Graph matching the matchable nodes when some nodes are unmatchable

Vince Lyzinski† and Daniel L. Sussman‡

†Department of Applied Mathematics and Statistics, Johns Hopkins University
‡Department of Mathematics and Statistics, Boston University

May 8, 2017

Abstract

Often in inference across multiple graphs there is a latent correspondence between the nodes across networks, and graph matching—the problem of finding an alignment between the nodes of two graphs that best preserves common structure across graphs—seeks to uncover this latent correspondence in the presence of unknown or errorfully known vertex labels. However, in many realistic applications only a core subset of the nodes are matchable in each graph with the remaining nodes being junk vertices, only participating in one of the two networks. Under a statistical model for this situation, we show that correctly matching the core nodes is still possible provided the number of junk nodes does not grow too rapidly, allowing for asymptotically perfect core recovery in the presence of nearly linear (in core size) junk. We also consider the setting of junk present in only one network, showing that a smaller core network can be asymptotically perfectly matched to the corresponding core in a network with exponentially many (in core size) junk vertices present. These theoretical results are further corroborated in both simulated and real data experiments. Although it is possible to correctly match the core vertices across networks, many graph matching algorithms do not identify these proper matches (versus junk-to-junk or core-to-junk matches). As such, we present a procedure for a posteriori detecting the matched core-to-core vertices after the networks have been matched, and demonstrate the effectiveness of our core detection procedure across both simulated and real data.
1 Introduction and Background

The graph matching problem seeks to find an alignment between the vertex sets of two graphs that best preserves common structure across graphs. At its simplest, it can be formulated as follows: Given two adjacency matrices $A$ and $B$ corresponding to $n$-vertex graphs, the graph matching problem seeks to minimize $\|A - PBP^T\|_F$ over permutation matrices $P$; i.e., the graph matching problem seeks a relabeling of the vertices of $B$ that minimizes the number of induced edge disagreements between $A$ and $PBP^T$. Variants and extensions of this problem have been extensively studied in the literature, with applications across areas as diverse as biology/neuroscience [Zaslavskiy et al., 2009, Chen et al., 2016], computer vision [Robles-Kelly and Hancock, 2007, Escolano et al., 2011, Lin et al., 2010], and pattern recognition [Lee et al., 2010, Sang and Xu, 2012, Zhou and De la Torre, 2012] among others. For an extensive survey of many of the recent applications and approaches to the graph matching problem, see the sequence of survey papers [Conte et al., 2004, Foggia et al., 2014, Emmert-Streib et al., 2016]. While recent results [Babai, 2015] have whittled away at the complexity of the related graph isomorphism problem—determining whether a permutation matrix $P$ exists satisfying $A = PBP^T$—at its most general, where $A$ and $B$ are allowed to be weighted and directed, the graph matching problem is known to be NP-hard. Indeed, in this case, the graph matching problem is equivalent to the notoriously difficult quadratic assignment problem [Loiola et al., 2007, Cela, 2013].

Recently, there has been a flurry of activity studying the closely related problem of graph matchability [Pedarsani and Grossglauser, 2011, Lyzinski et al., 2014, 2015, 2016a, Pedarsani and Grossglauser, 2011, Lyzinski, 2016]: Given a latent alignment between the vertex sets of two graphs, can graph matching uncover this alignment in the presence of shuffled vertex labels. This problem arises in a variety of contexts, from network deanonymization to the graph analogue of signal recovery in the presence of an errorful channel, to name a few. Most of the existing results are concerned with recovering a latent alignment present across the entire graphs (i.e., each vertex in one graph has a matched pair in the other graph). While this has served to establish a novel theoretical understanding of the matchability problem, this assumption is often not met in real data. Often, the graphs are of differing orders and/or only a fraction of the vertices are matchable with the remaining vertices having no true pair across networks. Social networks offer a compelling example of this, where matching across different social network platforms (or within a single time varying social network) requires the understanding that not all users will be participants across the two networks [Li and Campbell, 2015]. The challenge for the obtained matching is then to
simultaneously correctly align the matchable subgraphs across $A$ and $B$, and also correctly identify these vertices as matchable. Note that a similar problem was tackled in [Pedarsani and Grossglauser, 2011, Yartseva and Grossglauser, 2013] in the context of a different graph matching approach.

In this manuscript we will probabilistically tackle the problem of aligning the matchable subgraphs of $A$ and $B$. Working in the correlated heterogeneous Erdős-Rényi random graph setting (Section 1.1), we prove that, under mild model assumptions, an oracle graph matching algorithm will with high probability align the matchable cores across networks (Section 2). Moreover, the neighborhood structure across the matched graphs can be used *a posteriori* to rank vertices in the two networks according to which are mostly likely to be matchable (Section 4). We also study the analogue of the core matchability problem in the setting where the graphs are of differing order, and a smaller core graph $A$ is matched to the larger $B$ (Section 3). Our results in this setting prove that, under mild model assumptions, a suitably reweighted graph matching formulation will asymptotically almost surely align $A$ to the corresponding matchable core in $B$ even if the order of $A$ is logarithmic in the order of $B$ (Theorem 8). Lastly, in Section 5 we experimentally verify our theoretical results on both simulated and real data sets using a state-of-the-art graph matching procedure.

**Notation:** The following notation will be used throughout the manuscript: for $k \in \mathbb{Z} > 0$, we will let $J_k$ denote the hollow $k \times k$ matrix with all 1’s on its off-diagonal, $[0]_k$ will denote the $k \times k$ matrix of all 0’s, and $[k]$ will denote the set $\{1, 2, \ldots, k\}$. To ease exposition, unless otherwise specified all graphs considered herein will be undirected, without self-loops and unweighted. We will consider $A$ and $B$ interchangeably as adjacency matrices and as the corresponding graphs consisting of vertices and edges.

### 1.1 Correlated heterogeneous Erdős-Rényi graphs

Formally the graph matching problem we will consider is defined as follows.

**Definition 1.** Let $A, B \in \{0, 1\}^{n \times n}$ be symmetric adjacency matrices. The graph matching problem is to find

$$\arg\min_{P \in \Pi(n)} \|A - PBP^T\|_F^2 = \arg\min_{P \in \Pi(n)} - \text{trace}(APB^TP^T),$$

where $\Pi(n)$ is the set of $n \times n$ permutation matrices.

**Remark 2.** To ease exposition, we present the graph matching problem for unweighted, loop-less graphs. However, extending the problem to handle these data features is generally
straightforward and is achieved by replacing the adjacency matrices with another matrix as appropriate. We discuss this in more detail in Section 6.

In the presence of a latent vertex alignment, \( \phi : [n] \mapsto [n] \), we wish to understand the extent to which graph matching \( A \) and \( B \) will recover \( \phi \); i.e., if \( P_\phi \) is the permutation matrix corresponding to \( \phi \), will \( \{P_\phi\} = \arg\min_{P \in \Pi(n)} \|A - PBP^T\|_F^2 \)? In order to study this problem from a probabilistic perspective, we introduce a bivariate random graph model with a natural vertex alignment across graphs: the bivariate correlated heterogeneous Erdős-Rényi (ER) random graph model.

**Definition 3.** For \( R \) and \( \Lambda \) symmetric, hollow matrices in \([0,1]^{n \times n}\), we say \( A,B \) are \( R \)-correlated heterogeneous Erdős-Rényi(\( \Lambda \)) random graphs (abbreviated CorrER(\( \Lambda , R \))) if:

i. \( A \) and \( B \) are marginally ER(\( \Lambda \)); i.e., for all \( u, v \in [n] \), \( u < v \), \( A_{uv} \sim \text{Bern}(\Lambda_{uv}) \) and \( B_{uv} \sim \text{Bern}(\Lambda_{uv}) \), with \( A_{uv} = A_{vu} \) and \( B_{uv} = B_{vu} \).

ii. For all \( u, v, w, r \in [n] \), \( u < v \), \( w < r \), it holds that \( A_{uv} \) and \( B_{wr} \) are independent unless \( u = w \) and \( v = r \), in which case the correlation between \( A_{uv} \) and \( B_{uv} \) is \( R_{u,v} \geq 0 \).

Alternatively, if \( A \) and \( B \) are not identically distributed then we define the distribution CorrER(\( \Lambda_1, \Lambda_2, R \)) analogously except that \( A \sim ER(\Lambda_1) \) and \( B \sim ER(\Lambda_2) \). Note that if the graphs are not identically distributed, there are restrictions on feasible correlations: Indeed, if \( X \sim \text{Bern}(p) \) and \( Y \sim \text{Bern}(q) \) are \( \varrho \)-correlated with \( p > q \), then the correlation must satisfy \( \varrho \leq \sqrt{q(1-p)/(p(1-q))} \).

At one extreme, if \( R = [0]_n \) then the graphs are independent ER(\( \Lambda \)), and at the other extreme, if \( R = J_n \) then \( A \) and \( B \) are a.s. isomorphic. For \( 0 < R < 1 \) (entry-wise), the correlation matrix \( R \) endows the pair of random graphs with a natural vertex alignment: namely the identity mapping \( \text{id} : [n] \mapsto [n] \). Moreover, this model generalizes the homogeneous ER model of [Yartseva and Grossglauser, 2013, Kazemi et al., 2015], and similarly allows for the addition of “junk” vertices—those without a probabilistic match across graphs—by setting \( R = R_k \oplus [0]_{n-k} \) for some \( k \leq n \).

