Optimal Algorithms for Fixed-Graph Multi-Attribution Privacy

Raman Arora
Department of Computer Science
Johns Hopkins University
Baltimore, MD 21218
arora@cs.jhu.edu

Kaibo Zhang
Department of Computer Science
Johns Hopkins University
Baltimore, MD 21218
kzhang90@jhu.edu

Abstract

Traditional differential privacy assumes each training example affects only one user's privacy, but many real-world datasets contain examples attributed to multiple users. We study *fixed-graph differential privacy*, a framework for learning from such multi-attribution data where the attribution structure is public but example contents are private. We establish fundamental limits and optimal algorithms for this setting. First, we analyze approximation algorithms for the NP-hard contribution bounding problem. An LP-guided greedy algorithm achieves an $O(\min\{r^{1/k}, h^{1/(k+1)}\})$-approximation, where k is the contribution cap, r is the maximum number of users per example, and h is the number of users whose degree exceeds k; if $h = 0$, all examples can be retained. One weighted traversal achieves both established packing guarantees. We also prove that the greedy algorithm of Ganesh et al. (2025) has an exactly tight r-approximation ratio. For $k = 1$, known $\Omega(r)$ hardness makes this dependence on r optimal up to constants, assuming $\mathrm{NP} \not\subseteq \mathrm{BPP}$. Second, we introduce a *network density parameter* c that measures how concentrated the attribution network is, and establish information-theoretic error lower bounds scaling as $\tilde{\Omega}(c^2 d/\varepsilon^2)$ for estimating d-dimensional means from n examples. We show that c alone does not determine the achievable error, and introduce a retention-refined density profile c_β under which, in the standard small-ε regime, upper bounds from contribution bounding with the Gaussian mechanism match the lower bounds on every instance up to logarithmic factors. We further characterize when contribution bounding introduces selection bias and provide conditions under which bias vanishes. Finally, for i.i.d. samples, the privacy lower bound for hypothesis testing weakens from c to $\sqrt{c/n}$ scaling; this weaker rate is attained on clustered networks, and we show it is not attainable in general.

1 Introduction

Differential privacy [25, 24] has emerged as the gold standard for privacy-preserving machine learning, providing formal guarantees that individual data contributors cannot be identified from trained models. Traditional privacy frameworks assume each training example affects only the privacy of the person who contributed that example. However, this assumption fails for many real-world datasets where individual examples contain private information about *multiple* users. Consider email data, where a single message contains information about the sender, all recipients, and anyone mentioned in the conversation. Collaborative documents reveal information about all co-authors. Medical records contain information about patients, physicians, and family members. Social network posts expose information about the poster and all tagged individuals. In each case, a training example is *attributed* to multiple users whose private information it contains. These attributions form a hypergraph with users as vertices and examples as hyperedges.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

The recent work of Ganesh et al. [33] introduced this *multi-attribution privacy* problem and proposed *fixed-graph differential privacy*, a framework where the attribution hypergraph structure (V, E) is public, but example contents $\{x_e\}_{e \in E}$ are private. Their key algorithmic primitive is *contribution bounding*, which selects a subset $S \subseteq E$ such that each user appears in at most k selected examples. This enables standard differentially private algorithms like DP-SGD to provide user-level privacy guarantees. Ganesh et al. [33] proposed greedy algorithms and demonstrated empirical feasibility on transformer training; however, they left critical theoretical questions unanswered, including:

- What approximation guarantees do contribution bounding algorithms achieve?
- What are the fundamental sample complexity limits for private learning on networked data?
- How do aspects of network structure such as degree distributions, clustering, and sparsity impact privacy-utility tradeoffs?
- When does contribution bounding introduce systematic bias, and how can it be controlled?

Contributions. We develop theoretical foundations for fixed-graph differential privacy, studying approximation algorithms, statistical limits, and selection bias. Our main contributions include:

1. **Approximation Guarantees for Contribution Bounding (Section 3).**
   We analyze contribution bounding through its connection to hypergraph b-matching and column-sparse packing (Section 3.4). Our LP-guided greedy algorithm processes hyperedges in a random order weighted by an optimal LP solution and achieves an $O(\min\{r^{1/k}, h^{1/(k+1)}\})$-approximation, where $h = |V_{>k}|$ counts vertices of degree greater than k. If $h = 0$, the full edge set is feasible. A single traversal attains both established approximation orders without choosing a sampling scale. In particular, the guarantee is constant when $k = \Omega(\log r)$, independently of the number of users. We also analyze the greedy algorithm of Ganesh et al. [33], proving an exactly tight r-approximation bound. For $k = 1$, the $\Omega(r)$ hardness of Lee et al. [49], assuming $\mathrm{NP} \not\subseteq \mathrm{BPP}$, shows that this dependence on r is optimal up to constants. Finally, we establish per-instance bounds showing that structured networks with small average hyperedge size or community structure admit better approximation ratios than worst-case analysis suggests.
2. **Fundamental Sample Complexity Limits (Section 4).** We introduce a network density parameter c that characterizes how concentrated the attribution network is, and establish information-theoretic lower bounds. Reducing from known lower bounds for 1-way marginals [9], we prove that any (ε, δ)-fixed-graph differentially private algorithm has error at least $\tilde{\Omega}(c^2 d/\varepsilon^2)$ for estimating d-dimensional means, and more generally we characterize the optimal error on every instance through a retention-refined profile c_β, with lower and upper bounds stated at a common retention level. For $0 < \varepsilon \leq 1$, we obtain Gaussian upper bounds that match the lower bounds up to logarithmic factors in their common parameter regime. We characterize when contribution bounding's structure-based selection introduces bias, providing sufficient conditions for zero bias (data independent of structure, or selection preserving type distributions) and quantitative bounds showing bias is controlled by the distribution shift and type-proportion changes.
3. **Improved Bounds for i.i.d. Data (Appendix E).** When examples are sampled i.i.d. from a distribution (rather than worst-case), we obtain qualitatively better guarantees. For hypothesis testing between Bernoulli distributions separated by distance β, any $(\varepsilon, 0)$-DP algorithm requires error $\Omega(\exp(-\Theta(\varepsilon\beta\sqrt{n\log(n)/c})))$. The network dependence in the privacy lower bound changes from c to $\sqrt{c/n}$, a polynomial reduction when $c \gg \frac{1}{n}$. For clustered networks with $k = n/\Delta$ disjoint hubs, a group-based testing algorithm achieves matching upper bounds with the same $\exp(-\Theta(\varepsilon\beta\sqrt{n/c}))$ error rate. The proof technique uses a novel distance metric $d(X, Y)$ based on the minimum vertex cover of differing examples.

Technical Overview. Our technical contributions rest on three key insights:

1. **Contribution bounding and hyperedge size.** The LP-guided algorithm uses an optimal fractional solution to choose a weighted random order and accepts each feasible hyperedge. Its analysis considers two prefixes of an independent sampling sequence: one controls the chance that any vertex of the current edge is saturated, giving the $r^{1/k}$ bound; the other controls overload across the h potentially binding vertices, giving the $h^{1/(k+1)}$ bound. The order of first appearances in this sequence is exactly weighted sampling without replacement. Repeated appearances never change the greedy output, so the complete traversal inherits both prefix guarantees.
   For arbitrary processing orders, rejected edges can be charged to saturated vertices, yielding the r-approximation bound independently of the maximum degree Δ. At cap $k = 1$, contribution

bounding inherits $\Omega(r)$ hardness from r-Set Packing under NP $\not\subseteq$ BPP [49]. Thus greedy's dependence on r is optimal up to constants here; larger budgets allow the stronger LP guarantee.

2. **Retention and network concentration.** We introduce a network density parameter c that measures how dispersed examples can be across users in the best case. Intuitively, it captures the unavoidable concentration of examples around high-degree users. This single parameter ranges from $c = 1/n$ (maximally dispersed, as in matching graphs) to $c = 1$ (maximally concentrated, as in star networks), with $c = \Theta(\Delta/n)$ for regular networks. Our lower bounds of $\tilde{\Omega}(c^2 d/\varepsilon^2)$ hold on every network. The achievable error is characterized on every instance by the retention-refined profile c_β (Definition 9) rather than by c alone, which can be far from tight on networks that mix a dense core with dispersed examples (Appendix C.5). The distinction matters because minimizing concentration alone may select only a small fraction of the data. Requiring retention of at least $(1-\beta)n$ examples lets us compare lower and upper bounds at a common retention level and balance the bias from discarding data against the privacy noise. The peeling algorithm makes this tradeoff computationally useful: it removes excess edges to meet a degree cap, with a bound on the number of deletions, and chooses the cap using only the public graph.
3. **Systematic selection bias arises from structural correlation.** Since contribution bounding selects examples using only the public graph structure (V, E) without seeing data, systematic selection bias can arise when network structure (hyperedge size, vertex degrees) correlates with data distributions. We formalize this through a type-based framework: bias is controlled by (a) how much selection changes the distribution over structural types and (b) how much the data distributions differ across types. This provides sufficient conditions for zero bias and quantitative bounds otherwise.

1.1 Related Work

Ganesh et al. [33] introduced fixed-graph differential privacy for data in which an example may contain information about several users. They proposed greedy contribution bounding algorithms and demonstrated their use in private transformer training. We build on this framework by studying the approximation guarantees of contribution bounding and its consequences for private mean estimation. Contribution bounding limits how many selected examples one user can affect. Privacy noise depends on this cap and the number of retained examples; selection can also shift the subset mean away from the full-data mean. Our analysis accounts for both effects and asks how the attribution structure determines the accuracy that is achievable.

Contribution bounding has a direct interpretation in combinatorial optimization. It is the maximum unweighted b-matching problem in a hypergraph, with capacity $b = k$ at each vertex. Equivalently, it is a packing integer program whose constraint matrix has at most r nonzero entries in each column [62, 48, 7]. Randomized rounding gives the established $O(r^{1/k})$ and $O(h^{1/(k+1)})$ approximation orders, where h counts the nontrivial degree constraints. We show that one LP-guided greedy traversal achieves both. At capacity $k = 1$, contribution bounding reduces to r-Set Packing, for which Lee et al. [49] prove $\Omega(r)$ hardness under NP $\not\subseteq$ BPP. Our statistical analysis studies how the retained examples and their contribution cap affect private estimation of the mean of the original dataset.

Repeated participation is central to user-level differential privacy, which protects all examples contributed by one user. For worst-case data, group privacy gives a direct way to account for multiple contributions [24, 17]. Under i.i.d. assumptions, dedicated user-level algorithms can obtain substantially better guarantees [50, 8, 35]. Multi-attribution introduces a further difficulty because the examples associated with different users can overlap. Our lower bounds use a reduction from one-way marginals estimation [9], the problem of privately estimating the coordinate-wise means of a dataset. The attribution structure determines how this lower-bound problem can be embedded in the fixed graph. The retention-refined density profile then relates the resulting privacy cost to the subsets available to a contribution bounding algorithm. Additional related work is discussed in Appendix A.

2 Preliminaries and Problem Formulation

Here, we introduce notation, review differential privacy, and formalize multi-attribution learning.

Notation. We use $[n]$ to denote the set $\{1, 2, \ldots, n\}$. For a vector $v \in \mathbb{R}^d$, we write $\|v\|_2$ for its Euclidean norm. We denote the set of probability distributions over a space $\mathcal{X}$ by $\Delta(\mathcal{X})$.

A *hypergraph* $\mathcal{H} = (V, E)$ consists of a vertex set V and a collection of hyperedges E, where each hyperedge $e \in E$ is a subset of V. Hyperedges may contain any number of vertices. For a vertex $v \in V$, its *degree* is $\deg(v) = |\{e \in E : v \in e\}|$, the number of hyperedges containing v. The

maximum degree of the hypergraph is $\Delta = \max_{v \in V} \deg(v)$. The *size* of a hyperedge e is $|e|$, and the *maximum hyperedge size* is $r = \max_{e \in E} |e|$.

Next, we briefly review differential privacy and establish conventions used throughout the paper.

Definition 1 (Differential Privacy [25])**.** *Two datasets $D, D' \in \mathcal{X}^*$ are* adjacent*, denoted $D \sim D'$, if they differ in exactly one element. A randomized mechanism $\mathcal{M} : \mathcal{X}^* \to \mathcal{Y}$ satisfies* (ε, δ)-differential privacy *if for all adjacent datasets $D \sim D'$ and all measurable $S \subseteq \mathcal{Y}$, $\Pr[\mathcal{M}(D) \in S] \leq e^{\varepsilon} \Pr[\mathcal{M}(D') \in S] + \delta$. We write $\mathcal{M}(D) \approx_{\varepsilon,\delta} \mathcal{M}(D')$ to denote this relation.*

Definition 2 (Group Privacy [25])**.** *A mechanism $\mathcal{M}$ satisfying (ε, δ)-DP provides $(k\varepsilon, ke^{(k-1)\varepsilon}\delta)$-*group privacy *for groups of size k: for any two datasets D, D' differing in at most k elements, $\Pr[\mathcal{M}(D) \in S] \leq e^{k\varepsilon} \Pr[\mathcal{M}(D') \in S] + ke^{(k-1)\varepsilon}\delta$.*

We now formalize the multi-attribution setting introduced by Ganesh et al. [33].

Definition 3 (Attributed Dataset)**.** *An* attributed dataset *is a tuple $D = (V, E, \{x_e\}_{e \in E})$ where V is a set of users (vertices), E is a collection of hyperedges over V, and for each $e \in E$, $x_e \in \mathcal{X}$ is a training example attributed to all users in e. We write $\mathcal{H}(D) = (V, E)$ for the underlying hypergraph and $|D| = |E|$ for the number of examples.*

In the email dataset from Ganesh et al. [33], V represents email addresses, each hyperedge e contains the sender and all recipients of an email, and x_e is the email text.

Definition 4 (Fixed-Graph DP [33])**.** *Two attributed datasets $D = (V, E, \{x_e\})$ and $D' = (V, E, \{x'_e\})$ are* fixed-graph adjacent for user v *if they have the same hypergraph $\mathcal{H}(D) = \mathcal{H}(D')$ and $x_e = x'_e$ for all $e \not\ni v$. A mechanism $\mathcal{M}$ satisfies* (ε, δ)-fixed-graph differential privacy *if for all users $v \in V$ and all datasets D, D' that are fixed-graph adjacent for v, $\mathcal{M}(D) \approx_{\varepsilon,\delta} \mathcal{M}(D')$.*

Intuitively, fixed-graph DP protects against changing all data associated with any single user, regardless of how many examples are attributed to that user.

When is the model appropriate? The fixed-graph guarantee protects contents while treating the attribution structure as given, so it fits settings where the structure is already releasable to the recipients of the trained model or has been explicitly designated nonsensitive while contents remain confidential. A clean example is publicly registered project or committee membership paired with confidential team-generated records attributed to the listed team as a whole; the roster is public by construction and the document contents remain confidential. Visibility of the structure to the curator alone does not justify the model. When the existence of an edge is itself sensitive, a stronger model that protects the attribution structure is required, with edge-level or node-level adjacency depending on the threat model; node-level differential privacy protects a user's participation by allowing the removal of that user and all incident examples. The key algorithmic primitive for achieving fixed-graph DP is contribution bounding.

Definition 5 (Contribution Bounding)**.** *Given an attributed dataset $D = (V, E, \{x_e\})$ and a budget $k \in \mathbb{N}$, the* contribution bounding problem *asks for a sub-hypergraph (V, S) with $S \subseteq E$ that maximizes $|S|$ subject to the* contribution constraint *that for all users $v \in V$, $|\{e \in S : v \in e\}| \leq k$.*

Ganesh et al. [33] show NP-hardness by reducing maximum independent set to contribution bounding. For a graph $G = (V_G, E_G)$, create a hyperedge for each vertex and a user for each edge, with attribution given by incidence. At $k = 1$, feasible selections correspond to independent sets of the same size. For a contribution-bounded dataset $(V, S, \{x_e\}_{e \in S})$, group privacy (Definition 2) translates DP-SGD or DP-MF guarantees to fixed-graph DP [33].

3 Approximation Algorithms for Contribution Bounding

We analyze LP-guided and arbitrary-order greedy contribution bounding, relate their guarantees to packing results, and show that greedy's dependence on r is optimal up to constants at cap $k = 1$.

3.1 LP-Guided Greedy Algorithm

We begin by formulating contribution bounding as an integer linear program.

Definition 6 (Contribution Bounding IP)**.** *For hypergraph (V, E) and budget k, the integer program is:*

$$\textit{maximize} \sum_{e \in E} y_e \quad \textit{subject to} \quad \sum_{e : v \in e} y_e \leq k \quad \forall v \in V, \;\; y_e \in \{0, 1\} \quad \forall e \in E \tag{1}$$

where $y_e = 1$ indicates hyperedge e is selected.

Algorithm 1 LP-Guided Greedy Contribution Bounding

Require: Hypergraph (V, E), integer budget $k \geq 1$
Ensure: Subset $S \subseteq E$ with maximum degree at most k
If $\deg_E(v) \leq k$ for every $v \in V$, return E
Solve the LP in Definition 7 to obtain an optimal vector y^*
For each e with $y_e^* > 0$, draw $U_e \sim \mathrm{Unif}(0,1)$ independently and set $\tau_e \leftarrow -\log U_e / y_e^*$
Sort positive-weight edges by increasing τ_e; append zero-weight edges in any fixed order
Initialize $S \leftarrow \emptyset$ and $c_v \leftarrow 0$ for every $v \in V$
for each edge e in this order **do**
  if $c_v < k$ for every $v \in e$ **then**
    $S \leftarrow S \cup \{e\}$ and $c_v \leftarrow c_v + 1$ for every $v \in e$
  end if
end for
return S

Definition 7 (Contribution Bounding LP)**.** *The LP relaxation replaces $y_e \in \{0,1\}$ in Definition 6 with $0 \leq y_e \leq 1$ for every $e \in E$.*

Let $\mathrm{OPT}_{\mathrm{IP}}$ and $\mathrm{OPT}_{\mathrm{LP}}$ denote the optimal integer and fractional objective values, respectively. Since every integer solution is feasible for the relaxation, $\mathrm{OPT}_{\mathrm{IP}} \leq \mathrm{OPT}_{\mathrm{LP}}$. We use an optimal fractional solution y^* to choose the order in which a greedy algorithm processes the edges. Among the unprocessed edges with positive weight, the next edge is chosen with probability proportional to y_e^*. It is accepted whenever all its incident vertices have remaining capacity. After these edges have been processed, the algorithm considers the zero-weight edges in any fixed order. Thus every edge is considered once and feasibility is maintained throughout.

This weighted order can be generated by drawing independent $U_e \sim \mathrm{Unif}(0,1)$ and sorting the positive-weight edges by $-\log U_e / y_e^*$. These priorities are independent exponential random variables with rates y_e^*; their order has the required successive sampling probabilities. See Algorithm 1.

Theorem 1 (LP-guided greedy approximation)**.** *Let $k \geq 1$ be an integer, let r be the maximum hyperedge size, and let $h = |V_{>k}|$, where $V_{>k} = \{v \in V : \deg_E(v) > k\}$. Algorithm 1 returns a feasible set S. If $h = 0$, it returns E. Otherwise, let $A_k = \min\{r^{1/k}, h^{1/(k+1)}\}$. There is an absolute constant $c_0 > 0$ such that*

$$\Pr\left[|S| \geq c_0 \frac{\mathrm{OPT}_{\mathrm{LP}}}{A_k}\right] \geq \frac{1}{10}, \qquad \mathbb{E}|S| = \Omega\left(\frac{\mathrm{OPT}_{\mathrm{LP}}}{A_k}\right).$$

In particular, the algorithm is an $O(A_k)$-approximation in expectation. In every run it also satisfies $|S| \geq \mathrm{OPT}_{\mathrm{IP}}/r$ when $r \geq 1$. If the returned optimal LP vector is integral, S is optimal.

Proof sketch. Write $F = \mathrm{OPT}_{\mathrm{LP}}$ and consider independent draws from $p(e) = y_e^*/F$, with the same greedy acceptance rule. We analyze two prefixes of this sequence. For a prefix of length proportional to $F/r^{1/k}$, condition on the current draw. The LP constraints and a Chernoff bound control the chance that any of its at most r vertices is already saturated; the chance of a repeated edge is also small. Each draw is therefore accepted with probability at least a constant. For a prefix of length proportional to $F/h^{1/(k+1)}$, a union bound over $V_{>k}$ gives constant probability that the entire set of distinct sampled edges is feasible. A first-moment bound gives a constant probability of retaining a constant fraction of the prefix length. These arguments yield the two guarantees, including the one-draw cases when a prefix length would round down to zero.

To transfer these bounds to Algorithm 1, observe that the order of first appearances in the independent sequence is exactly the weighted order used by the algorithm. Repeated appearances do not change the greedy output: an accepted edge is already present, and an edge rejected earlier remains infeasible as the selected degrees increase. The complete weighted traversal therefore contains the accepted set from either prefix under this coupling. The final zero-weight pass cannot decrease its size. Since every edge is processed, the usual greedy charging argument gives the deterministic r-approximation. If y^* is integral, its positive support is feasible and all of it is accepted, attaining $F = \mathrm{OPT}_{\mathrm{IP}}$. Appendix B.1 gives the full proof. □

The two terms in A_k describe complementary regimes. The bound is constant when $k = \Omega(\log r)$, even on a network with many users, and it can also be small when only a few vertices have degree

Algorithm 2 Greedy Contribution Bounding

Require: Hypergraph (V, E), integer budget $k \geq 1$
Ensure: Subset $S \subseteq E$ with maximum degree at most k

```
Initialize S ← ∅ and c_v ← 0 for every v ∈ V
for each e ∈ E in an arbitrary order do
    if c_v < k for every v ∈ e then
        S ← S ∪ {e} and c_v ← c_v + 1 for every v ∈ e
    end if
end for
return S
```

above k. These approximation orders are known for packing problems; the result here establishes both for a single LP-guided greedy procedure (Section 3.4). After solving the LP, a traversal takes $O(|E| \log |E| + \sum_{e \in E} |e|)$ time. Repeating it independently and keeping the largest feasible output raises the success probability to $1 - \zeta$ using $O(\log(1/\zeta))$ traversals and a single LP solve.

3.2 Greedy Algorithm

The greedy algorithm has been analyzed and improved in multiple papers for $k = 1$ [16, 32, 64]. We now analyze the greedy contribution bounding algorithm for arbitrary $k \geq 1$ proposed by Ganesh et al. [33]. A charging argument gives an r-approximation, and a worst-case construction shows that this ratio is tight; the algorithm is simple to implement and provides deterministic guarantees.

Theorem 2 (Greedy Approximation)**.** *For any hypergraph (V, E) with maximum hyperedge size $r = \max_{e \in E} |e|$ and budget k, Algorithm 2 produces a solution S satisfying $|S| \geq \frac{\mathrm{OPT_{IP}}}{r}$. In particular, the greedy algorithm achieves an r-approximation, and Proposition 4 shows this ratio is exactly tight.*

Proof sketch. Let S be the greedy solution, S^* optimal, and call a vertex *saturated* if $d_S(v) = k$. Each rejected edge $e^* \in S^* \setminus S$ contains a saturated vertex. Charge every optimal edge once: a rejected edge to one saturated witness vertex; a common edge $e \in S^* \cap S$ touching a saturated vertex to one such vertex; a common edge with no saturated vertex to itself. At each saturated vertex v, at most $d_{S^*}(v) \leq k$ optimal edges are charged, while exactly k selected edges contain v, so the charges at v can be assigned injectively to distinct selected edges through v. A selected edge therefore receives at most one charge through each of its incident saturated vertices, or a single self-charge if it has none, for at most $|e| \leq r$ charges in total. Hence $|S^*| \leq r|S|$. Complete proof in Appendix B.2. □

3.3 Hardness of Approximation

At cap $k = 1$, contribution bounding is exactly r-Set Packing, so the following standard results apply.

Theorem 3 (Hardness from r-Set Packing)**.** *For contribution bounding with maximum hyperedge size r and budget $k = 1$:*

- *(a)* ***General*** *r: Unless $\mathrm{NP} \subseteq \mathrm{BPP}$, no polynomial-time algorithm achieves approximation factor $o(r)$ [49].*
- *(b)* ***Graphs*** *($r = 2$)**: The problem is poly-time solvable via Maximum Matching [26].***
- *(c)* ***Bounded*** *$r = O(1)$**: For every fixed $r \geq 3$, the problem is APX-hard [45], so it admits no polynomial-time approximation scheme unless $\mathrm{P} = \mathrm{NP}$.***

The reduction and proof are given in Appendix B.3. We construct instances showing that the r-approximation ratio of greedy (Theorem 2) is exactly tight.

Proposition 4 (Tight Instance for Greedy)**.** *For any integers $r \geq 2$ and $n \geq r$ with r dividing n, there exists a contribution bounding instance with budget $k = 1$, maximum hyperedge size r, and $\mathrm{OPT_{IP}} = n$, such that the greedy algorithm with adversarial tie-breaking selects only n/r hyperedges. The approximation ratio is exactly r.*

Replicating each hyperedge k times extends the construction to any cap $k \geq 1$; see Appendix B.3.

3.4 Connections to Hypergraph b-Matching and Column-Sparse Packing

Contribution bounding is maximum unweighted hypergraph b-matching with $b = k$, or a 0–1 packing integer program of column sparsity r. The large-capacity rounding theorem of Bansal et al. [7] gives an $O(r^{1/k})$-approximation for integer $k \geq 1$. The complementary $O(h^{1/(k+1)})$ order follows from

standard rounding for 0–1 packing after removing the automatically satisfied constraints at vertices of degree at most k. Theorem 1 obtains both orders from one LP-guided greedy traversal. For graphs ($r = 2$), every cap admits an exact polynomial-time solution by b-matching [59]. These combinatorial guarantees control how many examples can be retained; their implications for sensitivity, selection bias, and private estimation are developed in Section 4. Appendix B.4 gives further comparisons with rounding, local search, and hardness results.

3.5 Structured Networks and Per-Instance Bounds

While our worst-case analysis is tight, structured networks can admit better per-instance bounds. When the average size of selected hyperedges is small, or when the network has community structure with intra-community hyperedges smaller than global maximum size r, the greedy algorithm achieves substantially better approximation ratios. Details and algorithms appear in Appendix B.5.

4 Fundamental Limits for Network-Aware Private Learning

4.1 Error Lower Bounds

Our lower bounds reduce from the one-way marginals bound of Bassily et al. [9, Lemma 5.1], restated as Lemma 16 in the appendix. We introduce a network density parameter to describe how the attribution structure enters this reduction.

Definition 8 (Network Density Parameter)**.** *For a hypergraph (V, E) with $n = |E|$ examples, define for each subset size $m \in [1, n]$:*

$$f(m) = \min_{\substack{\tilde{E} \subseteq E \\ |\tilde{E}|=m}} \left[\max_{v \in V} |\{e \in \tilde{E} : v \in e\}| \right] / m. \tag{2}$$

The network density parameter *is $c = \min_{m \in [1,n]} f(m)$.*

Intuitively, $f(m)$ measures the maximum degree achieved in any subset of m examples, normalized by m. The parameter c captures the most "spread out" subset, the one where the maximum degree is smallest relative to subset size. High values of c (close to 1) indicate concentrated networks where some vertex appears in most examples, while low values ($c \approx 1/n$) indicate spread-out networks where no vertex dominates any subset.

Example 1 (Computing c for Common Network Types)**.** *The table gives c for several canonical networks. Detailed calculations appear in Appendix C.1.*

Network Type	**Network Density c**	**Example**
r-uniform, Δ-regular	$\Theta(\Delta/n)$	d-regular graphs ($r = 2$)
Approximately regular	$\Theta(\Delta/n)$	Dense random graphs
Star networks	1	All edges contain a center
Clustered (disjoint hubs)	$1/k$	k groups, $c = \Delta/n$ if equal-sized
Matching/disjoint edges	$1/n$	Each vertex in exactly one edge
Author-paper hypergraphs	$\Theta(\Delta/n)$	$r = O(1)$ authors per paper

These examples show how c measures the smallest maximum degree relative to subset size that can be achieved by selecting examples. On every family in the table, the refined profile c_β is constant in β up to boundary effects, which is why the bounds of Sec. 4 are informative there; mixed structures on which c_β varies sharply are analyzed in Appendix C.5. The key insight is that c captures network structure in a way that Δ alone cannot. Networks with the same maximum degree Δ can have vastly different values of c depending on whether high-degree vertices are common (large c) or rare (small c).

