Private Algorithms with Private Predictions

Kareem Amin[†] kamin@google.com
Travis Dick[†] tdick@google.com
Mikhail Khodak[*] khodak@cmu.edu
Sergei Vassilvitskii[†] sergeiv@google.com

Abstract

When applying differential privacy to sensitive data, a common way of getting improved performance is to use external information such as other sensitive data, public data, or human priors. We propose to use the algorithms with predictions (a.k.a. *learning-augmented algorithms*) framework—previously applied largely to improve time complexity or competitive ratios—as a powerful way of designing and analyzing privacy-preserving methods that can take advantage of such external information to improve *utility*. For three important tasks—(multiple) quantile release, covariance estimation, and data release—we construct prediction-dependent differentially private methods whose utility scales with natural measures of prediction quality. Our analysis enjoys several advantages, including minimal assumptions about the data, a natural way of adding robustness, and the provision of useful surrogate losses for two novel "meta" algorithms that learn predictions from other (potentially sensitive) data. We conclude with experiments in a diverse set of multi-dataset quantile release settings that demonstrate how a learning-augmented approach to incorporating external information can lead to large error reductions while preserving privacy.

1 Introduction

The differentially private (DP) release of statistics about a sensitive dataset $\mathbf{x} \in \mathbb{R}^n$ is an inevitably error-prone task because we are by definition precluded from revealing exact information about the instance at hand [31]. However, DP instances rarely occur in a vacuum: even in the simplest practical settings, we usually know basic information such as the fact that all individuals have a nonnegative age. Often, the dataset we are considering is drawn from a similar population as a public dataset $\mathbf{x}' \in \mathbb{R}^N$ and should thus have similar statistics, a case known as the *public-private* setting [11, 53]. Alternatively, in what we call *sequential release*, we aim to release information about each of a sequence of datasets $\mathbf{x}_1, \ldots, \mathbf{x}_T$ one-by-one. These could be generated by a stationary or other process that allows information derived from prior releases to inform predictions of future releases. In all of these settings, we might hope to incorporate external information to reduce error, but approaches for doing so tend to be *ad hoc* and assumption-heavy.

We propose that the framework of algorithms with predictions [59]—provides the right tools for deriving DP algorithms in this setting, and instantiate this idea for multiple quantile release[1] [33, 42], covariance estimation [4, 12, 27], and data release [36, 53]. Algorithms with predictions is an expanding field of algorithm design that constructs methods whose instance-dependent performance improves with the accuracy of some prediction about the instance. The goal is to bound the cost $C_{\mathbf{x}}(\mathbf{w})$ of running on instance $\mathbf{x}$ given a prediction $\mathbf{w}$ by some metric $U_{\mathbf{x}}(\mathbf{w})$ of the *quality* of the prediction on that instance. Motivated by practical success [48, 54] and as a type of beyond-worst-case analysis [64], such algorithms can target a wide variety of cost measures, e.g. competitive ratios

[†]Google Research - New York; [*]Carnegie Mellon University, work done in part at Google Research - New York
[1]Part of this work is in the proceedings of the 40[th] International Conference on Machine Learning (ICML 2023) [45].

in online algorithms [5, 9, 20, 24, 30, 38, 40, 49, 56, 63, 71], space complexity in streaming algorithms [28], and time complexity in graph algorithms [19, 26, 65] and distributed systems [50, 52, 66]. Departing from such work, we instead aim to design learning-augmented algorithms whose cost $C_{\mathbf{x}}(\mathbf{w})$ captures the error of some statistic—e.g. quantiles—computed privately on instance an $\mathbf{x}$ given a prediction $\mathbf{w}$. We are interested in bounding this cost in terms of the quality of the external information provided to our algorithm, which we denote by $U_{\mathbf{x}}(\mathbf{w})$.

While incorporating external information into DP is well-studied, c.f. public-private methods [11, 53] and private posterior inference [25, 32, 67], by deriving and analyzing a learning-augmented algorithm for multiple quantiles we show numerous comparative advantages, including:

1. Minimal data assumptions, sometimes even fewer than needed by the unaugmented baseline.
2. Existing tools for studying the robustness of algorithms to noisy predictions [56].
3. Co-design of algorithms with predictions with methods for *learning* those predictions from data [44], which we show is crucial for both the public-private and sequential release settings.

We derive learning-augmented extensions of the state-of-the-art `ApproximateQuantiles` (AQ) method [42] for quantile release and of the covariance estimation algorithms `SeparateCov` [27] and `IterativeEigenvectorSampling` [4]; for data release we show how our framework applies to `MWEM` [36], for which using a non-uniform (i.e. prediction-based) prior has been studied in past work [53]. In all cases our instance-dependent guarantees (nearly) match past worst-case bounds while being much better if a natural measure $U_{\mathbf{x}}(\mathbf{w})$ of prediction quality is small. We also show how these algorithms can be made robust to poor predictions $\mathbf{w}$ and how they can be efficiently and privately *learned* by optimizing $U_{\mathbf{x}}$ across related datasets $\mathbf{x}$. In addition, our analysis yields several contributions of independent interest for differential privacy:

1. The first robust algorithm for (single or multiple) private quantile release that avoids assuming the data is bounded on some interval, specifically by using a heavy-tailed prior.
2. Prediction-free trace-sensitive guarantees for `SeparateCov` (for both the pure and zCDP versions) that strictly improve upon the original bounds of Dong et al. [27] for the same algorithm.
3. A non-Euclidean extension of DP-FTRL [41] that is the first DP online convex optimization method that can be easily customized to obtain better regret guarantees on different geometries.

Finally, we conclude with an empirical study where we use our framework to design algorithms to reduce the error of private quantile release in both the public-private and sequential release settings described above. Our technical approach takes advantage of a novel connection between DP quantiles and censored regression to obtain both guarantees and practical algorithms. The experimental results highlight the effectiveness of our framework for ensuring robust performance in the face of noisy predictions and for designing surrogate loss functions that can be optimized to yield useful predictions.[2]

2 Problem formulation

The basic requirement for a learning-augmented algorithm is that the cost $C_{\mathbf{x}}(\mathbf{w})$ of running it on an instance $\mathbf{x}$ with prediction $\mathbf{w}$ should be upper bounded—usually up to constant or logarithmic factors—by a metric $U_{\mathbf{x}}(\mathbf{w})$ of the quality of the prediction on the instance. We denote this by $C_{\mathbf{x}} \lesssim U_{\mathbf{x}}$. In our work the cost $C_{\mathbf{x}}(\mathbf{w})$ will be the error of a privately released statistic, as compared to some ground truth. We will use the following privacy notion:

[2]Code to reproduce our results is available at `https://github.com/mkhodak/private-quantiles`.

Definition 2.1 ([31]). *Algorithm $\mathcal{A}$ is* (ε, δ)**-differentially private** *if for all subsets S of its range,* $\Pr\{\mathcal{A}(\mathbf{x}) \in S\} \leqslant e^{\varepsilon} \Pr\{\mathcal{A}(\tilde{\mathbf{x}}) \in S\} + \delta$ *whenever* $\mathbf{x} \sim \tilde{\mathbf{x}}$ *are* **neighboring datasets**.

Using ε-DP to denote $(\varepsilon, 0)$-DP, the broad goal of this work will be to reduce the error $C_{\mathbf{x}}(\mathbf{w})$ of ε-DP multiple quantile release while fixing the privacy level ε. For easier comparison to past prediction-free results, we will define neighboring datasets differently depending on the application; specifically, for quantile release we use **add-remove privacy**, where $\mathbf{x}$ can be obtained from $\tilde{\mathbf{x}}$ by adding or removing an entry, while for covariance estimation and data release we use **swap privacy**, in which $\mathbf{x}$ can be obtained from $\tilde{\mathbf{x}}$ by replacing one entry with another.

A good guarantee for a learning-augmented algorithm will have several important properties that formally separate its performance from naive upper bounds $U_{\mathbf{x}} \gtrsim C_{\mathbf{x}}$. The first, *consistency*, requires it to be a reasonable indicator of strong performance in the limit of perfect prediction:

Definition 2.2. *A guarantee $C_{\mathbf{x}} \lesssim U_{\mathbf{x}}$ is* $c_{\mathbf{x}}$**-consistent** *if $C_{\mathbf{x}}(\mathbf{w}) \leqslant c_{\mathbf{x}}$ whenever $U_{\mathbf{x}}(\mathbf{w}) = 0$.*

Here $c_{\mathbf{x}}$ is a prediction-independent quantity that should depend weakly or not at all on problem difficulty (in the case of quantiles, the minimum separation between data points). Consistency is often presented via a tradeoff with *robustness* [56], which bounds how poorly the method can do when the prediction is bad, in a manner similar to a standard worst-case bound:

Definition 2.3. *A guarantee $C_{\mathbf{x}} \lesssim U_{\mathbf{x}}$ is* $r_{\mathbf{x}}$**-robust** *if it implies $C_{\mathbf{x}}(\mathbf{w}) \leqslant r_{\mathbf{x}}$ for all predictions $\mathbf{w}$.*

Unlike consistency, robustness usually depends strongly on the difficulty of the instance $\mathbf{x}$, with the goal being to not do much worse than a prediction-free approach. Note that the latter is trivially robust but not (meaningfully) consistent, since it ignores the prediction; this makes clear the need for considering the two properties via a tradeoff between them. As discussed further in Section 5, this existing language for quantifying robustness is one of the advantages of using the framework of learning-augmented algorithms for incorporating external information into DP methods.

