PRIME: A Modular Approach for Private Synthetic Data

Miguel Fuentes[1*], **Brett Mullins**[3*], **Cecilia Ferrando**[2], **Cameron Musco**[2], **Daniel Sheldon**[2]

[1]Capital One [2]University of Massachusetts Amherst [3]Allegheny College

miguel.fuentes@capitalone.com

Abstract

Generating differentially private (DP) synthetic tabular data remains a challenge, particularly when leveraging state-of-the-art query-answering mechanisms like RESIDUAL-PLANNER or AIM+GREM. These mechanisms offer high utility through dense measurement collections $\mathcal{M}$; however, they are incompatible with existing synthesis tools. Specifically, graphical-model-based synthesis via PRIVATE-PGM becomes intractable as the collection of private measurements becomes dense.

We introduce PRIME (Private Re-weighting of Initial Model Estimates), a scalable framework that decouples the synthesis process from the complexity of the measurement set by restricting the support to a subset of the overall domain. PRIME transforms the synthesis problem into a convex re-weighting task over a support set of size K. This formulation achieves an $\mathcal{O}(K \cdot |\mathcal{M}|)$ per-iteration complexity, providing a drop-in solution for synthesis from any collection of privatized measurements. We show that, when provided the same measurements, PRIME achieves error comparable to PRIVATE-PGM while handling dense measurement sets where PRIVATE-PGM fails. We prove that PRIME's expected loss approaches PRIVATE-PGM's optimum at rate $\mathcal{O}(e^{D_2}/K)$, where D_2 measures the mismatch between the support generator and the target distribution. This theoretical relationship is demonstrated in an ablation study. Further empirical evaluation demonstrates that PRIME enables high-fidelity synthesis from query-answering mechanisms that were previously unusable for data generation.

1 Introduction

Differentially private (DP) query answering aims to release accurate statistical summaries of sensitive data under rigorous privacy guarantees, and recent mechanisms have improved the accuracy-efficiency tradeoff for large data domains. RESIDUAL-PLANNER [22] is an optimal scalable data-independent matrix mechanism for marginal queries, and AIM+GREM [7] integrates residual queries into an iterative, data-dependent framework. Synthetic data, however, offers a more versatile representation that supports unanticipated downstream tasks—such as exploratory data analysis or machine learning—beyond static query responses. Crucially, because DP is immune to post-processing, these high-fidelity measurements can be leveraged indefinitely without further privacy loss. This raises the question: can we transform the output of specialized query-answering mechanisms into high-quality synthetic data?

Doing so requires selecting, among the many distributions consistent with a set of measurements, a single canonical one. The Maximum Entropy (MaxEnt) principle resolves this by identifying the unique highest-entropy distribution consistent with the measurements, and sampling from it. Probabilistic Graphical Models (PGMs) provide an exact solution to this problem, and PRIVATE-PGM-based methods [15] have consistently demonstrated state-of-the-art performance in recent synthesis

*Work done while at the University of Massachusetts Amherst.

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

benchmarks [3]. These methods operate over a measurement graph that connects two attributes whenever they co-occur in some measurement. The cost of exact inference is governed by this graph's treewidth — the largest number of attributes inference must handle simultaneously — and scales exponentially in it, a severe bottleneck that renders the approach intractable when paired with the dense measurement structures of RESIDUAL-PLANNER and AIM+GREM. While alternative synthesizers such as GEM [14] or RAP [1] exist, they typically do not target the MaxEnt objective and lack support for correlated noise. What is required is a framework that preserves MaxEnt reconstruction, remains independent of treewidth, and accommodates the correlated noise structures used with residual queries.

We introduce PRIME (**P**rivate **R**eweighting of **I**nitial **M**odel **E**stimates), a modular synthetic data generator that provides a tractable approximation of MaxEnt synthesis, and prove that this approximation becomes exact as the support size grows. We build on Liu et al. [13], who re-weighted a public dataset to align with a private distribution (PRIME requires no public data). We generalize the re-weighting logic into a two-stage framework: a finite support D_S is sampled from a reference model, then a convex optimization problem finds the MaxEnt weight vector w^* over D_S consistent with the measurements. Because the support is fixed, the per-iteration complexity becomes $\mathcal{O}(K \cdot |\mathcal{M}|)$, where $K = |D_S|$ is the support size and $\mathcal{M}$ is the set of measurements, entirely eliminating the dependence on treewidth. PRIME handles both standard marginals and residual queries, avoiding the expensive covariance matrix inversion that naive optimization against correlated noise would require.

PRIME uses a modular architecture (Figure 1) that separates support initialization from inference, letting the reference model be treated as an interchangeable component and chosen to fit the situation: a simple, fast-to-sample model can supply a large support cheaply, or a more accurate model can reach the same coverage with far fewer samples. By framing synthesis in this way, PRIME provides a systematic framework for future improvements, as we demonstrate in Section 4.3. This modularity isolates the optimization cost—which is fixed at $\mathcal{O}(K \cdot |\mathcal{M}|)$ per iteration—from the statistical challenge of domain coverage.

In summary, our main contributions are as follows:

- **A Scalable, Modular Synthesis Framework:** We propose PRIME, a differentially private synthetic data generator that decouples support initialization from inference, providing a tractable approximation of the MaxEnt objective while avoiding the exponential scaling that bottlenecks existing graphical-model synthesizers.
- **Efficient Optimization for Correlated Noise:** We overcome the hurdle of optimizing against correlated noise via an algebraic reformulation using the `recon` operator, avoiding dense covariance matrix inversions and letting PRIME utilize the residual measurements produced by modern DP query-answering mechanisms.
- **Unlocking SOTA Query Mechanisms for Synthesis:** We demonstrate that PRIME can generate high-quality synthetic data from RESIDUAL-PLANNER and AIM+GREM, which other generators do not support, achieving data quality competitive with the state-of-the-art AIM mechanism.
- **Robust Scaling and Systematized Ablations:** We show that PRIME remains tractable as measurement density increases—precisely the regime where PRIVATE-PGM fails—and that support quality is an important performance lever, establishing a performance ceiling via oracle support.
- **A Finite-Sample Theoretical Guarantee:** We prove that PRIME's expected loss converges to PRIVATE-PGM's optimum at rate $\mathcal{O}(e^{D_2}/K)$, where D_2 is the Rényi divergence between the support generator and the target distribution and K is the support size (Section 3.6), formally characterizing how support quality governs the gap to PRIVATE-PGM's objective.

2 Background and Preliminaries

This section establishes the notation and theoretical foundations for our work. We define the data model and differential privacy, formalize residual queries, analyze the maximum entropy principle, and discuss the computational bottlenecks of graphical model-based synthesis.

2.1 Data Model, Marginal Queries, and Workloads

We assume data has a discrete domain, or has already been discretized. Let $\mathcal{A} = \{a_1, \ldots, a_m\}$ be a set of discrete attributes, where each attribute a_i has a finite domain $\Omega_i = [n_i]$. The data universe is

the Cartesian product $\mathcal{X} = \Omega_1 \times \cdots \times \Omega_m$, and a dataset D is a multiset of n records drawn from $\mathcal{X}$, written $D = \{x^{(1)}, \ldots, x^{(n)}\}$ with each $x^{(i)} \in \mathcal{X}$.

Let $\gamma \subseteq [m]$ be a subset of attribute indices with domain $\Omega_\gamma = \times_{i \in \gamma} \Omega_i$. The marginal query function $M_\gamma : \mathcal{X}^n \to \mathbb{R}^{|\Omega_\gamma|}$ computes the vector of counts for each configuration in Ω_γ:

$$M_\gamma(D) = \left(\sum_{x \in D} \mathbb{I}(x_\gamma = t) \right)_{t \in \Omega_\gamma}, \tag{1}$$

where x_γ is the restriction of record x to the attributes in γ.

The true marginal is $y_\gamma = M_\gamma(D)$, and a noisy measurement under Gaussian noise is $\tilde{y}_\gamma = y_\gamma + \xi$, where $\xi \sim \mathcal{N}(0, \sigma^2 I)$.

A workload $\mathcal{W} = \{\gamma_1, \ldots, \gamma_L\}$ is a collection of marginal queries of interest, representing the queries a private mechanism was designed to answer. We evaluate a synthetic dataset D_F against the original private data using the *Mean Workload Error*:

$$\text{Error}(D_F) = \frac{1}{n \cdot |\mathcal{W}|} \sum_{\gamma \in \mathcal{W}} \|y_\gamma - M_\gamma(D_F)\|_1. \tag{2}$$

2.2 Differential Privacy and Post-Processing

Two datasets $D, D' \in \mathbb{D}$ are *neighboring* if they differ by the addition or removal of at most one record. A randomized mechanism $\mathcal{M}$ satisfies (ϵ, δ)-differential privacy [5] if, for all neighboring D, D' and all measurable output sets S,

$$\Pr[\mathcal{M}(D) \in S] \leq e^\epsilon \Pr[\mathcal{M}(D') \in S] + \delta.$$

A key property of DP is *post-processing immunity*: a function of the output of an (ϵ, δ)-DP mechanism is itself (ϵ, δ)-DP, incurring no additional privacy cost. Since PRIME operates on the outputs of a DP mechanism, it inherits those privacy guarantees and requires no further privacy accounting.

2.3 Residual Queries and Correlated Noise

A marginal over a set of attributes τ can be used to derive the value of any lower-order marginal over $\gamma \subseteq \tau$. This shared information complicates the efficient measurement of a workload of queries. Residual queries address this by orthogonally decomposing each marginal into components that isolate the pure higher-order interaction specific to τ, excluding all lower-order contributions. This non-overlapping structure enables more efficient measurement allocation and is the foundation of both RESIDUAL-PLANNER [22] and AIM+GREM [7].

We adopt the formulation of residual queries from Fuentes et al. [7], which is equivalent to [22, 18]. Two linear operators are central to this framework. The *decomposition operator* `decomp` extracts the residual $r_\tau = \texttt{decomp}(y_\tau)$ from a marginal via a matrix A_τ formed as a Kronecker product of subtraction matrices. The *reconstruction map* `recon` inverts this transformation, projecting a residual back into marginal space via the Moore-Penrose pseudoinverse A_τ^+ (see Section A).

Residual mechanisms measure $\tilde{r}_\tau = \texttt{decomp}(y_\tau + \xi) = r_\tau + \texttt{decomp}(\xi)$, where $\xi \sim \mathcal{N}(0, \sigma^2 I)$, so $\tilde{r}_\tau \sim \mathcal{N}(r_\tau, \Sigma_\tau)$ with:

$$\Sigma_\tau = \sigma^2 A_\tau A_\tau^\top. \tag{3}$$

The residual covariance Σ_τ is non-diagonal, reflecting the correlations introduced by the decomposition operator. Crucially, Σ_τ possesses a structure inherited from the underlying linear operators. In Section 3.3, we show that the Mahalanobis loss under Σ_τ can be evaluated efficiently using this structure (Lemma 1). By exploiting this structure, PRIME accounts for the correlated noise without the prohibitive cost of explicitly forming or operating on the covariance matrix.

2.4 Maximum Entropy Inference and the Treewidth Bottleneck

Let $\mathcal{P}$ be the space of all possible distributions over the domain $\mathcal{X}$ and let $\mathcal{L}(P)$ be an objective function that quantifies the discrepancy between P and the observed data. There are many common

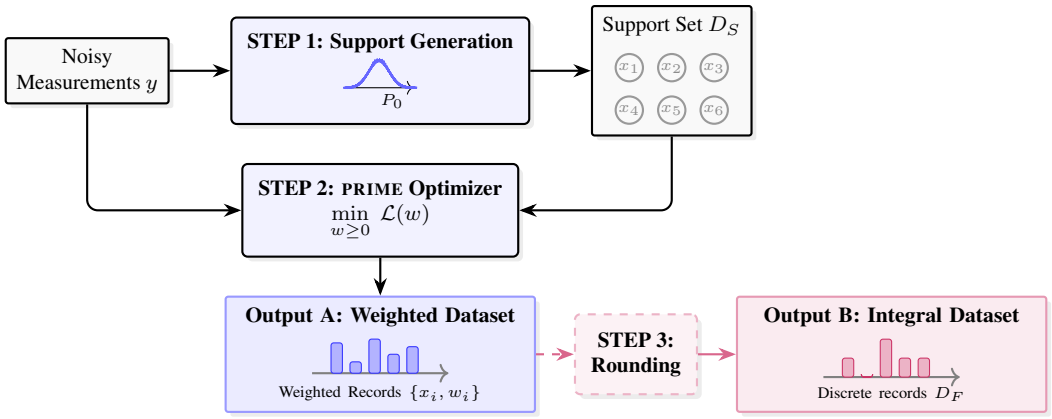


Figure 1: The PRIME workflow: (1) support records D_S are proposed via a distribution P_0, (2) weights w are fit to noisy measurements y, (3) a discrete dataset D_F is sampled.

cases where the minimizer of $\mathcal{L}(P)$ will not be unique. For instance, if we measure only two one-way marginals for binary variables X_1 and X_2 (e.g., $P(X_1 = 1) = 0.5$ and $P(X_2 = 1) = 0.5$), any joint distribution that preserves these marginals is a valid minimizer. This includes the independent case, perfect correlation, or any distribution in between, creating an infinite set of solutions indistinguishable to the objective function. To resolve this, we seek the maximum-entropy (MaxEnt) solution:

$$P^* = \underset{P \in \Omega}{\operatorname{argmax}} H(P), \quad \text{where } \Omega = \{P \in \mathcal{P} : \mathcal{L}(P) \text{ is minimized}\}. \tag{4}$$

This tie-breaking choice is referred to as the maximum-entropy principle and is a canonical choice for resolving under-determined problems [10].

