Appendix

A Proofs from the Paper

A.1 Proof of Proposition 1

Proof. We start by writing potential outcomes for an arbitrary unit i from (2) as $Y_i(t, \mathbf{t}_{-i}) = f_i(t, \mathbf{t}_{-i}) + \epsilon_i$, where ϵ_i is some pre-treatment value of the potential outcome, and $f_i(t, \mathbf{t}_{-i})$ is the treatment response function, dependent both on unit i's treatment and on everyone else's. Using A0-A3 we can write (2) as:

$$\begin{aligned} Y_i(t, \mathbf{t}_{-i}) &= t\tau_i + f_i(\mathbf{t}_{-i}) + \epsilon_i \qquad \text{(By A1)} \\ &= t\tau_i + f_i(\mathbf{t}_{\mathcal{N}_i}) + \epsilon_i \qquad \text{(By A2)} \\ &= t\tau_i + f(G^{\mathbf{t}}_{\mathcal{N}_i}) + \epsilon_i \qquad \text{(By A3).} \end{aligned} \tag{7}$$

This proves Equation (3). The first line agrees with the additivity assumption, which permits a contribution $f_i(\mathbf{t}_{-i})$ to $Y_i(t, \mathbf{t}_{-i})$ arising from anywhere within the rest of the graph. The second line states that this contribution is limited to i's neighborhood. The third line states that the contribution depends only the neighborhood structure and not anything else about the neighbors.

To prove identification for the ADE, we must show that the individual treatment effect is identified for a treated unit i. We need to show that: $\mathbb{E}[Y_i(1, \mathbf{0}) - Y_i(0, \mathbf{0})] = \mathbb{E}[Y_i | T_i = 1, G^{\mathbf{T}}_{\mathcal{N}_i} = g^{\mathbf{t}}_i] - \mathbb{E}[Y_i | T_i = 0, G^{\mathbf{T}}_{\mathcal{N}_i} = g^{\mathbf{t}}_i]$. We have first:

$$\begin{aligned} &\mathbb{E}[Y_i(1, \mathbf{0}) - Y_i(0, \mathbf{0})] \\ &= \mathbb{E}[t\tau_i + f(G^{\mathbf{0}}_{\mathcal{N}_i}) + \epsilon_i - f(G^{\mathbf{0}}_{\mathcal{N}_i}) - \epsilon_i] \\ &= \mathbb{E}[\epsilon_i + \tau_i - \epsilon_i] = \tau_i. \end{aligned} \tag{8}$$

Second, we have:

$$\begin{aligned} &\mathbb{E}[Y_i | G^{\mathbf{T}}_{\mathcal{N}_i} \simeq g^{\mathbf{t}}_i, T_i = 1] - \mathbb{E}[Y_i | G^{\mathbf{T}}_{\mathcal{N}_i} \simeq g^{\mathbf{t}}_i, T_i = 0] \\ =&\mathbb{E}[Y_i(1, \mathbf{T}_{-i})T_i + \\ &\quad + Y_i(0, \mathbf{T}_{-i})(1 - T_i) | G^{\mathbf{T}}_{\mathcal{N}_i} \simeq g^{\mathbf{t}}_i, T_i = 1] \\ -&\mathbb{E}[Y_i(1, \mathbf{T}_{-i})T_i \\ &\quad + Y_i(0, \mathbf{T}_{-i})(1 - T_i) | G^{\mathbf{T}}_{\mathcal{N}_i} \simeq g^{\mathbf{t}}_i, T_i = 0] \\ =&\mathbb{E}[\tau_i + f(g^{\mathbf{t}}_i) + \epsilon_i | G^{\mathbf{T}}_{\mathcal{N}_i} \simeq g^{\mathbf{t}}_i, T_i = 1] \\ &\quad - \mathbb{E}[f(g^{\mathbf{t}}_i) + \epsilon_i | G^{\mathbf{T}}_{\mathcal{N}_j} = g^{\mathbf{t}}_i, T_i = 0] \\ =&\alpha + f(g^{\mathbf{t}}_i) + \tau_i - \alpha - f(g^{\mathbf{t}}_i) \\ =&\tau_i. \end{aligned} \tag{9}$$

The first equality follows from the definition of Y_i, the second equality from the result in (3) and A3, and the third equality follows from independence of T and Y given in A0: $\mathbb{E}[\epsilon_i | T_i] = \mathbb{E}[\epsilon_i] = \alpha$ for all i. Finally, we can use both of the results above to obtain the ADE:

$$\begin{aligned} &\frac{1}{n^{(1)}} \sum_{i=1}^{n} \mathbb{E}[T_i \times (\mathbb{E}[Y_i | G^{\mathbf{T}}_{\mathcal{N}_i} \simeq g^{\mathbf{t}}_i, T_i = 1] \\ &\qquad - \mathbb{E}[Y_i | G^{\mathbf{T}}_{\mathcal{N}_i} \simeq g^{\mathbf{t}}_i, T_i = 0])] \\ &= \frac{1}{n^{(1)}} \sum_{i=1}^{n} \mathbb{E}[T_i \tau_i] \qquad \text{(By (9))} \\ &= \frac{1}{n^{(1)}} \sum_{i=1}^{n} \mathbb{E}[T_i \times (\mathbb{E}[Y_i(1, \mathbf{0}) - Y_i(0, \mathbf{0})])] \qquad \text{(By (8))} \\ &= \frac{1}{n^{(1)}} \sum_{i=1}^{n} \Pr(T_i = 1)(\mathbb{E}[Y_i(1, \mathbf{0}) - Y_i(0, \mathbf{0})]) \\ &= \frac{1}{n^{(1)}} \sum_{i=1}^{n} \frac{n^{(1)}}{n} (\mathbb{E}[Y_i(1, \mathbf{0}) - Y_i(0, \mathbf{0})]) \\ &= \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}[Y_i(1, \mathbf{0}) - Y_i(0, \mathbf{0})], \end{aligned}$$

where $\Pr(T_i = 1) = \frac{n^{(1)}}{n}$ by assumption of complete randomization. □

B Additional Theoretical Results

We study the expected error for one AME match on subgraphs under two assumptions: that the true weights for the AME objective (the weighted Hamming distance) are known, and that the candidate units for matching all have independently generated neighborhoods, and none of the units in these neighborhoods are being matched. Additional information on this setting is available in the proof.

Proposition 2. *(Oracle AME Error With Independent Neighborhoods) Suppose that there are N independently generated graphs, each with n vertices, and all i.i.d. from the same distribution over graphs: $\Pr(G_1 = g_1, \ldots, G_N = g_N) = \prod_{i=1}^{N} p(g_i)$. Assume matches are only allowed between units in different graphs. Suppose additionally that $n^{(1)}$ randomly chosen units within each graph are assigned treatment, so that $\Pr(\mathbf{T}_i = \mathbf{t}_i) = \binom{n}{n^{(1)}}^{-1}$. Assume further that interference functions obey the following: $|f(g) - f(h)| \leq K \mathbf{w}^T \mathbb{I}[S(g) \neq S(h)]$, where $\mathbf{w}$ is a vector of positive, real-valued, importance weights for the subgraphs counts, such that $\|\mathbf{w}\|_1 = M_n$ for some constant $0 < M_n < \infty$, and such that the condition above is satisfied for $\mathbf{w}$, and $\mathbb{I}[S(g) \neq S(h)]$ is the Hamming distance: a vector of the same length as $\mathbf{w}$ that is 0 at position k if graphs g and h have the same count of subgraph k, and 1 otherwise. Assume that baseline responses have variance $Var(\epsilon_i) = \sigma^2 \ \forall i$. Then, for a treatment unit i, if j solves the AME problem, i.e.,*