A last desirable property of the prediction quality measure $U_{\mathbf{x}}(\mathbf{w})$ is that it should be useful for making good predictions. One way to formalize this is to require $U_{\mathbf{x}_t}$ to be *learnable* from multiple instances $\mathbf{x}_t$. For example, we could ask for *online* learnability, i.e. the existence of an algorithm whose predictions $\mathbf{w}_t \in W$ in some action space W given instances $\mathbf{x}_1, \ldots, \mathbf{x}_{t-1}$ yield *regret* sublinear in T:

Definition 2.4. *The* **regret** *of actions $\{\mathbf{w}_t \in W\}_{t=1}^T$ on losses $\{U_{\mathbf{x}_t}\}_{t=1}^T$ is* $\max_{\mathbf{w} \in W} \sum_{t=1}^{T} U_{\mathbf{x}_t}(\mathbf{w}_t) - U_{\mathbf{x}_t}(\mathbf{w})$.

Sublinear regret implies average prediction quality as good as that of the optimal prediction in hindsight, up to an additive term that vanishes as $T \to \infty$. Since $U_{\mathbf{x}_t}$ roughly upper-bounds the error $C_{\mathbf{x}_t}$, this means that asymptotically the average error is governed by the average prediction quality $\min_{\mathbf{w} \in W} \frac{1}{T} \sum_{t=1}^{T} U_{\mathbf{x}_t}(\mathbf{w})$ of the optimal $\mathbf{w} \in W$. A crucial observation here is that sublinear regret can often be obtained by making the function $U_{\mathbf{x}}$ amenable to familiar gradient-based online convex optimization methods such as online gradient descent [44]. Doing so also enables instance-dependent linear prediction: setting $\mathbf{w}_t$ using a learned function of some instance features $\mathbf{f}_t$. In Section 6 we apply our non-Euclidean extension of DP-FTRL (c.f. Theorem 6.1) to show online and PAC learnability of the prediction quality measures $U_{\mathbf{x}}$ for all three DP tasks we consider.

The usefulness of both the learning-theoretic and robustness-consistency analysis is demonstrated in Section 7 on two applications where it is reasonable to have external information about the sensitive dataset(s). In the **public-private** setting, the prediction $\mathbf{w}$ is obtained from a public dataset $\mathbf{x}'$ that is assumed to be similar to $\mathbf{x}$ but is not subject to privacy-protection. In **sequential release**, we privately release information about each dataset in a sequence $\mathbf{x}_1, \ldots, \mathbf{x}_T$; the release at time t can depend on $\mathbf{x}_t$ and on a prediction $\mathbf{w}_t$, which can be derived (privately) from past

observations. We show that sequential release can be posed directly as a private online learning problem, while the public-private setting can be approached via online-to-batch conversion [17]. Both can thus be solved by treating the prediction quality measures $U_{\mathbf{x}_t}$ as surrogate objectives for the actual cost functions $C_{\mathbf{x}}$ and applying standard optimization techniques [44].

3 Overview of theoretical results

We now summarize the main results for the three tasks we consider, focusing on the prediction-dependent performance bounds $U_{\mathbf{x}} \gtrsim C_{\mathbf{x}}$ that we show for our learning-augmented private algorithms. These will be stated more formally in Section 4. We also highlight the utility of these results in ensuring robustness and enabling learning, which will be further detailed in Sections 5 and 6, respectively.

3.1 Related work

There has been significant work on incorporating external information to improve DP methods. A major line of work is the public-private framework, where we have access to public data that is related in some way to the private data [3, 10, 11, 51, 53]. The use of public data can be viewed as using a prediction, but such work starts by making (often strong) distributional assumptions on the public and private data; we instead derive instance-dependent upper bounds with minimal assumptions that we then apply to such public-private settings. Furthermore, our framework allows us to ensure robustness to poor predictions without distributional assumptions, and to derive learning algorithms using training data that may itself be sensitive. Another approach is to treat DP mechanisms (e.g. the exponential) as Bayesian posterior sampling [25, 32, 67]. Our work can be viewed as an adaptation where we give explicit prior-dependent utility bounds. To our knowledge, no such guarantees exist in the literature. Moreover, our approach does not necessitate specifying the external information in the form of (explicit) priors, e.g. for covariance estimation we use matrix predictions.

Our approach for augmenting DP with external information centers the algorithms with predictions framework, where past work has focused on using predictions to improve metrics related to time, space, and communication complexity. We make use of existing techniques from this literature, including robustness-consistency tradeoffs [56] and the online learning of predictions [44]. Tuning DP algorithms has been an important topic in private machine learning, e.g. for hyperparameter tuning [18] and federated learning [6], but these have not to our knowledge considered incorporating per-instance predictions.

3.2 Preliminaries

We use $[n]$ to denote the sequence $\big(1, \cdots, n\big)$, $\mathbf{x}_{[i]}$ to denote the ith element of a vector $\mathbf{x}$, and $\mathbf{X}_{[i,j]}$ to denote the jth element of the ith row of a matrix $\mathbf{X}$. For both vectors and matrices we will use $\|\cdot\|_p$ to denote the entry-wise p-norm, and for the latter we will use $|||\cdot|||_p$ to denote the Schatten p-norm; thus $\|\cdot\|_2 = \|\cdot\|_F$ is the Frobenius norm, $|||\cdot|||_\infty$ is the spectral norm, and $|||\cdot|||_1 = \|\cdot\|_{\mathrm{Tr}}$ is the trace or nuclear norm. We will use $\|\cdot\|_*$ to refer to the dual norm of $\|\cdot\|$, and for any dataset $\mathbf{X}$ we use $|\mathbf{X}|$ to denote the number of entries. We use $\mathbf{0}_d$ and $\mathbf{1}_d$ to denote the d-dimensional all-zero and all one vectors, 1_S to denote the indicator on set S, $x \sim \mathrm{Lap}(b)$ to denote sampling a Laplace r.v. with mean zero and scale b, $\mathbf{x} \sim \mathcal{N}(\mathbf{0}_d, \sigma^2)$ to denote sampling a Gaussian vector with mean $\mathbf{0}_d$ and covariance $\sigma^2\mathbf{I}_d$, and $\triangle_d$ to denote the simplex on d elements. For simplicity, for any probability measure $\mu : (a, b) \mapsto \mathbb{R}_{\geqslant 0}$ we use $\mu(I) = \int_I \mu(o)do$ to denote the probability it assigns to any interval $I \subset (a, b)$. Unless otherwise specified, $\tilde{\mathcal{O}}$ will be used to ignore logarithmic factors in standard asymptotic notation.

3.3 Multiple quantile release

In the quantile problem, given a quantile q and a sorted dataset $\mathbf{x} \in \mathbb{R}^n$ of n distinct points, the goal is to release a number o that upper bounds exactly $\lfloor qn \rfloor$ of the entries. The error metric, $\mathrm{Gap}_q(\mathbf{x}, o)$, is the number of entries between the released number o and $\lfloor qn \rfloor$. A straightforward application of the well-known exponential mechanism [58] with utility $-\mathrm{Gap}_q$ outputs o that satisfies $\mathrm{Gap}_q(\mathbf{x}, o) \leqslant \frac{2}{\varepsilon} \log \frac{1}{\beta \Psi_{\mathbf{x}}^{(q)}}$ w.p. $\geqslant 1 - \beta$, where $\Psi_{\mathbf{x}}^{(q)}$ is the probability $\mu((\mathbf{x}_{[\lfloor qn \rfloor]}, \mathbf{x}_{[\lfloor qn \rfloor + 1]}])$ that the prior assigns to the optimal interval. We thus use $U_{\mathbf{x}}^{(q)}(\mu) = -\log \Psi_{\mathbf{x}}^{(q)}$ as our measure of prediction quality in the single-quantile setting, which allows us to recover standard guarantees that assume $\mathbf{x} \in (a, b)^n$ is bounded and set μ to be the uniform measure on (a, b). As our first major contribution, we show by studying $U_{\mathbf{x}}$ how to dispense with this assumption by instead using the Cauchy distribution with location $\frac{a+b}{2}$ and scale $\frac{b-a}{2}$. If the boundedness assumption holds then the resulting mechanism has nearly the same bound on Gap_q as the uniform measure, up to an additive $\frac{2}{\varepsilon} \log \pi$ factor, but if does not—e.g. if all points $\mathbf{x}_{[i]}$ in the dataset are a distance $R > \frac{b-a}{2}$ away from $\frac{a+b}{2}$—then we still have the guarantee $\mathrm{Gap}_q = \tilde{\mathcal{O}}(\frac{\log R}{\varepsilon})$ w.h.p. (c.f. Corollary 4.1). In contrast, the error of the released quantile when using the uniform measure in the latter scenario is $\Omega(n)$ a.s.

The main technical challenge is then to extend the single-quantile guarantee to the case where we must estimate $m > 1$ quantiles $q_1, \ldots, q_m \in (0, 1)$ while making use of m priors $\mu_1, \ldots, \mu_m$. In particular, we want a guarantee on the maximum gap that encodes how useful each prior μ_i is for its quantile q_i *and* that grows sublinearly in m, ideally recovering the $\max_i \mathrm{Gap}_{q_i} = \mathcal{O}(\frac{\mathrm{polylog}(m)}{\varepsilon})$ bound of Kaplan et al. [42] in the prediction-free limit. Although it requires several major modifications to AQ, we are able to nearly achieve this goal, devising a method (c.f. Algorithm 5) that guarantees a bound of $\tilde{\mathcal{O}}(\frac{r(m)}{\varepsilon} \log \sum_{i=1}^m e^{U_{\mathbf{x}}^{(q_i)}(\mu_i)})$ on the maximum gap w.h.p. (c.f. Theorem 4.3), where $r(m)$ is sub-polynomial but super-polylogarithmic in m. This yields a quality measure $U_{\mathbf{x}}$ for $\mu_1, \ldots, \mu_m$ that aggregates the single-quantile measures $U_{\mathbf{x}}^{(q_i)}(\mu_i)$ via their log-sum-exp, a convenient form that allows us to easily extend single-quantile robustness and learning-theoretic results to multiple quantiles.