When the objective $\mathcal{L}(P)$ factors over a collection of marginal measurements $\mathcal{M}$, the MaxEnt solution is a Markov Random Field (MRF) whose graph structure $\mathcal{G}$ is defined by the measurements: an edge exists between two variables X_i and X_j if and only if they co-occur in at least one marginal in the objective function. PRIVATE-PGM [15] utilizes this property by optimizing over the family of MRFs with that graph structure to arrive at the MaxEnt distribution. The computational complexity of computing gradients for this MRF is exponential in the *treewidth* of $\mathcal{G}$—the size of the largest clique in its junction tree minus one. Modern query answering mechanisms such as RESIDUAL-PLANNER and AIM+GREM challenge this paradigm by measuring exhaustive or highly overlapping sets of attribute interactions. This results in dense measurement graphs where the treewidth scales with the number of attributes d, causing memory and time requirements for inference to become intractable.

3 Private Re-weighting of Initial Model Estimates

The input to PRIME is a collection of noisy measurements derived from a differentially private mechanism. We assume the mechanism provides a set of noisy marginals $\mathcal{C}_{\text{marg}} = \{(\gamma_i, \tilde{y}_i, \sigma_i^2)\}$ and/or noisy residuals $\mathcal{C}_{\text{res}} = \{(\tau_j, \tilde{r}_j, \sigma_j^2)\}$, where each tuple contains a query identifier, a noisy observation, and the associated noise variance. PRIME produces a synthetic dataset D_F via three stages: (1) a finite support D_S is sampled from a reference model; (2) a weight vector w^* over D_S is obtained by solving a convex optimization problem; and (3) a discrete integer dataset D_F is sampled from the weighted support (D_S, w^*). Sections 3.1 to 3.6 describe each component in turn.

3.1 Problem Formulation: Restricted Maximum Entropy Synthesis

To overcome the treewidth bottleneck described in Section 2.4, PRIME adopts the MaxEnt objective but restricts the optimization domain. Rather than searching the exponentially large space of all distributions $\mathcal{P}$—which requires the structure of an MRF—we optimize over the space of distributions supported strictly on a finite set of samples $D_S = \{x_1, \ldots, x_K\}$.

We represent a distribution over this support as a weight vector $w \in \mathbb{R}^K_{\geq 0}$, where each w_k corresponds to the weight assigned to sample $x_k \in D_S$. Our primary task is to find a weight vector that minimizes

the discrepancy between expected measurements under w and the observed noisy data:

$$w^* = \underset{w \in \mathbb{R}^K_{\geq 0},\, \sum_k w_k = n}{\operatorname{argmin}} \mathcal{L}(w). \tag{5}$$

When the objective $\mathcal{L}(w)$ is under-determined, we invoke the Principle of Maximum Entropy to select the least committal weights among the set of minimizers Ω:

$$w^* = \underset{w \in \Omega}{\operatorname{argmax}} H(w), \quad \text{where } H(w) = -\sum_{k=1}^{K} \frac{w_k}{n} \log \frac{w_k}{n}. \tag{6}$$

PRIME serves as a tractable approximation of PRIVATE-PGM that optimizes for the same objective but restricts the domain to D_S. This formulation is independent of the measurement graph structure. By optimizing over weights rather than MRF parameters, the complexity of our solver is determined by the support size K and the cost of computing marginals over D_S, bypassing the treewidth bottleneck. Section 3.6 quantifies the cost of this restriction, bounding how far $\mathcal{L}(\hat{w})$ can fall from the full-domain optimum PRIVATE-PGM targets.

3.2 Support Initialization

The support $D_S = \{x^{(1)}, \ldots, x^{(K)}\}$ is the finite set of candidate records over which the optimization is performed. Its quality has an irrecoverable effect on synthesis: any record absent from D_S can never receive positive weight, meaning coverage gaps cannot be corrected downstream.

The selection of D_S involves two primary trade-offs. First, for a fixed generative source, larger supports improve coverage but increase the per-iteration optimization cost, which scales as $\mathcal{O}(K \cdot |\mathcal{M}|)$. Second, there is a trade-off between the complexity of the support generator and the required sample density. A simple model is computationally cheap to train and sample from but may require a massive K to ensure the support covers high-density regions of the true distribution. Conversely, a more sophisticated generator would require relatively fewer samples to achieve high fidelity. Because PRIME is modular, D_S can be drawn from any generative source; we study this in Section 4.3.

Our default support module is a product distribution P_0 built from 1-way marginals estimated from the base mechanism's measurements. This choice is theoretically grounded: P_0 is the MaxEnt distribution consistent with the 1-way marginals, making it a principled starting point using information already available from every base mechanism we consider. To construct the support, we sample $\tilde{K} = c \cdot \hat{n}$ records from P_0 (where $\hat{n}$ is the estimated dataset size and c is a support multiplier studied in Section H) and remove duplicates to obtain the final set D_S.

3.3 Losses for Marginal and Residual Measurements

For a weight vector $w \in \mathbb{R}^K_{\geq 0}$, we extend the marginal function to the weighted support as:

$$M_\gamma(D_S, w) = \left(\sum_{k=1}^{K} w_k \, \mathbb{I}(x_\gamma^{(k)} = t) \right)_{t \in \Omega_\gamma}. \tag{7}$$

Minimizing the Gaussian NLL for measurements with independent noise reduces to a weighted sum of squared errors:

$$\mathcal{L}_{\text{marg}}(w) = \frac{1}{2} \sum_{(\gamma_i, \tilde{y}_i, \sigma_i^2) \in \mathcal{C}_{\text{marg}}} \frac{1}{\sigma_i^2} \| M_{\gamma_i}(D_S, w) - \tilde{y}_i \|_2^2. \tag{8}$$

3.3.1 Residual Loss

Residual measurements involve correlated noise with covariance $\Sigma_j = \sigma_j^2 A_{\tau_j} A_{\tau_j}^\top$ (Equation (3)). For these measurements, the negative log-likelihood (NLL) is given by the squared Mahalanobis distance: $\|\tilde{r}_j - v\|^2_{\Sigma_j^{-1}} = (\tilde{r}_j - v)^\top \Sigma_j^{-1} (\tilde{r}_j - v)$, where v is the predicted residual vector. Naive computation of this loss requires forming and inverting Σ_j, costing $\mathcal{O}(|\Omega_{\tau_j}|^3)$—a complexity that is prohibitive for residuals involving multiple attributes. However, this cost is entirely avoidable.

Lemma 1 (Marg-space error)**.** *Let A_τ be the residual matrix with $\Sigma_\tau = \sigma^2 A_\tau A_\tau^\top$. For any vector v, the Mahalanobis distance can be computed via the marginal-space reconstruction:*

$$\|\tilde{r}_\tau - v\|^2_{\Sigma_\tau^{-1}} = \frac{1}{\sigma^2}\|\texttt{recon}(\tilde{r}_\tau - v)\|_2^2. \tag{9}$$

Proof. See Section B. □

By Lemma 1, the Mahalanobis loss reduces to a squared ℓ_2 norm after applying `recon`—a tensor operation costing only $\mathcal{O}(|\tau| \cdot n_\tau)$ [7]. Since no dense matrix is ever formed or inverted, residual loss evaluations become as efficient as marginal loss evaluations. Summing over all residual measurements $\mathcal{C}_{\text{res}}$, the total residual loss is:

$$\mathcal{L}_{\text{res}}(w) = \frac{1}{2}\sum_{(\tau_j,\, \tilde{r}_j,\, \sigma_j^2) \in \mathcal{C}_{\text{res}}} \frac{1}{\sigma_j^2}\left\|\texttt{recon}\big(\texttt{decomp}\big(M_{\tau_j}(D_S, w)\big) - \tilde{r}_j\big)\right\|_2^2 \tag{10}$$

`decomp` projects the predicted weighted marginal $M_{\tau_j}(D_S, w)$ into residual space, the difference with the noisy measurement $\tilde{r}_j$ is taken, and `recon` maps back before the squared ℓ_2 norm is computed, with the result scaled by $\frac{1}{\sigma_j^2}$ to account for noise variance.

3.4 Optimization via Entropy-Regularized L-BFGS

As noted in Section 3.1, $\mathcal{L}(w)$ may admit multiple minimizers, so we introduce an entropy regularizer and solve the strictly convex problem

$$w^*(\lambda) = \underset{w \geq 0,\, \sum_k w_k = n}{\operatorname{argmin}} \Big[\mathcal{L}(w) + \lambda \sum_{k=1}^{K} w_k \log w_k\Big], \quad \lambda > 0, \tag{11}$$

which has a unique global minimizer for every $\lambda > 0$. The following lemma establishes that this family of problems is consistent with the MaxEnt objective:

Lemma 2 (Regularization path)**.** *As $\lambda \to 0^+$, $w^*(\lambda)$ converges to the maximum-entropy minimizer of $\mathcal{L}(w)$ over $\{w \geq 0,\, \sum_k w_k = n\}$.*

Proof. See Section D.1. □

Lemma 2 motivates solving (11) for a decreasing sequence $\lambda_1 > \lambda_2 > \cdots \to 0$, warm-starting each stage from the previous solution [21]. We use L-BFGS [12] as the inner solver; its superlinear convergence on smooth, strongly convex objectives makes it well-suited to the regularized subproblems. In our implementation, we fix λ to a small constant and run a single L-BFGS solve, we also scale the objective based both on the number of records in the dataset and the number of measurements to make the regularization weight transferable across settings. Full optimizer settings and implementation details are provided in Section F.

Additional Details. Entropic mirror descent (EMD) is an alternative that implicitly recovers the MaxEnt minimizer without explicit regularization; we describe it in Section G. In Section D we show that each L-BFGS iteration requires $\mathcal{O}(K \cdot |\mathcal{M}|)$ work, with no treewidth dependence. PRIVATE-PGM, by contrast, scales as $\mathcal{O}(\exp(\text{tw}(G)))$ per iteration.

3.5 Discrete Sampling

The weight vector w^* produced by L-BFGS is continuous; most downstream uses require a discrete dataset D_F of exact integer counts summing to n. We convert w^* to integer counts via *systematic sampling* [4]: we normalize w^* to probabilities, compute the CDF, and place n evenly spaced evaluation points along it with a single shared uniform random offset, assigning each point to a record. This bounds rounding error to at most ± 1 per record and minimizes sampling variance. Full details are given in Section F.

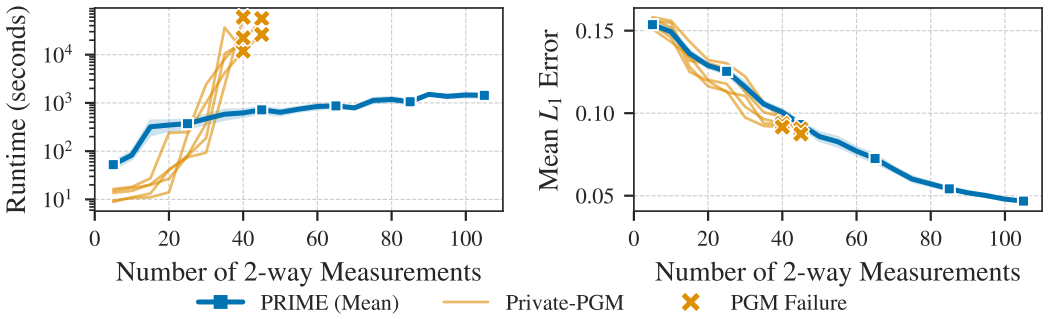


Figure 2: Computational scaling and utility of PRIME vs. PRIVATE-PGM as measurement graph density increases (Gaussian noise $\sigma = 10$). **Left:** PRIVATE-PGM runtime scales exponentially with treewidth until failure, while PRIME maintains tractability on fully connected graphs. **Right:** PRIME utility (mean L_1 error on 3-way marginals) continues to improve as more measurements are processed, well beyond PRIVATE-PGM's failure point.