The network density parameter c from Definition 8 characterizes fundamental limits. We can prove error lower bounds of $\tilde{\Omega}(c^2 d/\varepsilon^2)$ for approximate DP, and $\Omega(c^2 d^2/\varepsilon^2)$ for pure DP (see Appendix C.2 for details). However, achieving matching upper bounds requires a refinement. The issue is that c optimizes over *all* subset sizes $m \in [1, n]$, including very small subsets. While such small subsets may have favorable degree properties (making them well-suited for private learning), using only a small fraction of available data leads to poor generalization and large error on unused examples.

To address this tradeoff between privacy and generalization, we introduce a refined network density parameter c_β that restricts attention to large subsets of size at least $(1-\beta)n$. This retention requirement is needed to control the error introduced by discarding examples. For many practical networks, c_β

remains close to c when β is small. Crucially, c_β enables us to control selection bias. By using a $(1-\beta)n$-sized subset, the fraction of discarded examples is small enough that their contribution to total error can be bounded.

Definition 9 (Refined Network Density Parameter)**.** *For any $\beta \in [0,1]$, define*

$$c_\beta = \min_{m \in [(1-\beta)n, n]} f(m), \tag{3}$$

where $f(m)$ is from Definition 8. When $\beta = 1$, we recover the original network density parameter $c = c_1$. For $\beta \in [0,1)$, the restriction on m gives $c_\beta \geq c$.

Remark 1 (The retention–concentration frontier)**.** *Write $\kappa(m) := \min_{|S|=m} \max_{v \in V} \deg_S(v)$ for the smallest achievable cap at retention exactly m; κ is non-decreasing, since removing an edge from an optimal $(m+1)$-subset gives a feasible m-subset. The density profile and contribution bounding are then two parameterizations of one frontier, $c_\beta = \min_{m \geq (1-\beta)n} \kappa(m)/m$ indexed by retention and $\mathrm{OPT}(k) = \max\{m : \kappa(m) \leq k\}$ indexed by the cap. The exact identity $c_\beta(1-\beta)n = \kappa(\lceil (1-\beta)n \rceil)$ fails in general, since the minimizing m may exceed $(1-\beta)n$; on the matching hypergraph the left side is $1-\beta$ while $\kappa \equiv 1$. What holds, up to rounding, is the sandwich $c_\beta(1-\beta)n \leq \kappa(\lceil (1-\beta)n \rceil) \leq c_\beta n$, whose left inequality follows from $c_\beta \leq \kappa(m_0)/m_0$ at $m_0 = \lceil (1-\beta)n \rceil$ and whose right inequality follows from monotonicity of κ and $\kappa(m^\star) = c_\beta m^\star \leq c_\beta n$ at the minimizing $m^\star$; for $\beta \leq 1/2$ the two quantities agree within a factor of two.*

With this refined parameter, we can state a lower bound at every retention level, and a matching characterization that holds on every instance.

Theorem 5 (Network-Density-Dependent Error Lower Bound)**.** *Consider the mean estimation task on attributed datasets with hypergraph structure (V, E). Each hyperedge $e \in E$ is associated with a data point x_e in the d-dimensional bounded domain $\mathcal{X} = \{x \in \mathbb{R}^d \mid \|x\|_2 \leq 1\}$. Given a target error $\alpha > 0$, the learner needs to design an (ε, δ)-fixed-graph DP algorithm $\mathcal{A}$ achieving expected squared error at most α for every sample $X \in \mathcal{X}^n$:*

$$\sup_{X \in \mathcal{X}^n} \mathbb{E}_{\mathcal{A}} \Big\| \mathcal{A}(X) - \frac{1}{n} \sum_{e \in E} x_e \Big\|_2^2 \leq \alpha.$$

Then there exist absolute constants $C, \tilde{C}, \varepsilon_0 > 0$ such that for every $\beta \in (0, 1/2]$ and every $\varepsilon \leq \varepsilon_0$, the existence of such an algorithm $\mathcal{A}$ requires:

$$\alpha \;\geq\; \begin{cases} C \min\left\{\beta^2,\ \frac{c_\beta^2 d^2}{\varepsilon^2}\right\} & \text{if } \delta = 0, \\ \tilde{C} \min\left\{\beta^2,\ \frac{c_\beta^2 d}{\varepsilon^2}\right\} \text{ up to log factors} & \text{if } \delta = o\left(c_\beta\right). \end{cases} \tag{4}$$

In particular, $\alpha \geq C \sup_{\beta \in (0,1/2]} \min\{\beta^2,\ c_\beta^2 d^2/\varepsilon^2\}$ for pure DP, and analogously with d in place of d^2 for approximate DP over those β with $\delta = o(c_\beta)$. Choosing $\beta = C_0\sqrt{\alpha}$ for a suitable absolute constant C_0, for all $\alpha \leq 1/(4C_0^2)$, recovers the implicit bounds $\alpha = \Omega(c^2_{C_0\sqrt{\alpha}} d^2/\varepsilon^2)$ and $\alpha = \tilde{\Omega}(c^2_{C_0\sqrt{\alpha}} d/\varepsilon^2)$.

This theorem shows that the minimum achievable error depends on the refined network density c_β over large subsets, not just the worst-case maximum degree Δ; see Appendix C.2 for a proof. To compare the lower bound with the upper bounds below at a common retention level, define, for $D > 0$, the *rate functional*

$$R(D) \;:=\; \sup_{\beta \in (0,1/2]} \min\{\beta^2,\ c_\beta^2\, D\}. \tag{5}$$

Since β^2 increases and $c_\beta^2 D$ decreases, minimizing their sum over $\beta \in (0, 1/2]$, capped at one, gives $R(D)$ up to absolute constants. Define $\beta^\star = \inf\{\beta \in (0, 1/2] : \beta^2 \geq c_\beta^2 D\}$, with $\beta^\star = 1/2$ if this set is empty. Moreover, $R(LD) \leq LR(D)$ for $L \geq 1$. Lemma 19 in Appendix C.3 proves these claims, allowing jumps in c_β.

4.2 Matching Upper Bounds

We first establish an instance-dependent upper bound matching the lower bounds, then give error bounds for polynomial-time contribution bounding algorithms.

Theorem 6 (Instance-Dependent Matching Upper Bound)**.** *Consider any network (V, E) with $|E| = n$ and the refined network density c_β from Definition 9. Let $0 < \varepsilon \leq 1$ and $0 < \delta \leq e^{-1}$. For every $\beta \in (0, 1]$ there exists an (ε, δ)-fixed-graph DP algorithm $\mathcal{A}$ achieving, for every $X \in \mathcal{X}^n$,*

$$\mathbb{E}_{\mathcal{A}} \Big\| \mathcal{A}(X) - \frac{1}{n} \sum_{e \in E} x_e \Big\|_2^2 \;\leq\; C' \Big(\beta^2 + \frac{c_\beta^2\, d \log(1/\delta)}{\varepsilon^2} \Big) \tag{6}$$

for an absolute constant C'. Consequently, with $D_0 = d/\varepsilon^2$ and $D_1 = d\log(1/\delta)/\varepsilon^2$, the achievable error is $O(R(D_1))$, while Theorem 5 gives $\tilde{\Omega}(R(D_0))$ whenever $\delta = o(c_{\beta^\star})$ at the crossing point $\beta^\star$ of Lemma 19 defined with respect to D_0; since $R(D_1) \le \log(1/\delta)R(D_0)$, the two bounds match up to absolute constants and a $\log(1/\delta)$ factor on every instance. For pure DP and any $\varepsilon > 0$, replacing the Gaussian mechanism with the ℓ_2 K-norm mechanism gives $C'(\beta^2 + c_\beta^2 d^2/\varepsilon^2)$ for every β, hence $O(R(d^2/\varepsilon^2))$, matching the pure-DP lower bound up to constants (Appendix C.6).

Proof sketch. The proof constructs a two-phase algorithm and controls the optimization error and the selection bias separately. From the definition of $f(m)$ and c_β, there is a subset S with $|S| \ge (1-\beta)n$ in which every vertex appears in at most $k = |S| \cdot c_\beta$ hyperedges. Discarding at most βn examples bounds the selection bias by $(2\beta)^2$, and the contribution bound makes the ℓ_2-sensitivity of the subset mean at most $2c_\beta$, so the Gaussian mechanism contributes optimization error $O(c_\beta^2 d\log(1/\delta)/\varepsilon^2)$. Summing the two terms gives the display. See Appendix C.4 for a detailed proof. □

Remark 2 (Instance-Dependence and Computational Complexity)**.** *Theorem 6 matches the lower bound, but finding a subset S that minimizes $f(m)$ is computationally hard (Section 3.3). We therefore turn to polynomial-time approximation algorithms.*

We now analyze explicit polynomial-time contribution bounding algorithms. Their error depends on how many examples they retain at a given degree cap. The approximation guarantees in Section 3 bound the noise added to the selected mean; the selection bias is a separate term.

Theorem 7 (Error Bounds via Contribution Bounding)**.** *Let $0 < \varepsilon \le 1$ and $0 < \delta \le e^{-1}$. Let $S \ne \emptyset$ be chosen from the public structure with maximum degree at most k, and write $\theta_S^* = |S|^{-1}\sum_{e\in S} x_e$. Releasing $\hat{\theta} = \theta_S^* + Z$, where $Z \sim \mathcal{N}(0, \sigma^2 I_d)$ and $\sigma^2 = \frac{8k^2\ln(1.25/\delta)}{|S|^2\varepsilon^2}$, is (ε,δ)-fixed-graph DP. Conditional on S, $\mathbb{E}_Z\|\hat{\theta} - \theta^*\|_2^2 = d\sigma^2 + \|\theta_S^* - \theta^*\|_2^2$. Greedy guarantees $|S| \ge \mathrm{OPT}_{\mathrm{IP}}/r$. For $h = |V_{>k}| > 0$, let $A_k = \min\{r^{1/k}, h^{1/(k+1)}\}$. Keeping the largest of $O(1+\log(1/\zeta))$ independent outputs of Algorithm 1 gives $|S| = \Omega(\mathrm{OPT}_{\mathrm{IP}}/A_k)$ with probability at least $1-\zeta$. On this event, its noise contribution is*

$$d\sigma^2 = O\left(\frac{k^2 d\log(1/\delta)\, A_k^2}{\varepsilon^2 \mathrm{OPT}_{\mathrm{IP}}^2}\right).$$

If $h = 0$, use $S = E$ and the bias term vanishes. Appendix C.8.1 gives the proof and an unconditional noise bound obtained by controlling the failure event.

4.3 Polynomial-Time Algorithms in the High-Retention Regime

In the high-retention regime of Theorems 5 and 6, the goal is to retain a $(1-\beta)$ fraction of the data by deleting few hyperedges to reach a degree cap. A peeling algorithm achieves the degree cap with at most r times the minimum number of deletions.

Theorem 8 (Peeling with public cap selection)**.** *Assume $0 < \varepsilon \le 1$ and $0 < \delta \le e^{-1}$. Given a cap k, the peeling algorithm repeatedly picks any vertex of degree greater than k and deletes excess incident hyperedges. It runs in linear time, deletes at most $B_k := \sum_v (\deg_E(v) - k)^+$ hyperedges, and B_k is within a factor r of the minimum possible number of deletions. Consequently, for every β with $r\beta \le 1/2$, peeling at cap $\lceil c_\beta n\rceil$ retains at least $(1-r\beta)n$ hyperedges of maximum degree at most $\lceil c_\beta n\rceil$, and composing with the Gaussian mechanism gives error at most $C''\big(r^2\beta^2 + c_\beta^2 d\log(1/\delta)/\varepsilon^2\big)$. The full algorithm proceeds in a fixed order: it first runs peeling once at every integer cap $k \le \Delta$, producing candidates S_k and their public scores $\widehat{\mathrm{err}}(k) = 4\big((n-|S_k|)/n\big)^2 + 8\,(k/|S_k|)^2\, d\ln(1.25/\delta)/\varepsilon^2$, with the zero estimator entering as one further candidate with score 1; it selects the candidate with the smallest score using only the public structure. If the zero estimator is selected, it returns zero without accessing private contents. Otherwise, letting $\widehat{k}$ denote the selected cap, it releases the mean of $S_{\widehat{k}}$ with Gaussian noise. This is a polynomial-time (ε,δ)-fixed-graph DP algorithm whose error matches the noncomputational bound of Theorem 6 up to an r^2 factor, and therefore, for bounded r, matches $R(d\log(1/\delta)/\varepsilon^2)$ up to constants and the same logarithmic factors, with no assumptions on OPT or $|V|$. (Proof in Appendix C.7.)*

Remark 3 (On the r^2 factor)**.** *The r^2 factor is computational: the information-theoretic bound of Theorem 6 has none. It enters only through the deletion guarantee $B_k \le r \cdot \mathrm{OPT}_{\mathrm{del}}$, where $\mathrm{OPT}_{\mathrm{del}} = n - \mathrm{OPT}_{\mathrm{IP}}$ is the minimum number of deletions that reaches cap k. For graphs ($r = 2$) this minimum is computable exactly by simple b-matching [59], so the factor disappears. In general the deletion problem is a constrained set multicover: vertex v demands $(\deg_E(v) - k)^+$ deletions*

among its incident hyperedges, each hyperedge may be deleted once and covers at most r demands, and an approximation factor ρ for this deletion problem yields retention and error guarantees with ρ in place of r and ρ^2 in place of r^2. The retained-cardinality guarantee of Theorem 1 does not provide such a deletion guarantee: a multiplicative bound on retained cardinality can allow a large fraction of the examples to be discarded even when very few deletions are necessary.

4.4 Characterizing Selection Bias

The upper bounds in Section 4.2 decompose total error into privacy/estimation error and selection bias. Since contribution bounding selects examples using only the public structure (V, E) and not example contents $\{x_e\}$, bias can only arise from correlation between network structure and data distributions. We provide a complete characterization of when such bias occurs. Selection bias is zero when (a) data distributions are independent of network structure (e.g., hyperedge size, vertex degrees), or (b) the selection algorithm preserves the distribution over structural types.

When bias is nonzero, we model data as conditionally independent given structural types $t : E \to \{1, \ldots, J\}$ (e.g., hyperedge size or degree statistics), with $x_e \mid t(e) = j \sim \mathcal{D}_j$. Let q_j denote the proportion of type-j edges in the full dataset and p_j the proportion in the selected subset S. Under this data generative model, we provide quantitative bounds. In particular, for bounded losses, we bound the selection bias $\text{Bias}_{\text{sel}} \leq 4 \sum_j |p_j - q_j| \cdot \text{TV}(\mathcal{D}_j, \mathcal{D})$ in terms of how selection changes type proportions and type-conditional distribution shift. For strongly convex and β-smooth losses (here β is the smoothness constant, not the retention parameter), $\text{Bias}_{\text{sel}} \leq \frac{\beta^3 B^2 \|p-q\|_1^2}{2\lambda^2}$ in parameter space. Full details, proofs, and additional sufficient conditions appear in Appendix D. In particular, bias vanishes when structure carries no information about contents, or when selection preserves the proportions of structural types. Since selection operates on structure alone, type-preserving contribution bounding can be formulated using public structural types without affecting the privacy guarantee; exact preservation, however, can be fundamentally at odds with retention.

Exact type preservation can sharply reduce retention: on the mixed star-plus-singleton family, preserving the star fraction γ forces $|S| \leq k/\gamma$. Proposition 24 in Appendix D.1 gives the construction and a public subsampling rule for preserving type proportions.

5 Conclusion

We establish a theoretical foundation for fixed-graph differential privacy, the framework introduced by Ganesh et al. [33] for learning from multi-attribution data where training examples contain information about multiple users. Our main contributions are (1) a unified LP-guided contribution-bounding algorithm and an exactly tight analysis of greedy, drawing on and complementing the hypergraph b-matching literature; (2) information-theoretic lower bounds through network density c, together with a retention-refined profile c_β that characterizes the achievable error up to logarithmic factors; (3) a framework for selection bias with sufficient conditions for zero bias and quantitative bounds otherwise; and (4) improved $\sqrt{c/n}$-type guarantees under i.i.d. data. The approximation guarantees depend on the maximum hyperedge size r and the contribution cap, and can remain constant despite a large maximum degree Δ. At cap $k = 1$, $\Omega(r)$ hardness under $\text{NP} \not\subseteq \text{BPP}$ makes the greedy dependence on r optimal up to constants. Appendix F reports empirical results. Code is available at https://github.com/ramanarora/fixed-graph-dp.

Several questions remain open. For contribution caps $k \geq 2$, the hardness landscape remains unsettled; the $O(r^{1/k})$ upper bound and the conflicting hardness claim discussed in Appendix B.4 leave the dependence on the cap incompletely understood. For i.i.d. data, the $\sqrt{c/n}$ scaling is attainable on clustered networks, where the user-level reduction also gives improved mean estimation (Remark 8), but not on all networks (Proposition 27). Identifying the governing network parameter could close the gap between the coupling lower bound of Theorem 25 and the c_γ upper bound of Theorem 33. Exact type preservation can reduce retention by a factor of order $\gamma n/k$ (Proposition 24), motivating algorithms that balance retention with approximate preservation. A further question is whether the r^2 factor in Theorem 8 can be reduced. It is absent from the information-theoretic bound and disappears for graphs ($r = 2$), where b-matching solves the deletion problem exactly. For general hypergraphs, improving the approximation factor for the associated constrained set multicover problem would improve the error bound (Remark 3). As machine learning systems increasingly train on social data where information propagates through networks, these questions bear directly on the design of trustworthy private learning systems.

Acknowledgments and Disclosure of Funding

This work was supported, in part, by NSF BIGDATA award IIS-1838139 and NSF CAREER award IIS-1943251. The research reported in this paper was also supported by an Amazon Research Award, Fall 2025. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of Amazon.

References

[1] Martín Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 308–318, 2016. doi: 10.1145/2976749.2978318. URL https://arxiv.org/abs/1607.00133.

[2] Jayadev Acharya, Ziteng Sun, and Huanyu Zhang. Differentially private Assouad, Fano, and Le Cam. In *Proceedings of the 32nd International Conference on Algorithmic Learning Theory*, volume 132 of *Proceedings of Machine Learning Research*, pages 48–78. PMLR, 2021. URL https://proceedings.mlr.press/v132/acharya21a.html.

[3] Sushant Agarwal, Gautam Kamath, Mahbod Majid, Argyris Mouzakis, Rose Silver, and Jonathan Ullman. Private mean estimation with person-level differential privacy. In *Proceedings of the 2025 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2819–2880. Society for Industrial and Applied Mathematics, 2025. doi: 10.1137/1.9781611978322.92. URL https://arxiv.org/abs/2405.20405.

[4] Kareem Amin, Alex Kulesza, Andres Muñoz Medina, and Sergei Vassilvitskii. Bounding user contributions: A bias-variance trade-off in differential privacy. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 263–271. PMLR, 2019. URL https://proceedings.mlr.press/v97/amin19a.html.

[5] Borja Balle and Yu-Xiang Wang. Improving the Gaussian mechanism for differential privacy: Analytical calibration and optimal denoising. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 394–403. PMLR, 2018. URL https://proceedings.mlr.press/v80/balle18a.html.

[6] Borja Balle, Gilles Barthe, and Marco Gaboardi. Privacy amplification by subsampling: Tight analyses via couplings and divergences. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/3b5020bb891119b9f5130f1fea9bd773-Paper.pdf.

[7] Nikhil Bansal, Nitish Korula, Viswanath Nagarajan, and Aravind Srinivasan. Solving packing integer programs via randomized rounding with alterations. *Theory of Computing*, 8(24):533–565, 2012. doi: 10.4086/toc.2012.v008a024. URL https://theoryofcomputing.org/articles/v008a024/.

[8] Raef Bassily and Ziteng Sun. User-level private stochastic convex optimization with optimal rates. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 1838–1851. PMLR, 2023. URL https://proceedings.mlr.press/v202/bassily23a.html.

[9] Raef Bassily, Adam Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 464–473. IEEE, 2014. doi: 10.1109/FOCS.2014.56. URL https://arxiv.org/abs/1405.7085.

[10] Raef Bassily, Vitaly Feldman, Kunal Talwar, and Abhradeep Guha Thakurta. Private stochastic convex optimization with optimal rates. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/3bd8fdb090f1f5eb66a00c84dbc5ad51-Paper.pdf.

[11] Jeremiah Blocki, Avrim Blum, Anupam Datta, and Or Sheffet. Differentially private data analysis of social networks via restricted sensitivity. In *Proceedings of the 4th conference on Innovations in Theoretical Computer Science*, pages 87–96, 2013. doi: 10.1145/2422436.2422449. URL https://doi.org/10.1145/2422436.2422449.

[12] Christian Borgs, Jennifer Chayes, and Adam Smith. Private graphon estimation for sparse graphs. In *Advances in Neural Information Processing Systems*, volume 28, 2015. URL https://proceedings.neurips.cc/paper_files/paper/2015/file/7250eb93b3c18cc9daa29cf58af7a004-Paper.pdf.

[13] Christian Borgs, Jennifer Chayes, Adam Smith, and Ilias Zadik. Revealing network structure, confidentially: Improved rates for node-private graphon estimation. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 533–543, 2018. doi: 10.1109/FOCS.2018.00057. URL https://ieee-focs.org/FOCS-2018-Papers/pdfs/59f533.pdf.

[14] Mark Bun, Jonathan Ullman, and Salil Vadhan. Fingerprinting codes and the price of approximate differential privacy. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 1–10, 2014. doi: 10.1145/2591796.2591877. URL https://privacytools.seas.harvard.edu/sites/g/files/omnuum6656/files/privacytools/files/stoc2014.pdf.

[15] Clément L. Canonne, Gautam Kamath, Audra McMillan, Adam Smith, and Jonathan Ullman. The structure of optimal private tests for simple hypotheses. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 310–321. ACM, 2019. doi: 10.1145/3313276.3316336. URL https://arxiv.org/abs/1811.11148.

[16] Barun Chandra and Magnús M. Halldórsson. Greedy local improvement and weighted set packing approximation. *Journal of Algorithms*, 39(2):223–240, 2001. doi: 10.1006/jagm.2000.1155. URL https://doi.org/10.1006/jagm.2000.1155.

[17] Zachary Charles, Arun Ganesh, Ryan McKenna, H. Brendan McMahan, Nicole Mitchell, Krishna Pillutla, and Keith Rush. Learning with user-level differential privacy under fixed compute budgets. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 901–920. IEEE, 2025. doi: 10.1109/SATML64287.2025.00055. URL https://doi.org/10.1109/SATML64287.2025.00055. Earlier preprint: arXiv:2407.07737.

[18] Shixi Chen and Shuigeng Zhou. Recursive mechanism: Towards node differential privacy and unrestricted joins. In *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*, pages 653–664, 2013. doi: 10.1145/2463676.2465304. URL https://arxiv.org/abs/1304.4795.

[19] Christopher A. Choquette-Choo, H. Brendan McMahan, Keith Rush, and Abhradeep Thakurta. Multi-epoch matrix factorization mechanisms for private machine learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 5924–5963. PMLR, 2023. URL https://proceedings.mlr.press/v202/choquette-choo23a.html.

[20] Lynn Chua, Badih Ghazi, Yangsibo Huang, Pritish Kamath, Ravi Kumar, Daogao Liu, Pasin Manurangsi, Amer Sinha, and Chiyuan Zhang. Mind the privacy unit! User-level differential privacy for language model fine-tuning. In *First Conference on Language Modeling (COLM)*, 2024. URL https://openreview.net/pdf?id=Jd0bCD12DS.

[21] Vincent Cohen-Addad, Alessandro Epasto, Jason Lee, and Morteza Zadimoghaddam. Scalable contribution bounding to achieve privacy. *arXiv preprint arXiv:2507.23432*, 2025. URL https://arxiv.org/abs/2507.23432.

[22] Rachel Cummings, Vitaly Feldman, Audra McMillan, and Kunal Talwar. Mean estimation with user-level privacy under data heterogeneity. In *Advances in Neural Information Processing Systems*, volume 35, pages 29139–29151. Curran Associates, Inc., 2022. doi: 10.52202/068431-2113. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/bbba680acd2826f23928c6675a19f0e7-Abstract-Conference.html.

[23] Marek Cygan. Improved approximation for 3-dimensional matching via bounded pathwidth local search. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 509–518. IEEE, 2013. doi: 10.1109/FOCS.2013.61. URL https://arxiv.org/abs/1304.1424.

[24] Cynthia Dwork and Aaron Roth. *The Algorithmic Foundations of Differential Privacy*. Now Publishers, Inc., 2014. doi: 10.1561/0400000042. URL https://www.nowpublishers.com/articles/foundations-and-trends-in-theoretical-computer-science/TCS-042.

[25] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography*, volume 3876 of *Lecture Notes in Computer Science*, pages 265–284. Springer, 2006. doi: 10.1007/11681878_14. URL https://doi.org/10.1007/11681878_14.

[26] Jack Edmonds. Paths, trees, and flowers. *Canadian Journal of Mathematics*, 17:449–467, 1965. doi: 10.4153/CJM-1965-045-4. URL https://doi.org/10.4153/CJM-1965-045-4.

[27] Mourad El Ouali and Gerold Jäger. The b-matching problem in hypergraphs: Hardness and approximability. In *Combinatorial Optimization and Applications (COCOA 2012)*, volume 7402 of *Lecture Notes in Computer Science*, pages 200–211. Springer, 2012. doi: 10.1007/978-3-642-31770-5_18. URL https://link.springer.com/chapter/10.1007/978-3-642-31770-5_18.

[28] Mourad El Ouali, Antje Fretwurst, and Anand Srivastav. Inapproximability of b-matching in k-uniform hypergraphs. In *WALCOM: Algorithms and Computation*, volume 6552 of *Lecture Notes in Computer Science*, pages 57–69. Springer, 2011. doi: 10.1007/978-3-642-19094-0_8. URL https://link.springer.com/chapter/10.1007/978-3-642-19094-0_8.

[29] Alessandro Epasto, Mohammad Mahdian, Jieming Mao, Vahab Mirrokni, and Lijie Ren. Smoothly bounding user contributions in differential privacy. In *Advances in Neural Information Processing Systems*, volume 33, pages 13999–14010, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/a0dc078ca0d99b5ebb465a9f1cad54ba-Paper.pdf.

[30] Juanru Fang, Wei Dong, and Ke Yi. Shifted inverse: A general mechanism for monotonic functions under user differential privacy. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 1009–1022, 2022. doi: 10.1145/3548606.3560567. URL https://juanru-fang.github.io/ShiftedInverse.pdf.

[31] Zoltán Füredi, Jeff Kahn, and Paul D Seymour. On the fractional matching polytope of a hypergraph. *Combinatorica*, 13(2):167–180, 1993. doi: 10.1007/BF01303202. URL https://www.renyi.hu/~furedi/PUBS3/furedi_121_kahn_seymour_fractional_matching_polytope.pdf.

[32] Martin Fürer and Huiwen Yu. Approximating the k-set packing problem by local improvements. In *Combinatorial Optimization*, volume 8596 of *Lecture Notes in Computer Science*, pages 408–420. Springer, 2014. doi: 10.1007/978-3-319-09174-7_35. URL https://arxiv.org/abs/1307.2262.

[33] Arun Ganesh, Ryan McKenna, Hugh Brendan McMahan, Adam Smith, and Fan Wu. It's my data too: Private ML for datasets with multi-user training examples. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 18189–18205. PMLR, 2025. URL https://proceedings.mlr.press/v267/ganesh25a.html.

[34] Badih Ghazi, Ravi Kumar, and Pasin Manurangsi. User-level differentially private learning via correlated sampling. In *Advances in Neural Information Processing Systems*, volume 34, pages 20172–20184. Curran Associates, Inc., 2021. URL https://papers.neurips.cc/paper_files/paper/2021/hash/a89cf525e1d9f04d16ce31165e139a4b-Abstract.html.

[35] Badih Ghazi, Pritish Kamath, Ravi Kumar, Pasin Manurangsi, Raghu Meka, and Chiyuan Zhang. User-level differential privacy with few examples per user. In *Advances in Neural Information Processing Systems*, volume 36, pages 19263–19290, 2023. doi: 10.52202/075280-0846. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/3d57795f0e263aa69577f1bbceade46b-Paper-Conference.pdf.

[36] Moritz Hardt and Kunal Talwar. On the geometry of differential privacy. In *Proceedings of the Forty-Second ACM Symposium on Theory of Computing*, pages 705–714. ACM, 2010. doi: 10.1145/1806689.1806786. URL https://www.microsoft.com/en-us/research/publication/on-the-geometry-of-differential-privacy/.