We will say that \( (A, B) \sim \text{CorrER}(\Lambda, R) \) are matchable if \( \arg\min_{P \in \Pi(n)} \|AP - PB\|_F = \{I_n\} \), where \( I_n \) denotes the identity matrix. Under modest assumptions on \( \Lambda \), if \( R = J_n \) then \( A \) and \( B \) will be matchable a.s., and if \( R = [0]_n \) then \( A \) and \( B \) will be unmatchable a.s. For more general classes of non-negative \( R \), we seek to understand under what conditions the pair of graphs will be matchable.
In previous work addressing this question, the special case of $R = \varrho J_n$ (for $\varrho \geq 0$) was considered in the case of $\Lambda = p J_n$ in [Pedarsani and Grossglauser, 2011, Lyzinski et al., 2014] (i.e., the homogeneously correlated homogeneous Erdős-Rényi model), for stochastic block-models in [Onaran et al., 2016, Lyzinski, 2016], and for general heterogeneous ER($\Lambda$) in [Lyzinski et al., 2016a]. In each of these cases, the correlation considered was constant across each vertex-pair, with all vertices in $A$ possessing a latent matched pair in $B$. In Section 2, we will generalize these results by considering more general $R$, with a particular focus on the setting in which only a fraction of the vertices in $A$ possess a latent matched pair in $B$, so that $R_{i,j} = 0$ for vertices without a latent matched pair.

1.2 Bivariate Bernoulli Distribution

For the CorrER model, for each $\{u, v\} \in \binom{[n]}{2}$, $A_{uv}$ and $B_{uv}$ can be realized as a bivariate Bernoulli random variable. This will be a key insight in the proof of our main result, Theorem 5. Spelling this out further, a pair of Bernoulli random variables $(X,Y)$ has a BiBernoulli distribution with

$$(X,Y) \sim \text{BiBern} \begin{pmatrix} p_{11} & p_{10} \\ p_{01} & p_{00} \end{pmatrix}$$

if $\mathbb{P}[X = x, Y = y] = p_{xy}$ for each $(x,y) \in \{0,1\}^2$. A key property of BiBernoulli random variables is that they can be generated by a triple of independent Bernoulli random variables. For $(X,Y)$ as above, setting $Z_0 \sim \text{Bern}(p_{11} + p_{10})$, $Z_1 \sim \text{Bern}(p_{01}/(p_{01} + p_{00}))$, and $Z_2 \sim \text{Bern}(p_{11}/(p_{11} + p_{10}))$, with $Z_0, Z_1$ and $Z_2$ independent yields

$$(X,Y) \overset{\text{d}}{=} (Z_0, Z_0 Z_2 + (1 - Z_0) Z_1).$$

The following simple proposition will be useful throughout our proofs.

**Proposition 4.** If $(X,Y)$ are $\varrho$-correlated and identically distributed Bernoulli($p$), then

$$(X,Y) \sim \text{BiBern} \begin{pmatrix} p (p + \varrho (1 - p)) & p (1 - \varrho)(1 - p) \\ p (1 - \varrho)(1 - p) & (1 - p)((1 - p) + \varrho p) \end{pmatrix},$$

and we can realize this pair with independent Bernoulli random variables

$$Z_0 \sim \text{Bern}(p), \ Z_1 \sim \text{Bern}(p(1 - \varrho)), \text{ and } Z_2 \sim \text{Bern}((1 - \varrho)(1 - p)),$$

as above.

A fact we use later is that, with notation as in Proposition 4, we have

$$p (1 - p) \leq \text{Var}(Z_0 + Z_1 + Z_2) \leq 3p (1 - p).$$
2 Core-junk matchability

Often in applications, only a fraction of the vertices in $A$ possess a latent matched pair in $B$ with the remaining vertices having uncorrelated connectivity. In this setting, it holds that $R = R_c \oplus 0_{n_j}$ where $n = n_c + n_j$ and $R_c \in [0, 1]^{n_c \times n_c}$. We call the first $n_c$ vertices the core and the remaining $n_j$ vertices the junk and will refer to this as the core-junk ER model.

In this setting, we ask to what extent is the latent correspondence between the core vertices recoverable via graph matching? Formally, we will now say that a vertex $v$ is matchable if for all $P^* \in \arg\min_{P \in \Pi(n)} \|AP - PB\|_F$ it holds that $P^*_{vv} = 1$. A set of vertices will be matchable if all of the vertices are matchable, so that the core vertices are matchable if all $P^* \in \arg\min_{P \in \Pi(n)} \|AP - PB\|_F$ satisfy $P^* = I_{n_c} \oplus Q^*$ for some $Q^* \in \Pi(n_j)$. While if $n_j = 0$, this core matchability reduces to the classical matchability of Definition 3, as $n_j$ increases we expect that the matched signal present in the core vertices will be obfuscated by the (potentially adversarial) noise present in the junk.

Our main theorem establishes that if the junk in $(A, B) \sim \text{CorrER}(\Lambda, R)$ is essentially white noise, then the core vertices are matchable even in the presence of an almost linear amount of junk.

**Theorem 5.** Let $(A, B) \sim \text{CorrER}(\Lambda, R)$ satisfying

1. $R = R_c \oplus 0_{n_j}$ satisfies $R_c > 0$ entry-wise, corresponding to $n_c$ core vertices out of $n = n_c + n_j$ total vertices;
2. There exists $p \in (0, 1)$ such that $\Lambda = \begin{pmatrix} \Lambda_c & pJ_{n_c, n_j} \\ pJ_{n_j, n_c} & pJ_{n_j, n_j} \end{pmatrix}$.

Let $\mathcal{P}$ be the set of permutation matrices in $\Pi_n$ that permutes the labels of one or more core vertices, and let $\mathcal{Q}$ be the set of permutation matrices in $\Pi_n$ that fix all core-vertex labels. For any fixed permutation $P$ define the random variable $\delta(P) := \|A - BPB^T\|_F^2$.

Define $0 < \epsilon := \min_{i,j \in [n_c]; i \neq j} 2R_{i,j} \Lambda_{i,j}(1 - \Lambda_{i,j})$. There exists a constant $c > 0$ such that if $\epsilon^2 n_c > c n_j \log(n_j)$, then

$$\mathbb{P}\left( \exists P \in \mathcal{P} \text{ s.t. } \forall Q \in \mathcal{Q}, \delta(P) \leq \delta(Q) \right) = \exp\left\{-O\left(\epsilon^2 n_c\right)\right\}.$$ 

A consequence of Theorem 5 is that if $\epsilon$ is bounded below in $n$, then even in the presence of a nearly linear (in core size) amount of junk noise vertices, an exact solution to the graph matching problem will, with high probability, correctly align the core vertices. While, in general, exactly solving the graph matching problem is intractable, this result underlies the
potential efficacy of a well-performing approximate graph matching procedure for aligning cores across networks.

The key to the proof of Theorem 5 is the fact that the core vertices are stochastically matched (see Definition 12). This ensures that, in expectation, any fixed permutation of the core vertex labels of $B$ induces more edge disagreements than the optimal permutation fixing the core vertices.

**Lemma 6.** Let $(A, B) \sim \text{CorrER}(\Lambda, R)$ with $R = R_c \oplus 0_{n_j}$ with $R_c \in (0, 1]^{n_c \times n_c}$ for some $n_j$ and $n_c$. Let $P$ be a permutation matrix in $\Pi_n$ that permutes the labels of precisely $k$ core vertices (i.e., $\sum_{i=1}^{n_c} P_{i,i} = n_c - k$), and let $T_c$ denote the number of transpositions between core vertices induced by $P$. For $0 < \epsilon = \min_{i,j \in [n_c], i \neq j} 2R_{i,j}\Lambda_{i,j}(1 - \Lambda_{i,j})$, it follows that

$$
E\delta(P) - E\delta(I_n) = E\|A - PBPT\|_F^2 - E\|A - B\|_F^2 \geq \epsilon \left((n_c - k)k + \left(\frac{k}{2}\right) - T_c\right).
$$

An application of a McDiarmid-like inequality from [Kim et al., 2002] (see Proposition 15) then ensures that the number of excess edge disagreements induced by a core-shuffling permutation concentrates tightly about its mean. Combining the above, a union bound finishes the proof. The proof details are provided in the Appendix A.1.

**Remark 7.** Without the white noise junk assumption, the proof of Theorem 5 extends almost immediately to prove that there exists a constant $c > 0$ such that if $\epsilon^2 n_c > cn_j^2 \log(n_j)$, then

$$
P\left( \exists P \in \mathcal{P} \text{ s.t. } \forall Q \in \mathcal{Q}, \delta(P) \leq \delta(Q) \right) = \exp\{-O(\epsilon^2 n_c)\}. 
$$

This result makes no assumptions on the structure of the junk vertices or the core–to–junk edge structure beyond $(A, B) \sim \text{CorrER}(\Lambda, R)$ (i.e., $A$ and $B$ are identically distributed). Specifically, note that Lemma 13 makes no assumption on the $\Lambda$ but our reliance on equivalence classes in the our main proof fails (see Eq. (9) and the argument that follows). We expect that various structured noise assumptions, such as a stochastic blockmodel structure, would allow us to improve upon the junk order assumption (see also the discussion of heterogeneity in Section 6). Indeed, this is exemplified in the improved bound on junk size achieved in the white-noise junk setting—which is due to exploitable symmetries amongst the junk vertices.