3.6 Theoretical Guarantee

Section 3.1 restricts PRIVATE-PGM's MaxEnt objective (Section 2.4) from the full domain $\mathcal{P}$ to distributions supported on D_S. We now bound the cost of that restriction. Write $\mathcal{L}^\star_{\text{PGM}} = \min_{P \in \mathcal{P}} \mathcal{L}(P)$ for the optimal value of the full-domain objective, and let $\hat{p}_{\text{PGM}} \in \mathcal{P}$ be any minimizer—e.g., the MaxEnt distribution PRIVATE-PGM returns. Let $\hat{w}$ be any minimizer of Equation (5); every such minimizer attains the same value $\mathcal{L}(\hat{w})$, and by Lemma 2 this value is the $\lambda \to 0^+$ limit of the objective PRIME actually solves in Section 3.4. Let $D_2 = D_2(\hat{p}_{\text{PGM}} \| P_0)$ be the Rényi divergence between $\hat{p}_{\text{PGM}}$ and the support generator P_0 (Section 3.2), defined by $e^{D_2} = \sum_{x \in \mathcal{X}} \hat{p}_{\text{PGM}}(x)^2 / P_0(x)$, and let C_Q bound the largest loss contribution a single record can make to the stacked measurements: $C_Q \le n^2 \sum_i \sigma_i^{-2}$ (sum over all marginal and residual measurements), with equality when every measurement is a marginal—a residual's contribution is strictly smaller by Lemma 1, since `decomp` followed by `recon` is an orthogonal projection. Formal definitions are in Section E.

Assumption 1 (Absolute Continuity)**.** $\text{supp}(\hat{p}_{PGM}) \subseteq \text{supp}(P_0)$.

Because $\mathcal{X}$ is finite, Assumption 1 already implies $e^{D_2} < \infty$; it holds for any full-support P_0, including the product distribution of Section 3.2 whenever its one-way estimates are strictly positive.

Theorem 1 (Expected Optimization Gap)**.** *Let D_S be formed from $\tilde{K}$ i.i.d. draws from P_0, before deduplication (Section 3.2). Under Assumption 1, if $\tilde{K} \ge 8(e^{D_2} - 1)$ and $\mathcal{L}^\star_{PGM} \le C_Q$, then*

$$\mathcal{L}^\star_{PGM} \;\le\; \mathbb{E}[\mathcal{L}(\hat{w})] \;\le\; \mathcal{L}^\star_{PGM} + \frac{28\, C_Q\, e^{D_2}}{\tilde{K}}. \tag{12}$$

The lower bound holds because $\hat{w}/n$, viewed as a distribution on D_S, is feasible for the full-domain problem defining $\mathcal{L}^\star_{\text{PGM}}$. The upper bound is proved in Section E, where Theorem 2 gives a sharper version that does not require $\mathcal{L}^\star_{\text{PGM}} \le C_Q$.

The gap in (12) scales inversely with the *effective sample size* $K_{\text{eff}} = \tilde{K}/e^{D_2}$: support quality enters only through e^{D_2}, the mismatch between P_0 and $\hat{p}_{\text{PGM}}$, and as $\tilde{K} \to \infty$ (for fixed D_2), $\mathbb{E}[\mathcal{L}(\hat{w})]$ matches $\mathcal{L}^\star_{\text{PGM}}$ exactly. This bound is on the measurement loss $\mathcal{L}$ that PRIME optimizes directly, prior to the discrete sampling step of Section 3.5; it does not by itself bound the workload error of Equation (2) against the private dataset D, since D need not satisfy Assumption 1 relative to P_0. The ordering of support generators we observe empirically in Section 4.3 is consistent with this picture: a better-matched P_0 (lower D_2) tightens the guarantee at any fixed $\tilde{K}$.

4 Experiments

To evaluate the empirical performance of PRIME, we design experiments that address three key questions: (1) Does it scale gracefully to high-treewidth regimes where exact inference fails? (2) Can

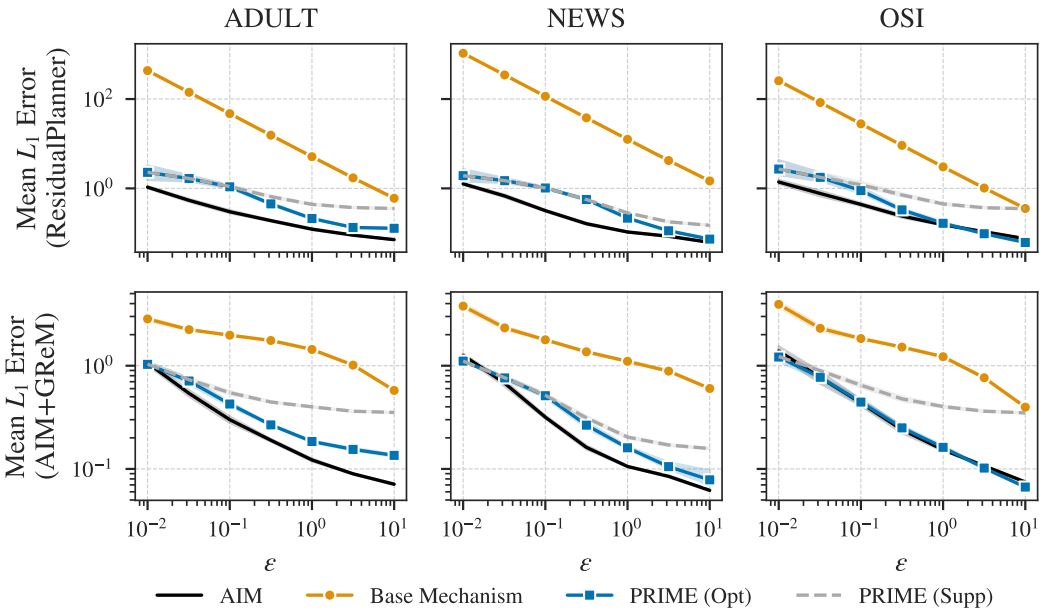


Figure 3: Average workload error of PRIME applied to RESIDUAL-PLANNER and AIM+GREM ($\delta = 10^{-9}$). PRIME (**OPT**, blue) consistently improves upon the base query answers (orange) and the unoptimized support data (**Supp**, grey), closing the gap with the PRIVATE-PGM-based AIM (black).

PRIME generate high-quality synthetic data from dense query-answering mechanisms? (3) How does the choice of support generator impact the final utility?

Experimental Setup. Unless otherwise specified, all reported error metrics represent the mean L_1 error evaluated across a workload of all 3-way marginals, and all shaded regions in our figures indicate the standard error across five independent trials. The one-way marginals used for the product distribution support generator are maximum likelihood estimates based on the noisy measurements of the base mechanism, truncated to enforce non-negativity and re-scaled to sum to the estimated number of records. Across all evaluations, PRIME is run with the L-BFGS optimizer and an entropy regularization weight of $\lambda = 10^{-8}$. This hyperparameter is analyzed in Section G.3.

4.1 Reconstruction Scaling

The theoretical motivation for PRIME is to bypass the treewidth bottleneck inherent to graphical-model synthesis. To test this empirically, we measure both the runtime and the resulting utility of PRIME and PRIVATE-PGM as the density of the measurement graph increases. Using the ADULT dataset, we construct a workload by iteratively accumulating 2-way marginal queries in batches of five, drawn uniformly at random without replacement from the set of all possible 2-way marginals. To isolate the computational scaling from any specific mechanism's privacy allocation, we apply independent Gaussian noise with a fixed scale of $\sigma = 10$ to every measured marginal. This value was chosen to inject a moderate level of noise relative to the dataset size ($n \approx 49,000$), ensuring a fixed baseline of uncertainty while largely preserving the underlying signal. Both algorithms were evaluated subject to a memory limit of 32GB and a timeout threshold of twelve hours. All experiments were run on the Unity Research Computing Platform, with 2 CPU cores and 32GB of memory per job.

As shown in Figure 2, the empirical results illustrate both the computational bottleneck of exact inference and the utility benefits of bypassing it. In the left panel, PRIVATE-PGM scales exponentially with the number of measurements and consistently fails—either by exceeding the memory limit or timing out—once the graph reaches approximately 40 measurements. In contrast, the runtime of PRIME scales linearly with the number of measurements, confirming that it can comfortably handle fully connected, high-treewidth measurement graphs. Furthermore, the right panel demonstrates the benefit of this scalability: by continuing to process more measurements that render PRIVATE-PGM intractable, PRIME is able to further drive down the mean L_1 error on 3-way marginals.

4.2 Synthetic Data Quality

We first assess whether PRIME can successfully repurpose the dense measurement graphs of state-of-the-art query-answering mechanisms into high-quality synthetic data. We apply PRIME to the private measurements generated by RESIDUAL-PLANNER and AIM+GREM, evaluating on the ADULT [11], NEWS [6], and OSI [20] datasets. For the PRIME support module, we use a product distribution support multiplier of $c = 10$. Since PRIVATE-PGM is not capable of generating synthetic data from these measurements, we compare against the state of the art synthetic data mechanism AIM [17, 3] which uses PRIVATE-PGM for synthesis. AIM selects marginal measurements that will maximize the quality of synthetic data while maintaining a tractable measurement graph for PRIVATE-PGM.

Figure 3 summarizes the mean workload error across the full privacy spectrum. Across all datasets and both base mechanisms, PRIME consistently produces synthetic data with substantially lower error than the base mechanisms' own direct query answers. Furthermore, PRIME closes much of the utility gap between these query-answering mechanisms and AIM. On the OSI dataset, PRIME even matches the synthetic data quality of AIM. These results are particularly notable because RESIDUAL-PLANNER and AIM+GREM are optimized strictly for query answering, yet PRIME extracts highly competitive synthetic data from their published measurements without any specialized tuning. A key driver of this performance is the quality of the initial support: we observe that the loss of the support before optimization is already an improvement over the baseline mechanism. This underscores the importance of the initialization strategy in PRIME, providing a foundation that allows the subsequent optimization to refine the more specific interactions in the measurements.

4.3 The Role of the Support Module

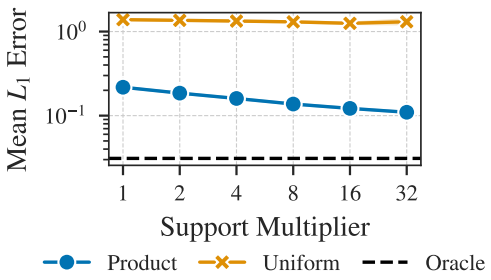


Figure 4: Support generator ablation using RESIDUAL-PLANNER measurements ($\epsilon = 3.16$, ADULT). The default **Product** distribution consistently outperforms a **Uniform** random baseline across all support multipliers. A non-private **Oracle** support ($K = n$) is included as a strong upper bound.

Because PRIME utilizes a modular architecture, the quality of the final synthetic dataset is inherently linked to the quality of the initial support, D_S. To isolate and quantify this contribution, we hold the weight optimization phase constant and vary the source and size of the support records. We evaluate PRIME on measurements from RESIDUAL-PLANNER at a fixed privacy budget of $\epsilon = 3.16$ using the ADULT dataset. We compare three support conditions: a Uniform baseline where records are sampled uniformly at random from the data universe $\mathcal{X}$; our default Product distribution, which is constructed directly from the mechanism's published 1-way marginals without requiring any additional privacy budget; and an Oracle support where records are sampled directly from the true private dataset. For the Uniform and Product supports, we vary the support size, represented as the support multiplier (K/n). Meanwhile, the Oracle support is held constant with a support multiplier of 1 (i.e., $K = n$). The Oracle support violates privacy and is included strictly as a strong upper bound.

Figure 4 demonstrates that the choice of support generator has a profound impact on final utility. While the default Product support vastly outperforms the weak Uniform baseline across all support fractions, a measurable performance gap remains between the Product distribution and the constant Oracle support. This indicates that while PRIME effectively extracts utility via optimization, its ultimate performance ceiling is dictated by the support's coverage of the true joint distribution. This ordering is consistent with Section 3.6: both Uniform and Product have full support, so Assumption 1 holds and each has a well-defined, finite Rényi divergence D_2 to PRIVATE-PGM's target $\hat{p}_{\text{PGM}}$; Product is built from $\hat{p}_{\text{PGM}}$'s own 1-way marginals, so it is intuitively closer to $\hat{p}_{\text{PGM}}$ (smaller D_2) than the Uniform baseline, and the guaranteed gap shrinks as e^{D_2} does. The Oracle support, by contrast, is drawn from the private dataset D itself; since D need not satisfy Assumption 1 relative to P_0, Oracle's advantage falls outside the scope of Theorem 1 and instead reflects that it is evaluated on the same distribution it is drawn from.

5 Related Work

PRIVATE-PGM [15] is the foundational graphical-model synthesis method, performing exact MaxEnt inference over a Markov random field of measurements via the junction tree algorithm. AIM [17] builds on PRIVATE-PGM with an adaptive, data-dependent measurement selection strategy, using PRIVATE-PGM as its synthesis backend. MST [16] sidesteps the treewidth problem by construction, restricting measurements to a spanning tree of the measurement graph; this guarantees tractability but limits which measurements can be incorporated. Other synthesizers are often introduced in the context of a novel end to end synthetic data mechanism such as the projection-based approach of RAP [1] or the generator network used in GEM [14]. These other synthesizers could be adapted to accept measurements with correlated noise; however their model structure makes optimization non-convex and remove any kind of guarantee to MaxEnt convergence.