Our quantile results exemplify the advantages of our approach to incorporating external information into DP algorithms that we discussed in the introduction: minimal assumptions, robustness-consistency tradeoffs, and learning. In-fact, the first outcome of our analysis was *removing* a boundedness assumption. This contrasts with past public-private work [11, 53], which makes distributional assumptions, and is why we can obtain guarantees in two very distinct settings in Section 7. We next highlight how our results imply convenient robustness-consistency tradeoffs and efficient learnability.

3.3.1 Robustness

Using the formalization of robustness and consistency in Definitions 2.2 and 2.3, algorithms with predictions provides a convenient way to deploy them by *parameterizing* the robustness-consistency tradeoff, in which methods are designed to be $r_{\mathbf{x}}(\lambda)$-robust and $c_{\mathbf{x}}(\lambda)$-consistent for a user-specified parameter $\lambda \in [0, 1]$ [9, 56]. For quantiles, we can obtain an elegant parameterized tradeoff by interpolating prediction priors with a "robust" prior. In particular, we can pick ρ to be a trusted prior such as the uniform or Cauchy and for any prediction μ use $\mu^{(\lambda)} = (1-\lambda)\mu + \lambda\rho$ instead. Then since $\Psi_{\mathbf{x}}^{(q)}$ is linear we have $\Psi_{\mathbf{x}}^{(q)}(\mu^{(\lambda)}) = (1-\lambda)\Psi_{\mathbf{x}}^{(q)}(\mu) + \lambda\Psi_{\mathbf{x}}^{(q)}(\rho)$, which implies the following guarantee:

Corollary 3.1 (of Lem. A.1; c.f. Cor. 5.1)**.** *For any quantile $q \in (0, 1)$, applying EM with prior $\mu^{(\lambda)} = (1-\lambda)\mu + \lambda\rho$ is $\left(\frac{2}{\varepsilon} \log \frac{1/\beta}{\lambda \Psi_{\mathbf{x}}^{(q)}(\rho)}\right)$-robust and $\left(\frac{2}{\varepsilon} \log \frac{1/\beta}{1-\lambda}\right)$-consistent.*

Thus w.h.p. error is simultaneously at most $\frac{2}{\varepsilon}\log\frac{1}{\lambda}$ worse than that of only using the robust prior ρ and we only have error $\frac{2}{\varepsilon}\log\frac{1/\beta}{1-\lambda}$ if the prediction μ is perfect, i.e. if it is only supported on the optimal interval. This is easy to extend to the multiple-quantile metric $U_{\mathbf{x}} = -\log\Psi_{\mathbf{x}}$. In fact, we can even interpolate between the polylog(m) prediction-free guarantee of past work and our learning-augmented guarantee with the worse dependence on m (c.f. Corollary 5.2); thus if the prediction is not good enough to overcome the worse rate we can still ensure that we do not do much worse than the original guarantee. These results show the advantage of our framework in designing algorithms that make robust use of possibly noisy predictions. Notably, related public-private work that studies robustness still assumes source and target data are Gaussian [11], whereas we make no distributional assumptions. We demonstrate the importance of our robustness techniques throughout the experiments in Section 7.

3.3.2 Learning

A last important use for prior-dependent bounds is as surrogate objectives for optimization. As we show in Section 7, being able to learn across upper bounds $U_{\mathbf{x}_1}, \ldots, U_{\mathbf{x}_T}$ of a sequence of (possibly sensitive) datasets $\mathbf{x}_t$ is useful for both the public-private and sequential release. Algorithms with predictions guarantees are often sufficiently nice to do this using off-the-shelf online learning [44], a property that largely holds for our upper bounds as well. Most saliently, the bound $U_{\mathbf{x}}^{(q)} = -\log\Psi_{\mathbf{x}}^{(q)}$ is a convex function of an inner product $\Psi_{\mathbf{x}}^{(q)}$ between the EM score and the prior μ; thus by discretizing one can learn over a large family of piecewise-constant priors, which themselves approximate Lipschitz priors over a bounded domain. The same is true of the multiple quantile bound $U_{\mathbf{x}}$ because it is the log-sum-exp over $U_{\mathbf{x}}^{(q_i)}$ and thus also convex. We therefore can apply an entropic variant of DP-FTRL to (privately) online learn the sequence $U_{\mathbf{x}_t}$ with low-regret w.r.t. any set of m Lipschitz priors (c.f. Theorem 6.2). However, in-practice we may not want to learn in the high dimensions needed by the discretization, and rather than fixed priors we may wish to learn a mapping from dataset-specific features.

Thus, in Section 7 we focus on the less-expressive family of location-scale models, which allows us to develop algorithms that are amenable to both analysis and implementation. In particular, we show that $U_{\mathbf{x}}$ has the same form as the negative log-likelihood of censored regression, which for log-concave location-scale families is convex in a convenient reparameterization of the location and scale [15, 62]. We can thus show DP online learning guarantees in the sequential release setting (c.f. Theorem 7.3) and derive an algorithm for public-private transfer whose error is bounded by the TV-distance between the order statistics of the public and private distributions (c.f. Theorem 7.2).

3.4 Covariance estimation

While encoding predictions via base measures of DP mechanisms is a natural starting point for learning-augmented algorithms, it is not the only way of doing so. We can instead start with existing algorithms whose errors have explicit or implicit dependence on some measure of complexity of the data and use this to convert them into algorithms with predictions. The errors will then have an (explicit) dependence on a related measure of the error between the data and a point (rather than distributional) prediction, leading to highly interpretable bounds $U_{\mathbf{x}}(\mathbf{w})$ on the utility loss.

Our application to covariance estimation exemplifies this approach. For this task we take advantage of recent "trace-sensitive" results, which bound the Frobenius error between the covariance matrix $\mathbf{C} = \mathbf{X}\mathbf{X}^T/n$ of a dataset $\mathbf{X} \in \mathbb{R}^{d\times n}$ by some function of its trace [4, 27]. Since the core component of these algorithms is a DP estimate of a symmetric $d \times d$ matrix, if we have a symmetric prediction $\mathbf{W} \in \mathbb{R}^{d\times d}$ we can try to use the methods to instead privately estimate the error $\mathbf{C} - \mathbf{W}$ and then add $\mathbf{W}$ to the result; we can then hope to show that the error depends on the trace norm

$\|\mathbf{C} - \mathbf{W}\|_{\mathrm{Tr}}$ of the error rather than the trace of $\mathbf{C}$. We achieve exactly this and more by extending the analysis in this prior work to the negative spectrum, in order to handle the possibly negative eigenvalues of $\mathbf{C} - \mathbf{W}$. The result below, for the learning-augmented extension of the state-of-the-art `SeparateCov` algorithm [27], is characteristic of these results (c.f. Section 4.2):

Corollary 3.2 (of Thm. 4.4; c.f. Cor 4.2)**.** *If $\mathbf{X} \in \mathbb{R}^{d \times n}$ has columns bounded by 1 in ℓ_2-norm then applying `SeparateCov` to $\mathbf{C} - \mathbf{W}$ and obtaining $\hat{\mathbf{C}}$ by adding $\mathbf{W}$ to the result is ε-DP and satisfies $\|\hat{\mathbf{C}} - \mathbf{C}\|_F^2 \leqslant \tilde{\mathcal{O}}\left(\frac{d}{\varepsilon^2 n^2} + \frac{d\sqrt{d}}{\varepsilon n} \min_{c \in \mathbb{R}} \|\mathbf{C} - \mathbf{W} - c\mathbf{I}_d\|_{\mathrm{Tr}}\right)$ w.h.p.*

Notably, for $\mathbf{W} = \mathbf{0}_{d \times d}$ this bound improves upon the corresponding prediction-free result of Dong et al. [27], who only show it for $c = 0$. A simple setting where this improvement is tangible is when the columns of $\mathbf{X}$ are drawn from a bounded distribution whose covariance is a scalar multiple of the identity, in which case w.h.p. $\min_{c \in \mathbb{R}} \|\mathbf{X}\mathbf{X}^T/n - c\mathbf{I}_d\|_{\mathrm{Tr}} \leqslant \tilde{\mathcal{O}}(d \min\{1, \sqrt{d/n}\})$ but $\|\mathbf{X}\mathbf{X}^T/n\|_{\mathrm{Tr}} \geqslant \tilde{\mathcal{O}}(d)$; therefore for constant ε the bound in Corollary 4.2 becomes $\tilde{\mathcal{O}}(\frac{d^2}{n}\sqrt{\min\{d, d^2/n\}})$ whereas the bound of Dong et al. [27, Lemma 19] is no better than $\tilde{\mathcal{O}}(d^2\sqrt{d}/n)$. In particular, for $d = O(1)$ our bound is asymptotically dominated by the error $\tilde{\mathcal{O}}(\frac{d^2}{n})$ of (non-privately) estimating the population covariance.

3.4.1 Robustness

Because of its non-convexity, we drop the minimum over $c \in \mathbb{R}$ for our robustness and learning-theoretic analyses of covariance estimation, using the looser bound at $c = 0$ to define our prediction quality metric $U_{\mathbf{X}}(\mathbf{W}) = \|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_{\mathrm{Tr}}$. To ensure robustness, we take the approach of privately checking if the quality $U_{\mathbf{X}}(\mathbf{W})$ of the prediction $\mathbf{W} \in \mathbb{R}^{d \times d}$ is better than $U_{\mathbf{X}}(\mathbf{0}_{d \times d})$, i.e. that of the prediction-free approach. In doing so we pay for robustness by a factor of $\sqrt{d}$ in the leading (non-trace-sensitive) term, although as we discuss later this may be an artifact of the setting.