[37] Michael Hay, Chao Li, Gerome Miklau, and David Jensen. Accurate estimation of the degree distribution of private networks. In *2009 Ninth IEEE International Conference on Data Mining*, pages 169–178. IEEE, 2009. doi: 10.1109/ICDM.2009.11. URL https://www.cs.colgate.edu/~mhay/assets/publications/hay2009accurate.pdf.

[38] Elad Hazan, Shmuel Safra, and Oded Schwartz. On the complexity of approximating k-set packing. *Computational Complexity*, 15(1):20–39, 2006. doi: 10.1007/s00037-006-0205-6. URL https://link.springer.com/article/10.1007/s00037-006-0205-6.

[39] Palak Jain, Adam Smith, and Connor Wagaman. Time-aware projections: Truly node-private graph statistics under continual observation. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 127–145. IEEE, 2024. doi: 10.1109/SP54263.2024.00196. URL https://arxiv.org/abs/2403.04630.

[40] Prateek Jain, Keith Rush, Adam Smith, Shuang Song, and Abhradeep Thakurta. Differentially private model personalization. In *Advances in Neural Information Processing Systems*, volume 34, pages 29723–29735, 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/f8580959e35cb0934479bb007fb241c2-Paper.pdf.

[41] Peter Kairouz, Brendan McMahan, Shuang Song, Om Thakkar, Abhradeep Thakurta, and Zheng Xu. Practical and private (deep) learning without sampling or shuffling. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 5213–5225. PMLR, 2021. URL https://proceedings.mlr.press/v139/kairouz21b.html.

[42] Peter Kairouz, H. Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, Rafael G. L. D'Oliveira, Hubert Eichner, Salim El Rouayheb, David Evans, Josh Gardner, Zachary Garrett, Adrià Gascón, Badih Ghazi, Phillip B. Gibbons, Marco Gruteser, Zaid Harchaoui, Chaoyang He, Lie He, Zhouyuan Huo, Ben Hutchinson, Justin Hsu, Martin Jaggi, Tara Javidi, Gauri Joshi, Mikhail Khodak, Jakub Konečný, Aleksandra Korolova, Farinaz Koushanfar, Sanmi Koyejo, Tancrède Lepoint, Yang Liu, Prateek Mittal, Mehryar Mohri, Richard Nock, Ayfer Özgür, Rasmus Pagh, Hang Qi, Daniel Ramage, Ramesh Raskar, Mariana Raykova, Dawn Song, Weikang Song, Sebastian U. Stich, Ziteng Sun, Ananda Theertha Suresh, Florian Tramèr, Praneeth Vepakomma, Jianyu Wang, Li Xiong, Zheng Xu, Qiang Yang, Felix X. Yu, Han Yu, and Sen Zhao. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1–2):1–210, 2021. doi: 10.1561/2200000083. URL https://arxiv.org/abs/1912.04977.

[43] Iden Kalemaj, Sofya Raskhodnikova, Adam Smith, and Charalampos E. Tsourakakis. Node-differentially private estimation of the number of connected components. In *Proceedings of the 42nd ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems (PODS)*, pages 183–194, 2023. doi: 10.1145/3584372.3588671. URL https://arxiv.org/abs/2304.05890.

[44] Gautam Kamath and Jonathan Ullman. A primer on private statistics. *arXiv preprint arXiv:2005.00010*, 2020. URL https://arxiv.org/abs/2005.00010.

[45] Viggo Kann. Maximum bounded 3-dimensional matching is MAX SNP-complete. *Information Processing Letters*, 37(1):27–35, 1991. doi: 10.1016/0020-0190(91)90246-E. URL https://www.csc.kth.se/~viggo/papers/ftp/max3dm.pdf.

[46] Shiva Prasad Kasiviswanathan, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. Analyzing graphs with node differential privacy. In *Theory of Cryptography*, volume 7785 of *Lecture Notes in Computer Science*, pages 457–476. Springer, 2013. doi: 10.1007/978-3-642-36594-2_26. URL https://link.springer.com/chapter/10.1007/978-3-642-36594-2_26.

[47] Antti Koskela, Joonas Jälkö, Lukas Prediger, and Antti Honkela. Tight differential privacy for discrete-valued mechanisms and for the subsampled Gaussian mechanism using FFT. In *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 3358–3366. PMLR, 2021. URL https://proceedings.mlr.press/v130/koskela21a.html.

[48] Piotr Krysta. Greedy approximation via duality for packing, combinatorial auctions and routing. In *Mathematical Foundations of Computer Science 2005*, volume 3618 of *Lecture Notes in Computer Science*, pages 615–627. Springer, 2005. doi: 10.1007/11549345_53. URL https://link.springer.com/chapter/10.1007/11549345_53.

[49] Euiwoong Lee, Ola Svensson, and Theophile Thiery. Asymptotically optimal hardness for k-set packing and k-matroid intersection. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, pages 54–61. ACM, 2025. doi: 10.1145/3717823.3718142. URL https://arxiv.org/abs/2409.17831.

[50] Daniel Levy, Ziteng Sun, Kareem Amin, Satyen Kale, Alex Kulesza, Mehryar Mohri, and Ananda Theertha Suresh. Learning with user-level privacy. In *Advances in Neural Information Processing Systems*, volume 34, pages 12466–12479. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/67e235e7f2fa8800d8375409b566e6b6-Paper.pdf.

[51] Xuechen Li, Florian Tramèr, Percy Liang, and Tatsunori Hashimoto. Large language models can be strong differentially private learners. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/attachment?id=bVuP3ltATMz&name=pdf.

[52] Yuhan Liu, Ananda Theertha Suresh, Felix Yu, Sanjiv Kumar, and Michael Riley. Learning discrete distributions: User vs item-level privacy. In *Advances in Neural Information Processing Systems*, volume 33, pages 20965–20976, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/f06edc8ab534b2c7ecbd4c2051d9cb1e-Paper.pdf.

[53] Yuhan Liu, Ananda Theertha Suresh, Wennan Zhu, Peter Kairouz, and Marco Gruteser. Algorithms for bounding contribution for histogram estimation under user-level privacy. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 21969–21996. PMLR, 2023. URL https://proceedings.mlr.press/v202/liu23ae.html.

[54] H. Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. Learning differentially private recurrent language models. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/pdf?id=BJ0hF1Z0b.

[55] Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275. IEEE, 2017. doi: 10.1109/CSF.2017.11. URL https://research.google/pubs/r%C3%A9nyi-differential-privacy/.

[56] Ojas Parekh and David Pritchard. Generalized hypergraph matching via iterated packing and local ratio. In *Approximation and Online Algorithms (WAOA 2014)*, volume 8952 of *Lecture Notes in Computer Science*, pages 207–223. Springer, 2015. doi: 10.1007/978-3-319-18263-6_18. URL https://arxiv.org/abs/1604.00322.

[57] Natalia Ponomareva, Hussein Hazimeh, Alex Kurakin, Zheng Xu, Carson Denison, H. Brendan McMahan, Sergei Vassilvitskii, Steve Chien, and Abhradeep Guha Thakurta. How to DP-fy ML: A practical guide to machine learning with differential privacy. *Journal of Artificial Intelligence Research*, 77:1113–1201, 2023. doi: 10.1613/jair.1.14649. URL https://www.jair.org/index.php/jair/article/view/14649.

[58] Sofya Raskhodnikova and Adam Smith. Lipschitz extensions for node-private graph statistics and the generalized exponential mechanism. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 495–504. IEEE, 2016. doi: 10.1109/FOCS.2016.60. URL https://cs-people.bu.edu/sofya/pubs/Lipschitz-deg-ext-FOCS16.pdf.

[59] Alexander Schrijver. *Combinatorial Optimization: Polyhedra and Efficiency*. Springer, 2003. URL https://link.springer.com/book/9783540443896.

[60] Adam Sealfon and Jonathan Ullman. Efficiently estimating Erdős–Rényi graphs with node differential privacy. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/955cb567b6e38f4c6b3f28cc857fc38c-Paper.pdf.

[61] Shuang Song, Kamalika Chaudhuri, and Anand D. Sarwate. Stochastic gradient descent with differentially private updates. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 245–248. IEEE, 2013. doi: 10.1109/GlobalSIP.2013.6736861. URL https://adsarwate.github.io/assets/pdf/papers/SongCS13sgd.pdf.

[62] Anand Srivastav and Peter Stangier. Weighted fractional and integral k-matching in hypergraphs. *Discrete Applied Mathematics*, 57(2–3):255–269, 1995. doi: 10.1016/0166-218X(94)00107-O. URL https://kups.ub.uni-koeln.de/54675/.

[63] Thomas Steinke and Jonathan Ullman. Between pure and approximate differential privacy. *Journal of Privacy and Confidentiality*, 7(2):3–22, 2016. doi: 10.29012/jpc.v7i2.648. URL https://journalprivacyconfidentiality.org/index.php/jpc/article/view/648.

[64] Theophile Thiery and Justin Ward. An improved approximation for maximum weighted k-set packing. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1138–1162. Society for Industrial and Applied Mathematics, 2023. doi: 10.1137/1.9781611977554.ch42. URL https://arxiv.org/abs/2301.07537.

[65] Florian Tramèr and Dan Boneh. Differentially private learning needs better features (or much more data). In *International Conference on Learning Representations*, 2021. URL https://openreview.net/pdf?id=YTWGvpFOQD-.

[66] Di Wang, Changyou Chen, and Jinhui Xu. Differentially private empirical risk minimization with non-convex loss functions. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6526–6535. PMLR, 2019. URL https://proceedings.mlr.press/v97/wang19c.html.

Supplementary to "Optimal Algorithms for Fixed-Graph Multi-Attribution Privacy"

The appendices are organized as follows. Appendix A expands the related-work discussion in Section 1.1. Appendix B provides the proofs for the contribution-bounding results in Section 3, together with further results on packing and structured networks. Appendix C gives the proofs and supporting calculations for the mean-estimation bounds in Section 4. Appendix D develops the selection-bias analysis outlined in Section 4.4. Appendix E complements the worst-case analysis with hypothesis-testing results for i.i.d. examples and their proofs. Appendix F presents experiments illustrating the approximation guarantees and privacy bounds in Sections 3 and 4.

A Detailed Related Work

Our work builds on several research threads at the intersection of differential privacy, approximation algorithms, and network analysis.

Differential privacy for machine learning. The foundational result of Bassily et al. [9, 10] established efficient algorithms and tight error bounds of $\tilde{O}(d/(\varepsilon^2 n^2))$ for differentially private empirical risk minimization, with extensions to non-convex losses [66]. The DP-SGD algorithm [1, 61] and its variants [41, 19] have become the standard approach for private deep learning, with recent advances extending these techniques to large language models [65, 51]. The extension from example-level to user-level privacy, where protection is provided for all data contributed by each user, has gained significant attention [54, 52, 50, 34, 22, 8]. Charles et al. [17] study user-level DP-SGD fine-tuning with group-level accounting, in which a user contributing up to Δ examples pays the worst-case group-privacy cost; Chua et al. [20] and Ghazi et al. [35] likewise treat user-level privacy with heterogeneous or few examples per user, and Agarwal et al. [3] give optimal person-level mean estimation. The choice of the contribution bound itself is studied by Liu et al. [53], and Cohen-Addad et al. [21] give a distributed contribution-bounding algorithm on the same user–record hypergraph as ours. Much of this literature assumes each example is attributed to a single user. Ganesh et al. [33] introduced fixed-graph differential privacy to handle multi-attribution data and proposed greedy contribution bounding algorithms, but left theoretical guarantees and fundamental limits as open questions. We analyze the approximation guarantees of their algorithms and establish that, for worst-case data on networks with $c = \Theta(\Delta/n)$, the Δ^2 overhead is information-theoretically necessary through our network density parameter c.

Contribution bounding and selection bias. The bias introduced by bounding contributions has also been studied in the single-attribution setting. Amin et al. [4] analyze the tradeoff between the bias from limiting contributions and the noise required for privacy, while Epasto et al. [29] develop alternatives to hard contribution caps. In the multi-attribution setting, selection depends on the public hypergraph and can change the proportions of examples with different structural characteristics. Our type-based analysis describes how these changes interact with differences in the data distributions. It gives sufficient conditions for zero systematic bias and quantitative bounds when those conditions do not hold.

Approximation algorithms for set systems. Contribution bounding is maximum unweighted hypergraph b-matching and a 0–1 packing integer program with column sparsity r and capacities k [62, 48, 7]. The established rounding guarantees are $O(r^{1/k})$ and $O(h^{1/(k+1)})$, where $h = |V_{>k}|$ is the number of degree constraints that are not automatically satisfied. Our analysis shows that one LP-guided greedy traversal attains both guarantees. At $k = 1$, the problem becomes r-Set Packing. The $\Omega(r/\log r)$ hardness of Hazan et al. [38] was strengthened to $\Omega(r)$ by Lee et al. [49], under $\mathrm{NP} \not\subseteq \mathrm{BPP}$. Local search gives a $(r + 1 + \eta)/3$ approximation for any fixed $\eta > 0$ [23, 32]. For bounded r, arbitrary-order greedy already gives a constant approximation independently of Δ; the LP analysis additionally captures how the guarantee improves with the contribution cap.

Privacy lower bounds and accounting. The fingerprinting framework of Bun et al. [14] has become the standard technique for proving lower bounds in approximate differential privacy. Their approach constructs hard instances where any accurate mechanism must distinguish between datasets differing in one individual, yielding fundamental sample complexity limits. Extensions include Steinke and Ullman [63] for pure versus approximate DP gaps and Kamath and Ullman [44] for a comprehensive survey. Our lower bounds take a different approach. Rather than constructing fingerprinting codes, we establish bounds through a reduction from known results for 1-way marginals estimation [9], with our network density parameter c characterizing how network structure amplifies privacy loss under group privacy composition. For privacy accounting, Mironov [55] introduced Rényi differential privacy for tighter composition, Balle and Wang [5] provided optimal Gaussian mechanism calibration, and Koskela et al. [47] developed privacy loss distributions for precise accounting. We leverage the group privacy accounting framework of Charles et al. [17] in our upper bounds, applying it in the multi-attribution setting where contribution bounding ensures each user appears in at most k selected examples.

Privacy amplification and network structure. Privacy amplification, where subsampling training data provides better privacy than naive composition, has been crucial for practical private learning. Balle et al. [6] provided tight amplification bounds showing that sampling a fraction p of examples amplifies privacy by roughly a factor of p. In the multi-attribution setting, amplification is more subtle because users' contributions overlap across examples. Selecting a random subset of examples does not yield independent samples per user. Ganesh et al. [33] use amplification for DP-SGD but note that careful analysis is required when combined with contribution bounding. Our work complements this by showing that the network density parameter c precisely captures how network structure limits the effectiveness of random sampling. Star networks with $c = 1$ admit no amplification, while matching graphs with $c = 1/n$ allow substantial amplification.

Graph differential privacy. Node-level differential privacy for graph statistics has been studied extensively [46, 11, 37, 58], with the goal of protecting against node insertion or deletion in the graph itself. Fang et al. [30] provide mechanisms for monotone functions under user-level differential privacy in relational settings where a user's tuples can be joined into shared records; node-private graphon and parameter estimation [12, 13, 60] and node-private statistics under continual observation [39] likewise concern private graph statistics. In our setting, the attribution hypergraph is public and privacy protects the contents of its examples. Thus, one user's privacy must account for every example attributed to that user, including examples shared with others. The public structure can be used to choose a subset before any private contents are accessed. This distinction changes both the neighboring datasets and the information available to the algorithm. The federated learning literature has explored personalized privacy [40], and recent work on collaborative learning with privacy [57, 42] has begun to explore how network topology affects optimization and privacy. Additional node-private estimators include Chen and Zhou [18], Kalemaj et al. [43].

B Detailed Proofs for Contribution Bounding Algorithms

This appendix gives the proofs for the approximation algorithms and hardness results in Section 3, followed by packing connections and bounds for structured networks.

B.1 Proof of LP-Guided Greedy Approximation

We prove Theorem 1 by analyzing two prefixes of an independent sampling sequence and coupling that sequence to the weighted order in Algorithm 1. Throughout, $k \geq 1$ is an integer. If $h = |V_{>k}| = 0$, the entire edge set is feasible and the algorithm returns it, including when $E = \emptyset$. Hence assume $h \geq 1$, so that $E \neq \emptyset$, $r \geq 1$, and $F := \mathrm{OPT}_{\mathrm{LP}} \geq 1$.

Let y^* be the optimal LP vector used by the algorithm and set $p(e) = y^*_e/F$. Consider an infinite sequence of independent draws from p. Starting from the empty set, accept a draw if it is not already selected and adding it preserves the contribution bound. Let S_t be the selected set after t draws. We first relate these sets to the implemented algorithm.

Lemma 9 (First-appearance coupling)**.** *The order of first appearances of positive-weight edges in the independent sequence has the same distribution as the positive-weight order in Algorithm 1. Under this coupling, the algorithm's final output contains S_t for every finite t.*

Proof. Suppose the set of edges seen so far is B, and consider an unseen edge e with $y_e^* > 0$. The next draw outside B equals e with probability

$$\sum_{j=0}^{\infty} \left(\sum_{f \in B} p(f) \right)^j p(e) = \frac{p(e)}{1 - \sum_{f \in B} p(f)} = \frac{y_e^*}{\sum_{f \notin B} y_f^*}.$$

These are exactly the successive probabilities of weighted sampling without replacement. They also describe the order of independent exponential priorities with rates y_e^*: the smallest priority has probability proportional to its rate, and the memoryless property gives the same rule after each priority is removed.

A repeated appearance never changes the greedy set. If the edge was accepted, it is already present. If it was rejected, some incident vertex was saturated at its first appearance and remains saturated thereafter. Deleting repeated appearances therefore leaves the accepted sequence unchanged. Every positive-weight edge appears eventually with probability one, since there are finitely many such edges. Thus the complete positive-weight traversal has the same final selected set as the infinite independent sequence. It contains each S_t, and processing the zero-weight edges can only add edges. □

Define the two analytical prefix lengths

$$T_r = \max\left\{1, \left\lfloor \frac{F}{4er^{1/k}} \right\rfloor \right\}, \qquad T_h = \max\left\{1, \left\lfloor \frac{F}{e(2h)^{1/(k+1)}} \right\rfloor \right\}.$$

The algorithm does not compute these quantities. They allow the same traversal to be analyzed in terms of either the maximum edge size or the number of potentially binding vertex constraints. We use the elementary inequality $\max\{1, \lfloor x \rfloor\} \geq x/2$ for $x > 0$.

The prefix controlled by the maximum edge size. Suppose first that $F/(4er^{1/k}) \geq 1$. Fix a position $i \leq T_r$ and condition on the current draw being e. The preceding draws remain independent. Since $y_e^* \leq 1$, the chance that e has appeared previously is at most

$$(i-1)p(e) \leq \frac{T_r}{F}.$$

For $v \in e$, let $Z_{v,i}$ be the number of preceding draws incident to v, counting repetitions. The LP constraint gives

$$\mathbb{E}[Z_{v,i} \mid e_i = e] = (i-1) \sum_{f : v \in f} \frac{y_f^*}{F} \leq \frac{kT_r}{F}.$$

If v is saturated before position i, at least k distinct incident edges have been accepted, and hence $Z_{v,i} \geq k$. The Chernoff bound $\Pr[Z \geq t] \leq (e\mathbb{E}Z/t)^t$ for a sum of independent Bernoulli variables and $t \geq \mathbb{E}Z$ yields

$$\Pr[v \text{ is saturated before position } i \mid e_i = e] \leq \left(\frac{eT_r}{F} \right)^k.$$

A union bound over the at most r vertices of e therefore gives

$$\Pr[e_i \text{ is not added} \mid e_i = e] \leq \frac{T_r}{F} + r \left(\frac{eT_r}{F} \right)^k \leq \frac{1}{4er^{1/k}} + 4^{-k} < \frac{1}{2}.$$

This bound is uniform over the current edge and position, so $\mathbb{E}|S_{T_r}| \geq T_r/2$. If instead $F/(4er^{1/k}) < 1$, then $T_r = 1$ and the first edge is accepted, which gives the same conclusion. Consequently,

$$\mathbb{E}|S_{T_r}| \geq \frac{T_r}{2} \geq \frac{F}{16er^{1/k}}. \tag{7}$$

For a probability bound, put $q = \Pr[|S_{T_r}| \geq T_r/4]$. Because $0 \leq |S_{T_r}| \leq T_r$,

$$\frac{T_r}{2} \leq \mathbb{E}|S_{T_r}| \leq \frac{T_r}{4}(1-q) + T_r q,$$

so $q \geq 1/3$. The floor inequality then gives

$$\Pr\left[|S_{T_r}| \geq \frac{F}{32er^{1/k}} \right] \geq \frac{1}{3}. \tag{8}$$

The prefix controlled by potentially binding constraints. Suppose first that $F/[e(2h)^{1/(k+1)}] \geq 1$. For each $v \in V_{>k}$, let X_v count the draws incident to v among the first T_h draws, including repetitions. The LP constraint gives $\mathbb{E}X_v \leq kT_h/F$. Since $T_h \leq F/[e(2h)^{1/(k+1)}]$, the Chernoff bound implies

$$\Pr[X_v > k] \leq \left(\frac{e\mathbb{E}X_v}{k+1}\right)^{k+1} \leq \left(\frac{eT_h}{F}\right)^{k+1} \leq \frac{1}{2h}.$$

Thus the event $\mathcal{E} = \{X_v \leq k \text{ for all } v \in V_{>k}\}$ has probability at least $1/2$.

Let N be the number of distinct edges in this prefix. Since $T_h/F \leq 1$ and $0 \leq y_e^* \leq 1$, the inequality $1 - e^{-x} \geq (1 - 1/e)x$ for $0 \leq x \leq 1$ yields

$$\begin{aligned}\mathbb{E}N &= \sum_{e \in E} \left(1 - (1 - p(e))^{T_h}\right) \\ &\geq \sum_{e \in E} \left(1 - \exp\left(-\frac{y_e^* T_h}{F}\right)\right) \geq \left(1 - \frac{1}{e}\right) T_h.\end{aligned}$$

Set $a = (2e-5)/(2e) > 0$ and $q = \Pr[N \geq aT_h]$. Using $0 \leq N \leq T_h$ gives

$$\left(1 - \frac{1}{e}\right) T_h \leq \mathbb{E}N \leq aT_h(1-q) + T_h q, \qquad q \geq \frac{(1-1/e) - a}{1-a} = \frac{3}{5}.$$

The two events need not be independent. A union bound suffices:

$$\Pr[\mathcal{E} \text{ and } N \geq aT_h] \geq \frac{1}{2} + \frac{3}{5} - 1 = \frac{1}{10}.$$

On $\mathcal{E}$, the set of distinct sampled edges is feasible. Vertices in $V_{>k}$ have at most k incident draws, and every other vertex has at most k incident edges in the entire input. The greedy rule therefore accepts every distinct edge in the prefix, so $|S_{T_h}| = N$ on $\mathcal{E}$.

If $F/[e(2h)^{1/(k+1)}] < 1$, then $T_h = 1$ and the first edge is accepted. Since $a < 1$, the same bound $\Pr[|S_{T_h}| \geq aT_h] \geq 1/10$ holds directly. In both cases, the floor inequality gives

$$\Pr\left[|S_{T_h}| \geq \frac{(2e-5)F}{4e^2(2h)^{1/(k+1)}}\right] \geq \frac{1}{10}, \tag{9}$$

$$\mathbb{E}|S_{T_h}| \geq \frac{(2e-5)F}{40e^2(2h)^{1/(k+1)}}. \tag{10}$$

Combining the guarantees. By Lemma 9, the final output S contains both prefix outputs under a single coupling. Equations (7) and (10) imply

$$\mathbb{E}|S| \geq \max\left\{\frac{F}{16er^{1/k}}, \frac{(2e-5)F}{40e^2(2h)^{1/(k+1)}}\right\}.$$

For the probability statement, let

$$c_0 = \min\left\{\frac{1}{32e}, \frac{2e-5}{4e^2\sqrt{2}}\right\}.$$

Since $2^{1/(k+1)} \leq \sqrt{2}$, Equations (8) and (9) show that the bound with denominator $A_k = \min\{r^{1/k}, h^{1/(k+1)}\}$ holds with probability at least $1/10$. We use whichever of the two deterministic parameter values is smaller; simultaneous success of the two prefix events is unnecessary. This proves the randomized approximation guarantees.

Deterministic guarantee and integral solutions. Algorithm 1 considers every edge and only rejects an edge when some incident vertex is saturated. Its final set is therefore maximal, and the charging argument in the proof of Theorem 2 applies to its order, giving $|S| \geq \mathrm{OPT}_{\mathrm{IP}}/r$. If y^* is integral, its positive support is feasible and has size F, so the first phase accepts all of it. No feasible set has more than F edges, and hence the returned set is optimal. This statement concerns the particular optimal vector returned by the LP solver; the existence of an integral optimum does not require every optimal LP vector to be integral.

Repetition and implementation. Conditioned on the chosen optimal vector, independent weighted orders give independent runs with the same success bound. Keeping the largest output of R runs gives failure probability at most $(9/10)^R$; $R = \lceil \log(1/\zeta)/\log(10/9) \rceil$ suffices for failure probability at most ζ. The LP is solved only once. Generating and sorting the exponential priorities takes $O(|E| \log |E|)$ time, and the feasibility checks and counter updates take $O(\sum_{e \in E} |e|)$ time per traversal. Different examples with the same attribution set are treated as distinct edges throughout.

B.2 Proof of Greedy Algorithm Approximation

We provide a complete proof of Theorem 2, establishing the approximation ratio of the greedy contribution bounding algorithm.

Theorem (Theorem 2, restated)**.** *For any hypergraph (V, E) with maximum hyperedge size $r = \max_{e \in E} |e|$ and budget k, the greedy algorithm (Algorithm 2) produces a solution S satisfying:*

$$|S| \geq \frac{\text{OPT}_{\text{IP}}}{r}.$$

Proof. Let S denote the solution returned by the greedy algorithm and let $S^* \subseteq E$ be an optimal solution with $|S^*| = \text{OPT}_{\text{IP}}$. Both S and S^* satisfy the contribution bound: every vertex v appears in at most k hyperedges from each set. We establish the approximation ratio through a counting argument. For each vertex $v \in V$, define $d_S(v) = |\{e \in S : v \in e\}|$ and $d_{S^*}(v) = |\{e \in S^* : v \in e\}|$. By feasibility, $d_S(v) \leq k$ and $d_{S^*}(v) \leq k$ for all v.

Lemma 10. *If an edge $e^* \in S^*$ is rejected by the greedy algorithm (i.e., $e^* \notin S$), then there exists at least one vertex $v \in e^*$ such that $d_S(v) = k$.*

Proof of Lemma 10. Edge e^* is rejected if and only if $\max_{v \in e^*} c_v \geq k$ when e^* is considered. Since counters never exceed k and never decrease, this vertex v has $d_S(v) = k$ in the final solution. □

Let $V_k = \{v \in V : d_S(v) = k\}$ be the set of saturated vertices. We charge every edge of S^* exactly once, as follows.

The charging scheme. If $e^* \in S^* \setminus S$, then by Lemma 10 it contains a saturated vertex; charge e^* to one such witness $v \in e^* \cap V_k$. If $e \in S^* \cap S$ and e contains a saturated vertex, charge e to one such vertex $v \in e \cap V_k$. If $e \in S^* \cap S$ contains no saturated vertex, let e charge itself.

Distributing the charges at a saturated vertex. Fix $v \in V_k$. Every edge charged to v belongs to S^* and contains v, so at most $d_{S^*}(v) \leq k$ edges are charged to v. Exactly $d_S(v) = k$ selected edges contain v. The charges at v can therefore be assigned injectively to distinct selected edges containing v: each edge charged to v is passed to its own selected edge through v.

Counting charges at a selected edge. Fix $f \in S$. Through each saturated vertex $v \in f$, the edge f receives at most one passed charge, by injectivity at v. If f has no saturated vertex, the only charge it can receive is its own self-charge, which occurs only when $f \in S^*$, for one charge in total. In either case the number of charges received by f is at most $\max\{1, |f \cap V_k|\} \leq |f| \leq r$.

Summing over S, since every edge of S^* is charged exactly once,

$$|S^*| \;=\; \text{total charge} \;\leq\; \sum_{f \in S} \max\{1, |f \cap V_k|\} \;\leq\; \sum_{f \in S} |f| \;\leq\; r|S|,$$

which gives $|S| \geq |S^*|/r = \text{OPT}_{\text{IP}}/r$.