### 3 Core matching in graphs of differing orders

Often, in practice, it is the case that $A$ and $B$ have differing numbers of vertices. In [Fishkind et al., 2017], this obstacle is overcome via a novel padding scheme in which isolated ver-
tices are appended to the (rewighted) smaller network before subsequently matching to the (rewighted) larger network. If $A$ is $n_c \times n_c$ and $B$ is $(n_c + n_j) \times (n_c + n_j)$, the modified padding procedure would seek the optimal permutation aligning $\tilde{A} = (2A - J_{n_c}) \oplus [0]_{n_j}$ to $\tilde{B} = 2B - J_{n_c+n_j}$. In this setting, the graph matching problem is formulated as

$$\min_{P \in \Pi_{n_c+n_j}} \| \tilde{A} - P\tilde{B}P^T \|_F.$$  

The effect of this padding scheme versus naively matching $A \oplus [0]_{n_j}$ to $B$ is immediate: matching $\tilde{A}$ to $\tilde{B}$ has the effect of matching $A$ to the best fitting induced subgraph of $B$ whereas matching $A \oplus [0]_{n_j}$ to $B$ effectively matches $A$ to the best fitting subgraph of $B$.

In the present core-junk matchability setting, we seek to understand how well these padding schemes will align a core $A$ to its corresponding matchable subgraph in $B$. Below, we tease out one of the principle gains of matching induced subgraphs versus subgraphs: namely a robustness of matchability to adversarial noise; see Theorems 8 and 9.

**Theorem 8.** Let $A, \tilde{B} \sim \text{CorrER}(R, \tilde{\Lambda})$ with $R, \tilde{\Lambda} \in [0, 1]^{n_c \times n_c}$. Let $B \sim \text{ER}(\Lambda)$ with $\Lambda \in [0, 1]^{(n_c+n_j) \times (n_c+n_j)}$, and suppose the principal $n_c \times n_c$ submatrices of $B$ and $\Lambda$ are $\tilde{B}$ and $\tilde{\Lambda}$, respectively. Define $\tilde{A} = (2A - J_{n_c}) \oplus [0]_{n_j}$ and $\tilde{B} = (2B - J_{n_c+n_j})$. If $M = \{ P \in \Pi_{n_c+n_j} \text{ s.t. } \sum_{i=1}^{n_c} P_{i,i} = n_c \}$, then

i. If $\Lambda \in [\alpha, \beta]^{(n_c+n_j) \times (n_c+n_j)}$ and there exists an $\epsilon > 0$ such that for all $\{u, v\} \in \binom{[n_c]}{2}$, $R_{u,v} \geq 1/2 + \epsilon$, then (with $\gamma := \min(\alpha(1-\alpha), \beta(1-\beta)))$

$$\frac{\log(n_j + n_c)}{\epsilon^2 n_c \gamma^2} = o(1)$$

implies that

$$\mathbb{P} \left( \exists P \in \Pi_{n_c+n_j} \setminus M \text{ s.t. } \| \tilde{A} - P\tilde{B}P^T \|_F \leq \| \tilde{A} - \tilde{B} \|_F \right) = 2 \exp \left\{ -O(\epsilon^2 n_c \gamma^2) \right\}.$$

ii. If $\Lambda = pJ_{n_c+n_j}$ and $R = qJ_{n_c}$, then

$$\frac{\log(n_j + n_c)}{q^2 n_c p^2 (1-p)^2} = o(1)$$

implies that

$$\mathbb{P} \left( \exists P \in \Pi_{n_c+n_j} \setminus M \text{ s.t. } \| \tilde{A} - P\tilde{B}P^T \|_F \leq \| \tilde{A} - \tilde{B} \|_F \right) = 2 \exp \left\{ -O(q^2 n_c p^2 (1-p)^2) \right\}.$$
This Theorem is proved in Appendix A.2.

In the general setting of Theorem 8 part i., there are numerous conditions on \( R \) that yield a similar conclusion as stated in the Theorem, namely that asymptotically almost surely (a.a.s.) the optimal matching correctly aligns \( A \) to the core vertices of \( B \). For example, if there exists an \( \epsilon > 0 \) such that, entry-wise, \( R \geq 1 + \epsilon - 4\alpha(1 - \beta) \), then \( \frac{\log(n_j + n_c)}{\epsilon^2 n_c} = o(1) \) suffices (see Eq. (10)).

In addition to its simplicity, we chose to present the \( R \geq 1/2 + \epsilon \) entry-wise condition to further highlight the advantages of the modified padding scheme over the naive padding scheme defined by matching \( \tilde{A} = A \oplus [0]_{n_j} \) to \( \tilde{B} = B \). Stated simply, while the modified padding scheme of Theorem 8 will a.a.s correctly align \( A \) with the core of \( B \) if \( R > 1/2 \) entry-wise, under the above naive padding scheme if \( R \) is bounded away from 1, there exists a choice of \( \tilde{\Lambda} \) such that a.a.s. the correct alignment across cores is not the optimal GM permutation. Indeed, by matching \( A \) to the best fitting induced subgraph of \( B \), the modified padding scheme achieves a robustness to adversarial junk vertices that the naive padding scheme cannot. Formally, we have the following:

**Theorem 9.** Let \( A, \tilde{B} \sim \text{CorrER}(R, \tilde{\Lambda}) \) with \( R, \Lambda \in [0, 1]^{n_c \times n_c} \). Let \( B \sim \text{ER}(\Lambda) \) with \( \Lambda \in [0, 1]^{(n_c + n_j) \times (n_c + n_j)} \), and the principal \( n_c \times n_c \) submatrices of \( B \) and \( \Lambda \) are \( \tilde{B} \) and \( \tilde{\Lambda} \), respectively. Define \( \tilde{A} = A \oplus [0]_{n_j} \) and \( \tilde{B} = B \). Let \( \mathcal{M} \) be as in Theorem 8. If \( n_c < n_j \) and there exists constants \( \varrho, \beta > 0 \) such that (entry-wise) \( \tilde{\Lambda} \leq \beta < 1 \), \( \tilde{\Lambda} = \omega(n_c^{-1} \log n_c) \), and \( R \leq \varrho < 1 \), then there exists a choice of \( \Lambda \) such that

\[
P \left( \exists P \in \Pi_{n_c + n_j} \setminus \mathcal{M} \text{ s.t. } \| \tilde{A} - P \tilde{B} P^T \|_F \leq \| \tilde{A} - \tilde{B} \|_F \right) = 1 - o(1). \tag{4}\]

This Theorem is proved in Appendix A.3.

### 4 Identifying core versus junk vertices

In Theorem 5, we established that under mild model assumptions, graph matching will a.s. correctly match the core vertices across graphs. We next establish that by considering the errors induced by the matching, these core vertices can be correctly identified post-matching. The main tool that we utilize is a graph matching variant of the permutation test.

Testing whether a vertex \( v \) is a core or junk vertex amounts to the testing the hypotheses

\[
H_0 : \quad \forall P \in \mathcal{P}, u \neq v, \quad \text{corr}(A_{vu}, (PB P^T)_{vu}) = 0, \\
versus \quad H_A : \quad \exists P \in \mathcal{P}, u \neq v, \quad \text{corr}(A_{vu}, (PB P^T)_{vu}) > 0. \tag{5}
\]
A natural test statistic for the hypotheses in Eq. (5) would seem to be the empirical correlation between row $v$ of $A$ and $P^*BP^{*T}$. However, note that depending on the structure of $\Lambda$, even if the theoretical correlation is 0, the empirical correlation could be large because of variance in the entries of $\Lambda$. Additionally, since $P^*$ is chosen to be the best permutation, even the theoretical correlation between rows of $A$ and $P^*BP^{*T}$ will be positive for all rows, though it will be smaller for junk vertices than core vertices.

In light of the theoretical difficulties presented by the correlation induced by $P^*$, rather than testing Eq. (5) via empirical correlation estimates, we will make use of the following relationship between edge-wise correlation and the number of induced errors between $A$ and $P^*BP^{*T}$. Namely, if $v$ is a core vertex then the number of errors induced by $P^*$ across the neighborhoods of $v$ in $A$ and $B$ (i.e., $\|(AP^*-P^*B)_{v,*}\|_1$) should be significantly smaller—due to the correlation structure provided by $R$—than the number of errors induced by a randomly chosen permutation $P$ (i.e., $\|(AP-PB)_{v,*}\|_1$). If $v$ is a junk vertex then, even with the nontrivial correlation induced by $P^*$, we will see that $\|(AP^*-P^*B)_{v,*}\|_1$ is significantly closer to the number of errors induced by a randomly chosen permutation $P$ (i.e., $\|(AP-PB)_{v,*}\|_1$) than in the core setting. With this in mind, we define $\Delta_v(P) = \|(AP-PB)_{v,*}\|_1$ and let $E_P$ and $\text{Var}_P$ denote the conditional expectation and variance of these quantities with respect to uniform sampling of $P$ over all permutation matrices. We will use the permutation-test inspired statistic

$$T(v, P^*) := \frac{\Delta_v(P^*) - E_P \Delta_v(P)}{\sqrt{\text{Var}_P \Delta_v(P)}}. \quad (6)$$

Rather than formalizing the test procedure based on rejecting the $H_0$ of Eq. (5) for suitably large values of $|T(v, P^*)|$—which would necessitate determining or estimating the critical region for $T(\cdot)$—we will instead use the test statistic $T(\cdot)$ to rank the vertices based on the likelihood they are core vertices. This procedure is outlined in Algorithm 1. In Sections 5, we will demonstrate the effectiveness of Algorithm 1 in both simulated and real data settings. A detailed theoretical analysis of Algorithm 1 is beyond the scope of the present work, and will be pursued in a follow-up paper.