Corollary 3.3 (of Thm. 4.4; c.f. Cor. 4.2)**.** *Running `SeparateCov` with the prediction $\mathbf{W}$ only if its trace distance $\|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_{\mathrm{Tr}}$ is smaller than $\|\mathbf{X}\mathbf{X}^T/n\|_{\mathrm{Tr}}$ according to the Laplace mechanism is $\tilde{\mathcal{O}}\left(\frac{d\sqrt{d}}{\varepsilon n}\left(\frac{1}{\varepsilon n} + \|\mathbf{X}\mathbf{X}^T/n\|_{\mathrm{Tr}}\right)\right)$-robust and $\tilde{\mathcal{O}}\left(\frac{d\sqrt{d}}{\varepsilon^2 n^2}\right)$-consistent.*

3.4.2 Learning

Similar to before, we can pose the problem of learning to release covariance estimates across multiple datasets as the online learning problem of obtaining low regret w.r.t. any matrix $\mathbf{W} \in \mathbb{R}^{d \times d}$ for the functions $U_{\mathbf{X}_t}(\mathbf{W}) = \|\mathbf{X}_t\mathbf{X}_t^T/n_t - \mathbf{W}\|_{\mathrm{Tr}}$ determined by the sequence of datasets $\{\mathbf{X}_t \in \mathbb{R}^{d \times n_t}\}_{t=1}^T$. We apply DP-FTRL with with a Schatten p-norm regularizer, which applies p-norm regularization to the spectrum of the matrix [29]; this yields a $\mathcal{O}(\sqrt{d})$-improvement in the regret—and a corresponding $\mathcal{O}(d)$-improvement in sample complexity—over regular DP-FTRL, highlighting the usefulness of our non-Euclidean analysis.

Theorem 3.1 (c.f. Thm. 6.3)**.** *There exists an (ε', δ')-DP online learner whose regret w.r.t. all symmetric $\mathbf{W} \in \mathbb{R}^{d \times d}$ is bounded w.h.p. by $\tilde{\mathcal{O}}\left(\sqrt{(1 + d/\varepsilon')T}\right)$. Furthermore, if the datasets $\mathbf{X}_t$ are all drawn i.i.d. from the same distribution and we set $\hat{\mathbf{W}} = \frac{1}{T}\sum_{t=1}^T \mathbf{W}_t$ to be the average iterate then $T = \tilde{\Omega}\left(\frac{1 + d/\varepsilon'}{\alpha^2}\right)$ samples suffice to ensure that w.h.p. its excess risk is at most α.*

3.5 Data release

In our last application we study private data release, where we seek to construct a synthetic dataset $\hat{\mathbf{x}} \in \mathbb{R}^d_{\geqslant 0}$ using sensitive data $\mathbf{x} \in \mathbb{Z}^d_{\geqslant 0}$ such that the maximum error of a finite set $\mathcal{Q}$ of linear queries $\mathbf{q} \in [-1, 1]^d$ is bounded. To do so we use the well-known `MWEM` method of [36], which has an implicit dependence on the KL-divergence $D_{KL}(\mathbf{x}/n||\mathbf{1}_d/d)$ between the data distribution and the uniform distribution it uses to initialize its iterative approach; by instead initializing with a prediction $\mathbf{w} \in \triangle_d$ in the d-dimensional simplex one can instead obtain a dependence on $D_{KL}(\mathbf{x}/n||\mathbf{w})$:

Lemma 3.1 (c.f. Lem. 4.2)**.** *Initializing `MWEM` with $\mathbf{w} \in \triangle_d$ and running it for m iterations on dataset $\mathbf{x}$ is ε-DP and w.p. $\geqslant 1 - \beta$ produces a synthetic dataset s.t. the largest mean squared error of any linear query in Q is bounded by $\mathcal{O}\left(\frac{n}{m} D_{KL}(\frac{\mathbf{x}}{n}||\mathbf{w}) + \frac{m^2}{\varepsilon^2 n} \log^2 \frac{m}{\beta} \log^4 |Q|\right)$, where $n = \|\mathbf{x}\|_1$.*

As in quantile release, for this task we can again ensure robustness via an interpolation-based approach, although here we are mixing finite-dimensional vectors rather than probability distributions. Note that using the uniform prior guarantees $\tilde{\mathcal{O}}\left(\sqrt[3]{\frac{n \log^2 d}{\varepsilon^2}}\right)$ error, so since the data-dimension d can be very large in this application, if we use small enough λ we can obtain a strong advantage under perfect predictions while ensuring performance similar to the prediction-free guarantee.

Corollary 3.4 (of Lem. 4.2; c.f. Cor. 5.4)**.** *There exists a fixed number of iterations s.t. using $\mathbf{w}^{(\lambda)} = (1 - \lambda)\mathbf{w} + \lambda \mathbf{1}_d/d$ instead of the prediction $\mathbf{w} \in \triangle_d$ to initialize `MWEM` is $\tilde{\mathcal{O}}\left(\sqrt[3]{\frac{n}{\varepsilon^2 \log d}} \log \frac{d}{\lambda}\right)$-robust, and $\tilde{\mathcal{O}}\left(\lambda \sqrt[3]{\frac{n \log^2 d}{\varepsilon^2}}\right)$-consistent, where n is the number of records.*

The observation that `MWEM` can be initialized non-uniformly is not novel, having been used by both the original authors and by subsequent public-private work [53]. However, our learning-theoretic analysis reveals interesting aspects that this prior work does not consider as closely, such as how the optimal choice for *other* parameters of the algorithm are influenced by the prediction quality. In-particular, when online learning the sequence of prediction quality measures $U_{\mathbf{x}_t}(\mathbf{w}) \simeq \frac{n_t}{m} D_{KL}(\mathbf{x}_t/n_t||\mathbf{w}) + \frac{m^2}{\varepsilon^2 n_t}$ that bound the error of data release—here n_t is the number of examples in $\mathbf{x}_t$ and m is the number of iterations—we note that the optimal setting of m depends on the similarity between instances: if $\min_{\mathbf{w}} \sum_{t=1}^T n_t D_{KL}(\mathbf{x}_t/n_t||\mathbf{w})$, i.e. the entropy of the average distribution $\left(\sum_{t=1}^T \mathbf{x}_t\right) / \sum_{t=1}^T n_t$, is small then we can take advantage of this by taking fewer iterations. However, we do not know this entropy *a priori*, so we can instead adapt to it by competing with the best step-size—which will encode the unknown entropy—by simultaneously running online learners both for $\mathbf{w}$ and for m, with the optimization domain of the latter being the m-simplex $\triangle_m$. We again apply entropic DP-FTRL to get the following guarantee:

Theorem 3.2 (c.f. Thm. 6.4)**.** *There exists an (ε', δ')-DP algorithm that adaptively sets the initializations $\mathbf{w}_t \in \triangle_d$ and number of iterations $m_t > 0$ s.t. the regret w.r.t. the optimal (initialization, iteration) pair $(\mathbf{w}, m)$ is $\tilde{\mathcal{O}}\left(\frac{dN^{\frac{4}{3}}}{\lambda \min\{1, \varepsilon^2\}} \sqrt{T/\varepsilon'}\right)$, where $N = \max_t n_t$ is the maximum number of entries in any dataset $\mathbf{x}_t$.*

3.6 Discussion

This concludes our overview of our theoretical results, where we highlight multiple ways of incorporating predictions—as priors in DP mechanisms, as offsets to be corrected using sensitive data, or as initializations for iterative methods—as well as two ways of making the methods robust to noisy predictions: (1) interpolating with a default prediction and (2) privately checking whether the quality of the default prediction is better. We also illustrate how learning-augmented analysis can yield new insights in the prediction-free setting, as demonstrated by our results for unbounded quantile release and trace-sensitive covariance estimation. Next we will go into further detail about these prediction-dependent guarantees, robustness-consistency tradeoffs, and learning-theoretic results in Sections 4, 5, and 6, respectively. Then in Section 7 we will present a theoretical and empirical investigation of of how to use predictions to improve multiple quantile release in both the public-private and sequential release settings.

4 Prediction-dependent utility bounds

As formulated in Section 2, the basic guarantee of learning-augmented private algorithm is an upper bound $U_{\mathbf{x}}(\mathbf{w})$ on the error $C_{\mathbf{x}}(\mathbf{w})$ of the statistic it releases about a dataset $\mathbf{x}$ when using a prediction $\mathbf{w}$. We now demonstrate how to design methods for different DP tasks that enjoy such guarantees. While for single quantile release and data release we take the straightforward approach of incorporating a prediction-dependent prior into the EM mechanism, we also show how to handle difficulties that arise when multiple mechanisms need to be combined for releasing multiple quantiles and how to incorporate matrix predictions instead of explicit distributional priors by estimating the additive error between true and predicted covariances. This section also discusses DP contributions of independent interest that arise from our study of measures of prediction quality, specifically our Cauchy-based approach for releasing quantiles without assuming boundedness (Corollary 4.1) and our improved bounds for the `SeparateCov` algorithm proposed by Dong et al. [27] (Corollary 4.2).

4.1 Quantile estimation via prediction-dependent priors

Given a quantile $q \in (0,1)$ and a sorted dataset $\mathbf{x} \in \mathbb{R}^n$ of n distinct points, we want to release $o \in [\mathbf{x}_{[\lfloor qn \rfloor]}, \mathbf{x}_{[\lfloor qn \rfloor+1]})$, i.e. such that the proportion of entries less than o is q. As in prior work [42], the error of o will be the number of points between it and the desired interval:

$$\mathrm{Gap}_q(\mathbf{x}, o) = |\,|\{i : \mathbf{x}_{[i]} < o\}| - \lfloor qn \rfloor\,| = |\max_{\mathbf{x}_{[i]} < o} i - \lfloor qn \rfloor| \tag{1}$$

$\mathrm{Gap}_q(\mathbf{x}, o)$ is constant on intervals $I_k = (\mathbf{x}_{[k]}, \mathbf{x}_{[k+1]}]$ in the partition by $\mathbf{x}$ of $\mathbb{R}$ (let $I_0 = (-\infty, \mathbf{x}_{[1]}]$ and $I_n = (\mathbf{x}_{[n]}, \infty)$), so we also say that $\mathrm{Gap}_q(\mathbf{x}, I_k)$ is the same as $\mathrm{Gap}_q(\mathbf{x}, o)$ for some o in the interior of I_k.