$j \in \arg\min_{\substack{k=1,\dots,n \\ T_k=0}} \mathbf{w}^T \mathbb{I}[S(h^{\mathbf{t}}_{\mathcal{N}_i}) \neq S(G^{\mathbf{T}}_{\mathcal{N}_k})]$, *under A0-A3:*

$$\mathbb{E}\left[|Y_i - Y_j - \tau_i| \big| T_i = 1, G^{\mathbf{t}}_{\mathcal{N}_i} = h^{\mathbf{t}}_{\mathcal{N}_i}\right] \leq \sqrt{2}\sigma + K\binom{n-1}{n^{(1)}}^{-1} \times \sum_{g \in \mathcal{G}_n} \sum_{\substack{\mathbf{t} \in \mathcal{T} \\ t_j = 0}} \mathbf{w}^T \mathbb{I}[S(h^{\mathbf{t}}_{\mathcal{N}_i}) \neq S(g^{\mathbf{t}}_{\mathcal{N}_j})] p(g) \times \left(\frac{n^{(1)}}{n} + \frac{n - n^{(1)}}{n} C(g^{\mathbf{t}}_{\mathcal{N}_j})\right)^{N-2},$$

where $\mathcal{G}_n$ is the set of all graphs with n units, and $C(h^{\mathbf{t}}_{\mathcal{N}_j}) \leq 1$ for all g and $\mathbf{t}$.

A proof is available in the following section. The first element in the right hand side of the inequality is the standard deviation of the baseline responses. One summation is over all possible graphs with n units, and the other summation is over possible treatment assignments. The expression inside the summation is the product of three terms. First, the weighted Hamming distance between a graph and the target graph we are trying to match. Second, the probability of observing that graph. Third, an upper bound on the probability that unit j is among the minimizers of the weighted Hamming distance. Note that $\mathbf{w}^T \mathbb{I}[S(g^{\mathbf{t}}_{\mathcal{N}_j}) \neq S(g^{\mathbf{t}}_{\mathcal{N}_j})] p(g)$ is bounded for fixed n for all g and $\mathbf{t}$. This implies that the bound converges to $2\sqrt{\sigma}$ as $N \to \infty$, as long as the size of neighborhood graphs is held fixed, because perfect matching is possible with large amounts of data in this regime.

B.1 Proof of Proposition 2

We briefly review some notation and assumptions to be used in this proof. For the purposes of theory, we study a simplified setting, in which we have to AME-match a unit i to one unit in a set of candidate units of size N such that: a) all the candidate units belong to disconnected graphs, which we refer to as candidate graphs. b) within each candidate graph there is only one pre-determined candidate unit c) candidate units have neighborhood graphs denoted by $G_{\mathcal{N}_j}$. d) all the candidate graphs are drawn independently from the same distribution over graphs: $\Pr(G_{\mathcal{N}_1} = g_1, \dots, G_{\mathcal{N}_N} = g_N) = \prod_{i=1}^N p(g_i)$. The support of p will be $\mathcal{G}_n$: the set of all graphs with exactly n units. We use $g_{\mathcal{N}_i}$ to denote the subgraph induced over g by the units in the set of neighbors of unit i, $\mathcal{N}_i \subseteq V(g)$, i.e., $g_{\mathcal{N}_i}$ is the graph consisting only of the vertices that share an edge with i, in g, and of the edges in g that are between these vertices. The ego i is not included in $g_{\mathcal{N}_i}$.

Assigned treatments are denoted by $\mathbf{T}$, where $\mathbf{T} \in \{0,1\}^n$, but in this setting treatment assignment is assumed to be independent within the N candidate graphs. Formally, the assumption we make is that $\Pr(\mathbf{T}_1 = \mathbf{t}_1, \dots, \mathbf{T}_N = \mathbf{t}_N) = \prod_{i=1}^N \binom{n}{n^{(1)}}^{-1}$, i.e., $n^{(1)}$ units are always treated uniformly at random within each of the N candidate graphs.

The direct treatment effect for any unit i is given by τ_i. We use $\mathbb{I}[S(g) \neq S(h)]$ to indicate the Hamming distance between subgraph counts of graphs g and h. This means that $\mathbb{I}[S(g) \neq S(h)]$ is a vector of size $|\mathcal{G}_n|$ that will be 1 in the ℓ^{th} entry if g and h have the same amount of occurrences of graph g_ℓ among their subgraphs. Note that this distance is coloring sensitive: two subgraphs that are isomorphic in shape but not labels will belong to different entries in this distance. The matched group of a *treated* unit i, denoted $\texttt{MG}_i$ is the set of all *control* units that match with i. In our setting $j \in \texttt{MG}_i$ if it solves the AME problem, that is $j \in \arg\min_{\substack{k=1\dots,n \\ T_k=0}} \mathbf{w}^T \mathbb{I}[S(G^{\mathbf{T}}_{\mathcal{N}_k}) \neq S(g^{\mathbf{t}}_{\mathcal{N}_i})]$. Finally, we assume that both the graph for the unit we want to match and the treatment assignment for that unit's graph are fixed: $\mathbf{t}_i$ is the treatment assignment in the graph of i, and $h^{\mathbf{t}_i}_{\mathcal{N}_i}$ is the neighborhood graph of i, where h denotes unit i's graph. All other notation is as in the main paper.

Proof. We start by upper-bounding our quantity of interest as follows:

$$\begin{aligned}
&\mathbb{E}[|Y_i - Y_j - \tau_i| \big| j \in \texttt{MG}_i] \\
&= \mathbb{E}[|Y_i(1, \mathbf{t}_{i-i}) - Y_j(0, \mathbf{T}_{j-j}) - \tau_i| \big| j \in \texttt{MG}_i] \\
&= \mathbb{E}[|\tau_i + f(h^{\mathbf{t}_i}_{\mathcal{N}_i}) + \epsilon_i - f(G^{\mathbf{T}_j}_{\mathcal{N}_j}) - \epsilon_j - \tau_i| \big| j \in \texttt{MG}_i] \\
&\leq \mathbb{E}[|f(h^{\mathbf{t}_i}_{\mathcal{N}_i}) - f(G^{\mathbf{T}_j}_{\mathcal{N}_j})| \big| j \in \texttt{MG}_i] \\
&\quad + \mathbb{E}[|\epsilon_i - \epsilon_j| \big| j \in \texttt{MG}_i] \\
&\leq K\mathbb{E}[\mathbf{w}^T \mathbb{I}[S(h^{\mathbf{t}_i}_{\mathcal{N}_i}) \neq S(G^{\mathbf{T}_j}_{\mathcal{N}_j})] \big| j \in \texttt{MG}_i] \\
&\quad + \mathbb{E}[|\epsilon_i - \epsilon_j| \big| j \in \texttt{MG}_i], \qquad (10)
\end{aligned}$$

where the notation $\mathbf{T}_{j-j}$ denotes the treatment indicator for candidate graph j for all units except j. The first equality follows from A0 since the event $j \in \texttt{MG}_i$ implies that $T_j = 0$, as only control units are allowed in the matched groups. The second equality follows from Proposition 1. The first inequality is an application of the triangle inequality. The last line follows from the condition on the interference functions. Consider the second term. We can use the Cauchy-Schwarz inequality to construct a simple upper bound on it:

$$\begin{aligned}
&\mathbb{E}[|\epsilon_i - \epsilon_j| \big| j \in \texttt{MG}_i] = \mathbb{E}[|\epsilon_i - \epsilon_j|] \\
&\leq \sqrt{\mathbb{E}[(\epsilon_i - \epsilon_j)^2]} \\
&= \sqrt{\mathbb{E}[\epsilon_i^2] + \mathbb{E}[\epsilon_j^2] - 2\mathbb{E}[\epsilon_i]\mathbb{E}[\epsilon_j]} = \sqrt{2}\sigma
\end{aligned}$$

where the last equality follows for the fact that the ϵ_i have mean 0 and are independent, with $Var(\epsilon_i) = \sigma^2$ for all i.