Two byproducts of the same computation are used for the per-instance bounds of Appendix B.5. First, the middle expression yields $|S^*| \leq \sum_{f \in S} |f| = \bar{r}_S |S|$, where $\bar{r}_S$ is the average size of the selected edges. Second, the coarser split $|S^*| = |S^* \cap S| + |S^* \setminus S|$ together with $|S^* \setminus S| \leq \sum_{v \in V_k} d_{S^*}(v) \leq k|V_k|$ gives $|S^*| \leq |S| + k|V_k|$. □

Remark 4 (The Role of r vs. Δ)**.** *The approximation ratio depends on r (maximum hyperedge size),* not Δ *(maximum vertex degree). The charging argument routes every optimal edge through a saturated vertex to a distinct selected edge, so each selected edge f absorbs at most $|f| \leq r$ charges; the count runs over hyperedges rather than vertices. While Δ bounds how many edges* contain *a*

given vertex, this quantity does not appear in the charging sum. Consequently, for practical networks with $r = O(1)$ (e.g., emails with $r \approx 5$ or papers with $r \approx 4$), the greedy algorithm achieves $O(1)$-approximation regardless of Δ.

B.3 Proof of Hardness Results

At cap $k = 1$, contribution bounding is exactly set packing. The following proof records the resulting hardness in terms of the maximum hyperedge size r.

Theorem (Theorem 3, restated)**.** *For contribution bounding with maximum hyperedge size r and budget $k = 1$:*

(a) Unless $\mathrm{NP} \subseteq \mathrm{BPP}$, no polynomial-time algorithm achieves approximation factor $o(r)$.
(b) When $r = 2$, the problem is polynomial-time solvable.
(c) For every fixed $r \geq 3$, the problem is APX-hard and admits no polynomial-time approximation scheme unless $\mathrm{P} = \mathrm{NP}$.

Proof. Contribution bounding with budget $k = 1$ asks for a maximum collection of pairwise disjoint hyperedges. This is exactly r-Set Packing when each hyperedge has size at most r.

Part (a): For every constant $\eta > 0$ and all sufficiently large r, Lee et al. [49] prove that r-Dimensional Matching is hard to approximate within $r/(12 + \eta)$ unless $\mathrm{NP} \subseteq \mathrm{BPP}$. Since r-Dimensional Matching is a special case of r-Set Packing, their theorem yields the claimed $\Omega(r)$ hardness for contribution bounding at cap one. It strengthens the $\Omega(r/\log r)$ bound of Hazan et al. [38], with the stated randomized complexity assumption.

Part (b): If every hyperedge has two vertices, contribution bounding at cap one is Maximum Matching, solvable in polynomial time by Edmonds' blossom algorithm [26]. Singleton hyperedges can also be included: attach a distinct auxiliary leaf to each singleton to obtain an ordinary matching instance with the same feasible subsets and objective values.

Part (c): For every fixed $r \geq 3$, the problem contains Maximum 3-Set Packing, which is APX-hard [45]. Thus no polynomial-time approximation scheme exists unless $\mathrm{P} = \mathrm{NP}$. The greedy r-approximation is constant when r is fixed. □

The cap-one lower bound and the greedy upper bound have the same linear dependence on r, up to constants. This does not establish matching hardness for larger caps, where the LP-guided algorithm has a stronger dependence on r. Table 1 summarizes the relevant guarantees. The construction below concerns the worst-case behavior of a particular greedy ordering; its extension to larger caps is separate from complexity-theoretic hardness.

Regime	**Upper bound**	**Hardness**	**Gap**
$r = 2$ (graphs), any k	Exact b-matching	—	Solved
Fixed $r \geq 3$, $k = 1$	$O(r) = O(1)$	APX-hard	Constant
General r, $k = 1$	$(r + 1 + \eta)/3$ [23]	$\Omega(r)$ [49]	Constant
General r, integer $k \geq 2$	$O(r^{1/k})$ [7]; also $O(h^{1/(k+1)})$	Matching hardness unresolved	Open

Table 1: Approximation guarantees and hardness for contribution bounding. The linear hardness at $k = 1$ assumes $\mathrm{NP} \not\subseteq \mathrm{BPP}$; the APX-hardness consequence assumes $\mathrm{P} \neq \mathrm{NP}$. Here $\eta > 0$ is fixed and $h = |V_{>k}|$; if $h = 0$, the full edge set is feasible. Appendix B.4 discusses the conflicting larger-cap hardness claim.

B.3.1 Proof of Proposition 4

Construction. Create an $n \times r$ grid of vertices $V = \{v_{j,i} : j \in [n], i \in [r]\}$.

Optimal hyperedges (rows): For each $j \in [n]$, define $O_j = \{v_{j,1}, \ldots, v_{j,r}\}$. Each O_j has size r, and the O_j are pairwise vertex-disjoint. Thus $\mathrm{OPT}_{\mathrm{IP}} \geq n$.

Blocking hyperedges (column 1): For $\ell = 1, \ldots, n/r$, define $B_\ell = \{v_{(\ell-1)r+1,1}, \ldots, v_{\ell r,1}\}$. Each B_ℓ has size r.

Greedy behavior: With adversarial ordering (blocking edges processed first), greedy selects B_1, saturating vertices $v_{1,1}, \ldots, v_{r,1}$. This blocks optimal edges $O_1, \ldots, O_r$ (each contains a saturated vertex). Select B_2, blocking $O_{r+1}, \ldots, O_{2r}$. Continuing, greedy selects all n/r blocking edges, blocking all n optimal edges. The greedy algorithm selects n/r edges, while $\text{OPT}_{\text{IP}} = n$. The ratio is $n/(n/r) = r$. □

Extension to arbitrary budget $k \geq 1$. Fix $k \geq 1$ and take the instance above with every hyperedge replicated k times; recall that E is a collection of hyperedges, so repeated hyperedges are permitted. All kn row copies together form a feasible solution, since each grid vertex then appears in exactly k selected hyperedges, hence $\text{OPT}_{\text{IP}} \geq kn$. Under adversarial ordering, greedy first processes the k copies of B_1 and accepts all of them, raising each of the vertices $v_{1,1}, \ldots, v_{r,1}$ to the budget k; every copy of $O_1, \ldots, O_r$ is then blocked. Continuing with the copies of $B_2, \ldots, B_{n/r}$, greedy accepts exactly kn/r hyperedges and blocks every row copy. The ratio is $kn/(kn/r) = r$, so the r bound of Theorem 2 is exactly tight for every budget k. □

B.4 Packing Connections and Known Guarantees

Contribution bounding is maximum unweighted hypergraph b-matching with $b = k$. Equivalently, its constraint matrix has entries in $\{0, 1\}$, every right-hand side equals k, and each column has at most r nonzero entries [62, 48, 7]. This correspondence lets us apply standard packing guarantees directly.

Two established rounding guarantees. Theorem 1.4 of Bansal et al. [7] gives an $O(r^{1/k})$ approximation for integer capacities $k \geq 1$. Expressions of the form $O((kr)^{1/k})$ have the same order because $k^{1/k}$ is uniformly bounded. Their discussion of 0–1 packing also records the standard $O(m^{1/(k+1)})$ guarantee, where m is the number of constraints. In our setting, vertices of degree at most k impose no restriction on any subset, so only $h = |V_{>k}|$ constraints need to be counted. The resulting two orders are therefore $O(r^{1/k})$ and $O(h^{1/(k+1)})$. Either can be obtained by independent sampling followed by alteration. Theorem 1 shows that a single complete LP-guided greedy traversal attains both: its proof analyzes two prefixes of an independent sampling sequence and transfers their guarantees through the order of first appearances. The approximation orders are established packing results; our contribution is their analysis for this implementation and their use in fixed-graph private estimation.

Exact solutions and the cap-one case. For graphs ($r = 2$), simple b-matching solves the problem exactly for every cap [59]. At $k = 1$, the natural LP has instances with an integrality gap linear in r; projective-plane examples attain $r - 1 + 1/r$ when the corresponding plane exists [31]. Thus a guarantee relative to this LP cannot improve the linear dependence on r in general. Local search gives a $(r + 1 + \eta)/3$ approximation for any fixed $\eta > 0$ [23, 32]. The $\Omega(r)$ hardness of Lee et al. [49], under $\text{NP} \not\subseteq \text{BPP}$, matches this order up to constants. For weighted hypergraph b-matching, Parekh and Pritchard [56] give an LP-based $(r - 1 + 1/r)$ approximation at every cap. Greedy remains useful as a simple algorithm whose guarantee holds for every processing order and whose running time is linear in the incidence representation.

Larger-cap hardness. El Ouali et al. [28] state an $\Omega(r/(k \log r))$ hardness bound for $2 \leq k \leq r/\log r$. At $k = 2$, this expression exceeds the randomized polynomial-time $O(\sqrt{r})$ guarantee above for large r. Its applicability to the present unweighted, uniform-capacity problem therefore requires clarification, and we do not rely on it. APX-hardness results for bounded-degree hypergraph b-matching are given by El Ouali and Jäger [27]. The cap-one theorem alone does not establish a matching lower bound for the $r^{1/k}$ dependence.

These packing results concern the number of retained examples. Sections 4 and 4.4 address the statistical consequences of choosing a subset: the contribution cap controls sensitivity, while the identity of the selected examples determines selection bias.

B.5 Structured Networks and Per-Instance Bounds

Structured networks can admit better *per-instance* bounds that help explain empirical performance.

B.5.1 Per-Instance Bound via Saturation

The worst-case charging analysis bounds the charge absorbed by each selected edge by r. For specific instances, the byproducts recorded at the end of the proof of Theorem 2 give sharper statements.

Theorem 11 (Per-Instance Bound via Saturation Count)**.** *Let S be the greedy solution and $V_k = \{v \in V : d_S(v) = k\}$ be the saturated vertices. Then, $|S| \geq \frac{\text{OPT}_{\text{IP}}}{1+k|V_k|/|S|}$.*

Proof. From the second byproduct in the proof of Theorem 2, $|S^*| \leq |S| + k|V_k|$. Rearranging gives the result. □

Corollary 12 (Per-Instance Bound via Average Hyperedge Size)**.** *The greedy solution satisfies $|S| \geq \frac{\text{OPT}_{\text{IP}}}{\bar{r}_S}$, where $\bar{r}_S = \frac{1}{|S|}\sum_{e\in S}|e|$ is the average size of selected hyperedges.*

Proof. The first byproduct in the proof of Theorem 2 is exactly $|S^*| \leq \sum_{e\in S}|e| = \bar{r}_S \cdot |S|$. □

Corollary 13 (A Priori Per-Instance Bound)**.** *Let $m = \lceil \frac{\text{OPT}_{\text{IP}}}{r} \rceil$ and let $\bar{r}$ be the average size of the m largest hyperedges in E. The greedy solution satisfies $|S| \geq \frac{\text{OPT}_{\text{IP}}}{\bar{r}}$.*

Proof. By Theorem 2, $|S| \geq m$. For any set of $|S| \geq m$ hyperedges, the total size is at most $|S| \cdot \bar{r}$: the m largest contribute at most $m \cdot \bar{r}$, and any additional hyperedges have size at most $|e_{(m)}| \leq \bar{r}$. Thus $\sum_{e\in S}|e| \leq |S| \cdot \bar{r}$, and the result follows from the charging bound $|S^*| \leq \sum_{e\in S}|e|$ recorded in the proof of Theorem 2. □

The bound above is a-priori in the sense that $\bar{r}$ depends only on the instance (the m largest hyperedges), not on the algorithm's execution. However, it requires knowledge of OPT_{IP} to compute m.

B.5.2 Community Structure with Bounded Local Hyperedge Size

Community structure alone does not improve worst-case bounds, since intra-community hyperedges may have the same maximum size r as global hyperedges. However, if intra-community hyperedges are smaller, we obtain genuine improvement.

Definition 10 (Community Structure)**.** *A hypergraph (V, E) has (K, γ)-community structure if V can be partitioned into K communities $V_1, \ldots, V_K$ such that at least $(1-\gamma)|E|$ hyperedges are contained entirely within some community. (We write K for the number of communities and γ for the inter-community fraction to avoid collision with the network density c and the retention parameter β.)*

Definition 11 (Local Hyperedge Size)**.** *For a hypergraph with (K, γ)-community structure, define:*

- $r_{\text{local}} = \max\{|e| : e \subseteq V_i \text{ for some } i\}$ *(maximum intra-community hyperedge size)*
- $r = \max_{e\in E}|e|$ *(global maximum hyperedge size)*

Algorithm 3 Community-Aware Contribution Bounding

Require: Hypergraph (V, E), budget k, community partition $\{V_1, \ldots, V_K\}$
Ensure: Subset $S \subseteq E$ satisfying contribution bound k

```
Initialize S ← ∅
for i = 1 to K do
    E_i ← {e ∈ E : e ⊆ V_i} {Intra-community hyperedges}
    S_i ← GreedyContributionBounding(V_i, E_i, k)
    S ← S ∪ S_i
end for
E_inter ← {e ∈ E : e ⊈ V_i for all i}
Greedily add hyperedges from E_inter to S while respecting budget k
return S
```

Theorem 14 (Improved Bound with Bounded Local Hyperedge Size)**.** *Let (V, E) have (K, γ)-community structure with local hyperedge size r_{local}. Algorithm 3 achieves $|S| \geq \frac{|S^*_{\text{intra}}|}{r_{\text{local}}} + |S_{\text{inter}}|$, where $S^*_{\text{intra}} = S^* \cap E_{\text{intra}}$ is the intra-community portion of the optimal solution. In particular, if the optimal solution satisfies $|S^*_{\text{inter}}| \leq \gamma' \cdot \text{OPT}_{\text{IP}}$, then $|S| \geq \frac{(1-\gamma') \cdot \text{OPT}_{\text{IP}}}{r_{\text{local}}}$.*

Proof. Let S^* be an optimal solution. Partition $S^* = S^*_{\text{intra}} \cup S^*_{\text{inter}}$ where:

- $S^*_{\text{intra}} = \{e \in S^* : e \subseteq V_i \text{ for some } i\}$
- $S^*_{\text{inter}} = \{e \in S^* : e \not\subseteq V_i \text{ for any } i\}$

We first bound the intra-community contribution. For each community V_i, let $E_i = \{e \in E : e \subseteq V_i\}$ and OPT_i be the optimal solution on (V_i, E_i).

The set $S^*_{\text{intra}} \cap E_i$ is a feasible solution for the subproblem on (V_i, E_i), so:

$$\text{OPT}_i \geq |S^*_{\text{intra}} \cap E_i|.$$

Summing over all communities:

$$\sum_{i=1}^{K} \text{OPT}_i \geq \sum_{i=1}^{K} |S^*_{\text{intra}} \cap E_i| = |S^*_{\text{intra}}|.$$

By Theorem 2 applied to each subproblem (with maximum hyperedge size r_{local}):

$$|S_i| \geq \frac{\text{OPT}_i}{r_{\text{local}}}.$$

Since communities are disjoint, the solutions do not interfere:

$$|S_{\text{intra}}| = \sum_{i=1}^{K} |S_i| \geq \frac{\sum_{i=1}^{K} \text{OPT}_i}{r_{\text{local}}} \geq \frac{|S^*_{\text{intra}}|}{r_{\text{local}}}.$$

The algorithm also adds inter-community edges in line 7:

$$|S| = |S_{\text{intra}}| + |S_{\text{inter}}| \geq \frac{|S^*_{\text{intra}}|}{r_{\text{local}}} + |S_{\text{inter}}|.$$

If we additionally have $|S^*_{\text{inter}}| \leq \gamma' \cdot \text{OPT}_{\text{IP}}$, then:

$$|S^*_{\text{intra}}| = \text{OPT}_{\text{IP}} - |S^*_{\text{inter}}| \geq (1 - \gamma') \cdot \text{OPT}_{\text{IP}}.$$

Combining this inequality with $|S| \geq |S_{\text{intra}}|$ gives the claimed bound. □

The condition $|S^*_{\text{inter}}| \leq \gamma' \cdot \text{OPT}_{\text{IP}}$ is *not* automatic for some small γ' from the (K, γ)-community structure definition. A sufficient condition is $\gamma|E| \leq \gamma' \text{OPT}_{\text{IP}}$, since $|S^*_{\text{inter}}| \leq |E_{\text{inter}}| \leq \gamma|E|$. Thus, the fraction of inter-community edges must be considered relative to the size of an optimal feasible selection.

Remark 5 (Practical Benefits of Community-Aware Algorithm)**.** *Even when $r_{\text{local}} = r$, Algorithm 3 offers practical benefits. First, intra-community subproblems are independent and can be solved in parallel. Second, solving K smaller problems may be faster than one large problem, improving scalability.*

C Detailed Proofs for Error Bounds

This section provides complete proofs for the information-theoretic lower bounds and matching upper bounds in Section 4.

C.1 Computing Network Density Parameter for Common Networks

Example (Example 1, restated: Computing c for Common Network Types - Detailed Calculations)**.**

We compute (or tightly bound) the network density parameter $c = \min_{m\in[1,n]} f(m)$ for several canonical structures. Throughout, for $\tilde{E} \subseteq E$ we write $\deg_{\tilde{E}}(v) := |\{e \in \tilde{E} : v \in e\}|$, and let $\Delta := \max_{v\in V} \deg_E(v)$ be the maximum degree in the full hypergraph.

A universal observation. For any hypergraph, taking $m = n$ forces $\tilde{E} = E$, hence

$$f(n) = \frac{\max_{v\in V} \deg_E(v)}{n} = \frac{\Delta}{n}, \qquad \text{so in particular } c \le \frac{\Delta}{n}. \tag{11}$$

- **r-uniform, Δ-regular networks.** Assume the hypergraph is r-*uniform* (each edge has size exactly r) and Δ-*regular* (each vertex has full degree exactly Δ). Then $nr = |V|\Delta$.
  Fix any $m \in [1, n]$ and any $\tilde{E} \subseteq E$ with $|\tilde{E}| = m$. The total number of incidences inside $\tilde{E}$ is
  $$\sum_{v\in V} \deg_{\tilde{E}}(v) = \sum_{e\in\tilde{E}} |e| = mr.$$
  By averaging, $\max_v \deg_{\tilde{E}}(v) \ge \frac{mr}{|V|}$, and therefore
  $$f(m) \;\ge\; \frac{(mr)/|V|}{m} \;=\; \frac{r}{|V|} \;=\; \frac{\Delta}{n}. \tag{12}$$
  On the other hand, (11) gives $f(n) = \Delta/n$, hence
  $$c = \min_{m\in[1,n]} f(m) = \frac{\Delta}{n}.$$
  In particular, for (simple) d-regular graphs ($r = 2$, $\Delta = d$) this yields $c = d/n$.

- **Approximately regular networks.** Suppose the hypergraph is r-uniform and a constant fraction of vertices have degree $\Theta(\Delta)$, so that the average degree is $\Theta(\Delta)$ (e.g., dense random graphs or networks with near-uniform participation). Then $nr = \Theta(|V|\Delta)$, and the same averaging argument yields $f(m) \ge \Theta(\Delta/n)$ for all m, while $f(n) = \Delta/n$. Thus,
  $$c = \Theta\left(\frac{\Delta}{n}\right).$$
  In networks without extreme concentration (e.g., approximately regular networks), c is small whenever $\Delta = o(n)$.

- **Star networks.** Consider the standard star graph on edges $e_i = \{v_c, v_i\}$ for $i = 1, \dots, n$ (each edge contains the center v_c). Then for any subset $\tilde{E}$ of size $m \ge 1$, the center appears in every chosen edge:
  $$\max_v \deg_{\tilde{E}}(v) = \deg_{\tilde{E}}(v_c) = m, \qquad \Rightarrow \qquad f(m) = \frac{m}{m} = 1.$$
  Therefore $c = \min_m f(m) = 1$.

- **Clustered networks (disjoint hubs).** Consider a hypergraph with k disjoint groups (clusters) $G_1, \dots, G_k$, where each cluster G_i contributes n_i edges, all containing a cluster hub v_i, and hubs are distinct. For simplicity assume equal sizes $n_i = n/k$. To minimize overlap in a subset of m edges, we distribute chosen edges as evenly as possible across the k clusters. Then the smallest possible maximum hub-load is
  $$\min_{|\tilde{E}|=m} \max_v \deg_{\tilde{E}}(v) = \left\lceil \frac{m}{k} \right\rceil, \qquad \Rightarrow \qquad f(m) = \frac{\lceil m/k \rceil}{m}.$$
  Thus the minimum over m occurs when $m = k$, yielding
  $$c = \min_{m\in[1,n]} f(m) = \frac{1}{k} \quad \text{(up to ceiling effects).}$$
  If each cluster size is Δ (so $n_i = \Delta$ and $k = n/\Delta$), then $c = 1/k = \Delta/n$.

- **Matching / disjoint-edge networks.** Suppose edges are pairwise disjoint (each vertex appears in exactly one edge). Then $\max_v \deg_{\tilde{E}}(v) = 1$ for any subset $\tilde{E}$, so

$$f(m) = \frac{1}{m}, \qquad c = f(n) = \frac{1}{n}.$$

This is the maximally dispersed regime.

- **Author-paper hypergraphs.** Consider a standard author–paper network: each paper is a hyperedge containing its authors. Assume: (i) each paper has $r = O(1)$ authors, (ii) the most prolific author writes $\Delta = o(n)$ papers, and (iii) a constant fraction of authors write $\Theta(\Delta)$ papers (e.g., large collaborations with roughly uniform senior participation).
Then the hypergraph is r-uniform and approximately regular, so

$$f(n) = \frac{\Delta}{n}, \qquad f(m) \geq \Theta\left(\frac{\Delta}{n}\right) \quad \forall m,$$

and hence

$$c = \Theta\left(\frac{\Delta}{n}\right).$$

This captures realistic scientific collaboration networks where no single author dominates the corpus, but repeated participation induces correlations.

- **Heavy-tailed (power-law) networks.** For any network with maximum degree $\Delta_{\max}$,

$$c \leq \frac{\Delta_{\max}}{n}.$$

Whether this upper bound is tight depends on structure: hub-dominated networks (e.g., stars) have $c = \Theta(1)$, while networks with many low-overlap edges can achieve $c = \Theta(\Delta_{\max}/n)$. Without additional assumptions, no sharper universal characterization is possible.

These examples illustrate that c precisely captures the extent to which correlations induced by repeated participation can be diluted by selecting favorable subsets of examples.

C.2 Network-Structure-Dependent Error Lower Bound

We first instantiate a lower bound in terms of c rather than the refined density parameter c_β. We then proceed to give a detailed proof of lower bound in Theorem 5 in terms of c_β.

Theorem 15 (Network-Structure-Dependent Error Lower Bound)**.** *Consider the mean estimation problem on attributed datasets with hypergraph structure (V, E) where $|E| = n$ examples, maximum degree Δ, network density parameter c (Definition 8), and d-dimensional bounded parameter space $\mathcal{X} = \{x \in \mathbb{R}^d \mid \|x\|_2 \leq 1\}$.*

For any network with density parameter c and sufficiently small ε (smaller than absolute constant), any (ε, δ)-fixed-graph differentially private algorithm $\mathcal{A}$ achieving expected squared error at most α for every sample $X \in \mathcal{X}^n$ must incur error:

$$\alpha = \begin{cases} \Omega\left(\min\left\{1, \frac{c^2 d^2}{\varepsilon^2}\right\}\right) & \text{if } \delta = 0, \\ \tilde{\Omega}\left(\min\left\{1, \frac{c^2 d}{\varepsilon^2}\right\}\right) & \text{if } \delta = o\left(c\right). \end{cases} \tag{13}$$

Specifically, for networks where the density parameter $c = \Theta(\Delta/n)$, the error satisfies:

$$\alpha = \begin{cases} \Omega\left(\min\left\{1, \frac{\Delta^2 d^2}{\varepsilon^2 n^2}\right\}\right) & \text{if } \delta = 0, \\ \tilde{\Omega}\left(\min\left\{1, \frac{\Delta^2 d}{\varepsilon^2 n^2}\right\}\right) & \text{if } \delta = o\left(\frac{\Delta}{n}\right). \end{cases} \tag{14}$$

As we show below, the proof constructs a reduction from the fixed-graph differentially private mean estimation to fundamental mean estimation lower bounds under differential privacy, and utilizing a prior result from [9] that we restate for completeness.

Lemma 16 (Lemma 5.1 from Bassily et al. [9]: Lower bounds for 1-way marginals)**.**

1. ***ε-differentially private algorithms:*** *Let $n, d \in \mathbb{N}_+$ and let ε be sufficiently small (smaller than absolute constant). For every ε-differentially private algorithm $\mathcal{A}$, there is a dataset*

$$X = \{x_1, \ldots, x_n\} \subseteq \left\{-\frac{1}{\sqrt{d}}, \frac{1}{\sqrt{d}}\right\}^d$$

such that, with probability at least $1/2$ (taken over the algorithm random coins), we have

$$\left\| \mathcal{A}(X) - \frac{1}{n}\sum_{i=1}^{n} x_i \right\|_2 = \Omega\left(\min\left(1, \frac{d}{\varepsilon n}\right)\right).$$

2. ***(ε, δ)-differentially private algorithms:*** *Let $n, d \in \mathbb{N}_+$ and let ε be sufficiently small (smaller than absolute constant), and $\delta = o\left(\frac{1}{n}\right)$. For every (ε, δ)-differentially private algorithm $\mathcal{A}$, there is a dataset*

$$X = \{x_1, \ldots, x_n\} \subseteq \left\{-\frac{1}{\sqrt{d}}, \frac{1}{\sqrt{d}}\right\}^d$$

such that, with probability at least $1/3$ (taken over the algorithm random coins), we have

$$\left\| \mathcal{A}(X) - \frac{1}{n}\sum_{i=1}^{n} x_i \right\|_2 = \tilde{\Omega}\left(\min\left(1, \frac{\sqrt{d}}{\varepsilon n}\right)\right).$$

Next, we sketch our main arguments in proving Theorem 15.

Key insight. The key insight relates to effective sample size and privacy amplification. Consider a network with density parameter c. By Definition 8, we can partition the examples into $k \approx 1/c$ groups, where each group shares a common high-degree vertex. These groups act as "super-samples," each containing $\Theta(cn)$ examples.

We now briefly explain how the reduction from fixed-graph DP to standard DP mean estimation works. Any fixed-graph (ε, δ)-DP algorithm on this network can be transformed into a standard (ε, δ)-DP algorithm for mean estimation with $k \approx 1/c$ samples. The transformation works by replicating each of k arbitrary data points within the corresponding network group. By Bassily et al. [9] (Lemma 5.1), mean estimation with k samples under (ε, δ)-DP incurs error:

$$\alpha = \tilde{\Omega}\left(\frac{d}{\varepsilon^2 k^2}\right) = \tilde{\Omega}\left(\frac{c^2 d}{\varepsilon^2}\right). \tag{15}$$

For networks where the maximum $c = \Theta(\Delta/n)$. Substituting:

$$\alpha = \tilde{\Omega}\left(\frac{(\Delta/n)^2 d}{\varepsilon^2}\right) = \Omega\left(\frac{\Delta^2 d}{\varepsilon^2 n^2}\right). \tag{16}$$

Before we give a complete proof with formal reduction argument, we prove the following lemma.