4.1.1 Warm-up: releasing one quantile

For single quantile release we choose perhaps the most natural way of specifying a prediction for a DP algorithm: via the base measure $\mu : \mathbb{R} \mapsto \mathbb{R}_{\geqslant 0}$ of the exponential mechanism:

Theorem 4.1 ([58])**.** *If the* **utility** $u(\mathbf{x}, o)$ *of an outcome o of a query over dataset $\mathbf{x}$ has* **sensitivity** $\max_{o, \mathbf{x} \sim \tilde{\mathbf{x}}} |u(\mathbf{x}, o) - u(\tilde{\mathbf{x}}, o)| \leqslant \Delta$ *then the* **exponential mechanism***, which releases o w.p. $\propto \exp(\frac{\varepsilon}{2\Delta} u(\mathbf{x}, o))\mu(o)$ for some base measure μ, is ε-DP.*

The utility function we use is $u_q = -\operatorname{Gap}_q$, so since this is constant on each interval I_k the mechanism here is equivalent to sampling k w.p. $\propto \exp(\varepsilon u_q(\mathbf{x}, I_k)/2)\mu(I_k)$ and then sampling o from I_k w.p. $\propto \mu(o)$. While the idea of specifying a prior for EM is well-known, the key idea here is to obtain a prediction-dependent bound on the error that reveals a useful measure of the *quality* of the prediction. In particular, we can show (c.f. Lemma A.1) that running EM in this way yields o that w.p. $\geqslant 1-\beta$ satisfies

$$\operatorname{Gap}_q(\mathbf{x}, o) \leqslant \frac{2}{\varepsilon} \log \frac{1/\beta}{\Psi_{\mathbf{x}}^{(q,\varepsilon)}(\mu)} \leqslant \frac{2}{\varepsilon} \log \frac{1/\beta}{\Psi_{\mathbf{x}}^{(q)}(\mu)} \tag{2}$$

where the quantity $\Psi_{\mathbf{x}}^{(q,\varepsilon)} = \int \exp(-\frac{\varepsilon}{2} \operatorname{Gap}_q(\mathbf{x}, o))\mu(o)do$ is the inner product between the prior and the EM score while $\Psi_{\mathbf{x}}^{(q)} = \lim_{\varepsilon\to\infty} \Psi_{\mathbf{x}}^{(q,\varepsilon)} = \mu((\mathbf{x}_{[\lfloor qn \rfloor]}, \mathbf{x}_{[\lfloor qn \rfloor+1]}])$ is the probability that the prior assigns to the optimal interval.

This suggests two metrics of prediction quality: the negative log-inner-products $U_{\mathbf{x}}^{(q,\varepsilon)}(\mu) = -\log \Psi_{\mathbf{x}}^{(q,\varepsilon)}(\mu)$ and $U_{\mathbf{x}}^{(q)}(\mu) = -\log \Psi_{\mathbf{x}}^{(q)}(\mu)$. Both make intuitive sense: we expect predictions μ that assign a high probability to intervals that the EM score weighs heavily to perform well, and EM assigns the most weight to the optimal interval. There are also many ways that these metrics are useful. For one, in the case of perfect prediction—i.e. if μ assigns probability one to the optimal interval $I_{\lfloor qn \rfloor}$—then $\Psi_{\mathbf{x}}^{(q,\varepsilon)}(\mu) = \Psi_{\mathbf{x}}^{(q)}(\mu) = 1$, yielding an upper bound on the error of only $\frac{2}{\varepsilon}\log\frac{1}{\beta}$. Secondly, as we will see, both are also amenable for analyzing robustness (the mechanism's sensitivity to *incorrect* priors) and learning. A final and important quality is that the guarantees using these metrics hold under no extra assumptions. Between the two, the first metric provides a tighter bound on the utility loss while the second does not depend on ε, which may be desirable.

It is also fruitful to analyze the metrics for specific priors. When $\mathbf{x}$ is in a bounded interval (a, b) and $\mu(o) = \frac{1_{o\in(a,b)}}{b-a}$ is the uniform measure, then $\Psi_{\mathbf{x}}^{(q)}(\mu) \geqslant \frac{\psi_{\mathbf{x}}}{b-a}$, where $\psi_{\mathbf{x}}$ is the minimum distance between entries; thus we recover past bounds, e.g. [42, Lemma A.1], that implicitly use this measure to guarantee $\operatorname{Gap}_q(\mathbf{x}, o) \leqslant \frac{2}{\varepsilon}\log\frac{b-a}{\beta\psi_{\mathbf{x}}}$. Here the support of the uniform distribution is correct by assumption as the data is assumed bounded. However, analyzing $\Psi_{\mathbf{x}}^{(q)}$ also yields a novel way of removing this assumption: if we suspect the data lies in (a, b), we set μ to be the Cauchy prior with location $\frac{a+b}{2}$ and scale $\frac{b-a}{2}$. Even if we are wrong about the interval, there exists an $R > 0$ s.t. the data lies in the interval $(\frac{a+b}{2} \pm R)$, so using the Cauchy yields $\Psi_{\mathbf{x}}^{(q)} \geqslant \frac{2(b-a)\psi_{\mathbf{x}}/\pi}{(b-a)^2+4R^2}$ and thus the following guarantee:

Corollary 4.1 (of Lem. A.1)**.** *If the data lies in the interval $(\frac{a+b}{2} \pm R)$ and μ is the Cauchy measure with location $\frac{a+b}{2}$ and scale $\frac{b-a}{2}$ then the output of the exponential mechanism satisfies $\operatorname{Gap}_q(\mathbf{x}, o) \leqslant \frac{2}{\varepsilon}\log\left(\pi\frac{b-a+\frac{4R^2}{b-a}}{2\beta\psi_{\mathbf{x}}}\right)$ w.p. $\geqslant 1-\beta$.*

If $R = \frac{b-a}{2}$, i.e. we get the interval right, then the bound is only an additive factor $\frac{2}{\varepsilon}\log\pi$ worse than before, but if we are wrong then performance degrades as $\mathcal{O}(\log(1+R^2))$, unlike the $\mathcal{O}(R)$ error of the uniform prior. Note our use of a heavy-tailed distribution here: a sub-exponential density decays too quickly and leads to error $\mathcal{O}(R)$ rather than $\mathcal{O}(\log(1+R^2))$. We can also adapt this technique if we know only a single-sided bound, e.g. if values must be positive, by using an appropriate half-Cauchy distribution.

4.1.2 Releasing multiple quantiles

To simultaneously estimate quantiles $q_1, \dots, q_m$ we adapt the `ApproximateQuantiles` method of Kaplan et al. [42], which assigns each q_i to a node in a binary tree and, starting from the root, uses EM with the uniform prior to estimate a quantile before sending the data below the outcome o to its left child and the data above o to its right child. Thus each entry is only involved in $\lceil \log_2 m \rceil$ exponential mechanisms, and so for data in (a, b) the maximum Gap_{q_i} across quantiles is $\mathcal{O}\left(\frac{\log^2 m}{\varepsilon} \log \frac{m(b-a)}{\beta \psi_{\mathbf{x}}}\right)$, which is much better than the naive bound of a linear function of m.

Given one prior μ_i for each q_i, a naive extension of (2) gets a similar polylog(m) bound (c.f. Lem A.2); notably we extend the Cauchy-unboundedness result to multiple quantiles (c.f. Cor. A.1). However the upper bound is not a deterministic function of μ_i, as it depends on restrictions of $\mathbf{x}$ and μ_i to subsets (o_j, o_k) of the domain induced by the outcomes of EM for quantiles q_j and q_k earlier in the tree. It thus does not encode a direct relationship between the prediction and instance data and is less amenable for learning.

We instead want guarantees depending on a more natural metric, e.g. one aggregating $\Psi_{\mathbf{x}}^{(q_i,\varepsilon_i)}(\mu_i)$ from the previous section across pairs (q_i, μ_i). The core issue is that the data splitting makes the probability assigned by a prior μ_i to data outside the interval (o_j, o_k) induced by the outcomes of quantiles q_j and q_k earlier in the tree not affect the distribution of o_i. One way to handle this is to assign this probability mass to the edges of (o_j, o_k), rather than the more natural conditional approach of `ApproximateQuantiles`. We refer to this as "edge-based prior adaptation" and use it to bound $\mathrm{Gap}_{\max} = \max_i \mathrm{Gap}_{q_i}(\mathbf{x}, o_i)$ via the harmonic mean $\Psi_{\mathbf{x}}^{(\varepsilon)}$ of the inner products $\Psi_{\mathbf{x}}^{(q_i,\varepsilon_i)}(\mu_i)$:

Theorem 4.2 (c.f. Thm. A.1)**.** *If $m = 2^k - 1$ for some k, quantiles $q_1, \dots, q_m$ are uniformly spaced, and for each we have a prior $\mu_i : \mathbb{R} \mapsto \mathbb{R}_{\geqslant 0}$, then running `ApproximateQuantiles` with edge-based prior adaptation (c.f. Algorithm 5) is ε-DP, and w.p. $\geqslant 1 - \beta$*

$$\mathrm{Gap}_{\max} \leqslant \frac{2}{\varepsilon} \phi^{\log_2(m+1)} \lceil \log_2(m+1) \rceil \log \frac{m/\beta}{\Psi_{\mathbf{x}}^{(\varepsilon)}} \qquad \text{for} \qquad \Psi_{\mathbf{x}}^{(\varepsilon)} = \left(\sum_{i=1}^{m} \frac{1/m}{\Psi_{\mathbf{x}}^{(q_i,\varepsilon_i)}(\mu_i)} \right)^{-1} \tag{3}$$

Here $\varepsilon_i = \frac{\varepsilon}{\lceil \log_2(m+1) \rceil}$ and $\phi = \frac{1+\sqrt{5}}{2}$ is the golden ratio.