Consider now the term $\mathbb{E}[\mathbf{w}^T\mathbb{I}[S(h^{\mathbf{t}_i}_{\mathcal{N}_i}) \neq S(G^{\mathbf{T}_j}_{\mathcal{N}_j})]|j \in \texttt{MG}_i]$. To upper-bound this, we write it out as follows using the definition of expectation:

$$\begin{aligned}
&\mathbb{E}[\mathbf{w}^T\mathbb{I}[S(h^{\mathbf{t}_i}_{\mathcal{N}_i}) \neq S(G^{\mathbf{T}_j}_{\mathcal{N}_j})]|j \in \texttt{MG}_i] \\
&= \sum_{g \in \mathcal{G}_n} \sum_{\mathbf{t} \in \mathcal{T}, t_j = 0} \mathbf{w}^T\mathbb{I}[S(h^{\mathbf{t}_i}_{\mathcal{N}_i}) \neq S(g^{\mathbf{t}_j}_{\mathcal{N}_j})] \\
&\qquad \times \Pr(G^{\mathbf{T}_j}_{\mathcal{N}_j} = g^{\mathbf{t}}_{\mathcal{N}_j}, \mathbf{T}_j = \mathbf{t}|j \in \texttt{MG}_i).
\end{aligned}$$

We want to find an upper bound on $\Pr(G^{\mathbf{T}_j}_{\mathcal{N}_j} = g^{\mathbf{t}}_{\mathcal{N}_j}, \mathbf{T}_j = \mathbf{t}|j \in \texttt{MG}_i)$. We start by writing this quantity out as a product of two probabilities:

$$\begin{aligned}
&\Pr(G^{\mathbf{T}_j}_{\mathcal{N}_j} = g^{\mathbf{t}}_{\mathcal{N}_j}, \mathbf{T}_j = \mathbf{t}|j \in \texttt{MG}_i) \\
&= \Pr(G^{\mathbf{T}_j}_{\mathcal{N}_j} = g^{\mathbf{t}}_{\mathcal{N}_j}|j \in \texttt{MG}_i, \mathbf{T}_j = \mathbf{t}) \\
&\times \Pr(\mathbf{T}_j = \mathbf{t}|j \in \texttt{MG}_i) \\
&= \Pr(G^{\mathbf{T}_j}_{\mathcal{N}_j} = g^{\mathbf{t}}_{\mathcal{N}_j}|j \in \texttt{MG}_i, \mathbf{T}_j = \mathbf{t}) \binom{n-1}{n^{(1)}}^{-1}.
\end{aligned}$$

Note that $\Pr(\mathbf{T}_j = \mathbf{t}|j \in \texttt{MG}_i) = \binom{n-1}{n^{(1)}}^{-1}$ because treatment is uniformly randomized with $n^{(1)}$ units always treated in each candidate graph, but $T_j = 0$ conditionally on $j \in \texttt{MG}_i$.

We use Bayes' rule to write out the first term in the final product as $\Pr(G^{\mathbf{T}}_{\mathcal{N}_j} = g^{\mathbf{t}}_{\mathcal{N}_j}|j \in \texttt{MG}_i, \mathbf{T}_j = \mathbf{t}) = \frac{\Pr(j \in \texttt{MG}_i|\mathbf{T}_j = \mathbf{t}, G^{\mathbf{T}_j}_{\mathcal{N}_j} = g^{\mathbf{t}}_{\mathcal{N}_j}) \Pr(G^{\mathbf{T}_j}_{\mathcal{N}_j} = g^{\mathbf{t}}_{\mathcal{N}_j}|\mathbf{T}_j = \mathbf{t})}{\Pr(j \in \texttt{MG}_i|\mathbf{T}_j = \mathbf{t})}$.

By assumption, if all neighborhood graphs are empty, all units are used for all matched groups, and we are restricting ourselves to assignments in which $T_j = 0$, therefore, $\Pr(j \in \texttt{MG}_i|\mathbf{T}_j = \mathbf{t}) = 1$. Second, by assumption $\Pr(G^{\mathbf{T}_j}_{\mathcal{N}_j} = g^{\mathbf{t}}_{\mathcal{N}_j}|\mathbf{T}_j = \mathbf{t}) = p(g)$. This is because treatment assignment is independent of the graph. We are left with having to find an expression for the likelihood, this can be written as:

$$\begin{aligned}
&\Pr(j \in \texttt{MG}_i|\mathbf{T}_j = \mathbf{t}, G^{\mathbf{T}_j}_{\mathcal{N}_j} = g^{\mathbf{t}}_{\mathcal{N}_j}) \\
&= \Pr(j \in \underset{\substack{k=1,\ldots,N,\\ k\neq i}}{\arg\min}\, \mathbf{w}^T\mathbb{I}[S(h^{\mathbf{t}_i}_{\mathcal{N}_i}) \neq S(G^{\mathbf{T}_k}_{\mathcal{N}_k})] \\
&\qquad |\mathbf{T}_j = \mathbf{t}, G^{\mathbf{T}}_{\mathcal{N}_j} = g^{\mathbf{t}}_{\mathcal{N}_j}) \\
&= \prod_{\substack{k=1\\ k\neq i,j}}^{N} \Bigg[\Pr(\mathbf{w}^T\mathbb{I}[S(h^{\mathbf{t}_i}_{\mathcal{N}_i}) \neq S(G^{\mathbf{T}_k}_{\mathcal{N}_k})] \geq \\
&\qquad \mathbf{w}^T\mathbb{I}[S(h^{\mathbf{t}_i}_{\mathcal{N}_i}) \neq S(g^{\mathbf{t}}_{\mathcal{N}_j})]|T_k = 0) \Pr(T_k = 0) \\
&\qquad + \Pr(T_k = 1) \Bigg] \\
&= \prod_{\substack{k=1\\ k\neq i,j}}^{N} \Bigg[\Pr\left(\mathbf{w}^T\mathbb{I}[S(h^{\mathbf{t}_i}_{\mathcal{N}_i}) \neq S(G^{\mathbf{T}_k}_{\mathcal{N}_k})]\right. \\
&\qquad \geq \mathbf{w}^T\mathbb{I}[S(h^{\mathbf{t}_i}_{\mathcal{N}_i}) \neq S(g^{\mathbf{t}}_{\mathcal{N}_j})]|T_k = 0) \frac{n - n^{(1)}}{n} \\
&\qquad \left. + \frac{n^{(1)}}{n} \right] \Bigg].
\end{aligned}$$