MWEM [8] is an online multiplicative weights (MW) algorithm that maintains a synthetic dataset supported by the entire data domain, making it intractable for most real-world datasets. The most closely related prior work is PMW$^{\text{PUB}}$ [13], which uses a collection of public data as its support and iteratively re-weights samples via MW to match private query answers, sharing a similar high-level paradigm as PRIME. However, it relies on the availability of external, in-distribution public data. Private Post-GAN Boosting [19] applies MW re-weighting over samples from the sequence of generators produced during GAN training. It mitigates the convergence issues related to private GAN training and was applicable to image domains but it inherits the limitations of private GAN generators which typically do not perform well in tabular data benchmarks [3]. PRIME distinguishes itself from all three in several ways. First, it does not require external public data and is effective given a simple generative model. Second, it replaces the online sequential MW update with a single batch convex optimization, which is more natural when all measurements are available upfront and explicitly handles correlated residual noise via Lemma 1.

Recent work has shown that large language models can generate schema-consistent pseudo-records without access to real public data [9], making LLM-derived surrogate data a candidate for PRIME's support module. This connects naturally to PRIME's modular design: any generative sources can be plugged in as the support generator without modifying the inference step.

6 Conclusion

A growing tension has emerged between the sophistication of modern DP query-answering mechanisms and the tools available for synthesis. While mechanisms like RESIDUAL-PLANNER and AIM+GREM achieve high accuracy by generating dense measurements, these same structures render graphical-model synthesis via PRIVATE-PGM computationally intractable. PRIME resolves this by pursuing the same MaxEnt objective via support-based convex optimization with $\mathcal{O}(K \cdot |\mathcal{M}|)$ complexity, independent of treewidth. Empirically, PRIME generates high-quality synthetic data, unlocks synthesis from previously unusable measurements, and scales where PRIVATE-PGM is infeasible, broadening the set of query answering mechanisms from which synthesis is achievable.

That said, performance is sensitive to support quality: the product distribution is a practical default but leaves a measurable gap relative to oracle support. Section 3.6 formalizes this gap: it shrinks at rate $\mathcal{O}(e^{D_2}/K)$, with support quality entering only through D_2, the mismatch between the support generator and the target distribution, and vanishes as $K \to \infty$. Closing this gap is therefore the most direct path to further improvement, and several directions are promising, each reducible to lowering D_2 or tolerating a larger K. LLM-based surrogate data [9] can generate schema-consistent pseudo-records from domain descriptions without access to real data. Tree-structured graphical models offer a more expressive alternative to the product distribution while remaining tractable. Finally, adaptive support refinement—iteratively concentrating resampling of D_S in high-probability regions identified by w^*, or pruning samples assigned very low weight—could improve support quality without requiring a more expressive generator. Together, these directions suggest a path toward DP synthetic data generation that is simultaneously expressive and scalable.

Another key direction for future work is integrating PRIME into an end-to-end synthetic data mechanism. In the same way that AIM was designed to account for the limitations and strengths of the PRIVATE-PGM synthesizer, new iterative synthetic data pipelines should be developed to select measurements that maximize the potential of the PRIME synthesizer.

Acknowledgments and Disclosure of Funding

The computational resources for this work were provided by the University of Massachusetts Amherst's partnership with the Unity Research Computing Platform, a multi-institutional cluster led by the University of Massachusetts and the University of Rhode Island. This material is based upon work supported by the National Science Foundation under Grant No. #2210979.

References

[1] Sergul Aydore, William Brown, Michael Kearns, Krishnaram Kenthapadi, Luca Melis, Aaron Roth, and Ankit A. Siva. Differentially private query release through adaptive projection. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 457–467. PMLR, 2021.

[2] Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Oper. Res. Lett.*, 31(3):167–175, May 2003. ISSN 0167-6377. doi: 10.1016/S0167-6377(02)00231-6. URL https://doi.org/10.1016/S0167-6377(02)00231-6.

[3] Kai Chen, Xiaochen Li, Chen Gong, Ryan Mckenna, and Tianhao Wang. Benchmarking differentially private tabular data synthesis: [experiments & analysis]. *Proc. ACM Manag. Data*, 3(6), December 2025. doi: 10.1145/3769764. URL https://doi.org/10.1145/3769764.

[4] Arnaud Doucet, Nando de Freitas, and Neil J. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Statistics for Engineering and Information Science. Springer New York, New York, NY, 2001. ISBN 978-0-387-95146-1.

[5] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography*, pages 265–284. Springer Berlin Heidelberg, 2006. ISBN 9783540327325. doi: 10/d4xx6w.

[6] Kelwin Fernandes, Pedro Vinagre, and Paulo Cortez. A proactive intelligent decision support system for predicting the popularity of online news. In Francisco Pereira, Penousal Machado, Ernesto Costa, and Amílcar Cardoso, editors, *Progress in Artificial Intelligence*, pages 535–546, Cham, 2015. Springer International Publishing. ISBN 978-3-319-23485-4.

[7] Miguel Fuentes, Brett Mullins, Yingtai Xiao, Daniel Kifer, Cameron N Musco, and Daniel Sheldon. Fast private adaptive query answering for large data domains. In Emtiyaz Khan, Yingzhen Li, Arno Solin, and Aaditya Ramdas, editors, *Proceedings of The 29th International Conference on Artificial Intelligence and Statistics*, volume 300 of *Proceedings of Machine Learning Research*, pages 3790–3798. PMLR, 02–05 May 2026. URL https://proceedings.mlr.press/v300/fuentes26a.html.

[8] Moritz Hardt and Guy N. Rothblum. A multiplicative weights mechanism for privacy-preserving data analysis. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 61–70. IEEE, October 2010. doi: 10/c9s2jd.

[9] Shlomi Hod, Lucas Rosenblatt, and Julia Stoyanovich. Do you really need public data? surrogate public data for differential privacy on tabular data, 2025. URL https://arxiv.org/abs/2504.14368.

[10] E. T. Jaynes. Information theory and statistical mechanics. *Phys. Rev.*, 106:620–630, May 1957. doi: 10.1103/PhysRev.106.620. URL https://link.aps.org/doi/10.1103/PhysRev.106.620.

[11] Ron Kohavi et al. Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD)*, volume 96, pages 202–207, 1996.

[12] Dong C. Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1):503–528, Aug 1989. ISSN 1436-4646. doi: 10.1007/BF01589116. URL https://doi.org/10.1007/BF01589116.

[13] Terrance Liu, Giuseppe Vietri, Thomas Steinke, Jonathan Ullman, and Steven Wu. Leveraging public data for practical private query release. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 6968–6977. PMLR, July 2021. doi: 10/g97pwz.

[14] Terrance Liu, Giuseppe Vietri, and Steven Z. Wu. Iterative methods for private synthetic data: Unifying framework and new methods. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 690–702. Curran Associates, Inc., 2021. doi: 10/g97pwr.

[15] Ryan Mckenna, Daniel Sheldon, and Gerome Miklau. Graphical-model based estimation and inference for differential privacy. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4435–4444. PMLR, June 2019. doi: 10/g97pwn.

[16] Ryan McKenna, Gerome Miklau, and Daniel Sheldon. Winning the NIST contest: A scalable and general approach to differentially private synthetic data. *Journal of Privacy and Confidentiality*, 11(3), December 2021. ISSN 2575-8527. doi: 10/g97pwp.

[17] Ryan McKenna, Brett Mullins, Daniel Sheldon, and Gerome Miklau. AIM: an adaptive and iterative mechanism for differentially private synthetic data. *Proc. VLDB Endow.*, 15(11): 2599–2612, July 2022. ISSN 2150-8097. doi: 10.14778/3551793.3551817. URL `https://doi.org/10.14778/3551793.3551817`.

[18] Brett Mullins, Miguel Fuentes, Yingtai Xiao, Daniel Kifer, Cameron Musco, and Daniel Sheldon. Efficient and private marginal reconstruction with local non-negativity. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 141356–141389. Curran Associates, Inc., 2024.

[19] Marcel Neunhoeffer, Steven Wu, and Cynthia Dwork. Private post-GAN boosting. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL `https://openreview.net/forum?id=6isfR3JCbi`.

[20] C Sakar and Yomi Kastro. Online shoppers purchasing intention dataset, 2018.

[21] Lin Xiao and Tong Zhang. A proximal-gradient homotopy method for the l1-regularized least-squares problem. In *Proceedings of the 29th International Conference on International Conference on Machine Learning*, ICML'12, page 1227–1234, Madison, WI, USA, 2012. Omnipress. ISBN 9781450312851.

[22] Yingtai Xiao, Guanlin He, Danfeng Zhang, and Daniel Kifer. An optimal and scalable matrix mechanism for noisy marginals under convex loss functions. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 20495–20539. Curran Associates, Inc., 2023.

A More on Residual Queries

Here we present residual operations and their running time as formulated in Fuentes et al. [7].

A.1 Construction of the Residual Matrix

Residual queries provide a mechanism to decompose the dense information contained within standard marginal queries into orthogonal, non-overlapping components. A standard marginal query over a set of attributes τ inherently contains redundant information from all lower-order attribute subsets; residuals isolate the pure interactions specific to exactly τ. To achieve this, we define the linear decomposition operator `decomp` and its corresponding reconstruction operator `recon`.

Let y_τ be the flattened marginal vector over attributes τ. The decomposition function is defined by the matrix-vector product:

$$\texttt{decomp}(y_\tau) = A_\tau y_\tau, \tag{13}$$

where A_τ is the residual transformation matrix. The reconstruction function maps a residual vector r_τ back into marginal space via the Moore-Penrose pseudoinverse of A_τ:

$$\texttt{recon}(r_\tau) = A_\tau^+ r_\tau. \tag{14}$$

For a single attribute k with discrete domain of size n_k, we define the subtraction matrix $S_{(n_k)} \in \mathbb{R}^{(n_k-1)\times n_k}$ as:

$$S_{(n_k)} = \begin{pmatrix} -1 & 1 & 0 & \cdots & 0 \\ -1 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -1 & 0 & 0 & \cdots & 1 \end{pmatrix} \in \mathbb{R}^{(n_k-1)\times n_k}. \tag{15}$$

The full residual transformation matrix A_τ is the Kronecker product of the subtraction matrices for each attribute in τ:

$$A_\tau = \bigotimes_{k\in\tau} S_{(n_k)}, \tag{16}$$

and the reconstruction matrix is correspondingly:

$$A_\tau^+ = \bigotimes_{k\in\tau} S_{(n_k)}^+. \tag{17}$$

These operations can also be extended so that a residual over attributes τ can be derived from any marginal over attributes $\gamma \supseteq \tau$; however, those operations are not needed for this work and are omitted for simplicity.

A.2 In-Axis Implementation of `decomp` and `recon`

While A_τ and A_τ^+ provide a clean algebraic formalism for the projection properties used in Section C, directly constructing these Kronecker matrices and performing matrix-vector multiplication becomes memory-intensive and error-prone for multi-attribute residuals.

Instead, `decomp` and `recon` are implemented natively as a sequence of efficient in-axis operations on multi-dimensional arrays, equivalent to tensor n-mode products. Rather than flattening the marginal into a 1D vector, it is preserved as an $|\tau|$-dimensional array. Applying the subtraction matrix $S_{(n_k)}$ algebraically is computationally equivalent to applying a simple differencing operation strictly along the k-th axis of this array.

By exploiting this multi-dimensional structure, the implementation avoids instantiating large Kronecker matrices entirely. This reduces the time complexity of both decomposition and reconstruction to $\mathcal{O}(|\tau| \cdot n_\tau)$, where $n_\tau = \prod_{k\in\tau} n_k$, allowing the framework to scale efficiently to high-dimensional residual queries.

B Proof of Lemma 1

Lemma (Marg-space error, restated from Lemma 1)**.** *Let A_τ be the residual matrix that transforms the marginal y_τ to the residual r_τ. Let $\tilde{r}_\tau = r_\tau +$ `decomp`(ξ) where $\xi \sim \mathcal{N}(0, \sigma^2 I)$, and define*

$\Sigma_\tau := \sigma^2 A_\tau A_\tau^\mathsf{T}$. *Then, for any estimate* v*,*

$$\|\tilde{r}_\tau - v\|^2_{\Sigma_\tau^{-1}} = \frac{1}{\sigma^2}\|\texttt{recon}(\tilde{r}_\tau - v)\|_2^2. \tag{18}$$

Proof. Let $u = \tilde{r}_\tau - v$. The reconstruction operation is the linear transformation given by the pseudoinverse of the residual matrix, $\texttt{recon}(u) = A_\tau^+ u$. The squared Mahalanobis norm of u with respect to Σ_τ^{-1} is:

$$\|u\|^2_{\Sigma_\tau^{-1}} = u^\mathsf{T}\Sigma_\tau^{-1}u = u^\mathsf{T}\big(\sigma^2 A_\tau A_\tau^\mathsf{T}\big)^+ u = \frac{1}{\sigma^2}u^\mathsf{T}\big(A_\tau A_\tau^\mathsf{T}\big)^+ u. \tag{19}$$

Applying the pseudoinverse identity $(MM^\mathsf{T})^+ = (M^+)^\mathsf{T}M^+$:

$$\|u\|^2_{\Sigma_\tau^{-1}} = \frac{1}{\sigma^2}u^\mathsf{T}\Big((A_\tau^+)^\mathsf{T}A_\tau^+\Big)u \tag{20}$$

$$= \frac{1}{\sigma^2}(A_\tau^+ u)^\mathsf{T}(A_\tau^+ u) \tag{21}$$

$$= \frac{1}{\sigma^2}\|\texttt{recon}(u)\|_2^2. \tag{22}$$

This completes the proof. □

C Gradient Derivation

To efficiently solve the optimization problem in Section 3, we derive the gradient of the full objective with respect to the support weights w. For a given query γ and fixed support D_S of size K, let $M_\gamma \in \{0,1\}^{|\Omega_\gamma|\times K}$ be the binary indicator matrix where $(M_\gamma)_{t,k} = 1$ if $x_\gamma^{(k)} = t$ and 0 otherwise, so that the estimated marginal is the matrix-vector product $M_\gamma w$.