The golden ratio is due to a Fibonacci-type recurrence bounding the maximum Gap_{q_i} at each depth of the tree. $\Psi_{\mathbf{x}}^{(\varepsilon)}$ depends only on $\mathbf{x}$ and predictions μ_i, and it yields a nice error metric $U_{\mathbf{x}}^{(\varepsilon)} = -\log \Psi_{\mathbf{x}}^{(\varepsilon)} = \log \sum_{i=1}^{m} e^{U_{\mathbf{x}}^{(q_i,\varepsilon_i)}}$. However, the dependence of the error on m is worse than that of `ApproximateQuantiles`, as $\phi^{\log_2 m}$ is roughly $\mathcal{O}(m^{0.7})$, although the bound is still sublinear and thus better than the naive baseline of running EM m times. Note that, as in the single-quantile case, we can construct a looser but ε-independent upper bound

$$U_{\mathbf{x}} = -\log \Psi_{\mathbf{x}} = \log \sum_{i=1}^{m} e^{U_{\mathbf{x}}^{(q_i)}} \geqslant U_{\mathbf{x}}^{(\varepsilon)} \tag{4}$$

using the harmonic mean $\Psi_{\mathbf{x}}$ of $\Psi_{\mathbf{x}}^{(q_i)}$. We will make heavy use of this prediction quality measure as a surrogate loss function in applications (c.f. Section 7).

The $\tilde{\mathcal{O}}(\phi^{\log_2 m})$ dependence on the number of quantiles m in Theorem 4.2 results from error compounding across depths of the tree, so we can try to reduce depth by going from a binary to a K-ary tree. This involves running EM $K - 1$ times at each node—and paying $K - 1$ more in budget—to split the data into K subsets; the resulting estimates may also be out of order. However, by showing that sorting them back into order does not increase the error and then controlling the maximum Gap_{q_i} at each depth via another recurrence relation, we prove the following:

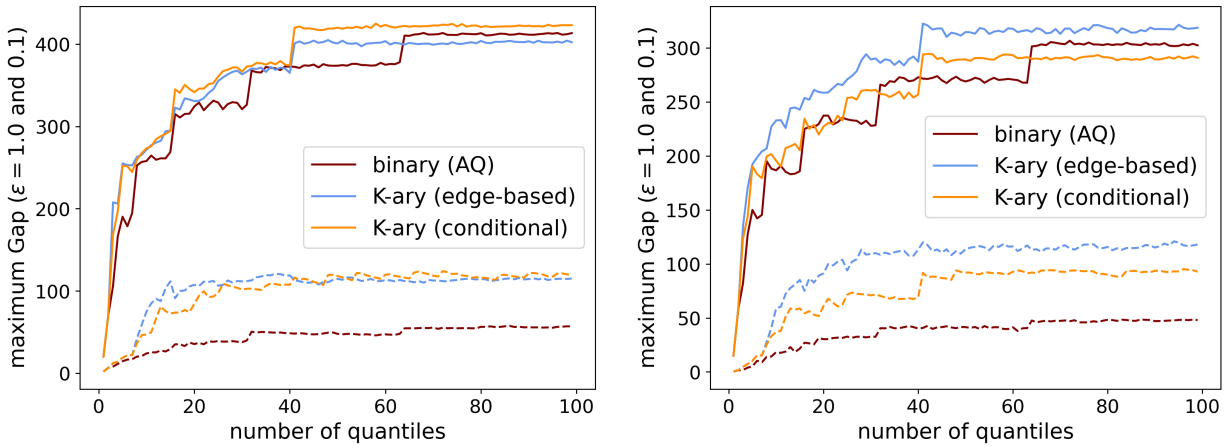

Figure 1: Maximum gap as a function of m for different variants of AQ when using the Uniform prior, evaluated on 1000 samples from a standard Gaussian (left) and the Adult "age" dataset (right). The dashed and solid lines correspond to $\varepsilon = 1$ and 0.1, respectively.

Theorem 4.3 (c.f. Thm. A.2)**.** *For any $q_1, \ldots, q_m$, using $K = \lceil \exp(\sqrt{\log 2 \log(m+1)}) \rceil$ and edge-based adaptation guarantees ε-DP and w.p. $\geqslant 1 - \beta$ has*

$$\mathrm{Gap}_{\max} \leqslant \frac{2\pi^2}{\varepsilon} \exp\left(2\sqrt{\log(2)\log(m+1)}\right) \log \frac{m/\beta}{\Psi_{\mathbf{x}}^{(\varepsilon)}} \tag{5}$$

The rate in m is both sub-polynomial and super-poly-logarithmic ($o(m^\alpha)$ and $\omega(\log^\alpha m)\ \forall\ \alpha > 0$); while asymptotically worse than the prediction-free original result [42], for almost any practical value of m (e.g. $m \in [3, 10^{12}]$) it does not exceed a small constant (e.g. nine) times $\log^3 m$. Thus if the error $-\log \Psi_{\mathbf{x}}^{(\varepsilon)}$ of the prediction is small—i.e. the inner products between priors and EM scores are large on (harmonic) average—then we may do much better with this approach.

We compare K-ary AQ with edge-based adaptation to regular AQ in Figure 1. The original is better at higher ε but similar or worse at higher privacy. We also find that conditional adaptation is only better on discretized data with repetitions, where neither method provides guarantees. Overall, we find that our prior-dependent analysis covers a useful algorithm, but for consistency with past work and due to its better performance at high ε we focus on the original binary approach in experiments.

4.2 Covariance estimation by estimating the prediction error

Encoding predictions as priors for EM and other mechanisms is a natural starting point for integrating external information into DP algorithms, but one might also wish to use a point prediction directly and hope to perform well if some distance measure between it and the output is small. While this is a less natural requirement for quantile release, where errors are measured using data points rather than metrics over the domain they live in, we show how this is easily achievable for the important problem of covariance estimation. In this setting we have a dataset $\mathbf{X} \in \mathbb{R}^{d \times n}$, where each of n records is a d-dimensional column with ℓ_2-norm bounded by 1, and we want to privately release an approximation $\hat{\mathbf{C}}$ of its covariance matrix $\mathbf{C} = \mathbf{X}\mathbf{X}^T/n$ such that the Frobenius distance between the two is small.

Given a prediction $\mathbf{W} \in \mathbb{R}^{d \times d}$ of $\mathbf{C}$, one can immediately construct the trivial, private, prediction-sensitive algorithm of just releasing $\mathbf{W}$, which has the obvious prediction-dependent performance guarantee of $\|\mathbf{W} - \mathbf{C}\|_F$. However, we can hope to use the data to get an error that both decreases

Algorithm 1: `SeparateCov` with predictions

Input: data $\mathbf{X} \in \mathbb{R}^{d\times n}$, symmetric prediction matrix $\mathbf{W} \in \mathbb{R}^{d\times d}$, privacy parameter $\varepsilon > 0$

$\mathbf{U}\Lambda\mathbf{U}^T \leftarrow \mathbf{X}\mathbf{X}^T/n - \mathbf{W}$

$\hat{\Lambda} \leftarrow \Lambda + \operatorname{diag}(\mathbf{z})$ where $\mathbf{z}_{[i]} \sim \operatorname{Lap}\left(\frac{4}{\varepsilon n}\right)$ `// add noise to prediction error eigenvalues`

$\tilde{\mathbf{C}} \leftarrow \mathbf{X}\mathbf{X}^T/n + \mathbf{Z}$ for $\mathbf{Z}_{[i,j]} = \mathbf{Z}_{[j,i]} \sim \operatorname{Lap}\left(\frac{2d\sqrt{2}}{\varepsilon n}\right)$

$\tilde{\mathbf{U}}\tilde{\Lambda}\tilde{\mathbf{U}}^T \leftarrow \tilde{\mathbf{C}} - \mathbf{W}$ `// get eigenvectors of noised prediction error`

Output: $\hat{\mathbf{C}} = \tilde{\mathbf{U}}\hat{\Lambda}\tilde{\mathbf{U}}^T + \mathbf{W}$ `// combine to estimate` $\mathbf{X}\mathbf{X}^T/n - \mathbf{W}$`, then add` $\mathbf{W}$

with n and is small if some distance between the prediction and ground truth is small. To do so, we make use of recent approaches that enjoy *trace-sensitive* guarantees, i.e. their utility improves if $\operatorname{Tr}(\mathbf{X}\mathbf{X}^T)$ is small [4, 27]; for example, the state-of-the-art method `SeparateCov` returns $\hat{\mathbf{C}}$ that is ε-DP and satisfies $\|\hat{\mathbf{C}} - \mathbf{C}\|_F^2 = \tilde{\mathcal{O}}(\frac{d}{\varepsilon^2 n^2} + \frac{d\sqrt{d}}{\varepsilon n}\operatorname{Tr}(\mathbf{X}\mathbf{X}^T/n))$ w.h.p. [27, Lemma 18]. This suggests a natural way to incorporate a symmetric prediction matrix $\mathbf{W}$: use the existing algorithm to privately estimate its difference $\mathbf{C} - \mathbf{W}$ with the ground truth, and then add $\mathbf{W}$ to the result; since $\mathbf{C} - \mathbf{W}$ is no longer PSD, the hope would be to obtain error that scales with its trace norm.