The second equality follows because k can never be in the matched group of unit i if $T_k = 1$, and, if $T_k = 0$, then k must be one of the minimizers of the weighted Hamming distance between neighborhood subgraph counts. The probability is a product of densities because of independence of candidate subgraphs. For an arbitrary unit, k, we define the following compact notation for the probability that k's weighted Hamming distance from i is larger than the weighted Hamming distance from j to i:

$$\begin{aligned}
&\Pr(\mathbf{w}^T\mathbb{I}[S(h^{\mathbf{t}_i}_{\mathcal{N}_i}) \neq S(G^{\mathbf{T}_k}_{\mathcal{N}_k})] \\
&\qquad \geq \mathbf{w}^T\mathbb{I}[S(h^{\mathbf{t}_i}_{\mathcal{N}_i}) \neq S(g^{\mathbf{t}}_{\mathcal{N}_j})]|T_k = 0) \\
&=: C_k(g^{\mathbf{t}}_{\mathcal{N}_j}) \leq 1.
\end{aligned}$$

Note that the last inequality follows from the fact that the expression above is a probability. Since graphs and treatment assignments, G_k and $\mathbf{T}_k$ are the only random variables in the probability denoted by $C_k(g^{\mathbf{t}}_{\mathcal{N}_j})$, and since they are all independent, and identically distributed, we can say that $C_1(g^{\mathbf{t}}_{\mathcal{N}_j}) = C_2(g^{\mathbf{t}}_{\mathcal{N}_j}) = \cdots = C_N(g^{\mathbf{t}}_{\mathcal{N}_j}) = C(g^{\mathbf{t}}_{\mathcal{N}_j})$. Because of this we have:

$$\begin{aligned}
&\Pr(j \in \texttt{MG}_i|G^{\mathbf{T}_j}_{\mathcal{N}_j} = g^{\mathbf{t}}_{\mathcal{N}_j}, \mathbf{T}_j = \mathbf{t}) \\
&= \left(\frac{n^{(1)}}{n} + \frac{n - n^{(1)}}{n} C(g^{\mathbf{t}}_{\mathcal{N}_j})\right)^{N-2}.
\end{aligned}$$

Putting all the elements we have together we get the

expression for the first term in the bound:

$$\begin{aligned}
&\mathbb{E}[\mathbf{w}^T\mathbb{I}[S(h^{\mathbf{t}_i}_{\mathcal{N}_i}) \neq S(G^{\mathbf{T}_j}_{\mathcal{N}_j})]|j \in \texttt{MG}_i] \\
&= \sum_{g\in\mathcal{G}_n}\sum_{\mathbf{t}\in\mathcal{T}:t_j=0} \mathbf{w}^T\mathbb{I}[S(g^{\mathbf{t}}_{\mathcal{N}_j}) \neq S(h^{\mathbf{t}_i}_{\mathcal{N}_i})] \\
&\qquad \times \Pr(G^{\mathbf{T}}_{\mathcal{N}_j} = g^{\mathbf{t}}_{\mathcal{N}_j}, \mathbf{T}_{\mathcal{N}_j} = \mathbf{t}_{\mathcal{N}_j}|j \in \texttt{MG}_i) \\
&= \sum_{g\in\mathcal{G}_N}\sum_{\mathbf{t}\in\mathcal{T}:t_j=0} \mathbf{w}^T\mathbb{I}[S(g^{\mathbf{t}}_{\mathcal{N}_j}) \neq S(h^{\mathbf{t}_i}_{\mathcal{N}_i})] \\
&\times \binom{n-1}{n^{(1)}}^{-1} p(g)\left(\frac{n^{(1)}}{n} + \frac{n-n^{(1)}}{n}C(g^{\mathbf{t}}_{\mathcal{N}_j})\right)^{N-2}.
\end{aligned}$$

□

B.2 Asymptotic Behavior

Here we expand on the asymptotic consequences of Proposition 2: note first, that, by assumption $\|\mathbf{w}\|_1 = M_n$, and that, therefore $\mathbf{w}^T\mathbb{I}[S(g) \neq S(h)] \leq M_n$ for any graphs $g, h \in \mathcal{G}_n$. That is to say, the weighted Hamming distance between any two graphs with n units will be upper-bounded by the sum of the weights. Recall also that $C(g^{\mathbf{t}}_{\mathcal{N}_j}) \leq 1$ for all g and $\mathbf{t}$ as this quantity is a probability, and let $C_{max} = \max_{\substack{g\in\mathcal{G}_n \\ \mathbf{t}\in\mathcal{T}}} C(g^{\mathbf{t}}_{\mathcal{N}_j})$. We can combine all these bounds with the upper bound in Proposition 2 to write:

$$\begin{aligned}
&\mathbb{E}[\mathbf{w}^T\mathbb{I}[S(h^{\mathbf{t}_i}_{\mathcal{N}_i}) \neq S(G^{\mathbf{T}_j}_{\mathcal{N}_j})]|j \in \texttt{MG}_i] \\
&= \sum_{g\in\mathcal{G}_\mathcal{N}}\sum_{\mathbf{t}\in\mathcal{T},t_j=0} \mathbf{w}^T\mathbb{I}[S(g^{\mathbf{t}}_{\mathcal{N}_j}) \neq S(h^{\mathbf{t}_i}_{\mathcal{N}_i})] \\
&\times \binom{n-1}{n^{(1)}}^{-1} p(g)\left(\frac{n^{(1)}}{n} + \frac{n-n^{(1)}}{n}C(g^{\mathbf{t}}_{\mathcal{N}_j})\right)^{N-2} \\
&\leq M_n\left(\frac{n^{(1)}}{n} + \frac{n-n^{(1)}}{n}C_{max}\right)^{N-2} \\
&\times \sum_{g\in\mathcal{G}_n}\sum_{\mathbf{t}\in\mathcal{T},t_j=0} \binom{n-1}{n^{(1)}}^{-1} p(g) \\
&= M_n\left(\frac{n^{(1)}}{n} + \frac{n-n^{(1)}}{n}C_{max}\right)^{N-2}.
\end{aligned}$$

The first equality follows from Proposition 2, the first inequality from the bounds previously discussed, and the second equality follows from the fact that the sum in the second to last line is a sum of probability distributions over their entire domain, and therefore is equal to 1. Under the condition that n, the number of units in each unit's candidate graph, stays fixed, and that $C_{max} < 1$, then, as $N \to \infty$, we have $M_n\left(\frac{n^{(1)}}{n} + \frac{n-n^{(1)}}{n}C_{max}\right)^{N-2} \to 0$, because the quantity inside the parentheses is always less than 1. This makes sense, because asymptotically, matches can be made exactly; i.e., units matched in the way described in our theoretical setting have isomorphic neighborhood subgraphs asymptotically. This also has a consequence that the bound in Proposition 2 converges to $\sqrt{2}\sigma$ asymptotically in N. This is the variance of the baseline errors and can be lowered by matching the same unit with multiple others. As noted before, for this argument to apply, candidate graphs must remain of fixed size n as they grow in number, so that the quantity M_n remains constant: this setting is common in cluster-randomized trials where a growing number of units is clustered into fixed-size clusters of size at most n. The asymptotic behavior of our proposed methodology is less clear in settings in which the analyst cannot perform such clustering before randomization and n is allowed to grow with N, and is an avenue for potential future research.