C.1 Marginal Loss Gradient

Using the indicator matrix representation, the gradient of the marginal loss follows straightforwardly from the squared error form:

$$\nabla_w \mathcal{L}_{\text{marg}}(w) = \sum_{(\gamma_i,\,\tilde{y}_i,\,\sigma_i^2)\,\in\,\mathcal{C}_{\text{marg}}} \frac{1}{\sigma_i^2}\,M_{\gamma_i}^\mathsf{T}(M_{\gamma_i}w - \tilde{y}_i). \tag{23}$$

C.2 Residual Loss Gradient

For the residual loss, recall that `decomp` and `recon` have matrix representations A_{τ_j} and $A_{\tau_j}^+$ respectively. Let $\Delta_j(w) = A_{\tau_j}(M_{\tau_j}w) - \tilde{r}_j$ denote the residual error at weight vector w. Applying the chain rule to the Mahalanobis loss:

$$\nabla_w \mathcal{L}_{\text{res}}(w) = \sum_{(\tau_j,\,\tilde{r}_j,\,\sigma_j^2)\,\in\,\mathcal{C}_{\text{res}}} \frac{1}{\sigma_j^2}(A_{\tau_j}M_{\tau_j})^\mathsf{T}(A_{\tau_j}^+)^\mathsf{T}A_{\tau_j}^+\big(\Delta_j(w)\big) \tag{24}$$

$$= \sum_{(\tau_j,\,\tilde{r}_j,\,\sigma_j^2)\,\in\,\mathcal{C}_{\text{res}}} \frac{1}{\sigma_j^2}\,M_{\tau_j}^\mathsf{T}(A_{\tau_j}^+A_{\tau_j})^\mathsf{T}A_{\tau_j}^+\big(\Delta_j(w)\big). \tag{25}$$

Because $A_{\tau_j}^+$ is the Moore-Penrose pseudoinverse of A_{τ_j}, the product $A_{\tau_j}^+A_{\tau_j}$ is an orthogonal projection matrix and therefore symmetric: $(A_{\tau_j}^+A_{\tau_j})^\mathsf{T} = A_{\tau_j}^+A_{\tau_j}$. Furthermore, by the pseudoinverse identity $A_{\tau_j}^+A_{\tau_j}A_{\tau_j}^+ = A_{\tau_j}^+$, the projection collapses:

$$(A_{\tau_j}^+A_{\tau_j})\,A_{\tau_j}^+\big(\Delta_j(w)\big) = A_{\tau_j}^+\big(\Delta_j(w)\big). \tag{26}$$

The gradient therefore reduces to:

$$\nabla_w \mathcal{L}_{\text{res}}(w) = \sum_{(\tau_j,\,\tilde{r}_j,\,\sigma_j^2)\,\in\,\mathcal{C}_{\text{res}}} \frac{1}{\sigma_j^2}\,M_{\tau_j}^\mathsf{T}\,A_{\tau_j}^+\big(\Delta_j(w)\big). \tag{27}$$

Back-propagating through the residual loss thus reduces to projecting the residual error back into marginal space via `recon`, followed by a transpose gather through the indicator matrix.

C.3 Entropy Regularization Gradient

The entropy regularization term in Equation (11) is $\mathcal{L}_{\text{ent}}(w) = \lambda \sum_k w_k \log w_k$, with gradient:

$$\nabla_w \mathcal{L}_{\text{ent}}(w) = \lambda\big(\mathbf{1} + \log w\big), \tag{28}$$

where the logarithm is applied element-wise.

D Additional Proofs

Proposition 1 (Per-iteration complexity)**.** *Each iteration of both EMD (Equation* (48)*) and L-BFGS (Equation* (11)*) costs $\mathcal{O}(K \cdot |\mathcal{M}|)$, where $K = |D_S|$ is the support size and $|\mathcal{M}| = |\mathcal{C}_{marg}| + |\mathcal{C}_{res}|$ is the total number of measurements. This bound is independent of the treewidth $\mathrm{tw}(G)$ of the measurement graph.*

Proof. We work under the following assumption, which holds in the practical regime where the support is sized proportionally to the dataset and queries are defined over low-order attribute subsets:

Assumption 2. *The marginal domain size satisfies $\max_i |\Omega_{\gamma_i}| = \mathcal{O}(K)$, and for residual queries, $\max_j |\tau_j| \cdot n_{\tau_j} = \mathcal{O}(K)$, where $n_{\tau_j} = |\Omega_{\tau_j}|$.*

Every iteration of both optimizers requires $\mathcal{O}(1)$ evaluations of the objective $\mathcal{L}(w)$ and its gradient $\nabla_w \mathcal{L}(w)$: EMD requires exactly one, and L-BFGS requires at most $\texttt{maxls} = 50$ evaluations per line search, a fixed constant independent of K or $|\mathcal{M}|$. The gradient decomposes additively over all measurements (Section C), so it suffices to bound the cost of a single measurement's contribution and multiply by $|\mathcal{M}|$.

Fix a measurement $i \in \mathcal{M}$ over query attributes γ_i (or τ_i for a residual), with marginal operator M_{γ_i} implemented as described in Section F.2. One gradient contribution involves three sequential operations.

1. **Forward pass (scatter-add).** The estimated marginal $\mu_i = M_{\gamma_i} w \in \mathbb{R}^{|\Omega_{\gamma_i}|}$ is computed by accumulating each of the K support weights into its corresponding bin via a vectorized scatter-add. The cost is $\mathcal{O}(K)$, since a static index map of length K is traversed once, independent of $|\Omega_{\gamma_i}|$.

2. **Error computation.** For a *marginal* measurement, the error vector $\Delta_i = \mu_i - \tilde{y}_i \in \mathbb{R}^{|\Omega_{\gamma_i}|}$ is computed in $\mathcal{O}(|\Omega_{\gamma_i}|)$. For a *residual* measurement, computing Δ_i additionally requires applying `decomp` and `recon` to project the estimated marginal into and out of residual space; by Section A.2, each in-axis tensor operation costs $\mathcal{O}(|\tau_i| \cdot n_{\tau_i})$. Under Assumption 2, both $\mathcal{O}(|\Omega_{\gamma_i}|)$ and $\mathcal{O}(|\tau_i| \cdot n_{\tau_i})$ are absorbed into $\mathcal{O}(K)$, so the error computation costs $\mathcal{O}(K)$ in both cases.

3. **Backward pass (gather).** The gradient contribution $M_{\gamma_i}^\top \Delta_i \in \mathbb{R}^K$ is computed by reading from each support record's corresponding marginal bin via a vectorized gather over the same static index map. This costs $\mathcal{O}(K)$.

Each of the three steps costs $\mathcal{O}(K)$, so one measurement contributes $\mathcal{O}(K)$ to the gradient. Summing over all $|\mathcal{M}|$ measurements gives a total gradient evaluation cost of $\mathcal{O}(K \cdot |\mathcal{M}|)$.

EMD update step. After the gradient evaluation, the multiplicative weight update (Equation (48)) applies an elementwise exponential and a single normalizing sum over the K-dimensional weight vector, costing $\mathcal{O}(K)$. The per-iteration cost is therefore $\mathcal{O}(K \cdot |\mathcal{M}|)$.

L-BFGS update step. The L-BFGS two-loop recursion operates on the K-dimensional gradient vector using a sliding window of m stored curvature pairs, where $m = 100$ is a fixed constant (Section F.3). This costs $\mathcal{O}(m \cdot K) = \mathcal{O}(K)$. The softmax reparameterization and its chain-rule correction (Section F.3) are likewise $\mathcal{O}(K)$. The per-iteration cost is therefore also $\mathcal{O}(K \cdot |\mathcal{M}|)$.

In both cases the bound contains no factor involving $\mathrm{tw}(G)$. The total runtime follows by multiplying by the number of iterations, which is analysed for EMD and L-BFGS respectively in Sections 3.4 and G.1. □

Comparison with PRIVATE-PGM. PRIVATE-PGM performs inference by running belief propagation on the junction tree of the measurement graph G [15], where nodes represent attributes and edges connect every pair of attributes that co-occur in at least one measurement. The dominant cost per belief-propagation sweep is marginalizing messages over the maximal cliques of the junction tree. By definition of treewidth, the largest such clique contains $\mathrm{tw}(G)+1$ nodes; assuming uniform per-attribute domain sizes $n_k = \bar{n}$ for simplicity, enumerating all joint configurations of this clique costs

$$\mathcal{O}\Big(\bar{n}^{\,\mathrm{tw}(G)+1}\Big) = \mathcal{O}\big(\exp\big((\mathrm{tw}(G)+1)\cdot\log\bar{n}\big)\big)\,,$$

which is exponential in the treewidth.

For modern mechanisms such as RESIDUAL-PLANNER [22] and AIM+GREM [7], which measure exhaustive or highly overlapping sets of attribute interactions, the resulting measurement graph is dense and $\mathrm{tw}(G)$ grows with the number of attributes d. The junction-tree inference therefore becomes intractable on standard hardware, even though the DP measurements themselves encode high-utility information.

D.1 Proof of Lemma 2

Let $R(w) = \sum_{k=1}^{K} w_k \log w_k$ (the negative entropy), let $L^* = \min_{w\in\Delta} \mathcal{L}(w)$ denote the minimum NLL value over the feasible set $\Delta = \{w \in \mathbb{R}^K : w \geq 0,\ \sum_k w_k = n\}$, and let $S^* = \{w \in \Delta : \mathcal{L}(w) = L^*\}$ be the set of NLL minimizers. We adopt the convention $0\log 0 = 0$, under which R is continuous on Δ.

We record several preliminary observations before proceeding to the main argument. The feasible set Δ is compact (it is closed and bounded in $\mathbb{R}^K$) and convex, so the minimum L^* is attained and S^* is non-empty. The set S^* is convex: since $\mathcal{L}$ is convex, $S^* = \{w \in \Delta : \mathcal{L}(w) \leq L^*\}$ is a sublevel set of a convex function intersected with the convex set Δ, and is therefore convex. For each $\lambda > 0$, the objective $\mathcal{L}(w) + \lambda R(w)$ is strictly convex on Δ, since R is strictly convex (the function $x \mapsto x\log x$ is strictly convex, and R is a sum of such functions); hence the minimizer $w^*(\lambda)$ over Δ is unique.

Step 1: Any limit point of $w^*(\lambda)$ lies in S^*. Since Δ is compact, every sequence $\lambda_j \to 0^+$ has a convergent subsequence; let $w^*(\lambda_j) \to \bar{w} \in \Delta$ along such a subsequence. By optimality of $w^*(\lambda_j)$, for any fixed $w \in S^*$:

$$\mathcal{L}(w^*(\lambda_j)) + \lambda_j R(w^*(\lambda_j)) \;\leq\; \mathcal{L}(w) + \lambda_j R(w) \;=\; L^* + \lambda_j R(w). \tag{29}$$

Taking $j \to \infty$: the left-hand side converges to $\mathcal{L}(\bar{w})$ by continuity of $\mathcal{L}$, and the term $\lambda_j R(w^*(\lambda_j))$ vanishes since R is bounded on the compact set Δ; the right-hand side converges to L^* since $\lambda_j R(w) \to 0$. We therefore obtain $\mathcal{L}(\bar{w}) \leq L^*$, and since L^* is the minimum of $\mathcal{L}$ over Δ, equality holds and $\bar{w} \in S^*$.

Step 2: Every limit point minimizes R over S^*. Since $\mathcal{L}(w^*(\lambda_j)) \geq L^*$, we can extend the inequality in (29) to the chain

$$L^* + \lambda_j R(w^*(\lambda_j)) \;\leq\; \mathcal{L}(w^*(\lambda_j)) + \lambda_j R(w^*(\lambda_j)) \;\leq\; L^* + \lambda_j R(w) \qquad \forall\, w \in S^*, \tag{30}$$

where the first inequality uses $L^* \leq \mathcal{L}(w^*(\lambda_j))$ and the second is (29). Subtracting L^* and dividing by $\lambda_j > 0$ gives

$$R(w^*(\lambda_j)) \leq R(w) \qquad \forall\, w \in S^*. \tag{31}$$

Taking $j \to \infty$ and using continuity of R, we obtain $R(\bar{w}) \leq R(w)$ for all $w \in S^*$, so $\bar{w}$ minimizes negative entropy (equivalently, maximizes entropy) over S^*.