**Remark 10.** The above method for classifying core versus junk vertices has an immediate application to the problem of across graph vertex nomination [Fishkind et al., 2015, Lyzinski et al., 2016b]. Consider the setting where i. either a few vertices of $A$ are of interest; or ii. all the vertices in $A$ are of interest. If the task is to find these interesting vertices in $B$ (or determine if they are not present), we can use $T(v)$ to i. determine whether the vertices of interest are core (i.e., present in $B$) or junk vertices (i.e., not present in $B$); or ii. rank the vertices of $B$ based on how likely they are to be core vertices. Intuitively, the larger $|T(v)|$,
Algorithm 1 Finding the core vertices

**Input:** Adjacency matrices $A, B \in \{0, 1\}^{(n_c+n_j) \times (n_c+n_j)}$; available seeded vertices $S$

1. Use available seeded vertices to match $A$ and $B$ yielding optimal permutation $P^*$;
2. Use $P^*$ to approximately compute $T(v, P^*)$ for each vertex $v$;
3. Rank the vertices via decreasing value of $|T(v, P^*)|$, which ranks them by decreasing likelihood of being core vertices;

**Output:** Rank list of vertices, with those at the top more likely to be core vertices

the more likely $v$ is to be a core vertex. In this setting, the sorted rank list based on $|T(\cdot)|$ gives us a nomination list, with vertices ranked highly those more likely to be present in graph $B$.

5 Experimental results

For our experimental results we employed the seeded graph matching (SGM) algorithm as described in [Fishkind et al., 2017]. As described above, this algorithm incorporates seeded vertices, which are a set of vertices for which the correspondence between the two graphs is assumed known. We use seeds for two reasons. First, in practice seeds may be available from other non-network information known about the vertices. Second, the SGM algorithm is not guaranteed to find the global minimum of the objective function Eq. (1) and hence is not guaranteed to demonstrate the results of our theory. By incorporating seeds, the algorithm is effectively pushed towards what our theory deems should be the global optimum. Altogether, the seeded graph matching algorithm helps to illustrate our theory and the gap between the theoretical results and the algorithmic performance.

5.1 Simulations

For our simulations, we will consider two model settings, homogeneous Erdős-Rényi (ER) and random dot product graphs (RDPG) [Nickel, 2006, Young and Scheinerman, 2007]. In both settings, the adjacency matrices will be distributed as $\text{CorrER}(\Lambda, R)$ where $\Lambda$ will have a structure as determined by the model. For the correlations we will take $R = gJ_{n_c} \oplus 0_{n_j}$ to simulate the core-plus-junk setting.
Figure 1: Core matching accuracy on the log-scale as a function of the number of junk vertices and the number of seeds, for various edge probability $p$ for correlated homogeneous Erdos-Renyi graphs with $\varrho = 0.5$.

5.1.1 Erdős-Rényi

For the ER case, we let $A, B \sim \text{CorrER}(pJ_n, R)$ where $p \in [0, 1]$, and $R = \varrho J_{n_c} \oplus 0_{n_j}$, corresponding to $n_c$ core vertices and $n_j$ junk vertices.

In order to investigate the empirical impact of the number of junk vertices $n_j$, the number of seeded core vertices $n_s$, and the edge probability $p$, we simulated 1000 graphs for each of $p \in \{0.1, 0.2, 0.5\}$ and each $n_j \in \{0, 5, 10, 20, 50\}$. Each graph was taken to have $n = 250$ vertices. For each graph, we varied the number of seeds with $n_s \in \{0, 5, 10, 20, 50\}$. In Figure 1 we fixed $\varrho = 0.5$ to illustrate the impact of the other parameters.

We plot the accuracy of matching non-seed core vertices, $\frac{1}{n_c - n_s} \sum_{i \in C \setminus S} P_{ii}$, where $S$ is the set of seed vertices and $P$ is the permutation matrix found by the algorithm. Note the accuracy is plotted on a log scale. The horizontal axis is the number of junk vertices, $n_j$. The line labeled as “Chance” is the core matching accuracy for a random permutation in the case when there are zero seeds, $\frac{1}{n_c} = \frac{1}{250 - n_j}$. This “Chance” accuracy would be slightly higher with seeds but as Figure 1 demonstrates, the comparison is only close for the 0 seed case. Confidence intervals for the mean accuracy based of the normal approximation are shown at 95% confidence as vertical lines at each point (note some confidence intervals are too short to be visible).

As expected, performance degrades as the number of junk vertices increases, however the degradation rate depends strongly on the number seeds and the edge probability. When
the number of seeds was 50, this was sufficient so that regardless of the number of junk vertices, the core matching was quite accurate. On the other hand with 0 seeds, the number of junk vertices again had a minimal impact as the performance with no junk seeds was quite poor. Again we want to emphasize that the performance with no seeds is due to algorithmic challenges rather than theoretical limitations. For moderate number of seeds performance degrades with the number of junk though not especially quickly.

In general the performance of the algorithm increased with $p$, especially for seed levels of 10 and 20. We also simulated graphs with correlations of $\rho = 0.9$ where only 5 seeds were needed to achieve near perfect accuracy at all junk levels. On the other hand, if the correlation $\rho = 0.2$, the accuracy was poor at all seed and junk levels.

In addition to considering the matching accuracy, we also evaluated the performance of our core identification algorithms described in section 4. For these simulations, we used the test statistic $T(v)$, see Eq (6), estimated by sampling permutations. We ranked the vertices according to $T(v)$ and classified each vertex as either core or junk based on the ranking and the known number of junk vertices $n_j$. Seeded vertices were not ranked.

Figure 2 shows the mean precision of this procedure at each rank, with low ranks being correct if that vertex was a core vertex and ranks near $n$ being correct if that vertex was a junk vertex. Here we only show results for $n_j \in \{20, 50\}$ and $n_s \in \{5, 10, 20\}$ in order
to illustrate the parameter values where the changes were most substantial. Note that for small number of junk vertices, even a random algorithm will achieve high precision for core vertices but it will have very poor precision for junk vertices. This chance performance is indicated by the horizontal lines for each of core and junk. When the number of seeds was either 5 or 10 the precision for junk vertices was very close to chance but if \( n_s = 20 \), the junk precision increased substantially in all cases. Core performance had similar behavior though performance at low ranks was substantially improved when \( n_s = 10 \) as compared to \( n_s = 5 \).

### 5.1.2 Padding in the Erdős-Rényi regime

To investigate the impact that order inhomogeneity can have on matching accuracy, we consider matching \( A \sim \text{ER}(50, p) \) to \( B \sim \text{ER}(50/r, p) \) for \( p \in [0.1, 0.3, 0.5] \) and \( r \in \)
In each experiment, we correlate $A$ to a 50 vertex subgraph in $B$, varying the correlation over $[0.1, 0.5, 0.9]$, and consider matching $A$ and $B$ using the SGM algorithm run with $n_s = [5, 10, 15, 20]$ seeds. For each choice of parameters, 25 Monte Carlo replicates were run, and results are plotted in Figure 3. From the figure, we see that matching accuracy—here defined as the fraction of the $50 - n_s$ unseeded vertices in $A$ correctly matched to their correlated pair in $B$—improves monotonically as correlation and $r$ increase for each choice of $n_s$ and $p$. This is unsurprising, as increasing the correlation and $r$ increases the signal present in the cores (relative to the noise in the junk) across networks.

We note also the uniformly poor performance in the correlation equal to 0.1 regime. While at first glance this seemingly contradicts the perfect matching theoretical guarantees provided by Theorem 8, we emphasize here that the theory holds asymptotically and the combination of $\rho$ and $n_c$ here is too small for the theoretical guarantees to kick in. Indeed, we observe that the matchings we obtain in the $\rho = 0.1$ case are better—according the the graph matching objective function—than the true labeling across the networks. The uniformly poor performance in matching accuracy in this low correlation regime is not an algorithmic failing, rather it is an indication that these graphs were not matchable [Lyzinski, 2016]; i.e., the true labeling is not optimal in the graph matching objective function. We are presently exploring methodologies for detecting this non-matchability.

### 5.1.3 Random Dot Product Graphs

To analyze the effect of vertex heterogeneity including degree heterogeneity and mixed membership community structure, we simulated correlated graphs from the random dot product graph distributions (RDPG). For the RDPG case, we have that $A, B \sim \text{CorrER}(X X^T, R)$ where $X = [X_1, X_2, \ldots, X_n]^T \in \mathbb{R}^{n \times d}$ is the matrix of latent positions and $X_1, \ldots, X_n \overset{iid}{\sim} F$ for some distribution $F$ satisfying $\mathbb{P}[X_i^T X_j \in [0, 1]] = 1$ for all $i, j$. Note that $A_{ij}, B_{ij} \sim \text{Bern}(X_i^T X_j)$.