Lemma 17. *From the definition of c, we can partition a subset of the n hyperedges into $k := \left\lfloor \frac{n}{2\left\lceil \frac{cn}{2} \right\rceil} \right\rfloor + 1$ disjoint groups $E_1, \ldots, E_k$, s.t. each group has exactly $\left\lceil \frac{cn}{2} \right\rceil$ data and all the hyperedges in each group share a common vertex. The total number of hyperedges in the groups $\in \left(\frac{n}{2}, n\right]$.*

Proof of Lemma 17. We prove the claim by contradiction. Assume we can find at most $k \leq \left\lfloor \frac{n}{2\left\lceil \frac{cn}{2} \right\rceil} \right\rfloor$ disjoint groups $E_1, \ldots, E_k$, s.t. each group has exactly $\left\lceil \frac{cn}{2} \right\rceil$ hyperedges and all the hyperedges in each group share a common vertex. Based on this assumption,

$$|E_1 \cup \cdots \cup E_k| = k\left\lceil \frac{cn}{2} \right\rceil \leq \left\lfloor \frac{n}{2\left\lceil \frac{cn}{2} \right\rceil} \right\rfloor \cdot \left\lceil \frac{cn}{2} \right\rceil \leq \frac{n}{2\left\lceil \frac{cn}{2} \right\rceil} \cdot \left\lceil \frac{cn}{2} \right\rceil = \frac{n}{2}.$$

By the definition of $f(m)$ and c, $\forall \tilde{E} \subseteq E, |\tilde{E}| = m$, there exists $v^* \in V$, s.t. $|\{e \in \tilde{E} : v^* \in e\}| \geq cm$. We apply $\tilde{E} = E \backslash (E_1 \cup \cdots \cup E_k)$ and get

$$|\{e \in \tilde{E} : v^* \in e\}| \geq cm = c|\tilde{E}| = c(n - |E_1 \cup \cdots \cup E_k|) \geq c\left(n - \frac{n}{2}\right) = \frac{cn}{2}.$$

Since $|\{e \in \tilde{E} : v^* \in e\}|$ is an integer, $|\{e \in \tilde{E} : v^* \in e\}| \geq \lceil \frac{cn}{2} \rceil$. Therefore, we can choose $\lceil \frac{cn}{2} \rceil$ hyperedges from $\{e \in \tilde{E} : v^* \in e\}$ to form E_{k+1}, and these hyperedges share a common vertex v^*. We have thus constructed $k+1$ disjoint groups $E_1, \ldots, E_{k+1}$, s.t. each group has exactly $\lceil \frac{cn}{2} \rceil$ hyperedges and all the hyperedges in each group share a common vertex. This is a contradiction with the assumption. Therefore, we have $\left\lfloor \frac{n}{2\lceil \frac{cn}{2} \rceil} \right\rfloor + 1$ disjoint groups s.t. each group has exactly $\lceil \frac{cn}{2} \rceil$ hyperedges and all the hyperedges in each group share a common vertex.

The total number of chosen hyperedges is $k \lceil \frac{cn}{2} \rceil$. From our analysis above, $k \lceil \frac{cn}{2} \rceil \leq n$ is obvious since the groups are disjoint. We also have

$$k \left\lceil \frac{cn}{2} \right\rceil = \left(\left\lfloor \frac{n}{2\lceil \frac{cn}{2} \rceil} \right\rfloor + 1 \right) \left\lceil \frac{cn}{2} \right\rceil > \frac{n}{2\lceil \frac{cn}{2} \rceil} \left\lceil \frac{cn}{2} \right\rceil = \frac{n}{2}.$$

□

Proof of Theorem 15. If there exists a fixed-graph ε-DP or (ε, δ)-DP ($\delta = o(c)$) algorithm $\mathcal{A}$ with error bound α, we will first construct a standard ε-DP or (ε, δ)-DP ($\delta = o\left(\frac{1}{k}\right)$) algorithm $\tilde{\mathcal{A}}$ that estimates the mean of arbitrary given $k = \left\lfloor \frac{n}{2\lceil \frac{cn}{2} \rceil} \right\rfloor + 1$ data $\tilde{X} = \{\tilde{x}_1, \ldots, \tilde{x}_k\} \in \mathcal{X}^k$. From Lemma 17, we have $k = \left\lfloor \frac{n}{2\lceil \frac{cn}{2} \rceil} \right\rfloor + 1$ disjoint groups of hyperedges $E_1, \ldots, E_k$, s.t. each group has exactly $\lceil \frac{cn}{2} \rceil$ hyperedges and all the hyperedges in each group share a common vertex. Now we construct a new sample X in the graph. For any $i \in [1, k]$, replicate the data $\tilde{x}_i$ for $\lceil \frac{cn}{2} \rceil$ times as the data in E_i. Set the rest of the data in $E \backslash (E_1 \cup \cdots \cup E_k)$ as $\mathbf{0}$. Let our new algorithm return $\tilde{\mathcal{A}}(\tilde{X}) = \frac{n}{k\lceil \frac{cn}{2} \rceil} \mathcal{A}(X)$. We can verify this new algorithm $\tilde{A}$ is ε-DP or (ε, δ)-DP ($\delta = o(c)$), because changing one data $\tilde{x}_i$ will only affect the data in E_i, which share a common vertex. By the definition of fixed-graph DP, changing all the data attributed to this single user preserves privacy. Therefore, this procedure will be ε-DP or (ε, δ)-DP. Since our algorithm $\mathcal{A}$ has expected squared error at most α for $X \in \mathcal{X}^n$, we know

$$\mathbb{E}_{\mathcal{A}} \left\| \mathcal{A}(X) - \frac{\lceil \frac{cn}{2} \rceil (\tilde{x}_1 + \cdots + \tilde{x}_k)}{n} \right\|_2^2 \leq \alpha.$$

From the definition of $\tilde{A}$, we have

$$\mathbb{E}_{\mathcal{A}} \left\| \tilde{\mathcal{A}}(\tilde{X}) - \frac{\tilde{x}_1 + \cdots + \tilde{x}_k}{k} \right\|_2^2 \leq \left(\frac{n}{k\lceil \frac{cn}{2} \rceil} \right)^2 \alpha < 4\alpha. \qquad (k\lceil \tfrac{cn}{2} \rceil > \tfrac{n}{2} \text{ from Lemma 17})$$

However, for this specific algorithm $\tilde{A}$, Lemma 16 says there exists $\tilde{X}$, s.t. the expected squared error is lower bounded by

$$\mathbb{E}_{\mathcal{A}} \left\| \tilde{\mathcal{A}}(\tilde{X}) - \frac{\tilde{x}_1 + \cdots + \tilde{x}_k}{k} \right\|_2^2 \geq \begin{cases} \Omega\left(\min\left(1, \frac{d^2}{\varepsilon^2 k^2}\right)\right) & \tilde{\mathcal{A}} \text{ is } \varepsilon\text{-DP}, \\ \tilde{\Omega}\left(\min\left(1, \frac{d}{\varepsilon^2 k^2}\right)\right) & \tilde{\mathcal{A}} \text{ is } (\varepsilon, \delta)\text{-DP with } \delta = o(c) = o\left(\frac{1}{k}\right). \end{cases}$$

Here $\delta = o\left(\frac{1}{k}\right)$ is because

$$ck = c\left(\left\lfloor \frac{n}{2\lceil \frac{cn}{2} \rceil} \right\rfloor + 1 \right) \leq c\left(\frac{n}{2\lceil \frac{cn}{2} \rceil} + 1 \right) \leq c\left(\frac{n}{cn} + 1 \right) = 1 + c \leq 2.$$

This also implies $k \leq \frac{2}{c}$, and thus we have

$$\begin{aligned} \alpha &> \frac{1}{4} \mathbb{E}_{\mathcal{A}} \left\| \tilde{\mathcal{A}}(\tilde{X}) - \frac{\tilde{x}_1 + \cdots + \tilde{x}_k}{k} \right\|_2^2 \\ &\geq \begin{cases} \Omega\left(\min\left(1, \frac{c^2 d^2}{\varepsilon^2}\right)\right) & \varepsilon\text{-fixed-graph DP}, \\ \tilde{\Omega}\left(\min\left(1, \frac{c^2 d}{\varepsilon^2}\right)\right) & (\varepsilon, \delta)\text{-fixed-graph DP with } \delta = o(c). \end{cases} \end{aligned}$$

□

Remark 6 (Error Decay with Sample Size). *For networks with fixed maximum degree Δ, the error $\alpha = \tilde{\Omega}(\Delta^2 d/(\varepsilon^2 n^2))$ decays quadratically with the number of samples n. This is the expected behavior as more samples lead to better accuracy, even under privacy constraints.*

However, for pathological networks where the maximum degree grows with n (e.g., $\Delta = \Theta(n)$ in star graphs where a central vertex connects to all examples), the density parameter $c = \Theta(\Delta/n) = \Theta(1)$ remains constant, yielding error $\alpha = \Theta(\min\{1, d/\varepsilon^2\})$ independent of n. Such networks are fundamentally hard for private learning as high connectivity prevents privacy amplification through additional samples.

Next, we proceed to give a proof of network-structure-dependent error lower bound (Theorem 5) in terms of the refined network density parameter c_β. First, we prove the following useful lemma.

Lemma 18. *From the definition of c, for any $\beta \in [0, 1)$, we can partition a subset of the n hyperedges into $k := \left\lfloor \frac{\beta n}{\lceil c_\beta(1-\beta)n \rceil} \right\rfloor + 1$ disjoint groups $E_1, \ldots, E_k$, s.t. each group has exactly $\lceil c_\beta(1-\beta)n \rceil$ data and all the hyperedges in each group share a common vertex. The total number of hyperedges in the groups $\in (\beta n, n]$.*

Proof of Lemma 18. We prove the claim by contradiction. Assume we can find at most $k \leq \left\lfloor \frac{\beta n}{\lceil c_\beta(1-\beta)n \rceil} \right\rfloor$ disjoint groups $E_1, \ldots, E_k$, s.t. each group has exactly $\lceil c_\beta(1-\beta)n \rceil$ hyperedges and all the hyperedges in each group share a common vertex. Based on this assumption,

$$\begin{aligned} |E_1 \cup \cdots \cup E_k| &= k \lceil c_\beta(1-\beta)n \rceil \\ &\leq \left\lfloor \frac{\beta n}{\lceil c_\beta(1-\beta)n \rceil} \right\rfloor \cdot \lceil c_\beta(1-\beta)n \rceil \\ &\leq \frac{\beta n}{\lceil c_\beta(1-\beta)n \rceil} \cdot \lceil c_\beta(1-\beta)n \rceil \\ &= \beta n. \end{aligned}$$

By the definition of $f(m)$ and c_β, $\forall \tilde{E} \subseteq E, |\tilde{E}| = m \geq (1-\beta)n$, there exists $v^* \in V$, s.t. $|\{e \in \tilde{E} : v^* \in e\}| \geq c_\beta m$. We apply $\tilde{E} = E \backslash (E_1 \cup \cdots \cup E_k)$ and get

$$|\{e \in \tilde{E} : v^* \in e\}| \geq c_\beta m = c_\beta |\tilde{E}| = c_\beta (n - |E_1 \cup \cdots \cup E_k|) \geq c_\beta \left(n - \beta n\right) = c_\beta (1-\beta) n.$$

Since $|\{e \in \tilde{E} : v^* \in e\}|$ is an integer, $|\{e \in \tilde{E} : v^* \in e\}| \geq \lceil c_\beta(1-\beta)n \rceil$. Therefore, we can choose $\lceil c_\beta(1-\beta)n \rceil$ hyperedges from $\{e \in \tilde{E} : v^* \in e\}$ to form E_{k+1}, and these hyperedges share a common vertex v^*. We have thus constructed $k+1$ disjoint groups $E_1, \ldots, E_{k+1}$, s.t. each group has exactly $\lceil c_\beta(1-\beta)n \rceil$ hyperedges and all the hyperedges in each group share a common vertex. This is a contradiction with the assumption. Therefore, we have $\left\lfloor \frac{\beta n}{\lceil c_\beta(1-\beta)n \rceil} \right\rfloor + 1$ disjoint groups s.t. each group has exactly $\lceil c_\beta(1-\beta)n \rceil$ hyperedges and all the hyperedges in each group share a common vertex. The total number of chosen hyperedges is $k \lceil c_\beta(1-\beta)n \rceil$. From our analysis above, $k \lceil c_\beta(1-\beta)n \rceil \leq n$ is obvious since the groups are disjoint. We also have

$$\begin{aligned} k \lceil c_\beta(1-\beta)n \rceil &= \left(\left\lfloor \frac{\beta n}{\lceil c_\beta(1-\beta)n \rceil} \right\rfloor + 1 \right) \lceil c_\beta(1-\beta)n \rceil \\ &> \frac{\beta n}{\lceil c_\beta(1-\beta)n \rceil} \lceil c_\beta(1-\beta)n \rceil \\ &= \beta n. \end{aligned}$$

□

Theorem (Theorem 5, restated). *Consider the mean estimation task on attributed datasets with hypergraph structure (V, E). Each hyperedge $e \in E$ is associated with a data point x_e in the d-dimensional bounded domain $\mathcal{X} = \{x \in \mathbb{R}^d \mid \|x\|_2 \leq 1\}$, and $\mathcal{A}$ is an (ε, δ)-fixed-graph DP algorithm with*

$$\sup_{X \in \mathcal{X}^n} \mathbb{E}_{\mathcal{A}} \left\| \mathcal{A}(X) - \frac{1}{n} \sum_{e \in E} x_e \right\|_2^2 \leq \alpha.$$

There exist absolute constants $C, \tilde{C}, \varepsilon_0 > 0$ such that for every $\beta \in (0, 1/2]$ and every $\varepsilon \le \varepsilon_0$: if $\delta = 0$ then $\alpha \ge C \min\{\beta^2,\ c_\beta^2 d^2/\varepsilon^2\}$, and if $\delta = o(c_\beta)$ then $\alpha \ge \tilde{C} \min\{\beta^2,\ c_\beta^2 d/\varepsilon^2\}$ up to logarithmic factors.

Proof of Theorem 5. Fix an arbitrary $\beta \in (0, 1/2]$. The argument applies to every such β; the special choice $\beta = C_0\sqrt{\alpha}$ is needed only to translate the resulting inequality into an implicit bound, which we record at the end. From Lemma 18 we obtain $k = \left\lfloor \frac{\beta n}{\lceil c_\beta (1-\beta) n \rceil} \right\rfloor + 1$ disjoint groups of hyperedges $E_1, \dots, E_k$, each of size $g = \lceil c_\beta(1-\beta)n \rceil$, all hyperedges within a group sharing a common vertex, and with total size $kg \in (\beta n, n]$.

Given arbitrary data $\tilde{X} = \{\tilde{x}_1, \dots, \tilde{x}_k\} \in \mathcal{X}^k$, construct a sample X on the graph by replicating $\tilde{x}_i$ on every hyperedge of E_i and setting the data on $E \backslash (E_1 \cup \cdots \cup E_k)$ to $\mathbf{0}$, and define $\tilde{\mathcal{A}}(\tilde{X}) = \frac{n}{kg}\mathcal{A}(X)$. The induced algorithm $\tilde{\mathcal{A}}$ is ε-DP, respectively (ε, δ)-DP, because changing one data point $\tilde{x}_i$ changes only the data attributed to the common vertex of E_i. Since $\mathcal{A}$ has expected squared error at most α and $kg > \beta n$,

$$\mathbb{E}\left\| \tilde{\mathcal{A}}(\tilde{X}) - \frac{\tilde{x}_1 + \cdots + \tilde{x}_k}{k} \right\|_2^2 \le \left(\frac{n}{kg}\right)^2 \alpha < \frac{\alpha}{\beta^2}.$$

Applying Lemma 16 to $\tilde{\mathcal{A}}$ with k samples gives

$$\frac{\alpha}{\beta^2} \ \ge\ \begin{cases} C_2 \min\left(1, \frac{d^2}{\varepsilon^2 k^2}\right) & \tilde{\mathcal{A}} \text{ is } \varepsilon\text{-DP}, \\ \tilde{\Omega}\left(\min\left(1, \frac{d}{\varepsilon^2 k^2}\right)\right) & \tilde{\mathcal{A}} \text{ is } (\varepsilon,\delta)\text{-DP with } \delta = o\left(\frac{1}{k}\right), \end{cases}$$

where $\delta = o(c_\beta)$ implies $\delta = o(1/k)$ because

$$c_\beta k = c_\beta \left(\left\lfloor \frac{\beta n}{\lceil c_\beta(1-\beta)n \rceil} \right\rfloor + 1 \right) \le c_\beta \left(\frac{\beta n}{c_\beta (1-\beta) n} + 1 \right) = \frac{\beta}{1-\beta} + c_\beta \le 2,$$

using $\beta \le \frac{1}{2}$ and $c_\beta \le 1$. Hence, in the pure case, $\alpha \ge C_2 \min\{\beta^2,\ \beta^2 d^2/(\varepsilon^2 k^2)\}$. It remains to bound β/k from below. Since $\beta \le 1/2$, $k \le \beta/((1-\beta)c_\beta) + 1 \le 2\beta/c_\beta + 1$. If $\beta \ge c_\beta/2$ then $2\beta/c_\beta \ge 1$, so $k \le 4\beta/c_\beta$ and $\beta/k \ge c_\beta/4$, giving $\alpha \ge (C_2/16)\min\{\beta^2,\ c_\beta^2 d^2/\varepsilon^2\}$. If $\beta < c_\beta/2$ then $k = 1$, and the display gives $\alpha \ge C_2 \min\{\beta^2, \beta^2 d^2/\varepsilon^2\} = C_2\beta^2 \ge C_2 \min\{\beta^2, c_\beta^2 d^2/\varepsilon^2\}$, using $d \ge 1 \ge \varepsilon_0 \ge \varepsilon$. The approximate case is identical with d in place of d^2, up to logarithmic factors.

Finally, choose $\beta = C_0\sqrt{\alpha}$ with $C_0 = \sqrt{2/C_2}$ and $\alpha \le 1/(4C_0^2)$, so that $\beta \le 1/2$ and $C_2\beta^2 = 2\alpha$. If this β satisfied $\beta < c_\beta/2$, the $k = 1$ case above would give $\alpha \ge C_2\beta^2 = 2\alpha$, which is impossible; hence $\beta \ge c_\beta/2$. In that regime the derivation above gives the two-branch bound in the sharper form $\alpha \ge \min\{C_2\beta^2,\ (C_2/16)\, c_\beta^2 d^2/\varepsilon^2\}$, and the first branch cannot be the minimum either, since $C_2\beta^2 = 2\alpha > \alpha$. The second branch is therefore forced, and we recover the implicit bound $\alpha \ge (C_2/16)\, c^2_{C_0\sqrt{\alpha}} d^2/\varepsilon^2 = \Omega(c^2_{C_0\sqrt{\alpha}} d^2/\varepsilon^2)$ for pure differential privacy; the approximate case is identical with d in place of d^2, with C_0 enlarged to exceed $\sqrt{2}$ times the reciprocal square root of the corresponding constant, giving $\alpha = \tilde{\Omega}(c^2_{C_0\sqrt{\alpha}} d/\varepsilon^2)$. □

C.3 Minimax Equality and Scaling: Proof of Lemma 19

Lemma 19 (Minimax equality and scaling)**.** *Let $f(\beta) = \beta^2$ and $g(\beta) = c_\beta^2 D$. If $g(1/2) \le 1/4$, then $R(D) = \inf_{\beta \in (0,1/2]} \max\{f(\beta), g(\beta)\} = (\beta^\star)^2$, where $\beta^\star = \inf\{\beta : f(\beta) \ge g(\beta)\}$; only continuity of f is used, so the step-function nature of c_β is harmless. If c_β jumps at $\beta^\star$, the two optimizers are approached by the one-sided limits $\beta \uparrow \beta^\star$ and $\beta \downarrow \beta^\star$, and since $c_\beta \ge c_{\beta^\star}$ to the left, the condition $\delta = o(c_{\beta^\star})$ suffices for the lower bound. If $g(1/2) > 1/4$, the minimax error is already constant, and after capping at the trivial rate $\alpha \le 1$ the two quantities agree within an absolute factor of four; we set $\beta^\star := 1/2$ in that case. Moreover $R(LD) \le L\, R(D)$ for every $L \ge 1$.*

Proof of Lemma 19. Write $f(\beta) = \beta^2$ and $g(\beta) = c_\beta^2 D$; f is continuous and increasing with $f(0^+) = 0$, and g is non-increasing since enlarging the feasible range of m can only decrease the minimum defining c_β.

Suppose first $g(1/2) \le f(1/2)$ and let $\beta^\star = \inf\{\beta \in (0, 1/2] : f(\beta) \ge g(\beta)\}$, which is well defined. By monotonicity, $f \le g$ on $(0, \beta^\star)$ and $f \ge g$ on $(\beta^\star, 1/2]$. Therefore

$$\sup_\beta \min\{f, g\} = \max\Big\{ \sup_{\beta<\beta^\star} f(\beta),\ \sup_{\beta>\beta^\star} g(\beta)\Big\} = \max\big\{f(\beta^{\star-}),\ g(\beta^{\star+})\big\} = (\beta^\star)^2,$$

using continuity of f for the first term and $g(\beta^{\star+}) \le f(\beta^{\star+}) = (\beta^\star)^2$ for the second. Symmetrically,

$$\inf_\beta \max\{f, g\} = \min\Big\{ \inf_{\beta<\beta^\star} g(\beta),\ \inf_{\beta>\beta^\star} f(\beta)\Big\} = \min\big\{g(\beta^{\star-}),\ f(\beta^{\star+})\big\} = (\beta^\star)^2,$$

since $g(\beta^{\star-}) \ge f(\beta^{\star-}) = (\beta^\star)^2$. Only continuity of f was used, so jumps of g are harmless. If c_β jumps at $\beta^\star$ itself, neither optimizer need be attained at $\beta^\star$; the supremum is approached as $\beta \uparrow \beta^\star$ and the infimum as $\beta \downarrow \beta^\star$, and since $c_\beta \ge c_{\beta^\star}$ for $\beta \le \beta^\star$, the condition $\delta = o(c_{\beta^\star})$ implies $\delta = o(c_\beta)$ for every β used on the lower-bound side.

If $g(1/2) > f(1/2)$, then $f < g$ throughout $(0, 1/2]$, so $\sup \min\{f, g\} = \sup f = 1/4$ while $\inf \max\{f, g\} = g(1/2) > 1/4$. Capping both quantities at the trivial rate $\alpha \le 1$, achieved by outputting 0, the ratio is at most $1/(1/4) = 4$. We set $\beta^\star := 1/2$ in this case.

For the scaling claim, fix β and $L \ge 1$. If $\beta^2 \le c_\beta^2 D$ then $\min\{\beta^2, Lc_\beta^2 D\} = \beta^2 \le L\beta^2 = L\min\{\beta^2, c_\beta^2 D\}$. If $c_\beta^2 D < \beta^2 \le Lc_\beta^2 D$ then $\min\{\beta^2, Lc_\beta^2 D\} = \beta^2 \le Lc_\beta^2 D = L\min\{\beta^2, c_\beta^2 D\}$. If $\beta^2 > Lc_\beta^2 D$ then both minima equal their second arguments and the claim is an identity. Taking suprema over β gives $R(LD) \le LR(D)$. □

Appendix C.5 works the characterization out on a mixed star-plus-singleton family where c_β changes by an unbounded factor under a constant rescaling of β; the crossing-point rate remains tight there.

C.4 Instance-Dependent Upper Bound (Theorem 6)

Theorem (Theorem 6, restated)**.** *Consider any network (V, E) with $|E| = n$ and the refined network density c_β from Definition 9. Let $0 < \varepsilon \le 1$ and $0 < \delta \le e^{-1}$. For every $\beta \in (0, 1]$ there exists an (ε, δ)-fixed-graph DP algorithm achieving, for every $X \in \mathcal{X}^n$,*

$$\mathbb{E}_{\mathcal{A}} \left\| \mathcal{A}(X) - \frac{1}{n} \sum_{e \in E} x_e \right\|_2^2 \ \le\ C' \left(\beta^2 + \frac{c_\beta^2\, d \log(1/\delta)}{\varepsilon^2} \right)$$

for an absolute constant C'.

Proof. Fix $\beta \in (0, 1]$. From the definition $c_\beta = \min_{m \in [(1-\beta)n, n]} f(m)$ and

$$f(m) = \min_{\substack{S \subseteq E \\ |S| = m}} \left[\max_{v \in V} |\{e \in S : v \in e\}| \right] / m,$$

there exists a subset $S \subseteq E$ with $|S| \ge (1 - \beta)n$ in which every vertex $v \in V$ appears in at most $k = |S| \cdot c_\beta$ hyperedges. While finding this subset may be computationally intractable in general, its existence suffices for this theorem; Theorem 8 gives the polynomial-time counterpart.

Given the subset S, we compute the empirical mean $\theta_S^* = \frac{1}{|S|} \sum_{e \in S} x_e$ and release $\hat\theta = \theta_S^* + Z$ with $Z \sim \mathcal{N}(0, \sigma^2 I_d)$. When the data of a single user changes, at most k hyperedges of S are affected, and since $\|x_e\|_2 \le 1$ the ℓ_2-sensitivity of θ_S^* is $\Delta_2 = 2k/|S| = 2c_\beta$. Under the stated parameter assumptions, the Gaussian mechanism with $\sigma = \frac{2c_\beta \sqrt{2\log(1.25/\delta)}}{\varepsilon}$ ensures (ε, δ)-fixed-graph differential privacy.

For the utility, write $\theta^* = \frac{1}{n} \sum_{e \in E} x_e$. The noise contributes $\mathbb{E}\|Z\|_2^2 = \sigma^2 d = O(c_\beta^2 d \log(1/\delta)/\varepsilon^2)$. For the selection bias, $n - |S| \le \beta n$ and $\|x_e\|_2 \le 1$ give

$$\|\theta_S^* - \theta^*\|_2 \ \le\ \left| \frac{1}{|S|} - \frac{1}{n} \right| \cdot \Big\| \sum_{e \in S} x_e \Big\|_2 + \frac{1}{n} \Big\| \sum_{e \in E \setminus S} x_e \Big\|_2 \ \le\ 2 \cdot \frac{n - |S|}{n} \ \le\ 2\beta.$$

Combining, $\mathbb{E}\|\hat\theta - \theta^*\|_2^2 \le 2\mathbb{E}\|Z\|_2^2 + 2\|\theta_S^* - \theta^*\|_2^2 \le C'(\beta^2 + c_\beta^2 d \log(1/\delta)/\varepsilon^2)$. □

Combining the display with Lemma 19, the achievable error optimized over β is $O(R(d\log(1/\delta)/\varepsilon^2))$, while Theorem 5 gives $\tilde{\Omega}(R(d/\varepsilon^2))$ under $\delta = o(c_{\beta^\star})$; by the scaling claim $R(LD) \le LR(D)$, the two match up to absolute constants and a $\log(1/\delta)$ factor on every instance. The refined network density profile therefore characterizes the achievable accuracy instance by instance.

C.5 A Family Where c_β Varies Sharply

Consider γn star hyperedges through a common center together with $(1-\gamma)n$ pairwise disjoint singleton hyperedges. For $\beta < \gamma$, any subset of size at least $(1-\beta)n$ contains at least $(\gamma-\beta)n$ star edges, and up to integer effects

$$c_\beta = \max\left\{\frac{\gamma-\beta}{1-\beta},\ \frac{1}{(1-\gamma)n+1}\right\};$$

for $\beta \ge \gamma$ the minimizing subset takes all singletons together with a single star edge, giving $c_\beta = 1/((1-\gamma)n+1)$, constant in β. The profile is monotone and changes by an unbounded factor under a constant-factor change of β near γ, which is exactly what invalidates any comparison of the lower and upper bounds at retention levels differing by a constant factor; the common-β characterization of Lemma 19 is unaffected.

Before the singleton floor is reached, the crossing of β^2 and $c_\beta^2 D$ solves $\beta(1-\beta) = \sqrt{D}(\gamma-\beta)$, giving

$$\beta^\star = \frac{1+\sqrt{D} - \sqrt{(1+\sqrt{D})^2 - 4\gamma\sqrt{D}}}{2},$$

which is $\gamma\sqrt{D}/(1+\sqrt{D})$ up to a factor $1+o(1)$ in the small-γ limit. The crossing lies on this branch, below γ, precisely when $D/n^2 \lesssim \gamma^2$, where the rate is $\Theta(\gamma^2)$ for $D \ge 1$; when $D/n^2 \gtrsim \gamma^2$ the crossing occurs on the floor branch and the rate is $\Theta(D/n^2)$. In all regimes, up to constants and truncation at one,

$$R(D) = \Theta\big(\max\{\gamma^2,\ D/n^2\}\big).$$

Both regimes can be verified without the general machinery. The center's data determines the contents of all γn star edges, and changing it moves the true mean by up to 2γ, so any fixed-graph DP algorithm with $\varepsilon = O(1)$ has $\alpha = \Omega(\gamma^2)$. The standard private mean-estimation lower bound, embedded in the $(1-\gamma)n$ disjoint singleton records, supplies the $\tilde{\Omega}(d/(\varepsilon^2 n^2))$ noise floor. For $0 < \varepsilon \le 1$ and $0 < \delta \le e^{-1}$, discarding the star and averaging the singletons with Gaussian noise achieves $\alpha = O(\gamma^2 + d\log(1/\delta)/(\varepsilon^2 n^2))$.