Step 3: Uniqueness and convergence of the full net. Because R is strictly convex, its minimizer over the convex set S^* is unique; call it $w^\dagger$. Steps 1 and 2 show that every convergent subsequence of $w^*(\lambda)$ converges to $w^\dagger$. Since $\{w^*(\lambda)\}_{\lambda>0}$ lies in the compact set Δ, every subsequence has a further convergent subsequence, and all such subsequences share the same limit $w^\dagger$; therefore the entire net converges: $w^*(\lambda) \to w^\dagger$ as $\lambda \to 0^+$. □

E Expected Loss Bound for PRIME

We now state and prove a finite-sample guarantee comparing PRIME's unregularized objective value to that of the full-domain minimizer targeted by PRIVATE-PGM.

Notation bridge. Let $P \in \Delta_{\mathcal{X}}$ denote a probability distribution over the data universe (viewed as a vector in $\mathbb{R}^{|\mathcal{X}|}$), and extend Equation (1) to distributions via $M_\gamma(P) = \big(\sum_{x:x_\gamma=t} P(x)\big)_{t\in\Omega_\gamma}$. Stack all marginal and residual measurements from $\mathcal{C}_{\text{marg}} \cup \mathcal{C}_{\text{res}}$ (Section 3) into a single linear operator Q and target vector y: the block of Q for marginal γ_i is $n \cdot M_{\gamma_i}(\cdot)$, the block for residual τ_j is $n \cdot \texttt{decomp}(M_{\tau_j}(\cdot))$, and y stacks $(\tilde{y}_i)_i$ and $(\tilde{r}_j)_j$ in the same order; the factor of n converts a probability distribution into the same units (expected record counts) as the measurements. Let Σ be the block-diagonal covariance with blocks $\sigma_i^2 I$ for each marginal and $\Sigma_{\tau_j} = \sigma_j^2 A_{\tau_j} A_{\tau_j}^\top$ (Equation (3)) for each residual, and define

$$L(P) = \frac{1}{2}\|QP - y\|^2_{\Sigma^{-1}}. \tag{32}$$

For a support $D_S = \{x^{(1)}, \ldots, x^{(K)}\}$ and weights $w \in \mathbb{R}^K_{\geq 0}$ summing to n, let $w/n \in \Delta_K$ denote the corresponding probability vector over D_S (summing weights of duplicate support points), viewed as an element of $\Delta_{\mathcal{X}}$ supported on D_S. Then $Q(w/n)$ recovers exactly the weighted marginal and residual estimates of Equation (7), so $\mathcal{L}(w) = L(w/n)$: the objective of Section 3.3 is the restriction of L to distributions supported on D_S, written as a function of w with the slight abuse of notation of Section 3. Note that L is exactly Section 2.4's objective $\mathcal{L}(P)$ made concrete for Gaussian measurement noise; we use the unscripted L only to distinguish the full-domain argument P from the support-restricted argument w of Section 3. Write $\hat{p}_{\text{PGM}} = \operatorname{argmin}_{P\in\Delta_{\mathcal{X}}} L(P)$ for a full-domain minimizer of this objective, i.e., the distribution PRIVATE-PGM (Section 2.4) targets, with optimal value $\mathcal{L}^\star_{\text{PGM}} := L(\hat{p}_{\text{PGM}}) = \min_{P\in\Delta_{\mathcal{X}}} L(P)$ and measurement vector $\mu_{\text{PGM}} = Q\hat{p}_{\text{PGM}}$. Write $\hat{w} = \operatorname{argmin}_{w\geq 0,\, \sum_k w_k = n} \mathcal{L}(w)$ for a minimizer of PRIME's objective over D_S, so that $\hat{w}/n$ minimizes L over distributions supported on D_S; every such minimizer attains the same value $\mathcal{L}(\hat{w})$, since $\mathcal{L}$ is constant on its own argmin set. Write $R(x) = \hat{p}_{\text{PGM}}(x)/P_0(x)$ for the likelihood ratio between $\hat{p}_{\text{PGM}}$ and the support generator P_0 (Section 3.2), $q(x)$ for the column of Q corresponding to record x, and $C_Q = \max_{x\in\mathcal{X}} \|q(x)\|^2_{\Sigma^{-1}}$ for the maximum squared Mahalanobis column norm. For a marginal measurement γ_i, $q(x)$ contributes the block $n \cdot e_{x_{\gamma_i}}$, a scaled standard basis vector, so its squared Mahalanobis contribution is exactly n^2/σ_i^2 for every x. For a residual measurement τ_j, the corresponding block is $n \cdot \texttt{decomp}(e_{x_{\tau_j}})$, and by Lemma 1 its squared Mahalanobis contribution is $(n^2/\sigma_j^2)\|A^+_{\tau_j} A_{\tau_j} e_{x_{\tau_j}}\|_2^2 \leq n^2/\sigma_j^2$, since $A^+_{\tau_j} A_{\tau_j}$ is an orthogonal projection. Summing over all measurements gives $C_Q \leq n^2 \sum_i \sigma_i^{-2}$ (the sum ranging over both marginal and residual measurements), with equality whenever $\mathcal{C}_{\text{res}} = \emptyset$.

We restate Assumption 1 from Section 3.6 for reference: absolute continuity of $\hat{p}_{\text{PGM}}$ with respect to P_0. Because $\mathcal{X}$ is finite, Assumption 1 already implies finiteness of $e^{D_2(\hat{p}_{\text{PGM}}\|P_0)} = \sum_{x\in\mathcal{X}} \hat{p}_{\text{PGM}}(x)^2/P_0(x)$: the sum has at most $|\mathcal{X}|$ terms, each finite since $P_0(x) > 0$ wherever $\hat{p}_{\text{PGM}}(x) > 0$. Assumption 1 can fail in practice: the one-way estimates used to build P_0 are truncated to enforce non-negativity (Section 4) without a strictly positive floor, so a configuration with $\hat{p}_{\text{PGM}}(x) > 0$ can receive $P_0(x) = 0$ if its estimate truncates to exactly zero.

We additionally adopt the sample-size condition

$$\tilde{K} \geq 8\big(e^{D_2} - 1\big), \qquad D_2 := D_2(\hat{p}_{\text{PGM}}\|P_0). \tag{33}$$

This condition is not needed for the correctness of any step of the proof below; it is needed for the bound to be meaningful. By Chebyshev's inequality in Step 2, $P(G^c) \leq 4(e^{D_2} - 1)/\tilde{K}$; this quantity is then substituted, inside a square root, into the Step 2 bias bound. Without (33), that fraction can exceed 1, so the substitution would bound a probability by a number greater than 1 — a vacuous step that artificially inflates the final constant. Condition (33) is exactly the requirement that keeps this bound on $P(G^c)$ at most $1/2$, making the substitution and the resulting constant meaningful. Throughout the proof below we write $K := \tilde{K}$ for brevity.

Theorem 2 (Expected Optimization Gap)**.** *Under Assumption 1 and condition* (33)*, with $\tilde{K}$ taken to be the number of i.i.d. draws from P_0 used to build D_S before deduplication (Section 3.2),*

$$\mathbb{E}[\mathcal{L}(\hat{w})] \leq \mathcal{L}^\star_{PGM} + \frac{8\sqrt{2\,\mathcal{L}^\star_{PGM}\,C_Q}\;e^{D_2}}{\tilde{K}} + \frac{16\,C_Q\,e^{D_2}}{\tilde{K}}. \tag{34}$$

Theorem (restated from Theorem 1)**.** *If $\mathcal{L}^\star_{PGM} \leq C_Q$, then*

$$\mathcal{L}^\star_{PGM} \;\leq\; \mathbb{E}[\mathcal{L}(\hat{w})] \;\leq\; \mathcal{L}^\star_{PGM} + \frac{28\,C_Q\,e^{D_2}}{\tilde{K}}. \tag{35}$$

Proof. The lower bound holds because $\hat{w}/n$ is feasible for the full-domain problem defining $\mathcal{L}^\star_{\text{PGM}} = \min_{P\in\Delta_\mathcal{X}} L(P)$. For the upper bound, substituting $\mathcal{L}^\star_{\text{PGM}}/C_Q \le 1$ into (34) gives coefficient $16 + 8\sqrt{2} \le 28$. □

The gap scales as C_Q/K_{eff}, where $K_{\text{eff}} = \tilde{K}/e^{D_2}$ is the effective sample size: support quality enters only through e^{D_2}, the mismatch between P_0 and $\hat{p}_{\text{PGM}}$.

Why $\tilde{K}$ is the pre-deduplication draw count. Section 3.2 draws $\tilde{K} = c \cdot \hat{n}$ i.i.d. records from P_0 and removes duplicates to form D_S. Duplicate draws contribute identical columns to Q, so merging their weights never changes the attainable set $\{Qw : w \in \Delta_{\tilde{K}}\}$: the minimum of L over weights on the full multiset of $\tilde{K}$ draws therefore equals its minimum over weights on the deduplicated support used in practice. Theorem 2 thus applies with the pre-deduplication draw count $\tilde{K}$, rather than the smaller, random number of unique points remaining after deduplication; the proof below writes $K := \tilde{K}$, as noted above.

Proof of Theorem 2. Step 1: Self-normalized oracle construction. Let $x_1, \dots, x_K \overset{\text{i.i.d.}}{\sim} P_0$ be the draws comprising D_S before deduplication. Define unnormalized importance weights $\tilde{w}_k = \frac{1}{K} R(x_k)$, their total mass $S = \|\tilde{w}\|_1$, and the self-normalized oracle

$$\bar{w} = \begin{cases} \tilde{w}/S & S > 0, \\ \frac{1}{K}\mathbf{1} & S = 0, \end{cases}$$

so that $\bar{w} \in \Delta_K$ in either case. Write $\tilde{\mu} = Q_S \tilde{w}$ and $\bar{\mu} = Q_S \bar{w}$, where Q_S is Q restricted to the columns indexed by $x_1, \dots, x_K$. By standard importance sampling, $\mathbb{E}[\tilde{\mu}] = \mu_{\text{PGM}}$ and $\mathbb{E}[S] = 1$ (Assumption 1 ensures R is well-defined P_0-a.s.). Since $\hat{w}/n$ minimizes L over Δ_K and $\bar{w} \in \Delta_K$ is feasible but not necessarily optimal,

$$\mathbb{E}[\mathcal{L}(\hat{w})] = \mathbb{E}[L(\hat{w}/n)] \le \mathbb{E}[L(\bar{w})]. \tag{36}$$

Step 2: Quadratic expansion. Since L is exactly quadratic in the measurements, expand around μ_{PGM}:

$$\mathbb{E}[L(\bar{w})] = L(\hat{p}_{\text{PGM}}) + \left\langle \Sigma^{-1}(\mu_{\text{PGM}} - y),\, \mathbb{E}[\bar{\mu}] - \mu_{\text{PGM}} \right\rangle + \frac{1}{2}\mathbb{E}\big[\|\bar{\mu} - \mu_{\text{PGM}}\|^2_{\Sigma^{-1}}\big]. \tag{37}$$

It remains to bound the bias $\|\mathbb{E}[\bar{\mu}] - \mu_{\text{PGM}}\|_{\Sigma^{-1}}$ and the second moment $\mathbb{E}[\|\bar{\mu} - \mu_{\text{PGM}}\|^2_{\Sigma^{-1}}]$. Throughout, partition on $G = \{S \ge 1/2\}$ and $G^c = \{S < 1/2\}$. On G^c, both $\bar{\mu}$ and μ_{PGM} are convex combinations of columns of Q (the former since $\bar{w} \in \Delta_K$ at every S, including $S = 0$; the latter since $\hat{p}_{\text{PGM}} \in \Delta_\mathcal{X}$), so by the triangle inequality, convexity, and Jensen's inequality, $\|\bar{\mu} - \mu_{\text{PGM}}\|_{\Sigma^{-1}} \le 2\sqrt{C_Q}$ deterministically, regardless of whether $S = 0$.