B.3 Heteroskedasticity in The Baseline Effects

In a network setting such as the one we study, it is possible that baseline effects of units do not have equal variance. Here we discuss how this setting affects our result in Proposition 2. Here, we maintain that $\mathbb{E}[\epsilon_i] = \alpha$ for all i, but we assume that $Var(\epsilon_i) = \sigma_i^2$, and that $Cov(\epsilon_i\epsilon_j) \neq 0$. Starting from the upper bound on the estimation error given in (10), we can see that the baseline effects only come in in the term: $\mathbb{E}[|\epsilon_i - \epsilon_j||j \in \texttt{MG}_i]$, we therefore focus our attention on this term, as the rest of this bound does not change when the variance of these terms changes. Note first, that $\mathbb{E}[|\epsilon_i - \epsilon_j||j \in \texttt{MG}_i] = \mathbb{E}[|\epsilon_i - \epsilon_j|]$ as the event $j \in \texttt{MG}_i$ is independent of the baseline effects. We can now apply the Cauchy-Schwarz inequality, in the same way as we do in the proof of Proposition 2, to obtain:

$$\begin{aligned}
\mathbb{E}[|\epsilon_i - \epsilon_j|] &\leq \sqrt{\mathbb{E}[(\epsilon_i - \epsilon_j)^2]} \\
&= \sqrt{\mathbb{E}[\epsilon_i^2] + \mathbb{E}[\epsilon_j^2] - 2\mathbb{E}[\epsilon_i]\mathbb{E}[\epsilon_j]} \\
&= \sqrt{\sigma_i^2 + \alpha^2 + \sigma_j^2 + \alpha^2 - 2\alpha^2} \\
&= \sqrt{\sigma_i^2 + \sigma_j^2}.
\end{aligned}$$

Clearly, this is not too different from the homoskedastic setting we study in the proposition: as long as neither of the unit variances is too large for inference, results in the heteroskedastic setting will suffer from similar bias as they would under independent baseline effects with equal variance.

Simulations, shown in Section J, also support the above rationale of comparable performance in the heteroskedastic case and demonstrate that FLAME-Networks still outperforms competing methods.

C Derivation of SANIA MIVLUE Used in Simulations

Theorem 6.2 of Sussman and Airoldi, 2017
Suppose potential outcomes satisfy SANIA and that the prior on the parameters (baseline outcome and direct treatment effect) has no correlation between units. If unbiased estimators exist, the MIVLUE weights are:

$$w_i(\mathbf{z}) = \frac{C_{i,d_i^{\mathbf{z}}}}{\sum_{d=0}^{n-1} C_{i,d}} \cdot \frac{2z_i - 1}{n\mathrm{P}(\mathbf{z}_i^{\mathrm{obs}} = z_i, d_i^{\mathbf{z}^{\mathrm{obs}}} = d_i^{\mathbf{z}})}$$

where

$$C_{i,d} = \left(\sum_{\mathbf{z}} \mathrm{P}(\mathbf{z}) \mathbf{1}\{d_i^{\mathbf{z}} = d\} \cdot \frac{\Sigma(\mathbf{z})_{ii}}{\mathrm{P}(z_i^{\mathrm{obs}} = z_i, d_i^{\mathbf{z}^{\mathrm{obs}}} = d)^2} \right)^{-1}$$

and $C_{i,d}$ is defined to be 0 if the probability in its denominator is 0.

C.1 Setup and Notation

We assume that there is a constant probability p of each unit being treated and that units are independently treated. Let unit i have d_i neighbors and a treated degree of $d_i^{\mathbf{z}}$. In our setting, $\Sigma(\mathbf{z})_{ii} = \Sigma_{\alpha,ii} + z_i\Sigma_{\beta,ii}$ where Σ_α and Σ_β are the covariance matrices on priors placed on the baseline outcome and the direct treatment effect, respectively. Additionally in our setting, their diagonals are constant and so we let $\sigma_\alpha^2 := \Sigma_{\alpha,ii}$ and $\sigma_\beta^2 := \Sigma_{\beta,ii}$.

C.2 Find $\mathrm{P}(z_i^{\mathbf{obs}} = z_i, d_i^{\mathbf{z}^{\mathbf{obs}}} = d_i^{\mathbf{z}})$

By the setup, the constituent probabilities are independent and the probability of treatment is constant across units and so: $\mathrm{P}(z_i^{\mathrm{obs}} = z_i, d_i^{\mathbf{z}^{\mathrm{obs}}} = d_i^{\mathbf{z}}) = [z_i p + (1 - z_i)(1 - p)]\binom{d_i}{d_i^{\mathbf{z}}} p^{d_i^{\mathbf{z}}}(1-p)^{d_i - d_i^{\mathbf{z}}}$

C.3 Find $C_{i,d}$

Below, we will consider $\mathbf{z}_{\mathrm{neighbor}(i)}$, the treatment assignment of the neighbors of i (excluding i), z_i, the treatment assignment of unit i, and $\mathbf{z}_{\mathrm{rest}(i)}$, the treatment assignment of the remaining units.

$$\begin{aligned}
C_{i,d} &= \left(\sum_{\mathbf{z}} \frac{\mathrm{P}(\mathbf{z})\mathbf{1}\{d_i^{\mathbf{z}} = d\}\Sigma(\mathbf{z})_{ii}}{\mathrm{P}(z_i^{\mathrm{obs}} = z_i, d_i^{\mathbf{z}^{\mathrm{obs}}} = d)^2} \right)^{-1} \\
&= \left(\sum_{\mathbf{z}: d_i^{\mathbf{z}} = d} \frac{\mathrm{P}(\mathbf{z}_{\mathrm{neighbor}(i)})\mathrm{P}(z_i)\mathrm{P}(\mathbf{z}_{\mathrm{rest}(i)})\Sigma(\mathbf{z})_{ii}}{\mathrm{P}(z_i^{\mathrm{obs}} = z_i, d_i^{\mathbf{z}^{\mathrm{obs}}} = d)^2} \right)^{-1} \\
&= \left(\frac{\mathrm{P}(\mathbf{z}_{\mathrm{neighbor}(i)})}{\mathrm{P}(z_i^{\mathrm{obs}} = z_i, d_i^{\mathbf{z}^{\mathrm{obs}}} = d)^2} \right)^{-1} \\
&\quad \times \left(\sum_{\mathbf{z}: d_i^{\mathbf{z}} = d} \Sigma(\mathbf{z})_{ii}\mathrm{P}(z_i)\mathrm{P}(\mathbf{z}_{\mathrm{rest}(i)}) \right)^{-1} \\
&= \left(\frac{\mathrm{P}(\mathbf{z}_{\mathrm{neighbor}(i)})}{\mathrm{P}(z_i^{\mathrm{obs}} = z_i, d_i^{\mathbf{z}^{\mathrm{obs}}} = d)^2} \right)^{-1} \\
&\quad \times \left(\sum_{\mathbf{z}: d_i^{\mathbf{z}} = d} (\sigma_\alpha^2 + z_i\sigma_\beta^2)\mathrm{P}(z_i)\mathrm{P}(\mathbf{z}_{\mathrm{rest}(i)}) \right)^{-1} \\
&= \left(\frac{\mathrm{P}(\mathbf{z}_{\mathrm{neighbor}(i)})}{\mathrm{P}(z_i^{\mathrm{obs}} = z_i, d_i^{\mathbf{z}^{\mathrm{obs}}} = d)^2} \right)^{-1} \\
&\quad \times \left(\sigma_\alpha^2 + \sum_{\mathbf{z}: d_i^{\mathbf{z}} = d} (z_i\sigma_\beta^2)\mathrm{P}(z_i)\mathrm{P}(\mathbf{z}_{\mathrm{rest}(i)}) \right)^{-1} \\
&= \left(\frac{\mathrm{P}(\mathbf{z}_{\mathrm{neighbor}(i)})}{\mathrm{P}(z_i^{\mathrm{obs}} = z_i, d_i^{\mathbf{z}^{\mathrm{obs}}} = d)^2} \right) \\
&\quad \times \left(\sigma^2\alpha + \sum_{\substack{\mathbf{z}: d_i^{\mathbf{z}} = d \\ z_i = 1}} (\sigma_{\beta,}^2)p\mathrm{P}(\mathbf{z}_{\mathrm{rest}(i)}) \right)^{-1} \\
&= \left(\frac{\mathrm{P}(\mathbf{z}_{\mathrm{neighbor}(i)})\left(\sigma_\alpha^2 + \sigma_\beta^2\right)}{\mathrm{P}(z_i^{\mathrm{obs}} = z_i, d_i^{\mathbf{z}^{\mathrm{obs}}} = d)^2} \right)^{-1} \\
&= \frac{\left([z_i p + (1 - z_i)(1 - p)]\binom{d_i}{d} p^d(1-p)^{d_i - d}\right)^2}{(\sigma_\alpha^2 + \sigma_\beta^2)p^d(1-p)^{d_i-d}} \\
&= \frac{[z_i p + (1 - z_i)(1 - p)]^2\binom{d_i}{d}^2 p^d(1-p)^{d_i-d}}{\sigma_\alpha^2 + \sigma_\beta^2}
\end{aligned}$$