C.6 Pure-DP Upper Bound via the K-norm Mechanism

Theorem 20. *For every $\varepsilon > 0$ and $\beta \in (0,1]$ there exists an ε-fixed-graph DP algorithm achieving, for every $X \in \mathcal{X}^n$, $\mathbb{E}\|\mathcal{A}(X) - \frac{1}{n}\sum_e x_e\|_2^2 \le C'(\beta^2 + c_\beta^2 d^2/\varepsilon^2)$. Consequently the achievable pure-DP error is $O(R(d^2/\varepsilon^2))$, matching the pure-DP branch of Theorem 5 up to constants.*

Proof. Take the subset S from the proof of Theorem 6, with $|S| \ge (1-\beta)n$ and sensitivity $\Delta_2 = 2c_\beta$ for the subset mean, and release $\hat{\theta} = \theta^*_S + Z$ where Z has density proportional to $\exp(-\varepsilon\|z\|_2/\Delta_2)$. This is the K-norm mechanism of Hardt and Talwar [36] for the ℓ_2 ball scaled by Δ_2; for any adjacent pair the densities differ by a factor at most $\exp(\varepsilon\|\theta^*_S - \theta'^*_S\|_2/\Delta_2) \le e^\varepsilon$ by the triangle inequality, so the mechanism is ε-DP. In polar coordinates the radius follows a Gamma distribution with shape d and scale Δ_2/ε, whence the exact second moment $\mathbb{E}\|Z\|_2^2 = d(d+1)(\Delta_2/\varepsilon)^2 = O(c_\beta^2 d^2/\varepsilon^2)$. Combining with the bias bound $\|\theta^*_S - \theta^*\|_2 \le 2\beta$ as before gives the display, and optimizing over β gives $O(R(d^2/\varepsilon^2))$ by Lemma 19. □

C.7 Peeling and Public Cap Selection: Proof of Theorem 8

Proof of Theorem 8. Throughout this proof, $0 < \varepsilon \le 1$ and $0 < \delta \le e^{-1}$.

Peeling. While some vertex v has degree greater than k, delete $\deg(v) - k$ of its incident hyperedges arbitrarily. Each vertex is processed at most once, since after processing its degree equals k and

degrees never increase; the deletions at v number at most $(\deg_E(v) - k)^+$ in the original hypergraph, giving the additive bound $B_k = \sum_v (\deg_E(v) - k)^+$. For the approximation factor, consider the potential $\Phi(F) = \sum_v (\deg_F(v) - k)^+$. Any feasible solution must reduce Φ from B_k to 0, and deleting a single hyperedge decreases Φ by at most $|e| \le r$, so the minimum deletion number is at least B_k/r.

Cap $\lceil c_\beta n \rceil$. Fix β with $r\beta \le 1/2$. By definition of c_β there is a subset of size $m^\star \ge (1-\beta)n$ with maximum degree $c_\beta m^\star \le c_\beta n \le \lceil c_\beta n \rceil$, so the minimum deletion number at this cap is at most βn, and peeling deletes at most $r\beta n$, retaining at least $(1-r\beta)n \ge n/2$ hyperedges with maximum degree at most $\lceil c_\beta n \rceil$. The subset mean then has sensitivity at most $2\lceil c_\beta n \rceil / ((1-r\beta)n) \le 4c_\beta + O(1/n)$, and the Gaussian mechanism as in Theorem 6 gives bias at most $2r\beta$ and noise $O(c_\beta^2 d \log(1/\delta)/\varepsilon^2)$, hence error at most $C''(r^2\beta^2 + c_\beta^2 d \log(1/\delta)/\varepsilon^2)$.

Public cap selection. The algorithm cannot compute c_β. Instead it first runs peeling once at every integer cap $k \le \Delta$, forming every candidate S_k and its public score, then selects the candidate with the smallest score before any example contents are accessed. The score of the cap-k candidate is $\widehat{\mathrm{err}}(k) = 4((n - |S_k|)/n)^2 + 8(k/|S_k|)^2 d \ln(1.25/\delta)/\varepsilon^2$, and the zero estimator (output 0) enters as one further candidate with score 1. These constants make the surrogate a valid upper bound on the true error of each candidate: the deletion bias satisfies $\|\theta^*_{S_k} - \theta^*\|_2 \le 2(n - |S_k|)/n$, so its square is at most the first term; the Gaussian mechanism at sensitivity $2k/|S_k|$ uses $\sigma = (2k/|S_k|)\sqrt{2\ln(1.25/\delta)}/\varepsilon$ under the stated assumptions , contributing noise exactly $\sigma^2 d$, the second term; and the zero estimator's true error is $\|\theta^*\|_2^2 \le 1$, its score. All scores are functions of the public structure and the cap alone, so the selection consumes no privacy budget, and the single noisy mean released on $S_{\widehat{k}}$ is the only differentially private output; each peeling run is linear time and there are at most Δ runs. At the candidate $k = \lceil c_\beta n \rceil$ the realized deletion fraction is at most $r\beta$ and the realized sensitivity ratio is at most $4c_\beta + O(1/n)$, so the selected minimum is at most $O(r^2\beta^2 + c_\beta^2 d \log(1/\delta)/\varepsilon^2)$ for every β with $r\beta \le 1/2$. If the optimizing $\beta^\star$ of Lemma 19 satisfies $\beta^\star \ge 1/(2r)$, then $R(D) \ge (\beta^\star)^2 \ge 1/(4r^2)$ when the curves cross and $R(D) = 1/4 \ge 1/(4r^2)$ otherwise, so the zero estimator, with error at most one, is within a factor $4r^2$ of optimal there. Altogether the selected candidate matches the noncomputational bound of Theorem 6 up to an r^2 factor, and therefore, for bounded r, matches $R(d\log(1/\delta)/\varepsilon^2)$ up to constants and the same logarithmic factors. □

C.8 Contribution-Bounding Error Bounds

C.8.1 Conditional Error and Privacy

We first prove the conditional identity in Theorem 7, then apply the retention guarantees of the two contribution-bounding algorithms. Throughout this subsection, $E \ne \emptyset$, $k \ge 1$ is an integer, $0 < \varepsilon \le 1$, and $0 < \delta \le e^{-1}$. The data satisfy $\|x_e\|_2 \le 1$, and $\theta^* = n^{-1}\sum_{e\in E} x_e$.

Theorem (Theorem 7, conditional form)**.** *Let $S \ne \emptyset$ be a random subset chosen using only the public hypergraph and randomness independent of the example contents. Suppose $\deg_S(v) \le k$ for every vertex. Conditional on S, release*

$$\widehat{\theta} = \theta^*_S + Z, \qquad \theta^*_S = \frac{1}{|S|}\sum_{e\in S} x_e, \qquad Z \sim \mathcal{N}(0, \sigma_S^2 I_d), \qquad \sigma_S^2 = \frac{8k^2 \ln(1.25/\delta)}{|S|^2\varepsilon^2}.$$

The release is (ε, δ)-fixed-graph DP and satisfies

$$\mathbb{E}_Z\left[\|\widehat{\theta} - \theta^*\|_2^2 \mid S\right] = \frac{8k^2 d \ln(1.25/\delta)}{|S|^2\varepsilon^2} + \|\theta^*_S - \theta^*\|_2^2,$$

$$\mathbb{E}_{S,Z}\|\widehat{\theta} - \theta^*\|_2^2 = \frac{8k^2 d \ln(1.25/\delta)}{\varepsilon^2}\,\mathbb{E}_S\left[\frac{1}{|S|^2}\right] + \mathbb{E}_S\|\theta^*_S - \theta^*\|_2^2.$$

If $k \ge \Delta$, use $S = E$ and calibrate to the actual degree cap Δ. The bias is zero and the error is $8\Delta^2 d \ln(1.25/\delta)/(n^2\varepsilon^2)$.

Proof. For any fixed realization of the public selection randomness, adjacent datasets yield the same S. Changing one user's data changes at most k selected examples, each by Euclidean distance at most

two, so the selected mean has sensitivity at most $2k/|S|$. The stated Gaussian calibration therefore ensures (ε, δ)-fixed-graph privacy conditional on this randomness. Averaging the privacy inequalities over the same data-independent randomness proves privacy of the full mechanism, including when several candidate subsets are generated and the largest is selected.

Conditional on S, the noise has mean zero. Expanding the square, its cross term with $\theta^*_S - \theta^*$ has expectation zero, while $\mathbb{E}_Z\|Z\|_2^2 = d\sigma_S^2$. This gives the first identity. Averaging over S gives the second. When the entire hypergraph is feasible, $S = E$ and $\theta^*_S = \theta^*$. □

For larger ε, the same conditional identity holds with an analytically calibrated Gaussian variance [5]; the displayed $1/\varepsilon^2$ rates below are asserted only in the stated range. Other example-level private learners can also follow the public selection. An (ε', δ')-DP learner satisfies $(k\varepsilon', ke^{(k-1)\varepsilon'}\delta')$-fixed-graph DP by group privacy, since a user's data can change at most k selected examples. The bounds here concern the Gaussian mean release, whose sensitivity incorporates this factor directly.

C.8.2 Error Bound Using LP-Guided Greedy

Theorem 21 (Upper Bounds via LP-Guided Contribution Bounding)**.** *Use Algorithm 1 at an integer cap $1 \le k \le \Delta$, followed by the Gaussian release above. If $k = \Delta$, the algorithm returns E and has zero selection bias and error $8\Delta^2 d\ln(1.25/\delta)/(n^2\varepsilon^2)$.*

Suppose $k < \Delta$, and write

$$h = |V_{>k}| > 0, \qquad A_k = \min\{r^{1/k}, h^{1/(k+1)}\}, \qquad M_k = \mathrm{OPT}_{\mathrm{IP}}.$$

For any $\zeta \in (0,1)$, keep the largest subset S from

$$R \ge \max\left\{1, \left\lceil \frac{\log(1/\zeta)}{\log(10/9)} \right\rceil \right\}$$

independent traversals, using the same optimal LP solution. There is an absolute constant $c_0 > 0$ such that $|S| \ge c_0 M_k/A_k$ with probability at least $1 - \zeta$. For every selected subset satisfying this event,

$$\mathbb{E}_Z\left[\|\widehat{\theta} - \theta^*_S\|_2^2 \mid S\right] = O\left(\frac{k^2 d\log(1/\delta)\, A_k^2}{\varepsilon^2 M_k^2}\right).$$

Without conditioning on success, the noise satisfies

$$\mathbb{E}_{S,Z}\|\widehat{\theta} - \theta^*_S\|_2^2 \le \frac{8k^2 d\ln(1.25/\delta)}{\varepsilon^2 M_k^2}\left(c_0^{-2}A_k^2 + \zeta r^2\right).$$

In particular, $R = O(1 + \log(r/A_k))$ independent traversals suffice for the unconditional bound

$$\mathbb{E}_{S,Z}\|\widehat{\theta} - \theta^*_S\|_2^2 = O\left(\frac{k^2 d\log(1/\delta)}{\varepsilon^2 M_k^2}\min\{r^{2/k}, h^{2/(k+1)}\}\right).$$

The total error adds $\mathbb{E}_S\|\theta^*_S - \theta^*\|_2^2$ to the noise term. If $A_k = O(1)$ and $M_k = \Omega(n)$, the noise bound simplifies to $O(k^2 d\log(1/\delta)/(\varepsilon^2 n^2))$.

Proof. Theorem 1 gives a success probability of at least $1/10$ for each traversal, conditional on any fixed optimal LP solution. Independent traversal randomness therefore makes the probability that every run fails at most $(9/10)^R \le \zeta$. Keeping the largest output preserves the successful retention guarantee. All of these operations depend only on public structure and randomness, so the privacy argument above applies without composition across runs: only one noisy mean is released.

Every complete traversal, including its final pass over zero-weight edges, produces a maximal feasible set. The greedy bound therefore gives $|S| \ge M_k/r$ on every realization. Let $\mathcal{G} = \{|S| \ge c_0 M_k/A_k\}$. Separating its complement yields

$$\mathbb{E}_S\left[\frac{1}{|S|^2}\right] \le \frac{A_k^2}{c_0^2 M_k^2}\Pr(\mathcal{G}) + \frac{r^2}{M_k^2}\Pr(\mathcal{G}^c) \le \frac{c_0^{-2}A_k^2 + \zeta r^2}{M_k^2}.$$

Substituting in the conditional error identity proves both noise bounds. Since $1 \le A_k \le r$, choosing

$$R \ge \max\left\{1, \left\lceil \frac{2\log(r/A_k)}{\log(10/9)} \right\rceil \right\}$$

ensures $(9/10)^R r^2 \le A_k^2$ and proves the unconditional bound of the stated order. This argument controls the inverse-square subset size directly; it does not infer it from expected cardinality. □

The condition $A_k = O(1)$ holds, for example, when $k = \Omega(\log r)$ or $h = 2^{O(k)}$. The approximation controls the noise added at a fixed cap. It does not by itself control the selection-bias term, which depends on the retained examples as described in Section 4.4.

C.8.3 Error Bound Using Greedy

Theorem 22 (Upper Bounds via Greedy Contribution Bounding)**.** *Use Algorithm 2 at an integer cap $1 \le k \le \Delta$, followed by the Gaussian release above. The mechanism is (ε, δ)-fixed-graph DP. If $k = \Delta$, it returns E and has zero selection bias and error $8\Delta^2 d \ln(1.25/\delta)/(n^2\varepsilon^2)$. If $k < \Delta$, its noise satisfies*

$$\mathbb{E}_{S,Z}\|\widehat{\theta} - \theta^*_S\|_2^2 \le \frac{8k^2 r^2 d \ln(1.25/\delta)}{\varepsilon^2 \mathrm{OPT}_{\mathrm{IP}}^2}.$$

The expectation over S is unnecessary for a fixed processing order; the bound also holds for any random order independent of the example contents. The total error adds $\mathbb{E}_S\|\theta^*_S - \theta^*\|_2^2$ to the noise term.

Proof. Theorem 2 gives $|S| \ge \mathrm{OPT}_{\mathrm{IP}}/r$ for every order, so substituting this deterministic bound in the conditional error identity proves the claim. If $k = \Delta$, all edges are accepted and the bias is zero. □

When $\mathrm{OPT}_{\mathrm{IP}} = \Omega(n)$, the greedy noise bound is $O(k^2 r^2 d \log(1/\delta)/(\varepsilon^2 n^2))$. If every edge shares one user and $k < \Delta$, then $\mathrm{OPT}_{\mathrm{IP}} = k$ and greedy selects exactly k edges. Its calibrated sensitivity is then two, so the Gaussian noise contribution is $8d \ln(1.25/\delta)/\varepsilon^2$, independent of n. This is the error of the stated Gaussian release; the zero estimator supplies the trivial total-error bound of one when this noise level is large.

D Characterizing Selection Bias

In the upper bounds of Section 4.2, contribution bounding selects a subset $S \subseteq E$ using *only* the public structure (V, E) (and internal randomness), never the example contents $\{x_e\}$. Thus, any bias can only arise from *correlation between network structure and the data distribution*. In the single-attribution setting, the bias introduced by bounding user contributions is well documented [4, 29]; the multi-attribution setting introduces a distinct mechanism, namely correlation between hypergraph structure and data, which we now formalize.

Two viewpoints: optimization vs. population. Our privacy and optimization guarantees hold for an arbitrary fixed dataset $\{x_e\}_{e\in E}$. Selection bias becomes relevant only when we interpret performance relative to a *target* risk (e.g., population risk). To make this explicit, we define bias for a generic target risk $L(\theta)$, which can be instantiated as: (i) the empirical risk on all edges E, or (ii) a population risk under a distribution $\mathcal{D}$.

Structural data model Throughout the paper, the hypergraph (V, E) is treated as fixed, and our primary guarantees are stated for a fixed dataset $\{x_e\}_{e\in E}$ via an optimization problem. In this subsection, we introduce a *data-generating model* only to formalize *selection-induced distribution shift*. Fix a *type map* $t : E \to \{1, \dots, J\}$ that depends only on (V, E), e.g., $t(e) = |e|$ or a statistic of the degrees of the vertices in e. Conditional on (V, E), each hyperedge $e \in E$ carries an example x_e drawn independently from a distribution that may depend on $t(e)$:

$$x_e \mid t(e) = j \sim \mathcal{D}_j, \qquad \{x_e\}_{e\in E} \text{ independent given } (V, E).$$

Let μ be the base edge distribution (typically uniform on E), and define the target (population) risk

$$L(\theta) \;:=\; \mathbb{E}_{e\sim\mu}\, \mathbb{E}_{x\sim\mathcal{D}_{t(e)}}[\ell(\theta; x)] \;=\; \sum_{j=1}^{J} q_j\, L_j(\theta), \qquad L_j(\theta) := \mathbb{E}_{x\sim\mathcal{D}_j}[\ell(\theta; x)],$$

where $q_j := \mu\{e : t(e) = j\}$ are the population mixture weights over types. Let $\theta^\star \in \arg\min_\theta L(\theta)$ be the *population* minimizer.

Selection induces a new edge distribution. A contribution-bounding/selection algorithm $\mathcal{A}$ outputs a (random) subset $S \subseteq E$. This induces a new sampling distribution μ_S over hyperedges (e.g., μ_S uniform on S when conditioning on S), and hence a *selected* population risk

$$L_S(\theta) \;=\; \mathbb{E}_{e\sim\mu_S}\,\mathbb{E}_{x\sim\mathcal{D}_{t(e)}}\big[\ell(\theta;x)\big], \qquad \theta_S^\star \in \arg\min_\theta L_S(\theta).$$

The empirical ERM on the selected sample is

$$\hat\theta_S \in \arg\min_\theta \frac{1}{|S|}\sum_{e\in S}\ell(\theta;x_e).$$

Definition 12 (Selection Bias vs. Estimation Error)**.** *We define the* selection bias *as the excess* population *risk caused by optimizing for the selected distribution:*

$$\mathrm{Bias}_{\mathrm{sel}}(S) \;:=\; L(\theta_S^\star) - L(\theta^\star) \ge 0.$$

This isolates distribution shift due to selection. The total excess population risk of the learned parameter $\hat\theta_S$ decomposes as

$$L(\hat\theta_S) - L(\theta^\star) \;=\; \underbrace{\big(L(\theta_S^\star) - L(\theta^\star)\big)}_{\textit{selection bias}} + \underbrace{\big(L(\hat\theta_S) - L(\theta_S^\star)\big)}_{\textit{estimation/generalization error}}.$$

In particular, even when $\mathrm{Bias}_{\mathrm{sel}}(S) = 0$, the second term is generally nonzero for finite $|S|$.

Theorem 23 (Concrete conditions for zero or small selection bias)**.** *Fix a type map $t : E \to \{1,\dots,J\}$ depending only on (V,E) and a base edge distribution μ (uniform on E unless stated). Let $q_j := \mu\{e : t(e) = j\}$ and $p_j := \mu_S\{e : t(e) = j\}$ be the population and selected type weights. Assume conditional independence $x_e \mid t(e) = j \sim \mathcal{D}_j$ and define $L(\theta) = \sum_j q_j L_j(\theta)$, $L_S(\theta) = \sum_j p_j L_j(\theta)$ with $L_j(\theta) = \mathbb{E}_{x\sim\mathcal{D}_j}[\ell(\theta;x)]$. Let $\theta^\star \in \arg\min_\theta L(\theta)$ and $\theta_S^\star \in \arg\min_\theta L_S(\theta)$, and define $\mathrm{Bias}_{\mathrm{sel}}(S;\mathcal{D}) := L(\theta_S^\star) - L(\theta^\star)$.*

- *(a)* ***Exchangeable / type-independent examples $\Rightarrow$ zero bias.*** *If $\mathcal{D}_j \equiv \mathcal{D}$ for all j, then $L_j(\theta) \equiv L(\theta)$ for all j. Thus, $\mathrm{Bias}_{\mathrm{sel}}(S;\mathcal{D}) = 0$ for* any *selection rule $\mathcal{A}$ that is independent of $\{x_e\}$. (This includes the common case "examples independent of hyperedge size" by taking $t(e) = |e|$.)*
- *(b)* ***Type-proportion preservation $\Rightarrow$ zero bias.*** *If the selection rule satisfies $p = q$ (i.e., it preserves the mixture weights over types), then $L_S(\theta) \equiv L(\theta)$ for all θ and $\mathrm{Bias}_{\mathrm{sel}}(S;\mathcal{D}) = 0$ regardless of heterogeneity among the $\mathcal{D}_j$.*
- *(c)* ***Single-type networks.*** *If $t(e)$ is constant over E (e.g., perfectly regular/structured instances under the chosen t), then $p = q = (1,0,\dots,0)$ and bias is zero regardless of $\mathcal{A}$.*
- *(d)* ***Mixture-shift bound (general, distributional).*** *Let $\mathcal{D} := \sum_j q_j \mathcal{D}_j$ be the population mixture. If loss $\ell(\theta;x) \in [0,1]$ is bounded for all θ, x, then for each j and all θ, $|L_j(\theta) - L(\theta)| \le 2\,\mathrm{TV}(\mathcal{D}_j,\mathcal{D})$, and hence*

$$\mathrm{Bias}_{\mathrm{sel}}(S;\mathcal{D}) \;\le\; 4\sum_{j=1}^{J}|p_j - q_j|\,\mathrm{TV}(\mathcal{D}_j,\mathcal{D}).$$

  This makes the mechanism explicit: bias is small when (i) the selection does not heavily skew type proportions ($p \approx q$), or (ii) the type-conditioned data distributions are similar ($\mathcal{D}_j \approx \mathcal{D}$).
- *(e)* ***Mixture-shift bound (parameter-space, strongly convex).*** *Assume each $L_j(\theta)$ is λ-strongly convex and β-smooth (in this theorem β denotes the smoothness constant). Let $\theta_j^\star \in \arg\min_\theta L_j(\theta)$ and suppose $\|\theta_j^\star - \theta^\star\|_2 \le B$ for all j. Then*

$$\|\theta_S^\star - \theta^\star\|_2 \;\le\; \frac{\beta B\|p-q\|_1}{\lambda},$$

  and consequently

$$\mathrm{Bias}_{\mathrm{sel}}(S;\mathcal{D}) \;\le\; \frac{\beta}{2}\|\theta_S^\star - \theta^\star\|_2^2 \;\le\; \frac{\beta^3 B^2\|p-q\|_1^2}{2\lambda^2}.$$

(f) ***Lipschitz risk in parameters (plug-in).*** *If* $|L(\theta) - L(\theta')| \le L_\theta \|\theta - \theta'\|_2$ *for all* θ, θ' *and* $\|\theta_S^\star - \theta^\star\|_2 \le B$*, then*

$$\mathrm{Bias}_{\mathrm{sel}}(S; \mathcal{D}) \le L_\theta B.$$

(g) ***Degree-correlated data (explicit instantiation).*** *Let* $t(e)$ *encode a degree statistic (e.g.* $t(e) = \max_{v \in e} \deg(v)$ *or a binned version). If* $\mathrm{TV}(\mathcal{D}_t, \mathcal{D}) \le \delta(t)$*, then the TV mixture bound in (d) gives*

$$\mathrm{Bias}_{\mathrm{sel}}(S; \mathcal{D}) \;\le\; 4 \sum_t |p(t) - q(t)|\, \delta(t).$$

In particular, if $\delta(t)$ *grows with degree and* $\mathcal{A}$ *downweights large-*t *edges, the bias scales with the resulting type-distortion* $|p - q|$*.*

See Appendix D.2 for a proof.

Connection to the upper bound decomposition. In the i.i.d. setting, the excess population risk naturally decomposes into (i) optimization/noise error on the selected sample and (ii) the distribution-shift term $\mathrm{Bias}_{\mathrm{sel}}(S; \mathcal{D})$ from Definition 12. In the purely optimization viewpoint, one may instead take $L(\theta)$ to be the empirical risk on all edges E (and L_S the empirical risk on S), yielding an analogous deterministic "selection gap" definition without introducing $\mathcal{D}$. Either way, the only way S can introduce bias is by changing which structural types are represented, since S is chosen independently of $\{x_e\}$ under fixed-graph DP.

Remark 7 (Open Questions)**.** *Several important questions remain open. Can we design contribution bounding algorithms that minimize selection bias? Is there a fundamental tradeoff between privacy, utility, and selection bias in multi-attribution settings? Can importance weighting or other debiasing techniques reduce selection bias without violating privacy? How does selection bias interact with non-convex optimization in deep learning?*

D.1 Exact Type Preservation versus Retention

Proposition 24 (Exact type preservation versus retention)**.** *Let types be hyperedge sizes. On the network with* γn *star hyperedges* $\{v_c, v_i\}$ *and* $(1-\gamma)n$ *pairwise disjoint singleton hyperedges, every feasible selection* S *at cap* k *contains at most* k *star hyperedges, so if* S *preserves the type proportions exactly (*$p = q$*) then* $|S| \le k/\gamma$*, whereas the unconstrained feasible optimum has* $(1-\gamma)n + k$ *hyperedges. Conversely, any feasible* S *with type counts* s_j *can be made type-preserving, up to rounding, by subsampling type* j *to* $\rho\, q_j |S|$ *hyperedges with* $\rho = \min_j s_j/(q_j|S|)$*; the result is feasible, uses only structural information, and retains* $\rho|S| = \min_j s_j/q_j$ *hyperedges.*

Proof. All star hyperedges contain v_c, so a feasible S contains at most k of them, and $p_{\mathrm{star}} = |S \cap \mathrm{star}|/|S| \le k/|S|$; exact preservation $p_{\mathrm{star}} = q_{\mathrm{star}} = \gamma$ forces $|S| \le k/\gamma$. Taking all singletons together with k star hyperedges is feasible, which gives the stated optimum. For the converse, subsets of feasible selections are feasible, the subsampling reads only structural types, and the retained size is $\sum_j \rho q_j |S| = \rho|S|$. □

Exact preservation thus costs a factor of order $\gamma(1-\gamma)n/k$ in retention on this family, and the natural target is approximate preservation at a quantified retention cost; whether a single selection rule can be simultaneously near-optimal in approximation ratio and in the bias bound of Theorem 23 remains open in general.

D.2 Proof of Characterizing Selection Bias (Theorem 23)

Proof of Theorem 23, part (d). We begin by bounding $|L_j(\theta) - L(\theta)|$ using total variation. Recall that the total variation distance between distributions P and Q satisfies:

$$|\mathbb{E}_{x \sim P}[f(x)] - \mathbb{E}_{x \sim Q}[f(x)]| \le 2\|f\|_\infty \cdot \mathrm{TV}(P, Q)$$

for any measurable function f.

Since $\ell(\theta; x) \in [0,1]$ for all θ, x, we have $\|\ell(\theta;\cdot)\|_\infty \le 1$. Applying this with $P = \mathcal{D}_j$ and $Q = \mathcal{D} = \sum_k q_k \mathcal{D}_k$:

$$\begin{aligned} |L_j(\theta) - L(\theta)| &= \left|\mathbb{E}_{x\sim\mathcal{D}_j}[\ell(\theta;x)] - \mathbb{E}_{x\sim\mathcal{D}}[\ell(\theta;x)]\right| \\ &\le 2\,\mathrm{TV}(\mathcal{D}_j, \mathcal{D}). \end{aligned} \tag{17}$$

To express the difference $L(\theta) - L_S(\theta)$, note that for any θ:

$$\begin{aligned} L(\theta) - L_S(\theta) &= \sum_j q_j L_j(\theta) - \sum_j p_j L_j(\theta) \\ &= \sum_j (q_j - p_j) L_j(\theta) \\ &= \sum_j (q_j - p_j)[L_j(\theta) - L(\theta)] + L(\theta)\sum_j (q_j - p_j). \end{aligned} \tag{18}$$

Since $\sum_j q_j = \sum_j p_j = 1$, the second term vanishes:

$$L(\theta) - L_S(\theta) = \sum_j (q_j - p_j)[L_j(\theta) - L(\theta)].$$

Therefore:

$$\begin{aligned} |L(\theta) - L_S(\theta)| &\le \sum_j |q_j - p_j| \cdot |L_j(\theta) - L(\theta)| \\ &\le 2\sum_j |q_j - p_j| \cdot \mathrm{TV}(\mathcal{D}_j, \mathcal{D}). \end{aligned} \tag{19}$$

We now apply this inequality to bound the selection bias. Since $\theta_S^* = \arg\min L_S$ and $\theta^* = \arg\min L$:

$$\begin{aligned} \mathrm{Bias}_{\mathrm{sel}}(S;\mathcal{D}) &= L(\theta_S^*) - L(\theta^*) \\ &= [L(\theta_S^*) - L_S(\theta_S^*)] + [L_S(\theta_S^*) - L_S(\theta^*)] + [L_S(\theta^*) - L(\theta^*)] \\ &\le [L(\theta_S^*) - L_S(\theta_S^*)] + [L_S(\theta^*) - L(\theta^*)], \end{aligned} \tag{20}$$

where the inequality uses $L_S(\theta_S^*) \le L_S(\theta^*)$ by optimality.