Second moment. Let $\eta = \tilde{\mu} - S\mu_{\text{PGM}}$, so $\mathbb{E}[\eta] = 0$, and on G, $\bar{\mu} - \mu_{\text{PGM}} = \eta/S$. Since the x_k are i.i.d. and $\mathbb{E}_{P_0}[R(x)(q(x) - \mu_{\text{PGM}})] = \mu_{\text{PGM}} - \mu_{\text{PGM}} = 0$, cross terms vanish and

$$\mathbb{E}[\|\eta\|^2_{\Sigma^{-1}}] = \frac{1}{K}\,\mathbb{E}_{P_0}\big[R(x)^2 \|q(x) - \mu_{\text{PGM}}\|^2_{\Sigma^{-1}}\big] \le \frac{4\,C_Q\,e^{D_2}}{K}, \tag{38}$$

using $\|q(x) - \mu_{\text{PGM}}\|^2_{\Sigma^{-1}} \le 4C_Q$ and $\mathbb{E}_{P_0}[R(x)^2] = e^{D_2}$, finite since $\mathcal{X}$ is finite and Assumption 1 holds. On G, since $S \ge 1/2 \Rightarrow 1/S^2 \le 4$,

$$\mathbb{E}\Big[\|\eta/S\|^2_{\Sigma^{-1}} \mathbf{1}_G\Big] \le 4\,\mathbb{E}[\|\eta\|^2_{\Sigma^{-1}}] \tag{39}$$

$$\le \frac{16\,C_Q\,e^{D_2}}{K}. \tag{40}$$

On G^c, the deterministic bound above gives $\mathbb{E}[\|\bar{\mu} - \mu_{\text{PGM}}\|^2 \mathbf{1}_{G^c}] \le 4C_Q \cdot P(G^c)$; Chebyshev's inequality with $\mathrm{Var}(S) = (e^{D_2} - 1)/K$ gives $P(G^c) = P(S - 1 < -1/2) \le 4(e^{D_2} - 1)/K$, so $\mathbb{E}[\|\bar{\mu} - \mu_{\text{PGM}}\|^2 \mathbf{1}_{G^c}] \le 16C_Q(e^{D_2} - 1)/K$. Summing the two contributions,

$$\mathbb{E}\big[\|\bar{\mu} - \mu_{\text{PGM}}\|^2_{\Sigma^{-1}}\big] \le \frac{32\,C_Q\,e^{D_2}}{K}. \tag{41}$$

Bias. Writing $\delta = S - 1$, the identity $\mathbb{E}[\bar{\mu}] - \mu_{\text{PGM}} = -\mathbb{E}[(\bar{\mu} - \mu_{\text{PGM}})\,\delta]$ holds on all of the sample space: on G it follows from $1/S = 1 - \delta/S$ applied to $\bar{\mu} - \mu_{\text{PGM}} = \eta/S$ together with $\mathbb{E}[\eta] = 0$; on G^c, including at $S = 0$ where $\delta = -1$, it is immediate from the definition of $\bar{\mu} - \mu_{\text{PGM}}$ and involves no division by S. On G, by Cauchy–Schwarz with $\|\bar{\mu} - \mu_{\text{PGM}}\|_{\Sigma^{-1}} = \|\eta\|_{\Sigma^{-1}}/S \le 2\|\eta\|_{\Sigma^{-1}}$,

$$\begin{aligned}\|\mathbb{E}[(\bar{\mu} - \mu_{\text{PGM}})\delta\,\mathbf{1}_G]\|_{\Sigma^{-1}} &\le 2\sqrt{\mathbb{E}[\|\eta\|^2_{\Sigma^{-1}}]}\,\sqrt{\mathbb{E}[\delta^2]} \\ &\le 2 \cdot \frac{2\sqrt{C_Q\,e^{D_2}}}{\sqrt{K}} \cdot \frac{\sqrt{e^{D_2} - 1}}{\sqrt{K}} \le \frac{4\sqrt{C_Q}\,e^{D_2}}{K}.\end{aligned}$$

On G^c, by the deterministic bound $\|\bar{\mu} - \mu_{\text{PGM}}\|_{\Sigma^{-1}} \le 2\sqrt{C_Q}$ and Cauchy–Schwarz,

$$\begin{aligned}\|\mathbb{E}[(\bar{\mu} - \mu_{\text{PGM}})\delta\,\mathbf{1}_{G^c}]\|_{\Sigma^{-1}} &\le 2\sqrt{C_Q}\,\mathbb{E}[|\delta|\mathbf{1}_{G^c}] \\ &\le 2\sqrt{C_Q}\sqrt{\mathbb{E}[\delta^2]}\sqrt{P(G^c)} \\ &\le \frac{4\sqrt{C_Q}\,(e^{D_2} - 1)}{K} \le \frac{4\sqrt{C_Q}\,e^{D_2}}{K}.\end{aligned}$$

By the triangle inequality on the two contributions,

$$\|\mathbb{E}[\bar{\mu}] - \mu_{\text{PGM}}\|_{\Sigma^{-1}} \le \frac{8\sqrt{C_Q}\,e^{D_2}}{K}. \tag{42}$$

Assembly. By Cauchy–Schwarz, the first-order term is bounded by

$$\left|\langle \Sigma^{-1}(\mu_{\text{PGM}} - y), \mathbb{E}[\bar{\mu}] - \mu_{\text{PGM}}\rangle\right| \le \|\mu_{\text{PGM}} - y\|_{\Sigma^{-1}} \cdot \|\mathbb{E}[\bar{\mu}] - \mu_{\text{PGM}}\|_{\Sigma^{-1}} \le \sqrt{2L(\hat{p}_{\text{PGM}})} \cdot \frac{8\sqrt{C_Q}\,e^{D_2}}{K}.$$

Substituting this and (41) into the quadratic expansion of Step 2 and combining with (36) yields (34). □

Remark 1. *The constant* 28 *is not sharp: asymptotically, for large* $K_{\mathit{eff}} = \tilde{K}/e^{D_2}$*, the central limit theorem for self-normalized estimators gives an effective constant approaching* $1/2$*. The stated constant is the price of a finite-sample guarantee that holds uniformly for every* $\tilde{K}$ *satisfying* (33)*, rather than only in the large-*K_{eff} *limit.*

From the $\lambda = 0$ minimizer to the solved problem. Theorem 2 bounds $\hat{w}$, a minimizer of $\mathcal{L}$ over D_S at $\lambda = 0$. PRIME instead solves the entropy-regularized problem (11) for a small $\lambda > 0$. By Lemma 2, the regularized minimizer $w^*(\lambda)$ converges to the maximum-entropy element of $\operatorname{argmin}_w \mathcal{L}(w)$ as $\lambda \to 0^+$; since $\mathcal{L}$ is constant on $\operatorname{argmin}_w \mathcal{L}(w)$ and continuous, $\mathcal{L}(w^*(\lambda)) \to \mathcal{L}(\hat{w})$ along this limit, so Theorems 1 and 2 describe the $\lambda \to 0$ limit of the objective value PRIME actually optimizes.

F Implementation Details

F.1 Objective Scaling

In differentially private synthetic data generation, raw target measurements (marginal counts) scale directly with the number of records in the underlying dataset. This variation in magnitude means that without normalization, an optimization problem formulated for a dataset of $10,000$ records will have a drastically different loss landscape than one formulated for $10,000,000$ records. Such scale discrepancies severely condition the objective, leading to numerical instability in quasi-Newton methods like L-BFGS and preventing the reuse of optimization hyperparameters across different datasets.

To ensure the optimization problem remains well-conditioned and scale-invariant, we apply a two-tiered scaling strategy: a robust per-measurement normalization followed by a global objective scaling.

For a given noisy target measurement vector y with noise variance σ^2, we define a local scale factor $s = 1/(\max(|y|) + \epsilon)$, where ϵ is a small constant to prevent division by zero. This factor normalizes

the target values to approximately $\mathcal{O}(1)$. Both the target and its variance are mapped into this well-conditioned space:

$$\tilde{y} = s \cdot y \tag{43}$$

$$\tilde{\sigma}^2 = s^2 \cdot \sigma^2 \tag{44}$$

During optimization, the estimated measurement $\mu(w)$ derived from the current support weights w is identically scaled, yielding the $\mathcal{O}(1)$ error vector $\Delta = s \cdot \mu(w) - \tilde{y}$. The negative log-likelihood loss contribution for this specific measurement is evaluated in the scaled space as:

$$\mathcal{L}_{\text{local}}(w) = \frac{1}{2\tilde{\sigma}^2} \|\Delta\|_2^2 \tag{45}$$

While mathematically equivalent to the unscaled objective $\frac{1}{2\sigma^2}\|\mu(w) - y\|_2^2$, this scaled formulation prevents floating-point overflow during gradient accumulation. To map the gradient back to the raw weight space, the accumulated difference is scaled by an additional factor of $s/\tilde{\sigma}^2$ before applying the transposed linear operator.

Finally, to completely decouple the learning dynamics from the dataset size and the sheer number of queries, we apply a global scaling factor. Let n be the total target mass (number of records) and $|\mathcal{C}|$ be the total number of measurements in the workload. The aggregate unregularized objective is divided by $S_{\text{global}} = n \cdot |\mathcal{C}|$:

$$\mathcal{L}(w) = \frac{1}{S_{\text{global}}} \sum_{i \in \mathcal{C}} \mathcal{L}_{\text{local},i}(w) \tag{46}$$

By bounding the overall magnitude of the objective and its gradients, this global normalization ensures that the L-BFGS line search and tolerance criteria (e.g., `ftol` and `gtol`) behave consistently. Crucially, it also renders regularization parameters scale-invariant, allowing the same hyperparameter configurations to transfer reliably across datasets of vastly different sizes.

F.2 Gradient Computation via Static Index Mapping

To facilitate efficient computation of the objective and its gradient, we implement the transformations between the support weights and the marginal spaces using vectorized primitive operations. Prior to the optimization process, we pre-compute a static integer index mapping that assigns each record in the support set $\mathcal{D}_S$ to its corresponding bin in the target marginal space. During the forward pass, the marginal estimates are computed via a *scatter-add* operation, which accumulates record weights into their respective bins using the pre-computed indices. Conversely, the backward pass distributes gradients from the marginal space back to individual records using a *gather* operation, which retrieves the gradient value associated with each record's assigned bin via direct integer indexing.

This implementation strategy offers significant performance advantages over standard grouping or search-based methods by decoupling the spatial assignment of data from the optimization loop. By pre-calculating the mapping, we eliminate the recurring computational overhead of sorting or hashing that would otherwise be required to identify record-bin memberships in each iteration. Furthermore, scatter and gather operations are highly optimized within modern array libraries to maximize memory throughput and cache locality. This approach ensures that the projection complexity remains linear with respect to the number of support records while minimizing memory allocation penalties through the reuse of static index arrays.

F.3 Log-Space Parameterization

Directly optimizing over the constrained space $w \in \mathbb{R}^K_{\geq 0}$ can be numerically unstable. Instead, we parameterize via unconstrained logits $\theta \in \mathbb{R}^K$ and a softmax transformation:

$$w_k = n \cdot \frac{\exp(\theta_k)}{\sum_{i=1}^K \exp(\theta_i)}, \tag{47}$$

which unconditionally enforces both non-negativity and the size constraint $\sum_k w_k = n$. We utilize the scipy implementation of L-BFGS and configure it with maximum history size `maxcor`=100, line search limit `maxls`=50, projected gradient tolerance `gtol`=10^{-8}, and function evaluation tolerance `ftol`=10^{-10}.

F.4 Systematic Sampling Procedure

To synthesize a discrete dataset of exact size n from the optimized continuous weights w^*, we employ systematic sampling (stochastic universal sampling), which bounds rounding error to ± 1 per record and reduces variance substantially relative to multinomial sampling. The procedure is:

1. Normalize to a probability distribution: $p_k = w_k^* / \sum_{i=1}^{K} w_i^*$.
2. Compute the CDF: $C_k = \sum_{i=1}^{k} p_i$, with $C_0 = 0$.
3. Define step size $s = 1/n$ and sample a single uniform offset $u \sim U[0, s]$.
4. Construct n evenly spaced evaluation points: $P_j = u + j \cdot s$ for $j \in \{0, \ldots, n-1\}$.
5. For each P_j, select record $x^{(k)} \in D_S$ such that $C_{k-1} \leq P_j < C_k$.

The resulting collection of n records constitutes the final unweighted synthetic dataset D_F, such that each record, marginally, is drawn from the correct distribution.

F.5 Datasets

ADULT [11], NEWS [6], and OSI [20] are all obtained from the UCI Machine Learning Repository.

G Entropic Mirror Descent vs. L-BFGS

G.1 Entropic Mirror Descent

Mirror descent is a generalization of gradient descent designed for constrained optimization over non-Euclidean geometries. Rather than taking a step in the Euclidean direction of the gradient, mirror descent measures proximity using a *Bregman divergence* $D_\Phi(w\|v) = \Phi(w) - \Phi(v) - \langle \nabla\Phi(v),\, w - v\rangle$ induced by a strictly convex potential Φ, and updates by minimizing a linearization of the objective penalized by this divergence. The choice of Φ tailors the geometry of the updates to the constraint set.

Entropic mirror descent (EMD) [2] instantiates this framework by letting the negative entropy $\Phi(w) = \sum_k w_k \log w_k$ serve as the mirror-descent potential. This potential induces a geometry that respects the non-negativity and sum constraints on w, and yields the multiplicative weight update

$$w_k^{(t+1)} \;\propto\; w_k^{(t)} \exp\Big(-\eta\, \nabla_k \mathcal{L}\big(w^{(t)}\big)\Big), \qquad \textstyle\sum_k w_k^{(t+1)} = n, \tag{48}$$

applied to the composite loss $\mathcal{L}(w) = \mathcal{L}_{\mathrm{marg}}(w) + \mathcal{L}_{\mathrm{res}}(w)$.

MaxEnt property. A key property of EMD is its *implicit* maximum-entropy bias: initialized at the uniform weights $w^{(0)} = \frac{n}{K}\mathbf{1}$, it converges to the maximum-entropy minimizer of $\mathcal{L}(w)$ without any entropy regularization in the objective [2]. Intuitively, the negative-entropy Bregman divergence penalizes deviations from the current iterate in proportion to relative entropy; starting from the highest-entropy point on the simplex, the iterates move no further than the measurements require. PRIME inherits the guarantee over the finite support D_S.