C.4 Find w_i

Plugging in the expressions we've found:

$$
\begin{aligned}
&w_i(\mathbf{z}) = \\
&\frac{C_{i,d_i^{\mathbf{z}}}}{\sum_{d=0}^{n-1} C_{i,d}} \cdot \frac{2z_i - 1}{n\mathrm{P}(\mathbf{z}_i^{\text{obs}} = z_i, d_i^{\mathbf{z}^{\text{obs}}} = d_i^{\mathbf{z}})} \\
&= \frac{\frac{[z_i p + (1-z_i)(1-p)]^2 \binom{d_i}{d_i^{\mathbf{z}}}^2 p^{d_i^{\mathbf{z}}} (1-p)^{d_i - d_i^{\mathbf{z}}}}{(\sigma_\alpha^2 + \sigma_\beta^2)}}{\frac{\sum_{d=0}^{n-1} \binom{d_i}{d}^2 p^d (1-p)^{d_i - d} (z_i p + (1-z_i)(1-p))^2}{(\sigma_\alpha^2 + \sigma_\beta^2)}} \\
&\quad \times \frac{2z_i - 1}{n[z_i p + (1-z_i)(1-p)]\binom{d_i}{d_i^{\mathbf{z}}} p^{d_i^{\mathbf{z}}} (1-p)^{d_i - d_i^{\mathbf{z}}}} \\
&= \frac{(z_i p + (1-z_i)(1-p))(2z_i - 1)}{n \sum_{d=0}^{n-1} \binom{d_i}{d}^2 p^d (1-p)^{d_i - d} (z_i p + (1-z_i)(1-p))^2} \\
&= \frac{(z_i p + (1-z_i)(1-p))(2z_i - 1)}{n(z_i p + (1-z_i)(1-p))^2} \\
&\quad \times \frac{1}{\sum_{d=0}^{n-1} \binom{d_i}{d}^2 p^d (1-p)^{d_i - d}}
\end{aligned}
$$

Note in the first fraction that $(z_i p + (1-z_i)(1-p))(2z_i - 1)$ equals p when $z_i = 1$ and $p - 1$ when $z_i = 0$. Also, $(z_i p + (1-z_i)(1-p))^2$ equals p^2 when $z_i = 1$ and $(1-p)^2$ when $z_i = 0$. Thus, the first term is $1/np$ when $z_i = 1$ and $-1/n(1-p)$ when $z_i = 0$ and so the overall expression for the weights is:

$$
w_i(\mathbf{z}) = \frac{z_i/np - (1-z_i)/(n(1-p))}{\sum_{d=0}^{n-1} \binom{d_i}{d}^2 p^d (1-p)^{d_i - d}}
$$

and the MIVLUE is given by $\sum_{i=1}^{n} w_i Y_i$.

D Subgraph Descriptions

Here, for graphs without self-loops, we define the interference components used in our simulations:

- Degree: the degree of a node is the number of edges it is a part of.
- Triangle: A graph with three mutually connected nodes (see Figure 5).
- Square: A graph with 4 nodes and 4 edges, such that each node is a part of exactly two distinct edges (see Figure 5).
- k-Star: A graph with $k+1$ nodes, the first k of which are all connected to the $(k+1)$st node and no others (see Figure 5).

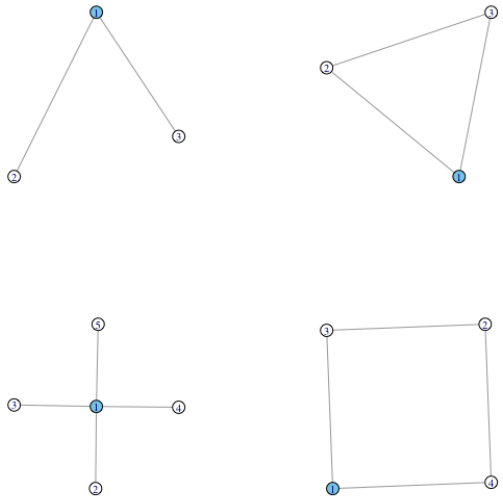

Figure 5: Four types of treated neighborhood subgraphs the colored unit might be a part of: triangle (top right), square (bottom right), 2-star (top left), 4-star (bottom left).

- Vertex Betweenness: The vertex betweenness of a vertex v is defined as:

$$
\sum_{i \neq v \neq j} \frac{\sigma_{ij}(v)}{\sigma_{ij}}
$$

  where σ_{ij} is the number of shortest paths between vertices i and j and $\sigma_{ij}(v)$ is the number of shortest paths between i and j that go through v. We use the normalized vertex betweenness which scales the above expression by $2/(n^2 - 3n + 2)$ where n is the number of nodes in the graph.

- Closeness Centrality: We use the normalized closeness centrality of a vertex v, defined as:

$$
\frac{n-1}{\sum_{i=1}^{n} d(\sigma_{vi})}
$$

  where $d(\sigma_{vi})$ is the length of the shortest path between v and i and n is the number of nodes in the graph.

E Data Pre-processing

We estimate the DTE using data from from Banerjee et al. (2013) on social networks for the 75 villages in India. A unit i is defined to be socially connected with

Units	Triangles	2-Stars	Edges	Telugu	Age	Treated	Outcome
Matched Group 1							
1	2 or 3	3	3	0	40	0	1
2	2 or 3	3	3	0	40	1	0
3	2 or 3	3	3	0	40	1	0
4	2 or 3	3	3	0	40	1	0
Matched Group 2							
1	0	3	3	1	30	0	0
2	0	3	3	1	34	0	0
3	0	3	3	1	25	0	1
4	0	3	3	1	20	1	0

Table 4: Sample Match Groups. Two sample matched groups generated by FLAME-Networks using data discussed in Section 4. The columns are the covariates used for matching, along with treatment status and outcome. The counts of subgraphs were coarsened into 10 bins defined by deciles of the empirical distribution of counts. The two groups have relatively good match quality overall. Note that the first group matches units exactly (given the binning). However, Matched Group 2 matches units approximately, with exact matches on subgraph counts and whether or not individuals speak Telugu, but inexact matches on age.

unit j if they are connected across at least 3 of the following four types of social connections: (1) i visits j's house, (2) j visits i's house, (3) i and j socialize and are relatives, (4) i and j socialize and are not related to each other. Along with subgraph counts, we also use age and whether or not individual i speaks Telugu as additional covariates to match on. For computational efficiency, we drop units with maximum degree of connection greater than 15, where the cut-off is selected based on computational efficiency.