**Remark 11.** This simulation does satisfy the assumptions of Theorem 5 because the junk vertices do not have a homogeneous Erdos-Renyi structure. Nonetheless, this simulation setting is more realistic than the Theorem setting since we expect junk vertices to have heterogeneous

We will take $F$ to be a the $(d - 1)$-dimensional marginal of a Dirichlet distribution on $d$ dimensions with parameter $\alpha = s \mathbf{1}$ for some $s > 0$. Note that when $\alpha$ is large, this distribution will concentrate around the point $(1/d, \ldots, 1/d)$ which has squared norm $(d - 1)/d^2$. Hence, when $s$ is large the distribution will approximate the homogeneous Erdős-
Figure 4: Three example graphs from RDPG distributions where the latent positions are distributed as the 3-dimensional marginals of a 4-dimension Dirichlet(α1) distribution. The vertices were sorted according to a community finding algorithm [Clauset et al., 2004] and according to degree. When α is small there is more community structure and degree heterogeneity which will impact the performance of graph matching and core detection.

Rényi case with parameter \( p = (d - 1)/d^2 \). These differences are illustrated in Figure 4 From this perspective, this simulation will illuminate the impact of vertex heterogeneity on the performance of core matching.

Figure 5 shows the matching accuracy as a function of the junk vertices for various seed levels. The dimension \( d \) was chosen to be 3. Each panel corresponds to a different choice of \( \alpha \in \{1, 2, 32\} \). Here it is evident that the overall patterns are similar to the ER case but we note that the performance increases substantially as the scale decreases. This is intuitive as the vertices are behaving more heterogeneously and so should be more easily matchable. The behavior for the mean precision was similar and hence we omit the results here.

### 5.2 Twitter data

In order to analyze the core-junk setting with real data, we considered two graphs derived from Twitter.\(^1\) The two graphs are based on the most active twitter users from April and May 2014. The graphs are weighted according the log of the number times a user mentioned another user during the given month. We use the same algorithm for the weighted graphs as for the unweighted graphs above, which seeks to minimize the same objective function. The empirical Pearson correlation between the entries in the two weighted adjacency matrices is

\(^1\)These graphs were provided as part of the DARPA XDATA project.


\[ \alpha = 1 \quad \alpha = 2 \quad \alpha = 32 \]

\[ 0 \quad 10 \quad 20 \quad 30 \quad 40 \quad 50 \]

\[ 0.01 \quad 0.10 \quad 1.00 \]

number of junk vertices
core matching accuracy
# seeds
20
10
5
0
Chance

Figure 5: Core matching accuracy on the log-scale as a function of the number of junk vertices and the number of seeds, for vertex heterogeneity parameters in the random dot product graph model described in Section 5.1.3.

\[ \approx 0.91. \] After keeping only the largest common connected component, the number of users was 431 in each graph.

Since all users were present in both graphs we mimicked the core-junk setting by sampling three disjoint sets of users $C$, $J_{April}$ and $J_{May}$ of sizes $n_c$, $n_j$ and $n_j$, respectively. We then set one graph to be the induced subgraph of the April graph corresponding to the vertices $C \cup J_{April}$. The second graph is the induced subgraph of the May graph corresponding to vertices in $C \cup J_{May}$. Hence, the core vertices in $C$ appear in both graphs and the remaining vertices appearing in only one graph are the junk vertices. The weighted adjacency matrices are shown in Figure 6.

Figure 7 shows the average matching accuracy across 1000 Monte Carlo replicates of graphs sampled with $n = 250$ and $n_j \in \{5, 10, 20, 50\}$. Each graph was matched with seed levels of $n_s \in \{0, 5, 10, 20, 50\}$ with the seeds chosen uniformly at random from the set of core vertices. The impact of increasing junk vertices from $n_j = 5$ to $n_j = 50$ led to an absolute decrease in the accuracy by between 19% and 24%, for $n_s = 50$ and $n_s = 0$, respectively.

Figure 8 shows the precision for identifying core and junk vertices for junk levels of $n_j = 20$ and 50. The seeds levels are $n_s = 20$ and 50. We also included an Oracle, where all core vertices were used as seeds and the remaining junk vertices were matched to themselves. We then used the same identification procedures after throwing away the seed information. This curve gives an upper bound for the precision, since all core vertices are correctly matched,
Figure 6: Weighted adjacency matrices of aligned Twitter graphs from April and May. The vertices were sorted according to a community finding algorithm [Clauset et al., 2004] and according to weighted degree.

however it still incorporates the fact that matching junk vertices to themselves introduces a small amount of empirical correlation between the junk vertices.

As can be seen in Figure 6, many vertices in this data set have very similar connectivity patterns. Hence if two different vertices that behave similarly are included as junk vertices in the two different graphs, these will likely be matched to each other and it may be hard to determine whether they are core or junk vertices.

6 Discussion

Understanding the limits of graph matchability is an essential step in robust multiple graph inference regimes. When graphs are not matchable—i.e., the true node correspondence cannot be recovered in the face of noise—paired graph inference methodologies that utilize the across graph correspondence (see, for example, [Tang et al., 2016, Asta and Shalizi, 2014]) cannot gainfully be employed. Nonmatchability limits analysis to methods which rely on graph statistics which are invariant to relabeling of the vertices, which can be useful but lack the full power of other methods. In this paper, we make the first foray into graph matchability when only a fraction of the graphs are matchable or when the graphs are of differing orders. Our theoretical results and subsequent simulations and experiments provide a basis for a deeper understanding the effect that junk vertices have on core matchability.

The information and computational limits for graph matching are still open problems for which we have pushed the boundaries, but significant more work is to be done. These
problems are in analogy to the recently addressed problems of detection and recovery for the planted partition and planted clique problems for a single graph [Feldman et al., 2013, Mossel et al., 2014, Abbe and Sandon, 2015]. For these settings, exact fundamental limits have been established and polynomial time algorithms have been shown to achieve or nearly achieve these limits. Obtaining similar results for the graph matching problem are key steps towards a robust statistical framework for multiple graph inference.

For the present work we do not expect that our results represent these fundamental limits. As an informal argument we can consider known results for the quadratic assignment problem for iid entries [Cela, 2013]. In the uncorrelated dense homogeneous Erdős-Rényi setting, these results imply that the best best solution to the graph matching problem, Eq. (1) will reduce the objective function $\Theta(n_j^{3/2})$ as compared to a random guess. On the other hand, in the dense homogeneous Erdős-Rényi setting with constant correlation, Lemma 13 shows that the best solution is $\Theta(n_c^2)$ better than a random guess. Under the heuristic that in order for the core vertices to be matched correctly, the signal from correlation must be greater than the possible improvements in the all noise setting, we can conjecture that a core matchability threshold at approximately $n_c = O(n^{3/4})$ may be possible. Using our present proof technique, we are unable to achieve this rate. This heuristic argument, though problematic, provides a potential guidepost for future work.

A key limitation of our current result is that they primarily apply in more homogeneous
settings. In particular, Theorem 5 assumes the junk is homogeneous and that the two graphs are identically distributed and Theorem 8 assumes that there is limited heterogeneity in the core and junk. Furthermore, using the naive padding scheme of Theorem 9, even allowing minimal heterogeneity in the junk can lead to graph matching failing with high probability.

A key difference between the naive padding and the improved padding schemes is that the adjacency matrices are appropriately centered. This suggests that an optimal centering may allow for improved theoretical results. Indeed, if both $A$ and $B$ are centered by their expectations than it can be easily seen that if $R > 0$ that $\mathbb{E}[(A_{uv} - \mathbb{E}[A_{uv}])(B_{wr} - \mathbb{E}[B_{wr}])] > 0$ if and only if $\{u, v\} = \{w, r\}$. Hence, if these expectations were known, matchability would be possible even if the networks are not identically distributed. Achieving similar results by estimating these expectation is another important area for future work.

While the results presented herein are for the case that the graphs are simple undirected graph with no edge-weights, the graph matching framework of Definition 1 is flexible, allowing us to accommodate many of the features—both those considered above, and additional eccentricities—innherent to real data settings. In the weighted, loopy setting, the matching can occur between the weighted adjacency matrices or normalized Laplacian matrices, $\mathcal{L}_A = D_A^{-1/2} A D_A^{-1/2}$ (where $D_A$ is the diagonal matrix with $(i, i)$-th entry equal to $\sum_j D_A(i, j)$) and
the similarly defined $L_B$. To match directed graphs, the graphs can either be made undirected (for example, by matching $A' = (A + A^T)/2$ to $B' = (B + B^T)/2$) or the directed adjacency matrices can be directly plugged into Eq. (1). Developing similar results to Theorems 5 and 8 in models (akin to our CorrER model) that incorporate these graph features as well as additional vertex and edge features is a natural next step.

Acknowledgments

Dr. Sussman’s contribution to this work was partially supported by a grant from MIT Lincoln Labs and the Department of Defense. We would also like to thank Dr. Carey E. Priebe for his helpful discussions in the writing of this manuscript.

A Proofs

A.1 Proof of Theorem 5

In this section we prove Theorem 5. We start by introducing the key concept of $\epsilon$-stochastically matched sequences of random variables.