We do exactly this in Algorithm 1, which uses the `SeparateCov` approach of separately estimating and combining eigenvalues and eigenvectors but applies it to $\mathbf{C} - \mathbf{W}$. The one potential issue is showing that their main error bound holds for symmetric matrices with negative eigenvalues, but this follows in Lemma 4.1 by applying their argument to both sides of the spectrum (c.f. Appendix B.1.2):

Lemma 4.1. *For $\mathbf{X} \in \mathbb{R}^{d\times n}$ and symmetric $\mathbf{W} \in \mathbb{R}^{d\times d}$, if $\tilde{\mathbf{U}}\tilde{\Lambda}\tilde{\mathbf{U}}^T = \mathbf{X}\mathbf{X}^T/n - \mathbf{W} + \mathbf{Z}$ for some symmetric $\mathbf{Z} \in \mathbb{R}^{d\times d}$ and $\hat{\Lambda} = \Lambda + \operatorname{diag}(\mathbf{z})$ for $\mathbf{U}\Lambda\mathbf{U}^T = \mathbf{X}\mathbf{X}^T/n - \mathbf{W}$ and some vector $\mathbf{z} \in \mathbb{R}^d$ then*

$$\|\tilde{\mathbf{U}}\hat{\Lambda}\tilde{\mathbf{U}}^T + \mathbf{W} - \mathbf{X}\mathbf{X}^T/n\|_F^2 \leqslant 4\left(\|\mathbf{z}\|_2^2 + \|\mathbf{Z}\|_\infty \|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_{\mathrm{Tr}}\right) \tag{6}$$

Our performance-dependent guarantee then follows via Laplace concentration in Theorem 4.4, which recovers the guarantee of [27, Lemma 18] when $\mathbf{W} = \mathbf{0}_{d\times d}$.[3] The result shows that if we have a good guess of the prediction matrix in terms of trace distance then the error can be made to depend mostly on the first term—which has a better dependence on both d and n—without sacrificing privacy. Note that the algorithm requires the same number of eigen-decompositions as the one without predictions [27] and only requires some extra matrix additions to implement.

Theorem 4.4. *If $\mathbf{X}$ has columns bounded by 1 in ℓ_2-norm then Algorithm 1 is ε-DP and w.p. $\geqslant 1-\beta$*

$$\|\hat{\mathbf{C}} - \mathbf{X}\mathbf{X}^T/n\|_F^2 \leqslant \frac{144d + \mathcal{O}(\log^2\frac{1}{\beta}\log^2 d)}{\varepsilon^2 n^2} + \frac{48d\sqrt{2d} + \mathcal{O}(d\log\frac{1}{\beta}\log d)}{\varepsilon n}\|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_{\mathrm{Tr}} \tag{7}$$

Proof. Following the analysis in [4, Theorem 1] (c.f. Lemma B.1) the ℓ_1-sensitivity of the eigenvalues of $\mathbf{X}\mathbf{X}^T/n - \mathbf{W}$ is $2/n$, and upper-bounding the ℓ_2-sensitivity of the covariance $\mathbf{X}\mathbf{X}^T/n$ of $\sqrt{2}/n$ [12, Lemma 3.2] shows that its ℓ_1-sensitivity is $d\sqrt{2}/n$. Thus the privacy guarantee follows from the composition of two Laplace mechanisms with budget $\varepsilon/2$ each. For the utility guarantee we use concentration of $\|\mathbf{y}\|_2 \leqslant 3\sqrt{d}/2 + \mathcal{O}\left(\log\frac{1}{\beta}\log d\right)$ w.p. $\geqslant 1 - \beta/2$ for i.i.d. $\mathbf{y}_{[i]} \sim \operatorname{Lap}(1)$ [27, Lemma 15] and $\|\mathbf{Y}\|_\infty \leqslant 3\sqrt{d} + \mathcal{O}\left(\log\frac{1}{\beta}\log d\right)$ w.p. $\geqslant 1 - \beta/2$ for i.i.d. $\mathbf{Y}_{[i,j]} \sim \operatorname{Lap}(1)$ for $i \geqslant j$ and $\mathbf{Y}_{[i,j]} = \mathbf{Y}_{[j,i]}$ for $i < j$ [27, Lemma 16]. Substituting $\mathbf{z} = \frac{4}{\varepsilon n}\mathbf{y}$ and $\mathbf{Z} = \frac{2d\sqrt{2}}{\varepsilon n}\mathbf{Y}$ into Lemma 4.1 yields the result. □

[3] Unlike Dong et al. [27] we square the Frobenius norm for the purposes of learning predictions later; in the single-instance setting this is immaterial. Whether one is more interested in one or the other is application-dependent.

Algorithm 2: `MWEM` with predictions

Input: dataset $\mathbf{x} \in \mathbb{Z}^d_{\geqslant 0}$ with n entries, query set $Q \subset [-1,1]^d$, prediction $\mathbf{w} \in \triangle_d$, number of iterations $m > 0$, privacy parameter $\varepsilon > 0$

$\mathbf{w}_1 \leftarrow \mathbf{w}$

for $i = 1, \ldots, m$ **do**

  sample $\mathbf{q}_i \in Q$ w.p. $\propto \exp\left(\frac{\varepsilon}{8m}|\langle \mathbf{q}, \mathbf{x} - n\mathbf{w}_i/\|\mathbf{w}_i\|_1\rangle|\right)$ `// exponential mechanism`

  $\mathbf{w}_{i+1} \leftarrow \mathbf{w}_i \odot \exp\left(\frac{\langle \mathbf{q}_i, \mathbf{x} - n\mathbf{w}_i/\|\mathbf{w}_i\|_1\rangle + \mathrm{Lap}(4m/\varepsilon)}{2n}\mathbf{q}_i\right)$ `// multiplicative weights update`

Output: $\hat{\mathbf{x}} = \frac{n}{m}\sum_{i=1}^m \mathbf{w}_i$ `// release average iterate`

In addition to its computational simplicity, there are two other aspects of Algorithm 1 that are important for understanding the utility of its output: (1) it adds the same amount of noise as the original `SeparateCov` method [27, Algorithm 1], despite our two-sided sensitivity analysis, and (2) it is invariant to perturbations of the prediction matrix by any scalar multiple of the identity, i.e. $\hat{\mathbf{C}}$ is the same when $\mathbf{W}$ is replaced by $\mathbf{W} + c\mathbf{I}_d$ for any $c \in \mathbb{R}$. Crucially, this means we can obtain a tighter bound for free by replacing the trace difference in the upper bound (7) by $\min_{c\in\mathbb{R}} \|\mathbf{X}\mathbf{X}^T/n - \mathbf{W} + c\mathbf{I}_d\|_{\mathrm{Tr}}$. Substituting $\mathbf{W} = \mathbf{0}_{d\times d}$ then yields the following corollary, which is a strict improvement upon the main pure-DP guarantee of Dong et al. [27, Lemma 19] for prediction-free `SeparateCov`:

Corollary 4.2. *If* $\mathbf{X}$ *has columns bounded by 1 in* ℓ_2*-norm then Algorithm 1 with* $\mathbf{W} = \mathbf{0}_{d\times d}$ *returns w.p.* $\geqslant 1 - \beta$ *an estimate* $\hat{\mathbf{C}} \in \mathbb{R}^{d\times d}$ *satisfying*

$$\|\hat{\mathbf{C}} - \mathbf{X}\mathbf{X}^T/n\|_F^2 \leqslant \frac{144d + \mathcal{O}(\log^2\frac{1}{\beta}\log^2 d)}{\varepsilon^2 n^2} + \frac{48d\sqrt{2d} + \mathcal{O}(d\log\frac{1}{\beta}\log d)}{\varepsilon n}\min_{c\in\mathbb{R}}\|\mathbf{X}\mathbf{X}^T/n - c\mathbf{I}_d\|_{\mathrm{Tr}} \quad (8)$$

While this improvement is for a prediction-free method, it is the direct result of the two-sided analysis we needed to incorporate predictions; as with our unbounded quantile release result, this is another example of how learning-augmented analysis is useful even in the prediction-free setting.

Lastly, we point the interested reader to several supplementary results that highlight the broad applicability of our framework. First, while our paper focuses on pure DP (except for learning), the main analysis of Dong et al. [27] is in the zCDP setting; in Appendix B.2 we show that similar guarantees hold there. Note that a prediction-free improvement similar to that of Corollary 4.2 can also be shown for `SeparateCov` under zCDP (c.f. Corollary B.1). Lastly, we show that prediction-dependent guarantee also holds for the older approach of Amin et al. [4], albeit with a modified algorithm and a more involved sensitivity analysis (c.f. Appendix B.3).

4.3 Initializing synthetic dataset construction with a predicted dataset

Our final application is to private data release, in which the goal is to privately respond to queries of a dataset, with the latter being defined via counts of items from some finite universe. For simplicity we will assume an indexing that allows us to specify datasets as vectors $\mathbf{x} \in \mathbb{Z}^d_{\geqslant 0}$, and we will consider a finite set Q of *linear* queries, i.e. ones that can be defined as an inner product of $\mathbf{x}$ with a vector $\mathbf{q} \in [-1,1]^d$. Here again we will incorporate a prediction into an existing algorithm, specifically the `MWEM` method of [36], which uses multiplicative weights to iteratively update a distribution over the data domain and to construct a synthetic dataset $\hat{\mathbf{x}} \in \mathbb{R}^d_{\geqslant 0}$ such that the maximum error $\max_{\mathbf{q}\in Q}|\langle \mathbf{q}, \mathbf{x} - \hat{\mathbf{x}}\rangle|$ of all queries is small. The natural approach here is to assume the prediction can be written as a distribution $\mathbf{w} \in \triangle_d$ and use it instead of the uniform initialization used by [36].