Applying our bound from above to both terms, we get

$$\begin{aligned} \mathrm{Bias}_{\mathrm{sel}}(S;\mathcal{D}) &\le |L(\theta_S^*) - L_S(\theta_S^*)| + |L_S(\theta^*) - L(\theta^*)| \\ &\le 2\sum_j |q_j - p_j| \mathrm{TV}(\mathcal{D}_j, \mathcal{D}) + 2\sum_j |q_j - p_j| \mathrm{TV}(\mathcal{D}_j, \mathcal{D}) \\ &= 4\sum_{j=1}^{J} |p_j - q_j| \mathrm{TV}(\mathcal{D}_j, \mathcal{D}). \end{aligned} \tag{21}$$

□

Proof of Theorem 23, part (e). We begin by bounding $\|\theta_S^* - \theta^*\|_2$ using first-order conditions. Since $\theta^* = \arg\min_\theta L(\theta)$ and $\theta_S^* = \arg\min_\theta L_S(\theta)$, the first-order optimality conditions give:

$$\nabla L(\theta^*) = \sum_j q_j \nabla L_j(\theta^*) = 0, \quad \nabla L_S(\theta_S^*) = \sum_j p_j \nabla L_j(\theta_S^*) = 0.$$

By λ-strong convexity of $L_S(\theta)$:

$$\|\nabla L_S(\theta_S^*) - \nabla L_S(\theta^*)\|_2 \ge \lambda \|\theta_S^* - \theta^*\|_2.$$

Since $\nabla L_S(\theta_S^*) = 0$:

$$\|\nabla L_S(\theta^*)\|_2 \geq \lambda \|\theta_S^* - \theta^*\|_2.$$

Now, express $\nabla L_S(\theta^*)$ using the mixture weights:

$$\begin{aligned}
\nabla L_S(\theta^*) &= \sum_j p_j \nabla L_j(\theta^*) \\
&= \sum_j p_j \nabla L_j(\theta^*) - \sum_j q_j \nabla L_j(\theta^*) \quad \text{(using } \sum_j q_j \nabla L_j(\theta^*) = 0\text{)} \\
&= \sum_j (p_j - q_j) \nabla L_j(\theta^*). \qquad (22)
\end{aligned}$$

This gives:

$$\|\theta_S^* - \theta^*\|_2 \leq \frac{1}{\lambda} \left\| \sum_j (p_j - q_j) \nabla L_j(\theta^*) \right\|_2. \qquad (23)$$

Next, we bound the gradient term using smoothness. By triangle inequality:

$$\left\| \sum_j (p_j - q_j) \nabla L_j(\theta^*) \right\|_2 \leq \sum_j |p_j - q_j| \|\nabla L_j(\theta^*)\|_2.$$

Since $\nabla L_j(\theta_j^*) = 0$ and each L_j is β-smooth:

$$\|\nabla L_j(\theta^*)\|_2 = \|\nabla L_j(\theta_j^*) - \nabla L_j(\theta^*)\|_2 \leq \beta \|\theta_j^* - \theta^*\|_2.$$

Therefore:

$$\left\| \sum_j (p_j - q_j) \nabla L_j(\theta^*) \right\|_2 \leq \beta \sum_j |p_j - q_j| \cdot \|\theta_j^* - \theta^*\|_2. \qquad (24)$$

Using $\|\theta_j^* - \theta^*\|_2 \leq B$:

$$\|\theta_S^* - \theta^*\|_2 \leq \frac{\beta}{\lambda} \sum_j |p_j - q_j| \cdot \|\theta_j^* - \theta^*\|_2 \leq \frac{\beta B}{\lambda} \|p - q\|_1.$$

Finally, we bound the selection bias. Since L is a convex combination of β-smooth functions L_j, it is also β-smooth. By the smoothness property:

$$L(\theta_S^*) \leq L(\theta^*) + \nabla L(\theta^*)^T (\theta_S^* - \theta^*) + \frac{\beta}{2} \|\theta_S^* - \theta^*\|_2^2.$$

Since $\nabla L(\theta^*) = 0$:

$$\text{Bias}_{\text{sel}}(S; \mathcal{D}) = L(\theta_S^*) - L(\theta^*) \leq \frac{\beta}{2} \|\theta_S^* - \theta^*\|_2^2.$$

Substituting our bound:

$$\text{Bias}_{\text{sel}}(S; \mathcal{D}) \leq \frac{\beta}{2} \left(\frac{\beta B}{\lambda} \|p - q\|_1 \right)^2 = \frac{\beta^3 B^2 \|p - q\|_1^2}{2\lambda^2}.$$

□

Proof of Theorem 23, part (f). By assumption, $|L(\theta) - L(\theta')| \le L_\theta \|\theta - \theta'\|_2$ for all θ, θ'. Applying this with $\theta = \theta_S^*$ and $\theta' = \theta^*$:

$$|L(\theta_S^*) - L(\theta^*)| \le L_\theta \|\theta_S^* - \theta^*\|_2.$$

Since $\mathrm{Bias}_{\mathrm{sel}}(S; \mathcal{D}) = L(\theta_S^*) - L(\theta^*) \ge 0$ (by optimality of θ^* for L):

$$\mathrm{Bias}_{\mathrm{sel}}(S; \mathcal{D}) \le L_\theta \|\theta_S^* - \theta^*\|_2.$$

If additionally $\|\theta_S^* - \theta^*\|_2 \le B$:

$$\mathrm{Bias}_{\mathrm{sel}}(S; \mathcal{D}) \le L_\theta B.$$

□

Proof of Theorem 23, part (g). This is a direct application of part (d) with $t(e)$ encoding a degree statistic (e.g., $t(e) = \max_{v \in e} \deg(v)$ or a binned version).

By the assumption $\mathrm{TV}(\mathcal{D}_t, \mathcal{D}) \le \delta(t)$ and part (d):

$$\mathrm{Bias}_{\mathrm{sel}}(S; \mathcal{D}) \le 4 \sum_t |p(t) - q(t)| \mathrm{TV}(\mathcal{D}_t, \mathcal{D}) \le 4 \sum_t |p(t) - q(t)| \delta(t),$$

where $p(t) = \mu_S\{e : t(e) = t\}$ and $q(t) = \mu\{e : t(e) = t\}$ are the selected and population proportions of edges of type t.

The bound makes explicit how bias scales with: (1) the degree-dependent distribution shift $\delta(t)$, and (2) the change in type proportions $|p(t) - q(t)|$ induced by the selection algorithm. If the algorithm downweights high-degree edges (where $\delta(t)$ is large), the bias is controlled by the resulting type distortion $\|p - q\|_1$. □

E Improved Bounds for i.i.d. Training Examples

E.1 Results and Algorithms

In the previous sections, we established bounds for arbitrary worst-case datasets under fixed-graph differential privacy. We now consider a fundamentally different setting where training examples are sampled independently from a distribution $\mathcal{D}$. This distributional assumption is weaker than requiring guarantees for all possible datasets, including adversarially chosen outliers with negligible probability. Under this assumption, the privacy lower bound has a weaker dependence on the network density. We establish bounds scaling as $\sqrt{\frac{c}{n}}$ rather than c, a polynomial improvement for $c \gg \frac{1}{n}$.

To illustrate this phenomenon in its simplest form, we analyze binary hypothesis testing; private simple hypothesis testing is well understood in the standard single-attribution setting [15], and our focus is on how network structure changes the achievable error. The input space is $\mathcal{X} = \{0, 1\}$ with two possible underlying distributions $\mathcal{D}_1 = \mathrm{Ber}(\frac{1}{2} - \beta)$ and $\mathcal{D}_2 = \mathrm{Ber}(\frac{1}{2} + \beta)$. The goal is to design a fixed-graph DP algorithm $\mathcal{M} : \mathcal{X}^n \to \{1, 2\}$ that correctly identifies the distribution. Formally, for any $i \in \{1, 2\}$, we require $\mathbb{E}_{X \sim \mathcal{D}_i^n} \mathbb{E}_{\mathcal{M}}[\mathbf{1}\{\mathcal{M}(X) \ne i\}] \le \alpha$.

Lower bound. Network density c still characterizes fundamental limits, but with improved scaling.
Theorem 25 (Lower Bound for i.i.d. Hypothesis Testing)**.** *Assume $\beta < 0.4$. Any $(\varepsilon, 0)$-fixed-graph DP mechanism $\mathcal{M} : \mathcal{X}^n \to \{1, 2\}$ satisfies $\alpha \ge \Omega\left(\exp\left(-60\varepsilon\beta\sqrt{\frac{n \log(n+1)}{c}}\right)\right)$.*

The proof (Appendix E.2) introduces a distance metric $d(X, Y)$ measuring the minimum number of vertices whose data must change to transform dataset X into Y. The key technical step is constructing a coupling between $\mathcal{D}_1^n$ and $\mathcal{D}_2^n$ with $\mathbb{E}[d(X, Y)] \le \sqrt{n \log(n)/c}$ by partitioning examples into $k \approx 1/c$ groups sharing common vertices (Lemma 29). Applying Le Cam's method with group privacy composition yields the $\sqrt{c/n}$ scaling. A second, purely statistical constraint applies to every mechanism, private or not: by the Bretagnolle–Huber inequality, any test based on n i.i.d. samples satisfies $\alpha \ge \frac{1}{4} e^{-n \mathrm{KL}(\mathcal{D}_1 \| \mathcal{D}_2)} \ge \exp(-O(n\beta^2))$ for $\beta \le 0.4$ (Lemma 31). The operative lower bound is the maximum of the privacy term of Theorem 25 and this sampling term.

Matching upper bound for clustered networks. For networks with specific structure, we achieve matching bounds. Consider a clustered network whose k groups have pairwise disjoint vertex sets, each group consisting of $n_i = n/k$ examples through a common hub vertex; the disjointness means every user's examples lie within a single group. Such networks have $c = 1/k$, so the privacy lower bound predicts error $\tilde{\Omega}(\exp(-\Theta(\varepsilon\beta\sqrt{nk})))$. We show that the maximum of the privacy and sampling lower bounds is achievable.

Algorithm 4 Group-Based Private Hypothesis Testing

Require: Network with k groups $G_1, \ldots, G_k$; examples $\{x_e\}_{e \in E}$; privacy ε

1: Compute $S_j = \mathbf{1}\left\{|\{e \subseteq G_j \mid x_e = 0\}| \le \frac{n}{2k}\right\}$ for $1 \le j \le k$

2: Compute $\hat{i} = \mathbf{1}\left\{\sum_{j=1}^{k} S_j + \text{Lap}\left(\mu = 0, b = \frac{1}{\varepsilon}\right) > \frac{k}{2}\right\} + 1$

3: **return** $\hat{i}$

Theorem 26 (Matching Upper Bound for Clustered Networks)**.** *Assume the k groups have pairwise disjoint vertex sets, so that every user's examples lie within a single group. There exist absolute constants $C_1, C_2 > 0$ such that if $C_1/n_i \le \beta \le C_2/\sqrt{n_i}$, where $n_i = n/k$, then Algorithm 4 satisfies $(\varepsilon, 0)$-fixed-graph DP with error $\alpha \le O\left(\exp\left(-\Theta\left(\varepsilon\beta\sqrt{nk}\right)\right) + \exp\left(-\Theta(n\beta^2)\right)\right)$.*

The algorithm exploits the vertex-disjoint group structure by computing a majority vote over k independent statistics S_j, one per group, with Laplace noise for privacy; disjointness is what makes $\sum_j S_j$ have sensitivity one, since changing the data attributed to any single vertex alters at most one S_j. Since $c = 1/k$ for such networks, the first term of the bound matches the privacy lower bound of Theorem 25 and the second matches the nonprivate sampling bound of Lemma 31, so the upper bound matches the maximum of the two lower bounds up to constants in the exponents and the $\sqrt{\log n}$ factor. The analysis (Appendix E.3) relies on calculating the expectation of S_j, which differs by $\Theta(\beta\sqrt{n/k})$ between the two distributions (Lemma 32).

The gap for arbitrary networks. For general networks, we do not have matching upper bounds. We can construct an $(\varepsilon, 0)$-DP algorithm achieving error $\alpha \le \frac{1}{2}\exp(-\varepsilon\beta/(2c_\gamma)) + \exp(-n(1-\gamma)\beta^2/2)$ for a refined density parameter $c_\gamma \ge c$ (Theorem 33, proof in Appendix E.4). However, since c_γ can be much larger than $1/n$, this upper bound does not match the lower bound for dense networks. Clustered networks are in fact fundamentally easier: the $\sqrt{c/n}$ rate is not attainable on all network structures.

Proposition 27 (The $\sqrt{c/n}$ rate is not attainable on the star)**.** *Let (V, E) be the star, $e_i = \{v_c, v_i\}$ for $i \in [n]$, so that $c = 1$. For every $\beta \in (0, 1/2)$, every n, and every (ε, δ)-fixed-graph DP mechanism $\mathcal{M} : \mathcal{X}^n \to \{1, 2\}$, the testing error satisfies $\alpha \ge (1-\delta)/(1+e^\varepsilon)$. The error is therefore bounded below by a constant independent of n and β, whereas the $\sqrt{c/n}$ rate of Theorem 25 would permit error $\exp(-\Theta(\varepsilon\beta\sqrt{n \log n}))$ on this network.*

Proof. Every hyperedge contains the center v_c, so by Definition 4 any two datasets $x, y \in \mathcal{X}^n$ are fixed-graph adjacent: they differ only in data attributed to v_c. Hence $\Pr[\mathcal{M}(y) = 1] \ge e^{-\varepsilon}\big(\Pr[\mathcal{M}(x) = 1] - \delta\big)$ for all x, y. Averaging over $x \sim \mathcal{D}_1^n$ and $y \sim \mathcal{D}_2^n$, and using $\Pr_{\mathcal{D}_2^n}[\mathcal{M} = 1] \le \alpha$ and $\Pr_{\mathcal{D}_1^n}[\mathcal{M} = 1] \ge 1 - \alpha$, gives $\alpha \ge e^{-\varepsilon}(1 - \alpha - \delta)$, that is, $\alpha \ge (1-\delta)/(1+e^\varepsilon)$. □

The proposition is consistent with Theorem 33, whose bound $\frac{1}{2}e^{-\varepsilon\beta/(2c_\gamma)}$ is $\Theta(1)$ on the star because $c_\gamma = 1$ there: on the star the general upper bound is tight and the coupling lower bound of Theorem 25 is loose. The open question is therefore not whether the $\sqrt{c/n}$ rate holds in general, which it does not, but which network parameter governs i.i.d. testing between the two bounds.

Remark 8 (Mean estimation on clustered networks via user-level privacy)**.** *On a clustered network whose k groups have pairwise disjoint vertex sets, fixed-graph DP coincides with user-level differential privacy with k users, each holding the $m = n/k$ examples of its group: a user-level neighbor*

changes one group's examples arbitrarily, which includes every fixed-graph neighbor, so any user-level DP algorithm is fixed-graph DP on such networks. The user-level literature therefore transfers verbatim. For i.i.d. mean estimation, Levy et al. [50] show that the privacy-induced error decreases as $1/\sqrt{m}$ in the samples per user and as $1/k$ in the number of users, up to dimension-dependent and logarithmic factors, with matching lower bounds; in our parameters, with $c = 1/k$ and $km = n$, the squared privacy term scales as $1/(k^2 m) = c/n$ rather than as the worst-case c^2, the same $\sqrt{c/n}$ improvement as in hypothesis testing. This settles the estimation question on clustered networks, while Proposition 27 shows that no such improvement is available on the star.

The key takeaway is that distributional assumptions weaken the privacy lower bound from c to $\sqrt{c/n}$ scaling, and that on clustered networks this weaker rate is attained: for a clustered network with hub degree Δ, where $c = \Delta/n$, the sample complexity of testing drops from $\Theta(\Delta)$ to $\Theta(\sqrt{\Delta})$ up to logarithmic factors, a $\sqrt{\Delta}$ improvement. Proposition 27 shows the improvement is structure-dependent and absent on the star, so independence alone does not guarantee the clustered-network rate.

E.2 Proof of Error Lower Bound in terms of c (Theorem 25)

We first state the key definition needed for the proof. This definition is similar to the Hamming distance between two datasets. Since we are considering fixed-graph DP, we need to cover the different elements by changing a subset of vertices $\tilde{V}$.

Definition 13. *For $X \in \mathcal{X}^n$, $Y \in \mathcal{X}^n$, define*

$$d(X, Y) = \min_{\tilde{V} \subseteq V} |\tilde{V}| \text{ s.t. } \tilde{V} \cap e \neq \emptyset, \forall e \text{ satisfying } x_e \neq y_e.$$

The following lemma is the (ε, δ)-DP Le Cam's method [2] adapted to fixed-graph multi-attribution privacy.

Lemma 28. *[(ε, δ)-DP Le Cam's method] For any two distributions $\mathcal{D}_1, \mathcal{D}_2$, $X \sim \mathcal{D}_1^n$, $Y \sim \mathcal{D}_2^n$. If there exists a coupling (X, Y) with $\mathbb{E}d(X, Y) \leq D$. Then for any $\varepsilon \geq 0, \delta \geq 0$, any (ε, δ)-fixed-graph DP mechanism $\mathcal{A} : \mathcal{X}^n \to \{1, 2\}$ must satisfy*

$$\mathbb{P}_{X \sim \mathcal{D}_1^n, \mathcal{A}}[\mathcal{A}(X) \neq 1] + \mathbb{P}_{Y \sim \mathcal{D}_2^n, \mathcal{A}}[\mathcal{A}(Y) \neq 2] \geq 0.9e^{-10\varepsilon D} - 20D\delta.$$

Proof. Let (X, Y) be distributed according to a coupling of $\mathcal{D}_1^n$ and $\mathcal{D}_2^n$ with $\mathbb{E}[d(X, Y)] \leq D$. The distribution of (X, Y) is denoted as μ. By Markov's inequality, $\Pr(d(X, Y) > 10D) < 0.1$. Let x and y be the realization of X and Y.

$$W := \{(x, y) \in \mathcal{X}^n \times \mathcal{X}^n \mid d(x, y) \leq 10D\}$$

be the set of pairs of realizations with distance at most $10D$.

For any $(x, y) \in \mathcal{X}^n \times \mathcal{X}^n$ with distance $d(x, y)$ and any $S \subseteq \{1, 2\}$, by the definition of $d(x, y)$ in Definition 13, one can find $d(x, y)$ vertices such that y is obtained from x by changing only the data attributed to these vertices. Since $\mathcal{A}$ is (ε, δ)-fixed-graph DP, changing all the data attributed to a single vertex maintains (ε, δ)-DP. By group privacy 2,

$$\Pr[\mathcal{A}(y) \in S] \leq e^{d(x,y)\cdot\varepsilon} \Pr[\mathcal{A}(x) \in S] + d(x, y)e^{(d(x,y)-1)\varepsilon}\delta.$$

Furthermore, if $d(x, y) \leq 10D$, we have

$$\Pr[\mathcal{A}(y) \in S] \leq e^{10\varepsilon D} \Pr[\mathcal{A}(x) \in S] + 10e^{10\varepsilon D} D\delta.$$

We apply this fact in the following inequality. Let $p_1 := \Pr(\mathcal{A}(X) = 2)$ and $p_2 := \Pr(\mathcal{A}(Y) = 1)$.

$$\begin{aligned}
1 - p_2 &= \Pr[\mathcal{A}(Y) = 2] \\
&= \int_{\mathcal{X}^n \times \mathcal{X}^n} \Pr(\mathcal{A}(y) = 2) \cdot d\mu(x, y) \\
&= \int_W \Pr(\mathcal{A}(y) = 2) \cdot d\mu(x, y) + \int_{\mathcal{X}^n \times \mathcal{X}^n \setminus W} \Pr(\mathcal{A}(y) = 2) \cdot d\mu(x, y)
\end{aligned}$$

$$
\begin{aligned}
&\leq \int_W \left(e^{10\varepsilon D}\Pr[\mathcal{A}(x)=2] + 10e^{10\varepsilon D}D\delta\right)\cdot d\mu(x,y) + \int_{\mathcal{X}^n\times\mathcal{X}^n\setminus W} 1\cdot d\mu(x,y)\\
&\leq \int_{\mathcal{X}^n\times\mathcal{X}^n} \left(e^{10\varepsilon D}\Pr[\mathcal{A}(x)=2] + 10e^{10\varepsilon D}D\delta\right)\cdot d\mu(x,y) + \Pr[d(x,y)>10D]\\
&< p_1 e^{10\varepsilon D} + 10D\delta\cdot e^{10\varepsilon D} + 0.1.
\end{aligned}
$$

Thus,

$$p_2 + p_1 e^{10\varepsilon D} > 0.9 - 10D\delta\cdot e^{10\varepsilon D}.$$

Similarly, we get

$$p_1 + p_2 e^{10\varepsilon D} > 0.9 - 10D\delta\cdot e^{10\varepsilon D}.$$

Adding the two inequalities and rearranging the terms,

$$p_1 + p_2 \geq \frac{1.8 - 20D\delta e^{10\varepsilon D}}{1+e^{10\varepsilon D}} \geq 0.9e^{-10\varepsilon D} - 20D\delta.$$

(since $1+e^{10\varepsilon D}\leq 2e^{10\varepsilon D}$ and $e^{10\varepsilon D}\leq 1+e^{10\varepsilon D}$)

□

Lemma 29. *We can split E into $k\leq\left\lfloor\frac{\log(n)}{c}\right\rfloor+1$ groups $E_1,\ldots,E_k$, such that each group share a common vertex.*

Proof. By Definition 8, the parameter c ensures that for any subset $\tilde{E}\subseteq E$ of size m, there exists a vertex appearing in at least cm hyperedges in $\tilde{E}$. We exploit this property to recursively decompose the hypergraph. Starting with $E_0=E$ (the full set of n examples), we construct groups $E_1,E_2,\ldots,E_k$ and corresponding vertices $v_1,v_2,\ldots,v_k$ as follows:

1. Apply Definition 8 to E_0: there exists vertex v_1 appearing in at least cn hyperedges. Let $E_1=\{e\in E_0 : v_1\in e\}$, so $|E_1|\geq cn$.
2. Apply Definition 8 to $E_0\setminus E_1$: there exists vertex v_2 appearing in at least $c|E_0\setminus E_1|$ hyperedges in this remaining set. Let $E_2=\{e\in E_0\setminus E_1 : v_2\in e\}$.
3. Continue recursively: for $i\geq 2$, apply the definition to $E_0\setminus(E_1\cup\cdots\cup E_{i-1})$ to find v_i and E_i.
4. Stop when all hyperedges are covered: $E_1\cup E_2\cup\cdots\cup E_k=E$.

It suffices to prove that $k\leq\left\lfloor\frac{\log(n)}{c}\right\rfloor+1$. We prove this by contradiction. Assume we have $k\geq\left\lfloor\frac{\log(n)}{c}\right\rfloor+2$. From the construction of the groups above, it is obvious that $|E_i|\geq c|E_0\setminus(E_1\cup\cdots\cup E_{i-1})|$. This implies $|E_0\setminus(E_1\cup\cdots\cup E_i)|\leq(1-c)|E_0\setminus(E_1\cup\cdots\cup E_{i-1})|$. We apply this property recursively and get

$$
\begin{aligned}
|E_k| &= |E_0\setminus(E_1\cup\cdots\cup E_{k-1})|\\
&\leq (1-c)^{k-1}n\\
&\leq e^{-c(k-1)}n && (1+x\leq e^x)\\
&\leq e^{-c\left(\left\lfloor\frac{\log(n)}{c}\right\rfloor+1\right)}n\\
&< e^{-c\left(\frac{\log(n)}{c}\right)}n = 1.
\end{aligned}
$$

This is impossible since $E_k\neq\emptyset$. We have proved $k\leq\left\lfloor\frac{\log(n)}{c}\right\rfloor+1$ by contradiction. □

Lemma 30. *There exists a coupling between $X\sim\mathcal{D}_1^n$ and $Y\sim\mathcal{D}_2^n$, such that $\mathbb{E}_{(X,Y)}d(X,Y)\leq\sqrt{n\left(\frac{\log(n)}{c}+1\right)\frac{4\beta^2}{1-2\beta}}$.*

Proof. We construct a coupling (X,Y) between $X\sim\mathrm{Ber}(\frac{1}{2}-\beta)^n$ and $Y\sim\mathrm{Ber}(\frac{1}{2}+\beta)^n$ and give an upper bound $\mathbb{E}[d(X,Y)]\leq D$.

From Lemma 29, we split E into $k \le \left\lfloor \frac{\log(n)}{c} \right\rfloor + 1$ groups $E_1, \ldots, E_k$, such that each group share a common vertex. For any fixed $i \in [1, k]$, we first couple the data in E_i: $X_i \sim \mathcal{D}_1^{|E_i|}$ and $Y_i \sim \mathcal{D}_2^{|E_i|}$. The KL-divergence between $\mathcal{D}_1 = \mathrm{Ber}(\frac{1}{2} - \beta)$ and $\mathcal{D}_2 = \mathrm{Ber}(\frac{1}{2} + \beta)$ is

$$D_{KL}(\mathcal{D}_1||\mathcal{D}_2) = \left(\frac{1}{2} - \beta\right)\log(\frac{\frac{1}{2}-\beta}{\frac{1}{2}+\beta}) + \left(\frac{1}{2} + \beta\right)\log(\frac{\frac{1}{2}+\beta}{\frac{1}{2}-\beta}) = 2\beta\log(\frac{\frac{1}{2}+\beta}{\frac{1}{2}-\beta}) \le \frac{8\beta^2}{1-2\beta}. \quad (\ln(1+x) \le x)$$

The KL-divergence between $X_i \sim \mathcal{D}_1^{|E_i|}$ and $Y_i \sim \mathcal{D}_2^{|E_i|}$ is

$$D_{KL}(\mathcal{D}_1^{|E_i|}||\mathcal{D}_2^{|E_i|}) = |E_i| D_{KL}(\mathcal{D}_1||\mathcal{D}_2) \le |E_i|\frac{8\beta^2}{1-2\beta}.$$

From Pinsker's inequality,

$$d_{TV}(\mathcal{D}_1^{|E_i|}||\mathcal{D}_2^{|E_i|}) \le \sqrt{\frac{1}{2} D_{KL}(\mathcal{D}_1^{|E_i|}||\mathcal{D}_2^{|E_i|})} \le \sqrt{|E_i|\frac{4\beta^2}{1-2\beta}}.$$

Therefore, there exists a coupling (X_i, Y_i), s.t. $\mathbb{P}[X_i \ne Y_i] \le \sqrt{|E_i|\frac{4\beta^2}{1-2\beta}}$. We couple each i independently, i.e., we make $(X_1, Y_1), (X_2, Y_2), \ldots, (X_k, Y_k)$ k independent random vectors. With this, we know that

$$\begin{aligned}
\mathbb{E}[|\{i : X_i \ne Y_i\}|] &= \sum_{i=1}^{k} \mathbb{P}[X_i \ne Y_i] \\
&\le \sum_{i=1}^{k} \sqrt{|E_i|\frac{4\beta^2}{1-2\beta}} \\
&\le \sqrt{k\sum_{i=1}^{k} |E_i|\frac{4\beta^2}{1-2\beta}} \\
&= \sqrt{kn\frac{4\beta^2}{1-2\beta}} \\
&\le \sqrt{n\left(\frac{\log(n)}{c} + 1\right)\frac{4\beta^2}{1-2\beta}}.
\end{aligned}$$

Since all the hyperedges in E_i share a common vertex, $d(X, Y) \le |\{i : X_i \ne Y_i\}|$. This is because if $X_i \ne Y_i$, we put into $\tilde{V}$ the common vertex in the hyperedges in E_i. This $\tilde{V}$ satisfies the constraint in Definition 13.