**Definition 12.** A pair of sequences $X$ and $Y$ are $\epsilon$-stochastically matched if for all $k$ and for all permutations $\sigma$ where $|\{i : \sigma(i) \neq i\}| = k$, it holds that

$$
\mathbb{E}\|X - \sigma Y\|_1 - \mathbb{E}\|X - Y\|_1 > k\epsilon.
$$

The key realization in the present setting of graph matching is the following. If $A$ and $B$ are the respective adjacency matrices of CorrER($\Lambda, R$) random graphs and if vec($A$) and vec($B$) are $\epsilon$-stochastically matched for suitable $\epsilon$, then $A$ and $B$ are matchable; i.e.,

$$\{I_n\} = \text{argmin}_{P \in \Pi(n)} \|AP - PB\|_F.$$

In this sense, $\epsilon$-stochastically matched is the vectorized version of graph matchability; see [Lyzinski et al., 2016b] for an early realization of this relationship. Exploiting this relation, we arrive at the following Lemma.

**Lemma 13.** Let $(A, B) \sim \text{CorrER}(\Lambda, R)$. Let $P$ be a permutation matrix in $\Pi_n$ that permutes precisely $k$ labels (i.e., $\sum_i P_{i,i} = n - k$), and let $T$ denote the number of transpositions induced by $P$. For $0 < \epsilon = \min_{i,j; i \neq j} 2R_{i,j}\Lambda_{i,j}(1 - \Lambda_{i,j})$, it follows that

$$
\mathbb{E}\delta(P) - \mathbb{E}\delta(I_n) = \mathbb{E}\|A - PBP^T\|_F^2 - \mathbb{E}\|A - B\|_F^2 \geq \epsilon \left( (n-k)k + \left(\frac{k}{2}\right) - T \right).
$$

21
Proof. With \( P \) as above, let \( \tau \) be a permutation in \( S_{(2)} \) such that \( \text{vec}(PBPT) = \tau(\text{vec}(B)) \). Let

\[
\tau = \tau_1 \circ \tau_2 \circ \cdots \circ \tau_\ell
\]

be the disjoint cycle decomposition of \( \tau \), and for each \( i \in [\ell] \), let \( t_i = |\tau_i| \) denote the length of the \( i \)-th cycle and let \( T_i \) denote the indices permuted by \( \tau_i \). Letting \( T \) be the number of transpositions induced by \( P \), we have that

\[
k_\tau := |\{ i : \tau(i) \neq i \}| = (n - k)k + \binom{k}{2} - T.
\]

For ease of notation, let \( X = \text{vec}(A) \) (resp., \( Y = \text{vec}(B) \)) and for each \( i \in \left( \binom{n}{2} \right) \), let \( p_i = \mathbb{E}(X_i) = \mathbb{E}(Y_i) \) so that (where, slightly abusing notation, we let \( \varrho_i = \text{corr}(X_i, Y_i) \))

\[
\mathbb{E} \| A - PBPT \|_F^2 - \mathbb{E} \| A - B \|_F^2 = \mathbb{E} \| X - \tau Y \|_1 - \mathbb{E} \| X - Y \|_1
\]

\[
\mathbb{E} \| A - PBPT \|_F^2 - \mathbb{E} \| A - B \|_F^2
\]

\[
= \sum_{i=1}^{\ell} \sum_{j \in T_i} \left( p_j + p_{\tau(j)} - 2p_jp_{\tau(j)} - 2(1 - \varrho_j(1 - p_j)p_j) \right)
\]

\[
= \sum_{i=1}^{\ell} \sum_{j \in T_i} \left( 2\varrho_jp_j(1 - p_j) + 2p_j(p_j - p_{\tau(j)}) \right)
\]

\[
= \sum_{i=1}^{\ell} \sum_{j \in T_i} 2\varrho_jp_j(1 - p_j) + \sum_{i=1}^{\ell} \sum_{j \in T_i} 2p_j(p_j - p_{\tau(j)}) \geq 0
\]

\[
\geq k_\tau \epsilon.
\]

The second term in Eq. (8) is non-negative by the rearrangement inequality.

Moreover, Lemma 13 can be immediately translated into the core-junk setting of Theorem 5, as noted in Lemma 6 which we restate and prove here.

**Lemma 14** (see Lemma 6). Let \((A, B) \sim \text{CorrER}(A, R)\) with \( R = R_c \oplus 0_{n_j} \) with \( R_c \in (0, 1]^{n_c \times n_c} \) for some \( n_j \) and \( n_c \). Let \( P \) be a permutation matrix in \( \Pi_n \) that permutes the labels of precisely \( k \) core vertices (i.e., \( \sum_{i=1}^{n_c} 1_{i, i} = n_c - k \)), and let \( T_c \) denote the number of transpositions between core vertices induced by \( P \). For \( 0 < \epsilon = \min_{i,j \in [n_c]} 2R_{i,j} \Lambda_{i,j}(1 - \Lambda_{i,j}) \), it follows that

\[
\mathbb{E} \delta(P) - \mathbb{E} \delta(I_n) = \mathbb{E} \| A - PBPT \|_F^2 - \mathbb{E} \| A - B \|_F^2 \geq \epsilon \left( (n_c - k)k + \binom{k}{2} - T_c \right).
\]
Proof of Lemma 6. Following the notation of the proof of Lemma 13, we note that the only entries of $X$ and $Y$ that are correlated via $R$ are those that correspond to edges between core vertices. If $P$ permutes the labels of $k$ core vertices, then $\sigma$ permutes the labels of
\[ k_\sigma = k(n_c - k) + \binom{k}{2} - T_c \]
entries of $Y$ corresponding to core edges. The proof then proceeds exactly as the proof of Lemma 13.

Combining the expectation bounds with the following McDiarmid-like concentration result will yield the proof of Theorem 5.

Proposition 15 (Proposition 3.2 from [Kim et al., 2002]). Let $X_1, \ldots, X_m$ be a sequence of independent Bernoulli random variables where $\mathbb{E}[X_i] = p_i$. Let $f : \{0, 1\}^m \rightarrow \mathbb{R}$ be such that changing any $X_i$ to $1 - X_i$ changes $f$ by at most
\[ M = \sup_i \sup_{X_1, \ldots, X_n} |f(X_1, \ldots, X_i, \ldots, X_n) - f(X_1, \ldots, 1 - X_i, \ldots, X_n)|. \]
Let $\sigma^2 = M^2 \sum_i p_i(1 - p_i)$ and let $Y = f(X_1, \ldots, X_n)$.

Then
\[ \mathbb{P}[|Y - \mathbb{E}[Y]| \geq t\sigma] \leq 2e^{-t^2/4} \]
for all $0 < t < 2\sigma/M$.

Proof of Theorem 5. Let $C = [n_c]$ be the set of core vertices, and let $J = [n] \setminus [n_c]$ denote the set of junk vertices. For a given permutation $\tau$ on $[n]$, we define the permutation $\tau_{id}$ uniquely as follows:
\[ \tau_{id}(i) = \begin{cases} i, & \text{if } i \in C \\ \tau^k(i), & \text{if } i \in J, \text{ where } k = \min\{\ell \geq 1 : \tau^\ell(i) \in J\}. \end{cases} \]
(9)

For example, if $n_c = 4$, $n_j = 4$, and
\[ \tau = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 3 & 6 & 7 & 1 & 8 & 5 & 4 & 2 \end{pmatrix}, \]
then
\[ \tau_{id} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 8 & 5 & 7 & 6 \end{pmatrix}. \]

For a permutation matrix $P$, we define $P_{id} \in Q$ analogously, where we recall here that $Q$ is the set of permutation matrices $Q$ in $\Pi_n$ satisfying $\sum_{i=1}^{n_c} Q_{i,i} = n_c$ (i.e., fixing all core
labels), and \( P = \Pi_n \setminus Q \). Since the event that there exists \( P \in P \) such that for all \( Q \in Q \), \( \delta(P) \leq \delta(Q) \), is a subset of the event that there exists \( P \) such that \( \delta(P) \leq \delta(P_{id}) \), it is sufficient to upper bound the probability of the latter event.

Suppose that \( P \in \Pi_n \) (with corresponding permutation \( \tau \in S_n \)) permutes \( k_c + k_j > 0 \) core labels, where

\[
\begin{align*}
k_c &= |\{i \in C : i \neq \tau(i) \in C\}|, \\
k_j &= |\{i \in C : i \neq \tau(i) \in J\}| \\
&= |\{i \in J : i \neq \tau(i) \in C\}|.
\end{align*}
\]

We see that \( X_P := \delta(P) - \delta(P_{id}) \) is a function of \( N_P \) independent Bernoulli random variables, where

\[
N_P = 3 \left( \frac{k_c + k_j}{2} + (n_c - k_c - k_j)(k_c + k_j) \right) + 2 \left( (k_c + k_j)n_j + (n_c - k_c - k_j)k_j \right) \frac{k_c + k_j}{2} + (n_j - k_j)k_j
\]

\[
\leq 3(k_c + 2k_j)n.
\]

Let \( S_P \) be the sum of these \( N_P \) Bernoulli random variables, and it follows that \( \text{Var}(S_P) \leq N_P/4 \).

