-structure formalism. A taxonomy of structural assumptions used in literature to make network inference tractable: We largely assumed that real-world network recovery is predominantly *unsupervised*, so that results verification is very difficult in practice. The MENDR dataset provides an initial foray into standardized reference problems, each having a "ground-truth", but this becomes much more complicated when real-world datasets do not.

1. Local, global, or resource/information-flow structural constraints;
2. Inverse-problem status (direct or indirect edge observation) ; and
3. Whether activation observations are pre-aggregated before estimation.

A method, *Forest Pursuit*, to address the need for a model with:

Local observation constraints,

An inverse-problem assumption for indirect edge observation, and

Dependence on the bipartite nature of node observations. One approach to verification would be to do forecasting on dependencies. For collaboration networks, for instance, if two authors publish together, we are assuming they are conditionally dependent on each

other (when there are no "hidden" node activations).

~~A dataset and benchmarking toolkit (MENDR)to reproducibly compare algorithmic~~ These links (the two-author papers) can be used as an incomplete "ground-truth", so we could theoretically test each algorithm's ability to predict dependencies when the two-author examples are either held out or predicted using all preceding papers.

One difficulty with this is the sheer number of true-negatives we expect in a sparse graph as the number of nodes increases. In general a sparse connected graph's edge count only must grow linearly with node count, but the non-edge count grows quadratically. This makes the chance that any two authors with a possible dependency *do* coauthor exclusively together vanishingly small, in general. Not all conditional dependencies within a department will lead to pair-papers, since some individuals will only publish in larger groups, for instance. To add to the trouble, relationship networks over time have a good chance of being *dynamic*: people move to new institutions, students graduate, etc. Using predictions of *future* two-author collaboration will risk running into errors from network dynamics like these. It may be possible to include network dynamics into the relationship inference itself, such as with Dynamic Topic Models [**Dynamictopicmodels_Blei2006**], but just as before we are left with the difficulty of verifying and validating a (now more complex) unsupervised model.

Within this train-of-thought, however, lies a possibility to create what are called *metamorphic tests* suitable for verification of unsupervised models [**METTLEMETamorphicTesting_Xie20**] Rather than look for a ground-truth the measure our network against, we can define properties of our network reconstruction that we know must hold given our modeling assumptions or domain knowledge. For the co-author network case, instead of predicting any two-author paper, instead we might construct a list of all student-advisor pairs. Because we know that their relationship is very likely to be dependent, these pairs should be recovered from an algorithm that is reconstructing author dependencies. Algorithms

could be tested against each other for performance on these kinds of metamorphic conditions, ensuring correctness in cases we specify. Such metamorphic tests would be a valuable addition to the network reconstruction community, especially if individual domains could compile lists of relevant conditions that *should* hold in any given reconstruction attempt.

10.1.2 Spreading process assumption

As discussed in Section 7.2, the marked-Random Spanning Forest model was motivated by a need to incorporate prior knowledge about the generating mechanism of our data (namely, *spreading processes*). Consequently, our validation through synthetic datasets provided in MENDR used random walks to construct node-activations for inferring structure. The results presented here verify Forest Pursuit's ability to recover ~~network structure from random walk activations, which has been applied to demonstrate the scaling and accuracy of *Forest Pursuit* over other methods.~~

~~Generalization of Forest Pursuit by developing a probabilistic model for it as a sparse dictionary learning technique, for which we provide an expectation maximization scheme to estimate.~~

~~Application of *Forest Pursuit* as case studies in scientific collaboration networks, classic literature analysis, technical language processing, and semantic verbal fluency tests.~~ structure in this setting, which also helps validate our design given the domain-based constraint (spreading-process generation).

~~We lay this foundation in the hope of further improving the ability of practitioners to explore the structure of their data in a principled manner~~However, other methods do exist for generating (correlated) binary activation data, and adding other types of datasets to the MENDR catalog would provide a mechanism to verify model recovery capability under other generation settings. The addition of thresholded multivariate-normal samples [**generationcorrelatedartificial_Leisch1998**] would be a reasonable next-step to

increasing the scope of possible verification for thesis algorithms. Validation in these cases would need more theoretical work to provide a framework for understanding the expected behavior of Forest Pursuit under non-spreading assumptions. However, we also hope to see further development of additive/local Desire Path Density models by the community that estimate relationships under other common generative schemes. It is possible that planar graphs and path-graphs are a class that appropriately model MVL (Ising) and Markov (sequential) generation schemes, respectively. More work is needed to show the behavior of Desire Path Densities with other graph classes.