We can thus give the upper bound

$$\mathbb{E}_{(X,Y)} d(X, Y) \le \mathbb{E}[|\{i : X_i \ne Y_i\}|] \le \sqrt{n\left(\frac{\log(n)}{c} + 1\right)\frac{4\beta^2}{1-2\beta}}.$$

□

Theorem (Theorem 25, restated)**.** *Assume $\beta < 0.4$. The error of any $(\varepsilon, 0)$-fixed-graph DP mechanism $\mathcal{M} : \mathcal{X}^n \to \{1, 2\}$ must satisfy*

$$\alpha \ge \Omega\left(\exp\left(-60\varepsilon\beta\sqrt{\frac{n\log(n+1)}{c}}\right)\right). \quad (25)$$

Proof of Theorem 25. From Lemma 28, $2\alpha \geq 0.9e^{-10\varepsilon D}$. From Lemma 30, we can choose $D = \sqrt{n\left(\frac{\log(n)}{c}+1\right)\frac{4\beta^2}{1-2\beta}}$. Therefore,

$$\begin{aligned}
\alpha &\geq 0.45e^{-10\varepsilon D} \\
&= 0.45\exp\left(-10\varepsilon\sqrt{n\left(\frac{\log(n)}{c}+1\right)\frac{4\beta^2}{1-2\beta}}\right) \\
&> 0.45\exp\left(-10\varepsilon\sqrt{n\left(\frac{\log(n)}{c}+1\right)20\beta^2}\right) && (\beta < 0.4) \\
&\geq 0.45\exp\left(-10\varepsilon\beta\sqrt{20n\left(\frac{\log(n)+1}{c}\right)}\right) && (c \leq 1) \\
&\geq 0.45\exp\left(-10\varepsilon\beta\sqrt{20n\left(\frac{\log(n)+1}{c}\right)}\right) \\
&\geq 0.45\exp\left(-10\varepsilon\beta\sqrt{20n\left(\frac{1.6\log(n+1)}{c}\right)}\right) \\
&\geq 0.45\exp\left(-60\varepsilon\beta\sqrt{\frac{n\log(n+1)}{c}}\right).
\end{aligned}$$

□

Lemma 31 (Nonprivate testing lower bound)**.** *For $\beta \leq 0.4$, any mechanism $\mathcal{M} : \mathcal{X}^n \to \{1,2\}$, private or not, satisfies $\alpha \geq \frac{1}{4}e^{-11n\beta^2}$.*

Proof. For any test, $\Pr_{\mathcal{D}_1^n}[\mathcal{M} \neq 1] + \Pr_{\mathcal{D}_2^n}[\mathcal{M} \neq 2] \geq 1 - \mathrm{TV}(\mathcal{D}_1^n, \mathcal{D}_2^n)$, so $\alpha \geq \frac{1}{2}\big(1 - \mathrm{TV}(\mathcal{D}_1^n, \mathcal{D}_2^n)\big)$. The Bretagnolle–Huber inequality gives $1 - \mathrm{TV}(P,Q) \geq \frac{1}{2}e^{-\mathrm{KL}(P\|Q)}$, and KL divergence tensorizes, so $\alpha \geq \frac{1}{4}e^{-n\,\mathrm{KL}(\mathcal{D}_1\|\mathcal{D}_2)}$. Finally, $\mathrm{KL}\big(\mathrm{Ber}(\frac{1}{2}-\beta)\,\|\,\mathrm{Ber}(\frac{1}{2}+\beta)\big) = 2\beta\log\frac{1+2\beta}{1-2\beta}$, and on $[0, 0.4]$ the ratio $\frac{1}{\beta}\log\frac{1+2\beta}{1-2\beta}$ is increasing with value $\log(9)/0.4 < 5.5$ at $\beta = 0.4$, so the divergence is at most $11\beta^2$. □

E.3 Proof of Matching Upper Bound for Clustered Networks (Theorem 26)

Lemma 32. *There exist absolute constants $C_1, C_2 > 0$ such that, under the assumption $\frac{C_1}{n_i} \leq \beta \leq \frac{C_2}{\sqrt{n_i}}$,*

$$\mathbb{E}_{X\sim\mathcal{D}_i^n}[S_j] = \begin{cases} \frac{1}{2} - \Theta(\beta\sqrt{n/k}) & \text{if } i = 1, \\ \frac{1}{2} + \Theta(\beta\sqrt{n/k}) & \text{if } i = 2. \end{cases}$$

Proof. If i=1,

$$\begin{aligned}
\mathbb{E}_{X\sim\mathcal{D}_1^n}[S_j] &= \Pr_{X\sim\mathcal{D}_1^n}\left[|\{e\in G_j \mid x_e=0\}| \le \frac{n}{2k}\right] \\
&= \Pr_{X\sim\mathcal{D}_1^n}\left[|\{e\in G_j \mid x_e=1\}| \ge \frac{n}{2k}\right] \\
&= \Pr_{X\sim\mathcal{D}_1^n}\left[\sum_{e\in G_j} x_e \ge \frac{n}{2k}\right] \\
&= \Pr_{X\sim\mathcal{D}_1^n}\left[\sum_{e\in G_j} (x_e-\mathbb{E}[x_e]) \ge \beta\frac{n}{k}\right] \\
&= \Pr_{X\sim\mathcal{D}_1^n}\left[\frac{\sum_{e\in G_j}(x_e-\mathbb{E}[x_e])}{\sqrt{\frac{n}{k}(\frac{1}{4}-\beta^2)}} \ge \frac{\beta}{\sqrt{\frac{1}{4}-\beta^2}}\sqrt{\frac{n}{k}}\right].
\end{aligned}$$

From the central limit theorem, $\frac{\sum_{e\in G_j}(x_e-\mathbb{E}[x_e])}{\sqrt{\frac{n}{k}(\frac{1}{4}-\beta^2)}}$ converges to the standard gaussian in distribution. The Berry-Esseen inequality provides an explicit quantitative bound, giving

$$\begin{aligned}
&\left|\Pr_{X\sim\mathcal{D}_1^n}\left[\frac{\sum_{e\in G_j}(x_e-\mathbb{E}[x_e])}{\sqrt{\frac{n}{k}(\frac{1}{4}-\beta^2)}} \ge \frac{\beta}{\sqrt{\frac{1}{4}-\beta^2}}\sqrt{\frac{n}{k}}\right]\right. \\
&\qquad \left. - \Pr_{g\sim\mathcal{N}(0,1)}\left[g \ge \frac{\beta}{\sqrt{\frac{1}{4}-\beta^2}}\sqrt{\frac{n}{k}}\right]\right| \le \frac{3}{\left(\frac{1}{4}-\beta^2\right)^{3/2}\sqrt{\frac{n}{k}}}.
\end{aligned} \tag{26}$$

We choose $C_2 \le \frac{1}{4}$ so that $\beta \le \frac{C_2}{\sqrt{n_i}} \le \frac{1}{4}$.

$$\frac{\beta}{\sqrt{\frac{1}{4}-\beta^2}}\sqrt{\frac{n}{k}} \le \frac{\beta}{\sqrt{\frac{1}{4}-\frac{1}{16}}}\sqrt{\frac{n}{k}} = \sqrt{\frac{16}{3}}\beta\sqrt{\frac{n}{k}} \le \sqrt{\frac{16}{3}}C_2 \le \frac{1}{\sqrt{3}}.$$

The density of standard gaussian for any $x\in[0,\frac{1}{\sqrt{3}}]$ is bounded by $p(x)\in\left[\frac{1}{\sqrt{2\pi}}e^{-1/6},\frac{1}{\sqrt{2\pi}}\right]$. Now we can give upper bound and lower bound on $\Pr_{g\sim\mathcal{N}(0,1)}\left[g\ge\frac{\beta}{\sqrt{\frac{1}{4}-\beta^2}}\sqrt{\frac{n}{k}}\right]$. We have

$$\begin{aligned}
\Pr_{g\sim\mathcal{N}(0,1)}\left[g \ge \frac{\beta}{\sqrt{\frac{1}{4}-\beta^2}}\sqrt{\frac{n}{k}}\right] &= \frac{1}{2} - \Pr_{g\sim\mathcal{N}(0,1)}\left[0\le g < \frac{\beta}{\sqrt{\frac{1}{4}-\beta^2}}\sqrt{\frac{n}{k}}\right] \\
&\le \frac{1}{2} - \frac{1}{\sqrt{2\pi}}e^{-1/6}\frac{\beta}{\sqrt{\frac{1}{4}-\beta^2}}\sqrt{\frac{n}{k}} \\
&\le \frac{1}{2} - \frac{1}{\sqrt{2\pi}}e^{-1/6}\cdot 2\beta\sqrt{\frac{n}{k}}
\end{aligned}$$

and

$$\begin{aligned}
\Pr_{g\sim\mathcal{N}(0,1)}\left[g \ge \frac{\beta}{\sqrt{\frac{1}{4}-\beta^2}}\sqrt{\frac{n}{k}}\right] &= \frac{1}{2} - \Pr_{g\sim\mathcal{N}(0,1)}\left[0\le g < \frac{\beta}{\sqrt{\frac{1}{4}-\beta^2}}\sqrt{\frac{n}{k}}\right] \\
&\ge \frac{1}{2} - \frac{1}{\sqrt{2\pi}}\frac{\beta}{\sqrt{\frac{1}{4}-\beta^2}}\sqrt{\frac{n}{k}} \\
&\ge \frac{1}{2} - \frac{1}{\sqrt{2\pi}}\cdot\sqrt{\frac{16}{3}}\beta\sqrt{\frac{n}{k}}.
\end{aligned}$$

Now we upper bound $\frac{3}{\left(\frac{1}{4}-\beta^2\right)^{3/2}\sqrt{\frac{n}{k}}}$. Since $\beta \le \frac{1}{4}$ and $\beta \ge \frac{C_1}{n_i}$,

$$\frac{3}{\left(\frac{1}{4}-\beta^2\right)^{3/2}\sqrt{\frac{n}{k}}} \le 3(16/3)^{3/2}\sqrt{\frac{k}{n}} \le \frac{3(16/3)^{3/2}}{C_1}\beta\sqrt{\frac{n}{k}}.$$

From equation 26,

$$\frac{1}{2} - \left(\frac{1}{\sqrt{2\pi}}\cdot\sqrt{\frac{16}{3}} + \frac{3(16/3)^{3/2}}{C_1}\right)\beta\sqrt{\frac{n}{k}} \le \mathbb{E}_{X\sim\mathcal{D}_1^n}[S_j] \le \frac{1}{2} - \left(\frac{2}{\sqrt{2\pi}}e^{-1/6} - \frac{3(16/3)^{3/2}}{C_1}\right)\cdot\beta\sqrt{\frac{n}{k}}.$$

Choosing constant C_1 large enough guarantees

$$\frac{1}{2} - \beta\sqrt{\frac{n}{k}} \le \mathbb{E}_{X\sim\mathcal{D}_1^n}[S_j] \le \frac{1}{2} - \frac{1}{2}\cdot\beta\sqrt{\frac{n}{k}}. \tag{27}$$

When $i = 2$, the proof proceeds similarly. □

Theorem (Theorem 26, restated)**.** *Assume the k groups have pairwise disjoint vertex sets, so that every user's examples lie within a single group. Algorithm 4 satisfies $(\varepsilon, 0)$-fixed-graph DP. Furthermore, there exist absolute constants $C_1, C_2 > 0$ such that if $C_1/n_i \le \beta \le C_2/\sqrt{n_i}$ with $n_i = n/k$, then the error of Algorithm 4 satisfies*

$$\alpha \le O\left(e^{-\Theta\left(\varepsilon\beta\sqrt{nk}\right)} + e^{-\Theta(n\beta^2)}\right).$$

Proof of Theorem 26. By the hypothesis that the groups have pairwise disjoint vertex sets, every user's examples lie within a single group, so if all the data incident to a single user changes, only one S_j can change. Therefore, $\sum_{j=1}^{k} S_j$ has sensitivity 1, and adding $\mathrm{Lap}\left(\mu = 0, b = \frac{1}{\varepsilon}\right)$ noise ensures $(\varepsilon, 0)$-fixed-graph DP. Without the disjointness hypothesis a single vertex could affect several group statistics and the sensitivity claim would fail.

Next, we study the error of this algorithm. If $i = 1$,

$$\begin{aligned}
\Pr\left[\hat{i} = 2\right] &= \Pr\left[\sum_{j=1}^{k} S_j + \mathrm{Lap}\left(\mu = 0, b = \frac{1}{\varepsilon}\right) > \frac{k}{2}\right] \\
&= \Pr\left[\sum_{j=1}^{k}(S_j - \mathbb{E}S_j) + \mathrm{Lap}\left(\mu = 0, b = \frac{1}{\varepsilon}\right) > k\left(\frac{1}{2} - \mathbb{E}S_1\right)\right] \\
&\le \Pr\left[\sum_{j=1}^{k}(S_j - \mathbb{E}S_j) + \mathrm{Lap}\left(\mu = 0, b = \frac{1}{\varepsilon}\right) > k\cdot\frac{1}{2}\beta\sqrt{\frac{n}{k}}\right] && \text{(equation 27)} \\
&\le \Pr\left[\mathrm{Lap}\left(\mu = 0, b = \frac{1}{\varepsilon}\right) > \frac{1}{4}\beta\sqrt{kn}\right] + \Pr\left[\sum_{j=1}^{k}(S_j - \mathbb{E}S_j) > \frac{1}{4}\beta\sqrt{kn}\right] \\
&\le \frac{1}{2}\exp\left(-\frac{1}{4}\varepsilon\beta\sqrt{nk}\right) + \exp\left(-\frac{n\beta^2}{8}\right). && \text{(Hoeffding's inequality)}
\end{aligned}$$

When $i = 2$, the proof proceeds similarly. □

E.4 Proof of Upper Bound for Arbitrary Network (Theorem 33)

Theorem 33. *For any $\gamma \in [0, 1)$, hypergraph (V, E) with $|E| = n$ examples and refined network density parameter c_γ from equation 3, there exists an $(\varepsilon, 0)$-fixed-graph DP algorithm achieving*

$$\alpha \le \frac{1}{2}e^{-\frac{\varepsilon\beta}{2c_\gamma}} + e^{-\frac{n(1-\gamma)\beta^2}{2}}.$$

From the definition of the refined network density parameter, it is obvious that $c_\gamma \geq c \geq \frac{1}{n}$, so the term $e^{-\frac{\varepsilon\beta}{2c_\gamma}}$ in the upper bound is always greater than the lower bound, and the gap becomes large when $c_\gamma \gg \frac{1}{n}$. Determining the optimal testing error for arbitrary attribution networks remains open.

Proof of Theorem 33. The proof constructs a two-phase algorithm that first chooses a good subset of hyperedges $S \subseteq E$. From the definition of c_γ, there exists $m \in [(1-\gamma)n, n]$ such that $f(m) = c_\gamma$, and thus there exists a subset $S \subseteq E$ with the following properties: $|S| = m \geq (1-\gamma)n$ and every vertex $v \in V$ appears in at most $k = m \cdot c_\gamma$ hyperedges in S. The algorithm retains the examples indexed by S for hypothesis testing.

The second phase of the algorithm adds Laplace noise to the empirical average. The algorithm returns

$$\hat{i} = \mathbf{1}\left\{\sum_{e\in S} x_e + \mathrm{Lap}\left(\mu = 0, b = \frac{k}{\varepsilon}\right) > \frac{|S|}{2}\right\} + 1.$$

We first verify the algorithm is $(\varepsilon, 0)$-fixed-graph DP. If all the data of a single user changes, since the user has at most $k = m \cdot c_\gamma$ incident hyperedges, this will affect at most k data in S. Therefore, the sensitivity of $\sum_{e\in S} x_e$ is at most $k = m \cdot c_\gamma$, and adding $\mathrm{Lap}\left(\mu = 0, b = \frac{k}{\varepsilon}\right)$ noise ensures $(\varepsilon, 0)$-fixed-graph DP.

Next, we study the error of this algorithm. If $i = 1$,

$$\begin{aligned}
\Pr\left[\hat{i} = 2\right] &= \Pr\left[\sum_{e\in S} x_e + \mathrm{Lap}\left(\mu = 0, b = \frac{k}{\varepsilon}\right) > \frac{|S|}{2}\right] \\
&= \Pr\left[\sum_{e\in S}(x_e - \mathbb{E}x_e) + \mathrm{Lap}\left(\mu = 0, b = \frac{k}{\varepsilon}\right) > \beta|S|\right] \\
&\leq \Pr\left[\mathrm{Lap}\left(\mu = 0, b = \frac{k}{\varepsilon}\right) > \frac{1}{2}\beta|S|\right] + \Pr\left[\sum_{e\in S}(x_e - \mathbb{E}x_e) > \frac{1}{2}\beta|S|\right] \\
&\leq \frac{1}{2}\exp\left(-\frac{\varepsilon\beta|S|}{2k}\right) + \exp\left(-\frac{|S|\beta^2}{2}\right). && \text{(Hoeffding's inequality)} \\
&\leq \frac{1}{2}e^{-\frac{\varepsilon\beta}{2c_\gamma}} + e^{-\frac{n(1-\gamma)\beta^2}{2}}. && (|S| \geq (1-\gamma)n)
\end{aligned}$$

When $i = 2$, the proof proceeds similarly. □

F Empirical Validation

We examine contribution bounding, Gaussian mean estimation, and hypothesis testing on synthetic hypergraphs. These families let us vary the attribution structure and compare the observed results with the theoretical guarantees. Linear programs are solved with HiGHS, all randomness uses fixed seeds, and the supplementary code includes the benchmark graphs, raw results, and scripts reproducing the figures. Experimental settings are given in the captions and the discussion below.

Contribution-bounding approximation. Figure 1(a) illustrates the tight greedy construction of Proposition 4. Processing the column edges first gives ratio exactly r at both budgets. The random-order curve is much closer to one on the same family, showing how the processing order affects retention.

Figure 1(b) compares random-order greedy with LP-weighted greedy on a family with increasingly many examples attributed to a common hub. For each planted-edge count $q \in \{5, 10, 20, 40, 80, 160\}$, we generate six hypergraphs with 240 vertices and 320 edges. The background consists of $320 - q$ uniformly sampled edges of size two or three; the remaining q edges join the hub to a uniformly sampled partner. Background edges may also contain the hub, so q denotes the number of planted edges rather than the maximum degree. We use budget $k = 2$ and retain the largest subset from eight

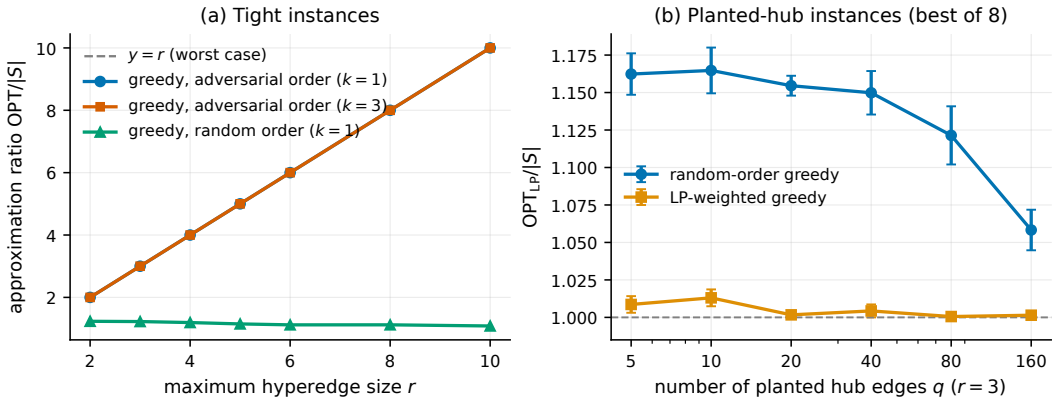


Figure 1: **Left.** Contribution bounding on the tight grid construction. Adversarial-order greedy attains ratio r at budgets $k = 1$ and $k = 3$; the random-order curve averages twenty runs at $k = 1$. **Right.** Contribution bounding on hypergraphs with 240 vertices, 320 edges of size two or three, and budget $k = 2$. Each instance contains q planted edges incident to a common hub. Both methods keep the largest output from eight runs; the LP is solved once per instance. Points and error bars give the mean and one standard deviation of $\mathrm{OPT}_{\mathrm{LP}}/|S|$ over six instances at each q.

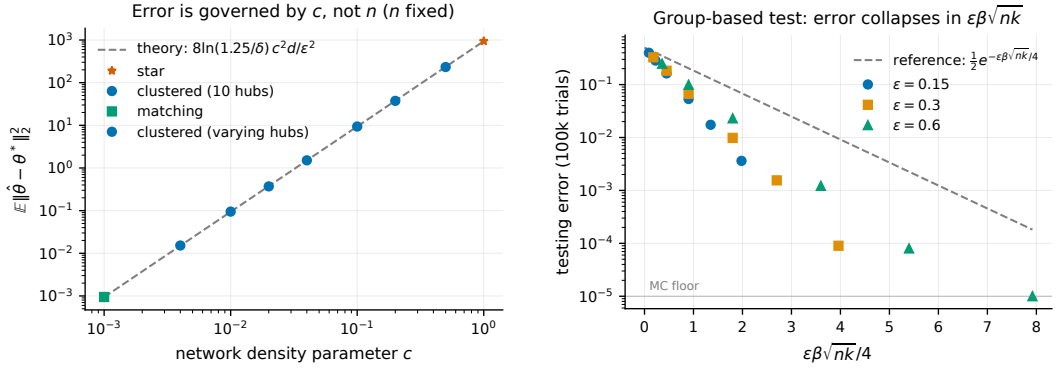


Figure 2: **Left.** Mean-estimation error of the two-phase mechanism across network families with matched $n = 1000$ and $d = 10$ at $\varepsilon = 1$, $\delta = 10^{-5}$ (2000 trials per point). The families are the star ($c = 1$), clustered networks with 2 to 250 hubs ($c = 1/\#\text{hubs}$), and the perfect matching ($c = 1/n$); the mechanism adds Gaussian noise at the fixed-graph sensitivity $2\Delta/n = 2c$. **Right.** Error under the first Bernoulli hypothesis of the group-based test (Algorithm 4) on clustered networks with group size $n_i = 400$, bias $\beta = 0.24/\sqrt{n_i}$, at the β-scaling of Theorem 26 (the proof's constants are not tracked), $k \in \{10, \ldots, 220\}$ groups, and $\varepsilon \in \{0.15, 0.3, 0.6\}$ (10^5 trials per point).

runs of each method on every graph. The mean ratio $\mathrm{OPT}_{\mathrm{LP}}/|S|$ ranges from 1.001 to 1.013 for LP-weighted greedy, compared with 1.058 to 1.165 for random-order greedy. Since $\mathrm{OPT}_{\mathrm{LP}} \geq \mathrm{OPT}_{\mathrm{IP}}$, these ratios also upper-bound the approximation ratios relative to the integer optimum. LP-weighted greedy retains more examples on all 36 instances in this experiment and remains close to the LP upper bound throughout the range of planted-edge counts. This family illustrates that large maximum degree need not lead to a poor contribution-bounding approximation.

The released code includes all benchmark graphs, the retained cardinalities from every run, the LP objectives, and the seeds. The two selectors use separate random streams and the same number of runs. LP solution time and traversal time are recorded separately. The comparison concerns retained cardinality at a fixed contribution cap; it does not assume that a larger selected subset always has smaller selection bias.

The density parameter governs error. The left panel of Figure 2 varies the network family at fixed n and plots the resulting estimation error against the density parameter c. The measured noise

errors follow $8\ln(1.25/\delta)\,c^2 d/\varepsilon^2$ across three orders of magnitude of c, consistent with the analysis in Section 4.2. When $c \lesssim \varepsilon/\sqrt{d\log(1/\delta)}$, this noise term is at most a constant. For these network families, Theorem 5 gives a matching dependence on c, d, and ε, up to logarithmic factors, in its stated parameter regime. For sufficiently large c, the lower bound saturates at a constant because of its $\min\{\beta^2, \cdot\}$ form. Once the Gaussian noise term exceeds one, the constant-zero candidate of Theorem 8 offers a smaller worst-case error bound. The plotted curve shows the Gaussian noise term, which can exceed the optimal estimation error in this regime. The star network, for which $c = 1$, is as hard as a single sample regardless of n, while the matching network achieves the example-level rate. The experiment makes concrete the central message of Section 4, that at fixed data volume the achievable error is a property of the attribution structure through c alone on these families.

Hypothesis testing under i.i.d. data. The right panel of Figure 2 shows the error under the first Bernoulli hypothesis for the group-based test, plotted against $\varepsilon\beta\sqrt{nk}/4$. Across the three privacy levels, the measurements follow a similar decreasing trend in this rescaled variable. The dashed exponential is a reference for the privacy-noise term in the proof of Theorem 26; the complete finite-sample bound also includes a sampling term. The plot illustrates the dependence on $\varepsilon\beta\sqrt{nk}$ in this experimental regime. Values below the Monte Carlo resolution 10^{-5} are displayed at that level.

Scope. The synthetic families isolate the effects of edge size, contribution concentration, and repeated participation. The contribution-bounding experiment measures retained cardinality at a fixed cap; the mean-estimation and testing experiments examine the corresponding noise and error terms on families with explicit structural parameters.

NeurIPS Paper Checklist

1. **Claims**
   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?
   Answer: [Yes]
   Justification: We listed our contributions in the introduction and proved our claims in the main paper.
   Guidelines:
   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.
2. **Limitations**
   Question: Does the paper discuss the limitations of the work performed by the authors?
   Answer: [Yes]
   Justification: We discuss the limitations (and future directions) in the Conclusion section.
   Guidelines:
   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.
3. **Theory assumptions and proofs**
   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?
   Answer: [Yes]
   Justification: We provide a full set of assumptions, a clear statement for each result, a proof sketch in the main paper, and a complete and correct detailed proof for each theoretical result in the Appendix.
   Guidelines:
   - The answer [N/A] means that the paper does not include theoretical results.

- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper includes an empirical validation (Appendix F). All experiments are synthetic with fixed random seeds; the appendix and the supplementary material specify every instance generator, solver, parameter, and script needed to reproduce each figure and table.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Complete code for all experiments, together with raw JSON results and a notebook reproducing every figure, is provided in the supplementary material with a README describing how to run it. All data are synthetic and generated by the included scripts from fixed seeds; no external data are used. The code, notebook, and accompanying data and results are also available at https://github.com/ramanarora/fixed-graph-dp.

   Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**
   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?
   Answer: [Yes]
   Justification: Every experimental configuration (instance sizes, hyperedge-size and cap parameters, network families, privacy budgets, trial counts, and random seeds) is stated in Appendix F and recorded in the accompanying scripts and result files; linear programs are solved with the HiGHS solver.
   Guidelines:
   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.
7. **Experiment statistical significance**
   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?
   Answer: [Yes]
   Justification: Monte Carlo quantities are reported with uncertainty throughout: the random-instance approximation panel reports means and standard deviations over 6 instances and 20 processing orders, the density-parameter figure averages 2000 trials per point, and the hypothesis-testing figure uses 10^5 trials per point with the Monte Carlo resolution limit stated and clipped values marked. The tight-instance ratios are deterministic and match the theoretical value exactly, so error bars are not meaningful for them.
   Guidelines:
   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.

- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**
   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?
   Answer: [Yes]
   Justification: All experiments run on a single CPU core in under five minutes in total; no GPUs or special infrastructure are used.
   Guidelines:
   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).
9. **Code of ethics**
   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?
   Answer: [Yes]
   Justification: The research presented in this paper complies with the NeurIPS Code of Ethics in all respects. Given the primarily theoretical nature of the results, ethical concerns are minimal.
   Guidelines:
   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).
10. **Broader impacts**
    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?
    Answer: [Yes]
    Justification: This is a theoretical paper focused on addressing a core problem. The problem domain has the potential for positive societal impact, as our framework enables private learning on sensitive data.
    Guidelines:
    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.

- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The released artifacts are small synthetic-experiment scripts that validate the theorems; they pose no risk of misuse, and no datasets or trained models are released. Accordingly, no specific safeguards are required.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [N/A]

    Justification: No external assets such as datasets, models, or third-party code are used beyond standard open-source scientific Python libraries, which are used in accordance with their licenses. All prior results are appropriately cited and credited in the manuscript.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The supplementary material introduces new code (experiment scripts, a small shared library, raw results, and a notebook), documented in an accompanying README; it will be released under an open-source license upon publication.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.

- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This work does not involve crowdsourcing experiments or research with human subjects. Therefore, this question is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This work does not involve human subjects or study participants. Therefore, there are no associated risks to disclose, and IRB approval is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: This work does not use LLMs as part of the core methodology. Any use of LLMs was limited to editing assistance and does not affect the scientific content, rigor, or originality of the paper.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.