We next apply Proposition 15 to bound the probability that \( P \) provides a better matching than \( P_{id} \). By the assumption that all edges involving junk vertices have identical probabilities, it holds that \( \mathbb{E}\delta(P_{id}) = \mathbb{E}\delta(I_n) \), so that by Proposition 6, if \( n_c \geq 4 \) then

\[
\mathbb{E}(X_P) \geq \epsilon \left( (n_c - k_c - k_j)(k_c + k_j) + \frac{k_c + k_j}{2} - T_c \right)
\]

\[
\geq \epsilon \left( (n_c - k_c - k_j - 1/2)(k_c + k_j) + \frac{k_c + k_j}{2} \right) \geq \epsilon(k_c + 2k_j)n_c/8,
\]

where \( \epsilon = \min_{i,j \in [n_c]} R_{ij} \Lambda_{i,j}(1 - \Lambda_{ij}) \). With notation as in Proposition 15, it holds that
$$\epsilon(k_c + 2k_j)n_c / 8 < \sigma^2 = 4 \text{Var}(S_P) \leq N_P \leq 3(k_c + 2k_j)n. \text{ By setting } t = \frac{\epsilon(k_c + 2k_j)n_c / 8}{\sigma}, \text{ we have}$$

$$\mathbb{P}(X_P \leq 0) \leq \mathbb{P}(|X_P - \mathbb{E}(X_P)| \geq \mathbb{E}(X_P))$$

$$\leq \mathbb{P}(|X_P - \mathbb{E}(X_P)| \geq \epsilon(k_c + 2k_j)n_c / 8)$$

$$= \mathbb{P}\left(|X_P - \mathbb{E}(X_P)| \geq \frac{\epsilon(k_c + 2k_j)n_c / 8}{\sigma}, \sigma\right)$$

$$\leq 2 \exp\left\{-C\frac{\epsilon^2(k_c + 2k_j)^2n_c^2}{\sigma^2}\right\}$$

$$\leq 2 \exp\left\{-C\frac{\epsilon^2(k_c + 2k_j)n_c^2}{n}\right\}.$$ 

with $C$ being an appropriate positive constant that may change line to line.

To use a union bound, note that the number of permutations with a given $k_c$ and $k_j$ is bounded above by $n_c^{2(k_c + k_j)} n_j^{2n_j}$. Hence, there exists a constant $c > 0$ such that if $n_c \epsilon^2 > cn_j \log(n_j)$, then

$$\mathbb{P}\left[\min_{P \in \mathcal{P}}(\delta(P) - \delta(P_{id})) \leq 0\right]$$

$$\leq \sum_{k_c=1}^{n_c} \sum_{k_j=1}^{n_j} 2^{2(k_c + k_j)} n_c^{2n_j} \exp\left\{-C\frac{\epsilon^2(k_c + 2k_j)n_c^2}{n}\right\}$$

$$\leq \sum_{k_c=1}^{n_c} \sum_{k_j=1}^{n_j} 2^{2(k_c + k_j)} \exp\left\{-C\frac{\epsilon^2(k_c + 2k_j)n_c^2}{n} + 2n_j \log(n_j)\right\}$$

$$\leq \sum_{k_c=1}^{n_c} \sum_{k_j=1}^{n_j} 2 \exp\{2(k_c + k_j) \log(n_c) - C\epsilon^2(k_c + k_j)n_c\}$$

$$= \exp\left\{-C\epsilon^2 n_c\right\},$$

as desired. \qed

### A.2 Proof of Theorem 8

**Proof.** Note that if $P \in \mathcal{M}$, then $\|\tilde{A} - \tilde{B}\|_F = \|\tilde{A} - P\tilde{B}P^T\|_F$. Let $Q \notin \mathcal{M}$ (with associated permutation $\tau$) satisfy $k_j = |\{i \leq n_c : i \neq \tau(i) > n_c\}|$ and $k_c = |\{i \leq n_c : i \neq \tau(i) \leq n_c\}|$, so that $Q$ fixes the labels of $n_c - k_c - k_j$ core vertices across $A$ and $B$. Define $X_Q := \|\tilde{A} - Q\tilde{B}Q^T\|_F - \|\tilde{A} - \tilde{B}\|_F$ and

$$\mathcal{E}_Q = \left\{\{u, v\} \in \left(\frac{[n_c]}{2}\right) : \{\tau(u), \tau(v)\} \neq \{u, v\}\right\},$$

$$25$$
and note that \( |\mathcal{E}_Q| \geq \frac{(n_c-1)(k_c+k_j)}{2} \). We then have
\[
\mathbb{E}X_Q = 2\mathbb{E}\left( \text{trace}(\bar{A}^T \bar{B}) - \text{trace}(\bar{A}^T Q \bar{B}Q^T) \right)
\]
\[=
4 \left( \sum_{\{u,v\} \in \mathcal{E}_Q} \mathbb{E}(A_{u,v}B_{u,v}) - \mathbb{E}(A_{u,v}B_{\tau(u),\tau(v)}) \right) .
\]
Now, for \( u, v \in [n_c], u \neq v, w, v \in [n], \) and \( w \neq r \), we have
\[
\mathbb{E}(\bar{A}_{u,v} \bar{B}_{w,r}) = \begin{cases} 
(1 - 2\Lambda_{u,v})^2 + 4\Lambda_{u,v}(1 - \Lambda_{u,v})R_{u,v} & \text{if } \{u, v\} = \{w, r\}, \\
(1 - 2\Lambda_{u,v})(1 - 2\Lambda_{w,r}) & \text{if } \{u, v\} \neq \{w, r\},
\end{cases}
\]
so that
\[
\mathbb{E}X_Q = 4 \left( \sum_{\{u,v\} \in \mathcal{E}_Q} 2(1 - 2\Lambda_{u,v})(\Lambda_{\tau(u),\tau(v)} - \Lambda_{u,v}) + 4\Lambda_{u,v}(1 - \Lambda_{u,v})R_{u,v} \right).
\]
In the general setting, under the assumptions of the theorem we have that
\[
2(1 - 2\Lambda_{u,v})(\Lambda_{\tau(u),\tau(v)} - \Lambda_{u,v}) + 4\Lambda_{u,v}(1 - \Lambda_{u,v})R_{u,v} \geq 4\epsilon \gamma ,
\]
so that
\[
\mathbb{E}X_Q \geq 8\epsilon (n_c - 1)(k_c + k_j)\gamma ;
\]
in the CorrER\((pJ, \varrho J)\) setting,
\[
\mathbb{E}X_Q = \sum_{\{u,v\} \in \mathcal{E}_Q} 16 p(1 - p)\varrho \geq 8 (n_c - 1)(k_c + k_j)p(1 - p)\varrho .
\]
Note that \( X_Q \) is a function of
\[
N_Q := 3 \left( (n_c - k_c - k_j)k_c + \binom{k_c}{2} \right) + 2 \left( (n_c - k_j)k_j + \binom{k_j}{2} \right) \leq 3n_c(k_c + k_j)
\]
independent Bernoulli random variables, and in the language of Proposition 15, we have that \( M = 4 \) and \( \sigma^2 \) satisfies\( \sigma^2 \leq 48n_c(k_c + k_j) \) and
\[
\sigma^2 \geq \begin{cases} 
16 \left( \binom{n_c}{2} - \binom{n_c-k_c-k_j}{2} \right) \gamma & \text{in the general setting;} \\
16 \left( \binom{n_c}{2} - \binom{n_c-k_c-k_j}{2} \right) p(1 - p) & \text{in the ER(p) setting.}
\end{cases}
\]
Setting
\[
t = \begin{cases} 
\frac{8\epsilon (n_c - 1)(k_c + k_j)\gamma}{\sigma} & \text{in the general setting;} \\
\frac{8 (n_c - 1)(k_c + k_j)p(1 - p)\varrho}{\sigma} & \text{in the ER(p) setting,}
\end{cases}
\]
in the general setting;
yields \( t < \sigma \) as required, and letting \( C \) be a positive constant that can vary line by line we have that (as in the proof of Theorem 5)
\[
\mathbb{P}(X_Q \leq 0) \leq \mathbb{P}(|X_Q - \mathbb{E}(X_Q)| \geq \mathbb{E}(X_Q)) \\
\leq \mathbb{P}\left(|X_P - \mathbb{E}(X_P)| \geq \frac{8\epsilon(n_c - 1)(k_c + k_j)\gamma}{\sigma} \cdot \sigma\right) \\
\leq 2 \exp\left\{-C\epsilon^2 n_c(k_c + k_j)\gamma^2\right\}; \quad (11)
\]
in the CorrER(\( pJ, qJ \)) setting, we similarly have
\[
\mathbb{P}(X_Q \leq 0) \leq 2 \exp\left\{-C\epsilon^2 n_c(k_c + k_j)p^2(1-p)^2\right\}. \quad (12)
\]
Define the equivalence relation “\( \sim \)” on \( \Pi_{n_c+n_j} \) via \( P \sim Q \) iff \( P,v = Q,v \) for all core vertices \( v \). Note that \( P \sim Q \) implies that To prove that there is no \( P \in \Pi_{n_c+n_j} \setminus \mathcal{M} \) satisfying \( X_P \leq 0 \), it suffices to consider a single \( P \) from each equivalence class under “\( \sim \)”. Summing Eqs. (11) and (12) over the (at most) \( n_c^2(k_j+k_c)n_j^{2k_j} \) equivalence classes for each \( k_c, k_j \) and then over \( k_c \) and \( k_j \), yields the desired result. \( \square \)

A.3 Proof of Theorem 9

Proof. Let \( Q \in \Pi(n_c+n_j) \) (with associated permutation \( \tau \)) satisfy \( k_j = |\{i \leq n_c: i \neq \tau(i) > n_c\}| = n_c \), so that all core vertices are mapped to junk vertices by \( Q \). Note that \( \mathcal{E}_Q = \binom{[n_c]}{2} \).