10.2 Modifications and extensions to Forest Pursuit

Desire Path Densities and *Forest Pursuit* are designed to be adaptable to several modalities of use[1]. However, there are key limitations of the model that could be addressed going forward, as well as future research directions inspired by our modeling paradigm.

10.2.1 Multiple sources and hidden nodes

One of the key assumptions to make the likelihood of the (marked) Random Spanning Forest tractable is to allow only one "source" node (the random walk starting node), and to sample it from a uniform categorical distribution. However, by explicitly adding a "root" node that is implied by the random forest distribution (see Figure 7.3), we would immediately achieve the possibility for multiple sources. A source would become any node incident to the root in a sampled spanning tree. To prevent every node from being activated, the spanning tree could be "pruned" at some depth away from the root (which is a parameter that we could model with *e.g.* a Geometric distribution). How many sources get selected in a minimum spanning tree could be controlled through the weights given to edges incident to the root (which, if the reader recalls, is already represented by β). Also possible could be the use of independent Bernoulli activations of nodes as sources (rather than a categorical selector)

[1] see for instance Section 9.2.4

10.2.3 Application areas and case studies

Lastly, a critical test of *Forest Pursuit* will be its application to a wider variety of domains under the scrutiny of each domain's experts. From the network community itself, for instance, recent interest has been shown in assessing the methodological reasons for observed assortativity [**PerceivedAssortativitySocial_Fisher2017**]. The conditions under which controlling for clique-bias *also* reduces assortativity would be a useful tool when deciding to use different network cleaning techniques.

Further afield, semantic Verbal Fluency tests discussed in Section 9.2 are often administered for the purposes of assisting in diagnosis of Alzheimer's and Schizophrenia patients. While experiments using inferred network structures [**newdissimilaritymeasure_Prescott2006**, **Semanticverbalfluency_AriasTrejo2021**] have been used to detecting early-onset neurological disease from topological differences, it could be useful to re-assess these outcomes when clique-bias has been better accounted for.

All of these applications would be further assisted by *Forest Pursuit*'s ability to

- Infer network estimates *quickly*, on streaming data, and
- Incorporate prior (and incoming) knowledge from domain experts on edges.

Together, these properties should make for an ideal *human-in-the-loop* analysis tool. Indeed, for any qualitative study on nodes and discovering relationships between them, decreasing the annotation load (number of edges to assess)—*while* increasing edge diversity—will be critical to correctly inventory the important categories of relationships or dependencies[3]. Building tools that enable practitioners and researchers to undertake complex grounded coding [**codingmanualqualitative_Saldana2021**] tasks like this is a rich area for possible Human-Systems Integration efforts going forward [**Humanlooptechnical_Fung2024**, **AIInformedApproaches_Harper2022**].

[3] see Section 9.2.2

10.3 Summary of contributions

To develop the practice of taking measurement error into account, we have proposed a combination of problem framing, measurement aggregation techniques, and methods for bias correction when recovering network structure from observed random walk visits.

This thesis provides:

- An interpretation of network recovery as an *inverse problem* comparable to sparse approximation, with concise definitions of data, edge, and node vector spaces from an underlying incidence-structure formalism.
- A taxonomy of structural assumptions used in literature to make network inference tractable:
    1) Local, global, or resource/information-flow structural constraints;
    2) Inverse-problem status (direct or indirect edge observation); and
    3) Whether activation observations are pre-aggregated before estimation.
- A method, *Forest Pursuit*, to address the need for a model with:
    1) *Local* observation constraints,
    2) An inverse-problem assumption for indirect edge observation, and
    3) Dependence on the bipartite nature of node observations.

- A dataset and benchmarking toolkit (MENDR) to reproducibly compare algorithmic ability to recover network structure from random walk activations, which has been applied to demonstrate the scaling and accuracy of *Forest Pursuit* over other methods.

- Generalization of Forest Pursuit by developing a probabilistic model for it as a sparse dictionary learning technique, for which we provide an expectation maximization

scheme to estimate.

- Application of *Forest Pursuit* as case studies in scientific collaboration networks, classic literature analysis, technical language processing, and semantic verbal fluency tests.

We lay this foundation in the hope of further improving the ability of practitioners to explore the structure of their data in a principled manner.