F Matched Groups

We provide sample matched groups in Table 4. These matched groups were produced by applying FLAME-Networks on the data discussed in Section 4. We report all the covariates used for matching. The first group is comprised of 40-year-old units who do not speak Telugu, and have 2 or 3 triangles, 3 2-stars, and 3 edges in their treated neighborhood graph. These units (given the binning) are matched exactly. The second group is comprised of units who speak Telugu with no triangles, 3 2-stars, 3 edges in their treated neighborhood graph. Note that units in this group are matched approximately, since they are not matched exactly on age.

G Additional Experiment: Multiplicative Interference

We explore settings in which interference is a nonlinear function of the interference components and their weights. Since matching is nonparametric, it is particularly appropriate for handling non-linearities in interference functions. Outcomes in this experiment have the form $Y_i(t, \mathbf{t}_{-i}) = t\tau_i + \alpha \prod_{p=1}^{P} m_{ip}^{\mathbb{I}[p \text{ is included}]} + \epsilon_i$. Table 5 shows which components are included in the outcome function for each setting. We use a small number of parameters in each setting, as their multiplicative interaction suffices to define a complex interference function. The simulations are run on an $ER(50, 0.05)$ graph.

Setting	d_i	Δ_i	$\bigstar_i^4$	B_i
1	x	x		
2	x			x
3		x		x
4		x	x	

Table 5: Parameters included in interference function Experiment 1. The marked components for each setting were the only ones included in those experiments.

Results for this experiment are presented in Figure 6. FLAME-Networks performed better than all baseline methods in this setting, both in terms of mean absolute error and, in most cases, in terms of standard deviation over simulations. The stratified and SANIA estimators perform especially poorly, because they cannot handle nonlinear interference settings, unlike FLAME-Networks.

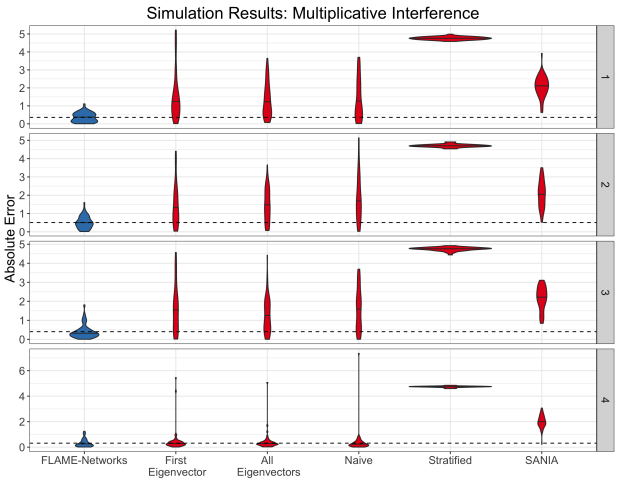

Figure 6: Results from experiments with a multiplicative interference function. Each violin plot represents the distribution over simulations of absolute estimation error over for each method. The panels are numbered according to the parameter setting the simulations were ran with. Violin plots are color-coded blue if the method had mean error either equal to or better than FLAME-Networks and red otherwise. The black line inside each violin is the median error. The dashed line is FLAME-Networks' mean error.

H Additional Experiment: Graph Cluster Randomization

We also explored the performance of FLAME-Networks in settings in which treatment is randomized within multiple clusters, which have few connections between them. More specifically, we simulate a network according to a stochastic block model with 5 clusters. In each cluster, there are 10 units, 5 of which are treated. The probability of edges within clusters is 0.3 and between clusters is 0.05. This results in graphs with few edges between clusters. We then evaluate our method as previously described, simulating the outcome with additive interference and homoskedastic errors. The results in Figure 7 demonstrate that FLAME-Networks outperforms competing methods in this setting as well.

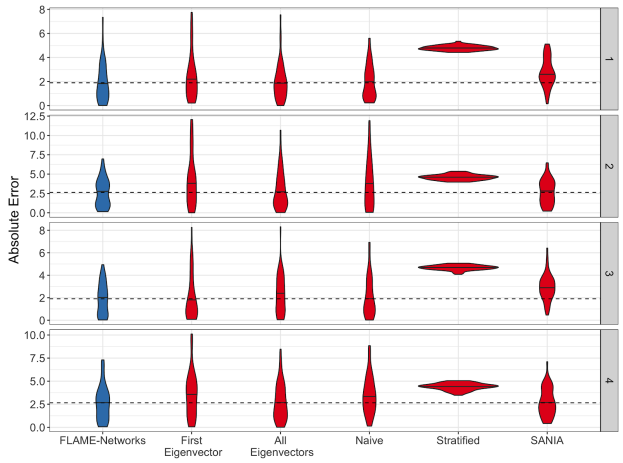

Figure 7: Results from experiments with additive interference on graphs in which treatment is randomly assigned within multiple clusters with few edges between them. Each violin plot represents the distribution over simulations of absolute estimation error over for each method. The panels are numbered according to the parameter setting the simulations were ran with. Violin plots are color-coded blue if the method had mean error either equal to or better than FLAME-Networks and red otherwise. The black line inside each violin is the median error. The dashed line is FLAME-Networks' mean error.

I Additional Experiment: Real Network

To ensure that FLAME-Networks also performs well on real networks, we consider an AddHealth network (Harris, 2013). Specifically, we use the addhealthc3 dataset from the **amen R** package, with all edges treated as undirected. There are 32 nodes and on every simulation, 16 are randomly selected to be treated. Outcome and additive interference are simulated as previously described. Errors are homoskedastic. The results in Figure 8 demonstrate that FLAME-Networks still outperforms competing methods.

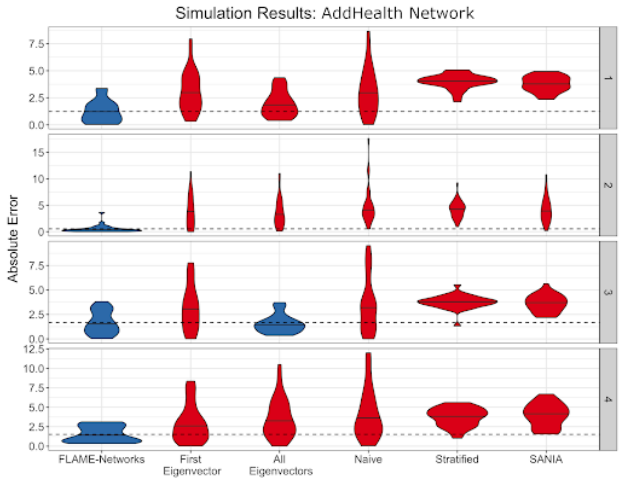

Figure 8: Results from experiments on a real, AddHealth network with additive interference. Each violin plot represents the distribution over simulations of absolute estimation error over for each method. The panels are numbered according to the parameter setting the simulations were ran with. Violin plots are color-coded blue if the method had mean error either equal to or better than FLAME-Networks and red otherwise. The black line inside each violin is the median error. The dashed line is FLAME-Networks' mean error.