We will construct an appropriate \( \Lambda \) such that \( Q \) will satisfy Eq. (4).

Note that for \( u, v \in [n_c], u \neq v, w, v \in [n], \) and \( w \neq r \), we have
\[
\mathbb{E}(\tilde{A}_{u,v} \tilde{B}_{w,r}) = \begin{cases} \\
\Lambda_{u,v}(\Lambda_{u,v} + (1 - \Lambda_{u,v})R_{u,v}) & \text{if } \{u, v\} = \{w, r\}, \\
\Lambda_{u,v} \Lambda_{w,r} & \text{if } \{u, v\} \neq \{w, r\},
\end{cases}
\]

So that
\[
\mathbb{E}X_Q \neq 4 \left( \sum_{\{u,v\} \in \mathcal{E}_Q} \Lambda_{u,v}(\Lambda_{u,v} + (1 - \Lambda_{u,v})R_{u,v}) - \Lambda_{u,v} \Lambda_{\tau(u),\tau(v)} \right) \\
\leq 4 \left( \sum_{\{u,v\} \in \mathcal{E}_Q} \Lambda_{u,v}(\Lambda_{u,v} + (1 - \Lambda_{u,v})q) - \Lambda_{u,v} \Lambda_{\tau(u),\tau(v)} \right)
\]
Letting each \( \Lambda_{\tau(u),\tau(v)} > \beta + (1 - \beta)q + \epsilon \) for \( \epsilon > 0 \) chosen to keep \( \Lambda_{\tau(u),\tau(v)} \in (0, 1) \) (which is possible by the assumption that \( \beta \) and \( q \) are both strictly less than 1), we have
\[
-\mathbb{E}X_Q \geq 4 \left( \sum_{\{u,v\} \in \mathcal{E}_Q} \Lambda_{u,v}(\beta + (1 - \beta)q + \epsilon) - \Lambda_{u,v}(\Lambda_{u,v} + (1 - \Lambda_{u,v})q) \right) = \omega(\epsilon n_c \log n_c).
\]
Applying Proposition 15 with $M = 2$, $\sigma^2 = \Theta(n_c^2)$, we have (where $C$ is a positive constant that can vary line–to–line)

$$\mathbb{P}(X_Q \geq 0) = \mathbb{P}(X_Q - \mathbb{E}X_Q \geq -\mathbb{E}X_Q)$$

$$\leq \mathbb{P}(|X_Q - \mathbb{E}X_Q| \geq C\epsilon n_c \log n_c)$$

$$\leq \exp\left\{-C\epsilon^2 (\log n_c)^2\right\} = o(1)$$

as desired.

\[\square\]

**References**

E. Abbe and C. Sandon. Community detection in general stochastic block models: Fundamental limits and efficient algorithms for recovery. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pages 670–688, October 2015.

D. Asta and C. R. Shalizi. Geometric network comparison. *arXiv preprint arXiv:1411.1350*, 2014.

L. Babai. Graph isomorphism in quasipolynomial time. *arXiv preprint arXiv:1512.03547*, 2015.

E. Cela. *The quadratic assignment problem: theory and algorithms*, volume 1. Springer Science & Business Media, 2013.

L. Chen, J. T. Vogelstein, V. Lyzinski, and C. E. Priebe. A joint graph inference case study: the c. elegans chemical and electrical connectomes. In *Worm*, volume 5. Taylor & Francis, 2016.

A. Clauset, M.E.J. Newman, and C. Moore. Finding community structure in very large networks. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 70(6 Pt 2): 066111, December 2004.

D. Conte, P. Foggia, C. Sansone, and M. Vento. Thirty years of graph matching in pattern recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 18(03):265–298, 2004.

F. Emmert-Streib, M. Dehmer, and Y. Shi. Fifty years of graph matching, network alignment and network comparison. *Information sciences*, 346–347:180–197, 2016.
F. Escolano, E. R. Hancock, and M. Lozano. Graph matching through entropic manifold alignment. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 2417–2424. IEEE, 2011.

V. Feldman, E. Grigorescu, L. Reyzin, S. Vempala, and Y. Xiao. Statistical algorithms and a lower bound for detecting planted cliques. In *Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing*, STOC ’13, pages 655–664, New York, NY, USA, 2013. ACM.

D. E. Fishkind, V. Lyzinski, H. Pao, L. Chen, and C. E. Priebe. Vertex nomination schemes for membership prediction. *Ann. Appl. Stat.*, 9(3):1510–1532, 09 2015. doi: 10.1214/15-AOAS834. URL http://dx.doi.org/10.1214/15-AOAS834.

D.E. Fishkind, S. Adali, H. G. Patsolic, L. Meng, V. Lyzinski, and C.E. Priebe. Seeded graph matching. *arXiv preprint arXiv:1209.0367*, 2017.

P. Foggia, G. Percannella, and M. Vento. Graph matching and learning in pattern recognition in the last 10 years. *International Journal of Pattern Recognition and Artificial Intelligence*, 28(01):1450001, 2014.

E. Kazemi, S. H. Hassani, and M. Grossglauser. Growing a graph matching from a handful of seeds. *Proceedings of the VLDB Endowment*, 8(10):1010–1021, 2015.

J. H. Kim, B. Sudakov, and V. H. Vu. On the asymmetry of random regular graphs and random graphs. *Random Structures and Algorithms*, 21:216–224, 2002.

J. Lee, M. Cho, and K. M. Lee. A graph matching algorithm using data-driven markov chain monte carlo sampling. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 2816–2819. IEEE, 2010.

L. Li and W. M. Campbell. Matching community structure across online social networks. *NIPS Workshop on Networks in the Social and Information Sciences*, 2015.

L. Lin, X. Liu, and S.-C. Zhu. Layered graph matching with composite cluster sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8):1426–1442, 2010.

E. M. Loiola, N. M. M. de Abreu, P. O. Boaventura-Netto, P. Hahn, and T. Querido. A survey for the quadratic assignment problem. *European journal of operational research*, 176(2):657–690, 2007.
V. Lyzinski. Information recovery in shuffled graphs via graph matching. *arXiv preprint arXiv:1605.02315*, 2016.

V. Lyzinski, D. E. Fishkind, and C. E. Priebe. Seeded graph matching for correlated Erdos-Renyi graphs. *Journal of Machine Learning Research*, 15:3513–3540, 2014. URL http://jmlr.org/papers/v15/lyzinski14a.html.

V. Lyzinski, D. L. Sussman, D. E. Fishkind, H. Pao, L. Chen, J. T. Vogelstein, Y. Park, and C. E. Priebe. Spectral clustering for divide-and-conquer graph matching. *Parallel Computing*, 47:70–87, 2015.

V. Lyzinski, D. E. Fishkind, M. Fiori, J. T. Vogelstein, C. E. Priebe, and G. Sapiro. Graph matching: Relax at your own risk. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(1):60–73, 2016a.

V. Lyzinski, K. Levin, D. E. Fishkind, and C. E. Priebe. On the consistency of the likelihood maximization vertex nomination scheme: Bridging the gap between maximum likelihood estimation and graph matching. *Journal of Machine Learning Research*, 17(179):1–34, 2016b.

E. Mossel, J. Neeman, and A. Sly. Belief propagation, robust reconstruction and optimal recovery of block models. In *Conference on Learning Theory*, pages 356–370. jmlr.org, 29 May 2014.

C. L. M. Nickel. *Random dot product graphs: A model for social networks*. PhD thesis, Johns Hopkins University, 2006.

E. Onaran, S. Garg, and E. Erkip. Optimal de-anonymization in random graphs with community structure. *arXiv preprint arXiv:1602.01409*, 2016.

P. Pedarsani and M. Grossglauser. On the privacy of anonymized networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1235–1243. ACM, 2011.

A. Robles-Kelly and E. R. Hancock. A riemannian approach to graph embedding. *Pattern Recognition*, 40(3):1042–1056, 2007.

J. Sang and C. Xu. Robust face-name graph matching for movie character identification. *IEEE Transactions on Multimedia*, 14(3):586–596, 2012.
M. Tang, A. Athreya, D. L. Sussman, V. Lyzinski, Y. Park, and C. E. Priebe. A semiparametric two-sample hypothesis testing problem for random dot product graphs. *Journal of Computational and Graphical Statistics*, Accepted for publication, 2016.

L. Yartseva and M. Grossglauser. On the performance of percolation graph matching. In *Proceedings of the first ACM conference on Online social networks*, pages 119–130. ACM, 2013.

S. Young and E. Scheinerman. Random dot product graph models for social networks. In *Proceedings of the 5th international conference on algorithms and models for the web-graph*, pages 138–149, 2007.

M. Zaslavskiy, F. Bach, and J.P. Vert. A path following algorithm for the graph matching problem. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(12):2227–2242, 2009.

F. Zhou and F. De la Torre. Factorized graph matching. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 127–134. IEEE, 2012.