Convergence. EMD converges at rate $\mathcal{O}(1/\sqrt{T})$ for convex objectives, with improved rates under strong convexity [2]. In practice, this first-order rate demands many iterations at the support sizes needed for low workload error, making EMD a useful theoretical baseline but an impractical optimizer; we quantify this gap in Section G.3.

G.2 Entropic Mirror Descent for PRIME

In the PRIME setting, EMD is applied to the NLL objective $\mathcal{L}(w) = \mathcal{L}_{\text{marg}}(w) + \mathcal{L}_{\text{res}}(w)$ under the simplex constraint $w \geq 0$, $\sum_k w_k = n$, initialized at the uniform point $w^{(0)} = \frac{n}{K}\mathbf{1}$. The multiplicative weight update at iteration t is:

$$w_k^{(t+1)} = n \cdot \frac{w_k^{(t)} \exp\big(-\eta\, \nabla_k \mathcal{L}(w^{(t)})\big)}{\sum_{i=1}^{K} w_i^{(t)} \exp\big(-\eta\, \nabla_i \mathcal{L}(w^{(t)})\big)}, \tag{49}$$

Table 1: KL Divergence from EMD solution, L1 Workload Error, and Computation Time for Entropic Mirror Descent (EMD) and L-BFGS across varying Support Sizes and Regularization Weights (λ). Values are mean $\pm$ standard error over 5 independent trials.

		Support Size		
Method	**Metric**	**48,842**	**122,105**	**244,210**
EMD	Error	0.2182 ± 0.0010	0.1828 ± 0.0011	0.1632 ± 0.0014
	Time (s)	$12,707.9 \pm 3,515.2$	$23,888.6 \pm 6,109.1$	$54,499.6 \pm 14,190.3$
LBFGS $\lambda = 10^{-10}$	KL Divergence	1.5386 ± 0.2391	2.0865 ± 0.0951	2.5655 ± 0.1337
	Error	0.2257 ± 0.0029	0.1856 ± 0.0010	0.1634 ± 0.0019
	Time (s)	256.5 ± 185.6	405.9 ± 102.0	983.3 ± 302.2
LBFGS $\lambda = 10^{-8}$	KL Divergence	0.7231 ± 0.0930	0.7714 ± 0.0998	0.7775 ± 0.0796
	Error	0.2174 ± 0.0026	0.1764 ± 0.0020	0.1515 ± 0.0019
	Time (s)	176.2 ± 45.6	340.7 ± 123.7	919.9 ± 190.3
LBFGS $\lambda = 10^{-6}$	KL Divergence	1.4343 ± 0.0109	1.6400 ± 0.0285	1.7534 ± 0.0168
	Error	0.3248 ± 0.0010	0.3197 ± 0.0008	0.3183 ± 0.0006
	Time (s)	10.6 ± 3.1	17.7 ± 4.7	46.4 ± 21.7
LBFGS $\lambda = 10^{-4}$	KL Divergence	2.4282 ± 0.0174	2.6825 ± 0.0129	2.8320 ± 0.0096
	Error	0.3735 ± 0.0011	0.3701 ± 0.0005	0.3694 ± 0.0007
	Time (s)	1.7 ± 0.4	3.0 ± 0.8	6.6 ± 1.4

where the step size $\eta > 0$ is selected by line search to satisfy a sufficient decrease condition, and gradients are computed via the scatter-add pattern described in Section F.2. Convergence is declared when the relative change in objective value falls below 10^{-6} or a maximum iteration count is reached. Code for both EMD and L-BFGS implementations of PRIME is publicly available at `https://github.com/Miguel-Fuentes/PRIME`.

G.3 Empirical Comparison of Final Solutions

To validate the efficiency of our proposed optimization strategy, we compare the performance of Entropic Mirror Descent (EMD) against the L-BFGS algorithm across varying support sizes ($48,842$, $122,105$, and $244,210$) and regularization weights (λ). As detailed in Table 1, both methods achieve comparable baseline error rates. However, their computational footprints diverge significantly.

EMD exhibits severe scaling limitations; for the largest support size, EMD requires over $54,000$ seconds (approximately 15 hours) to converge. In contrast, L-BFGS achieves competitive or slightly better L1 error in a fraction of the time. We also observe a clear trade-off governed by the regularization parameter λ in the L-BFGS implementation: $\lambda = 10^{-8}$ yields the lowest L1 error (0.1515 at the largest support size) with a runtime of 919.9 seconds, while increasing the regularization to $\lambda = 10^{-4}$ dramatically accelerates convergence to just 6.6 seconds, albeit at the cost of higher KL divergence and more than double the L1 error (0.3694 vs. 0.1515). Ultimately, L-BFGS closely approximates the fidelity of EMD while delivering speedups of up to several orders of magnitude, making it well suited for scaling to massive support sets.

H Support Size Ablation

To investigate the impact of the support size on the final query accuracy, we conducted an ablation study evaluating PRIME across three datasets (ADULT, NEWS, and OSI) and two baseline mechanisms (RESIDUAL-PLANNER and AIM+GREM). Figure 5 illustrates the mean L_1 error across a range of privacy budgets ($\varepsilon \in [10^{-2}, 10^{1}]$) for support sizes scaled to $1\times$, $5\times$, and $10\times$ the private dataset size estimated by the baseline.

The results demonstrate a clear difference in behavior depending on the privacy regime. In the high-privacy, high-noise regime ($\varepsilon \leq 10^{-1}$), the injected DP noise dominates the overall error. Consequently, the representational capacity of the support set is not the primary bottleneck, and all support sizes perform similarly.

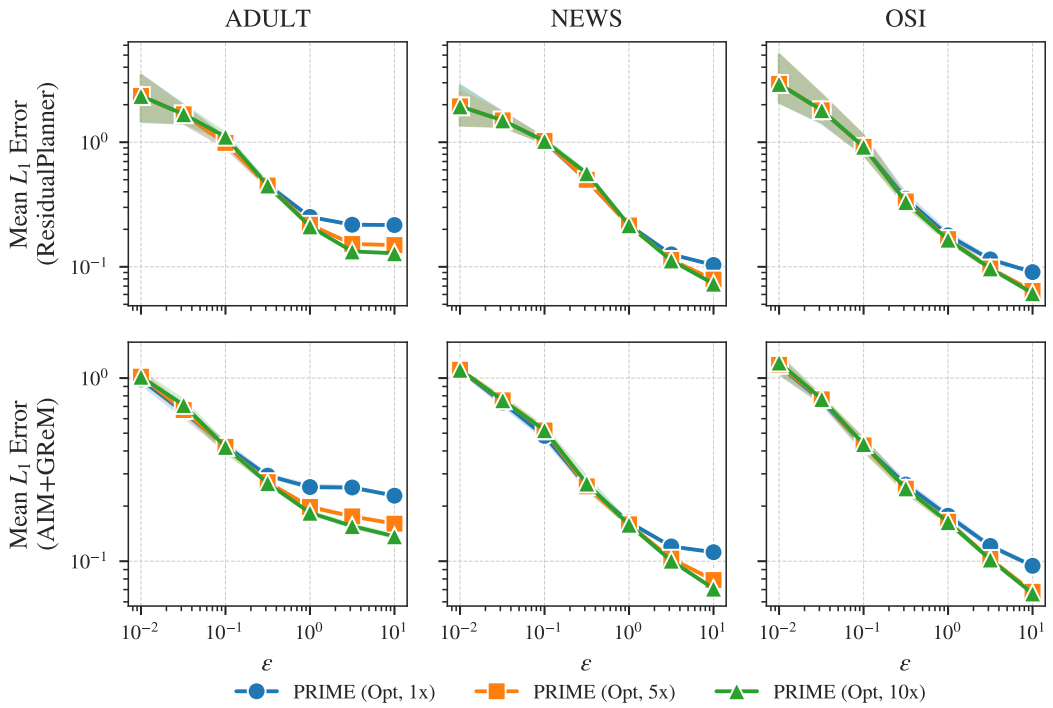


Figure 5: Mean L_1 workload error vs. privacy budget ε for PRIME at support multipliers $1\times$, $5\times$, and $10\times$ the estimated private dataset size, across three datasets (ADULT, NEWS, OSI) and two base mechanisms (RESIDUAL-PLANNER, AIM+GREM).

Conversely, in the lower-noise regimes ($\varepsilon \geq 10^0$), the representational capacity of the support set becomes critical. For a smaller support size ($1\times$), the mean L_1 error rapidly plateaus, indicating that the optimization is bottlenecked by the limited number of discrete points available to approximate the target distribution. Increasing the support size to $5\times$ and $10\times$ consistently alleviates this bottleneck, allowing the optimization to capture finer marginal details and achieve significantly lower errors. This trend holds consistently across all datasets and workloads evaluated, confirming that a larger, over-parameterized support set is essential for maximizing the utility of the optimized weights when sufficient privacy budget is available.

I Broader Impacts

The development of PRIME and the broader advancement of differentially private (DP) synthetic data generation carry both positive and negative potential societal impacts.

Positive Societal Impacts The primary motivation of this work is to facilitate the safe and privacy-preserving sharing of sensitive information. By overcoming the computational bottlenecks of existing graphical-model synthesizers, PRIME allows practitioners to generate high-utility synthetic data from state-of-the-art query-answering mechanisms. This democratization of data access has profound positive implications across numerous domains, from healthcare and finance to ecological modeling and public policy. It enables researchers and organizations to perform statistical inference and train machine learning models without compromising the privacy of individuals contributing to the underlying datasets. Furthermore, by reducing the computational burden and eliminating the exponential treewidth scaling, PRIME makes high-quality DP synthesis more accessible to researchers and organizations with limited computational resources.

Potential Negative Societal Impacts and Risks Despite the strong privacy guarantees of differential privacy, the release and downstream use of synthetic data present inherent risks. The most significant concern is the potential for synthetic data to preserve, obscure, or inadvertently amplify

historical biases present in the original data. Because PRIME relies on a finite support initialization step —such as a 1-way marginal product distribution —there is a risk that complex joint distributions, particularly those representing minority or outlier demographic groups, may be poorly represented in the final synthetic dataset. If downstream decision-makers or machine learning practitioners treat this synthetic data as absolute ground truth without accounting for the injected DP noise or representational limitations, it could lead to discriminatory outcomes or flawed policy decisions.

Mitigations To mitigate these risks, we strongly advocate for transparency in the release of synthetic datasets. Practitioners utilizing PRIME should accompany their data releases with comprehensive documentation detailing the privacy budget (ϵ, δ), the base query-answering mechanism utilized, and the specific support module employed. Downstream users must be educated on the inherent trade-offs between privacy, noise, and representational capacity. We encourage the routine evaluation of fairness and equity metrics on synthetic datasets prior to their use in sensitive or high-stakes downstream applications.

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: The abstract and introduction clearly outline the scalability and modularity claims of PRIME, which are empirically supported by the experiments in Section 4.

   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: Section 6 discusses the sensitivity of PRIME to support quality, formalizing the resulting gap relative to PRIVATE-PGM (Section 3.6) and the support size-runtime tradeoff, along with directions for closing this gap.

   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: Complete assumptions, statements, and proofs for Lemmas 1 and 2 and theorems 1 and 2 and Proposition 1 are provided in Sections B, D, D.1 and E.

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: Experimental setups, datasets, privacy budgets, memory limits, and hyperparameters are detailed in Sections 4 and F, with code publicly available at `https://github.com/Miguel-Fuentes/PRIME`.

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: Section G.2 states that the code is publicly available at `https://github.com/Miguel-Fuentes/PRIME`, and the datasets used (ADULT, NEWS, OSI) are publicly available (Section F.5).

   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: Full optimization details, including L-BFGS hyperparameters and objective scaling strategies, are specified in Sections 4 and F.

   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: All shaded regions in the figures and the $\pm$ values in Table 1 represent the standard error across five independent trials.

   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: Section 4.1 specifies that experiments were run on the Unity Research Computing Platform via SLURM, with each job requesting 2 CPU cores and 32GB of memory subject to a 12-hour timeout; per-run times for the λ-sensitivity ablation are reported in Table 1.

   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 develops algorithms for privacy-preserving synthetic data generation and fully conforms to the NeurIPS Code of Ethics.

   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: The broader societal impact of private synthetic data generation algorithms is discussed in Section I, along with considerations that are particularly relevant for the proposed method.

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 paper proposes an algorithmic improvement for differentially private data synthesis and does not release high-risk models or scraped datasets.

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: [Yes]

Justification: All datasets (ADULT, NEWS, OSI) and baseline mechanisms are appropriately credited and cited in the references section; the use is in line with their CC 4.0 license.

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 source code for the PRIME optimizer and related components is documented and will be made publicly available at `https://github.com/Miguel-Fuentes/PRIME`.

    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: The research methodology does not involve crowdsourcing or human subjects.

    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: The research does not involve human subjects.

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: The core methodology does not involve the use of LLMs; they are only mentioned briefly as a potential avenue for future work.

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.