J Additional Experiment: Heteroscedastic Errors

We also explored the performance of FLAME-Networks in settings in which the variance of the outcomes is not constant. We simulate a single ER(50, 0.07) graph and randomly treat 25 units. We consider additive interference, as in the body of the text, and all other simulation parameters are the same, expect for that now, each unit i has baseline outcome $\alpha_i \overset{ind}{\sim} N(0, v_i)$ with $v_i \overset{ind}{\sim} U(0,1)$. We see in Figure 9 that FLAME-Networks outperforms competitors; the fact that it is nonparametric allows it to handle more flexible baseline outcomes and variances.

K Additional Experiment: Matching on True Interference

Here, we compare FLAME-Networks to approaches that match directly on units' interference values.

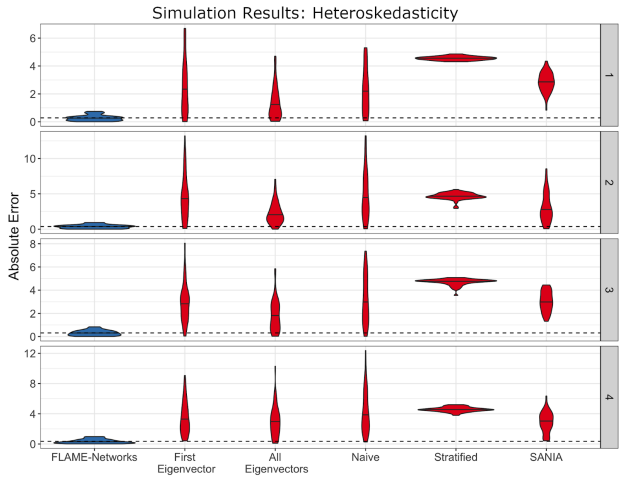

Figure 9: Results from experiments with an additive interference function involving heteroskedasticity in the baseline effects across units. Each violin plot represents the distribution over simulations of absolute estimation error over for each method. The panels are numbered according to the parameter setting the simulations were ran with. Violin plots are color-coded blue if the method had mean error either equal to or better than FLAME-Networks and red otherwise. The black line inside each violin is the median error. The dashed line is FLAME-Networks' mean error.

FLAME-Networks already has the advantage of interpretably matching on neighborhood graphs that can be visualized and compared, as opposed to uninterpretable scalar values of an interference function. Additionally, to perform well in practice, one would typically need to use equally uninterpretable machine learning methods to estimate units' interference values well. But even comparing FLAME-Networks to an approach that matches on the *true* (typically unknown) interference values, we see that our method does well in comparison, because it learns and matches on baseline effects as well as (approximate) interference values. Results using an ER(50, 0.07) graph with 25 units randomly treated, an additive interference function, and homoskedastic errors – as previously described – are shown in Figure 10.

L Additional Experiment: Covariate Weight

In this section, we show that increasing the influence that covariates have on the outcome function harms neither FLAME-Networks nor the competing methods. As in the results shown in the main text, however, the performance of FLAME-Networks is still superior, given that it naturally handles covariate data. The experimental setup is the same as in Experiment 2 in the main text and results are shown in Tables 6 and 7

Method	Median	25th q	75th q
FLAME-Networks	0.34	0.15	0.52
First Eigenvector	0.41	0.24	0.49
All Eigenvectors	0.36	0.32	0.74
Naive	0.61	0.19	0.85
SANIA	2.31	1.78	2.75
Stratified	4.56	4.55	4.63

Table 6: Additional results from the experimental setup of Experiment 2, but with $\beta = 20$. Median and 25th and 75th percentile of absolute error over 10 simulations.

Method	Median	25th q	75th q
FLAME-Networks	0.25	0.08	0.51
First Eigenvector	0.52	0.32	0.85
All Eigenvectors	0.53	0.29	0.83
Naive	0.78	0.32	1.16
SANIA	1.86	1.68	2.11
Stratified	4.78	4.74	4.80

Table 7: Additional results from the experimental setup of Experiment 2, but with $\beta = 25$. Median and 25th and 75th percentile of absolute error over 10 simulations.

M Match Quality

Here, we assess the quality of matches generated by FLAME-Networks versus matching on true interference, and the All Eigenvectors approach. To do so, for FLAME-Networks: for each (control) treated unit, we take the minimal Frobenius norm of the difference between that unit's neighborhood adjacency matrix and that of all the (treated) control units in its matched group[1], and average across all units. This gives an average graph distance for a single simulation. And to do so for the true interference matching and All Eigenvectors approaches: for every (control) treated unit, we take the closest (treated) control unit, find the graph distance (as above) between their neighborhood subgraphs, and average across units. This gives an average graph distance for a single simulation. Results from 50 simulations performed on an ER(50, 0.07) graph with additive interference and homoskedastic errors, as described previously, are shown in Figure 11, showing that FLAME-Networks produces more matches between units with similar neighborhood subgraphs than

[1]The Frobenius norm of the difference of the adjacency matrices, up to reordering of the vertices.

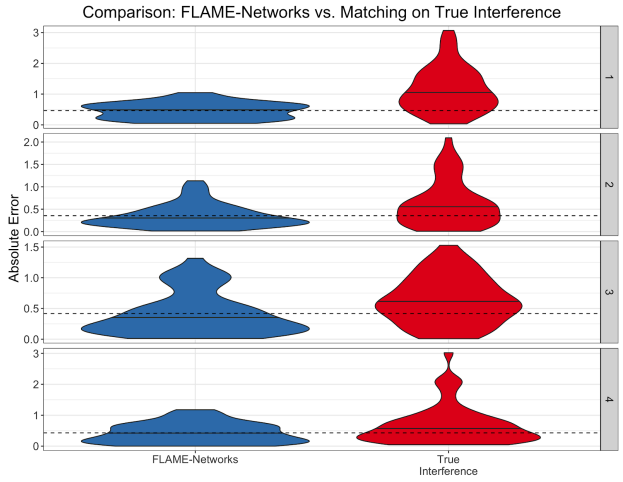

Figure 10: Results from experiments comparing FLAME-Networks to matching units on their true interference values. Each violin plot represents the distribution over simulations of absolute estimation error over for each method. The panels are numbered according to the parameter setting the simulations were ran with. Violin plots are color-coded blue if the method had mean error either equal to or better than FLAME-Networks and red otherwise. The black line inside each violin is the median error. The dashed line is FLAME-Networks' mean error.

matching on the true interference or the All Eigenvectors method.

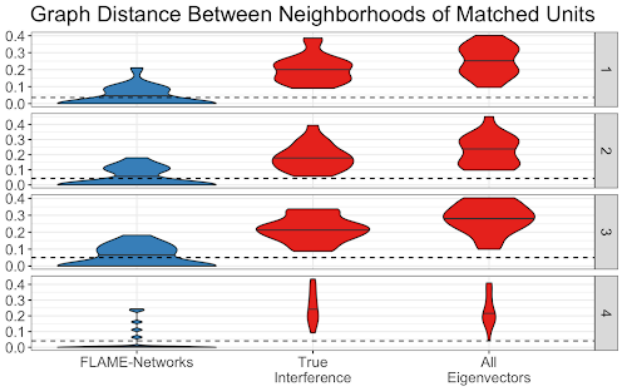

Figure 11: Results from experiments comparing the average distance between the neighborhood subgraphs of the units matched by different methods. Each violin plot represents the distribution over simulations of graph distance for each method. The panels are numbered according to the parameter setting the simulations were ran with. Violin plots are color-coded blue if the method had mean graph distance either equal to or better than FLAME-Networks and red otherwise. The black line inside each violin is the median graph distance. The dashed line is FLAME-Networks' mean graph distance.