Indeed this observation has been made in both the original work and by [53], who adapt the method to only operate over the support of a source dataset. A prediction-dependent guarantee also follows in a straightforward manner from the original analysis:[4]

Lemma 4.2. *Algorithm 2 is ε-DP and produces $\hat{\mathbf{x}} \in \mathbb{R}^d_{\geqslant 0}$ s.t. w.p. $\geqslant 1-\beta$*

$$\max_{\mathbf{q}\in Q} \frac{|\langle \mathbf{q}, \mathbf{x} - \hat{\mathbf{x}}\rangle|^2}{n} \leqslant \frac{8n}{m} D_{KL}\left(\frac{\mathbf{x}}{n}\middle\|\mathbf{w}\right) + \frac{16m^2}{\varepsilon^2 n}\left(3\log\frac{2m}{\beta} + 2\log^2|Q|\right)^2 \tag{9}$$

Our main purpose with this application is thus to discuss interesting issues arising in its robustness and especially in learning the prediction. We also conclude by noting the similarity of deriving prediction-based guarantees for all four methods—finding algorithms that implicitly use a default prediction such as a uniform distribution or zero matrix—even while the actual algorithms and uses of the predictions are quite different.

5 Robustness-consistency tradeoffs

While prediction-dependent guarantees work well if the prediction is accurate, without safeguards they may perform catastrophically poorly if the prediction is incorrect. In this section we provide robust alternatives to the methods we derived in the previous section, demonstrating the usefulness of the algorithms with predictions framework for understanding robustness when incorporating external information into DP algorithms.

5.1 Quantile estimation

While prediction-dependent guarantees work well if the prediction is accurate, without safeguards they may perform catastrophically poorly if the prediction is incorrect. Quantiles provide a prime demonstration of the importance of robustness, as using priors allows for approaches that may assign very little probability to the interval containing the quantile. For example, if one is confident that it has a specific value $x \in (a, b)$ one can specify a more concentrated prior, e.g. the Laplace distribution around x. Alternatively, if one believes the data is drawn i.i.d. from some a known distribution then μ can be constructed via its CDF using order statistics [23, Equation 2.1.5]. These reasonable approaches can result in distributions with exponential or high-order-polynomial tails, using which directly may work poorly if the prediction is incorrect.

Luckily, for our negative log-inner-product error metric it is straightforward to show a parameterized robustness-consistency tradeoff by simply mixing the prediction prior μ with a robust prior ρ:

Corollary 5.1. *For any prior $\mu : \mathbb{R} \mapsto \mathbb{R}_{\geqslant 0}$, robust prior $\rho : \mathbb{R} \mapsto \mathbb{R}_{\geqslant 0}$, and robustness parameter $\lambda \in [0, 1]$, releasing $o \in \mathbb{R}$ w.p. $\propto \exp(-\varepsilon \operatorname{Gap}_q(\mathbf{x}, o)/2)\mu^{(\lambda)}(o)$ for $\mu^{(\lambda)} = (1-\lambda)\mu + \lambda\rho$ is $\left(\frac{2}{\varepsilon}\log\frac{1/\beta}{\lambda\Psi^{(q,\varepsilon)}_{\mathbf{x}}(\rho)}\right)$-robust and $\left(\frac{2}{\varepsilon}\log\frac{1/\beta}{1-\lambda}\right)$-consistent w.p. $\geqslant 1-\beta$.*

Proof. Apply Lemma A.1 and linearity of $\Psi^{(q,\varepsilon)}_{\mathbf{x}}(\mu^{(\lambda)}) = (1-\lambda)\Psi^{(q,\varepsilon)}_{\mathbf{x}}(\mu) + \lambda\Psi^{(q,\varepsilon)}_{\mathbf{x}}(\rho)$. □

Thus if the interval is finite and we set ρ to be the uniform prior, using $\mu^{(\lambda)}$ in the algorithm will have a high probability guarantee at most $\frac{2}{\varepsilon}\log\frac{1}{\lambda}$-worse than the prediction-free guarantee of Kaplan et al. [42, Lemma A.1], no matter how poor μ is for the data, while also guaranteeing w.p. $\geqslant 1-\beta$ that the error will be at most $\frac{2}{\varepsilon}\log\frac{1/\beta}{1-\lambda}$ if μ is perfect. A similar result holds for the case

[4] Similar to covariance estimation, we consider the mean squared error for the purposes of learning the prediction.

of an infinite interval if we instead use a Cauchy prior. Corollary 5.1 demonstrates the usefulness of the algorithms with predictions framework for not only quantifying improvement in utility using external information but also for making the resulting DP algorithms robust to prediction noise.

The above argument for single-quantiles is straightforward to extend to the negative log of the harmonic means of the inner products. In-fact for the binary case with uniform quantiles we can trade-off between polylog(m)-guarantees similar to those of Kaplan et al. [42] and our prediction-dependent bounds:

Corollary 5.2. *Consider priors $\mu_1, \dots, \mu_m : \mathbb{R} \mapsto \mathbb{R}_{\geqslant 0}$, Cauchy prior $\rho : \mathbb{R} \mapsto \mathbb{R}_{\geqslant 0}$ with location $\frac{a+b}{2}$ and scale $\frac{b-a}{2}$, and robustness parameter $\lambda \in [0,1]$. Then running Algorithm 5 on quantiles that are uniform negative powers of two with $K = 2$, edge-based prior adaptation, $\varepsilon_i = \bar{\varepsilon} = \varepsilon/\lceil \log_2 m \rceil \ \forall\ i$, and priors $\mu_i^{(\lambda)} = \lambda\rho + (1-\lambda)\mu_i \ \forall\ i$ is $\left(\frac{2}{\varepsilon}\lceil \log_2 m \rceil^2 \log\left(\pi m \frac{b-a+\frac{4R^2}{b-a}}{2\lambda\beta\psi_{\mathbf{x}}}\right)\right)$-robust and $\left(\frac{2}{\varepsilon}\phi^{\log_2 m}\lceil \log_2 m \rceil \log\frac{m/\beta}{1-\lambda}\right)$-consistent w.p. $\geqslant 1-\beta$.*

Proof. Apply Lemma A.2, Theorem A.1, and linearity of the inner products in $\hat{\Psi}_{\mathbf{x}}^{(\varepsilon)}$ and $\Psi_{\mathbf{x}}^{(\varepsilon)}$. □

5.2 Covariance estimation

We take a different approach to making our prediction-based covariance estimation method robust to matrices $\mathbf{W}$ with large trace distance to $\mathbf{X}\mathbf{X}^T/n$. Instead of combining the prediction with a robust default, we simply spend some privacy to check whether $\|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_{\mathrm{Tr}}$ is larger than $\|\mathbf{X}\mathbf{X}^T/n\|_{\mathrm{Tr}}$ and if so run Algorithm 1 with the zero matrix instead. This has the following guarantee:

Corollary 5.3. *Pick $\lambda \in (0,1)$ and run Algorithm 1 with privacy $(1-\lambda)\varepsilon$ and symmetric prediction matrix $\mathbf{W}$ if $\|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_{\mathrm{Tr}} + z \leqslant \|\mathbf{X}\mathbf{X}^T\|_{\mathrm{Tr}}/n$ and $\mathbf{0}_{d\times d}$ otherwise, where $z \sim \mathrm{Lap}(\frac{4}{\lambda\varepsilon n})$. This procedure is ε-DP, $\tilde{\mathcal{O}}\left(\frac{d\sqrt{d}}{\varepsilon n}\left(\frac{1}{\varepsilon n} + \|\mathbf{X}\mathbf{X}^T/n\|_{\mathrm{Tr}}\right)\right)$-robust, and $\tilde{\mathcal{O}}\left(\frac{d\sqrt{d}}{\varepsilon^2 n^2}\right)$-consistent w.h.p.*

Proof. By Lemma B.1 the difference $\|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_{\mathrm{Tr}} - \|\mathbf{X}\mathbf{X}^T/n\|_{\mathrm{Tr}}$ has sensitivity $4/n$, so the comparison of $\|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_{\mathrm{Tr}} + z$ and $\|\mathbf{X}\mathbf{X}^T/n\|_{\mathrm{Tr}}$ is equivalent to using the Laplace mechanism with $\lambda\varepsilon$-DP to estimate this difference and then taking the sign. Composing this with the privacy guarantee of Theorem 4.4 yields ε-DP. Since $\Pr\{|z| \geqslant \frac{4}{\lambda\varepsilon n}\log\frac{2}{\beta}\} \leqslant \beta/2$, the matrix $\mathbf{W}_z \in \{\mathbf{W}, \mathbf{0}_{d\times d}\}$ passed to Algorithm 1 satisfies $\|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}_z\|_{\mathrm{Tr}} \leqslant \min\{\|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_{\mathrm{Tr}}, \|\mathbf{X}\mathbf{X}^T/n\|_{\mathrm{Tr}}\} + \frac{4}{\lambda\varepsilon n}\log\frac{2}{\beta}$ w.p. $\geqslant 1-\beta/2$. Applying the utility guarantee of Theorem 4.4 w.p. $1-\beta/2$ for constant $\lambda \in (0,1)$ yields the result. □

Adding this check for robustness make the data-independent term worse by a factor of $\sqrt{d}$; note that the data-dependent term can still be up to $\tilde{\mathcal{O}}(\varepsilon n\|\mathbf{X}\mathbf{X}^T/n\|_{\mathrm{Tr}})$ times larger, so this does not remove the usefulness of the prediction guarantee. The additional cost results from the large dependence on d of this latter term in the original bound, which is itself be caused by a mismatch between the ℓ_1-sensitivity measure and the ℓ_2-bound on the columns. Specifically, if instead the ℓ_1-norms of the columns are assumed bounded by one then the ℓ_1-sensitivity of $\mathbf{X}\mathbf{X}^T/n$ is $2/n$, making the numerator of the second term in Theorem 4.4 be $\tilde{\mathcal{O}}(\sqrt{d})$ and thus causing no (asymptotic) cost due to robustness.[5] Similarly, under the original assumption the corresponding term in the ℓ_2-sensitivity-based zCDP guarantee is also $\tilde{\mathcal{O}}(\sqrt{d})$ (c.f. Theorem B.2) and leads to a term that is $\mathcal{O}(\sqrt{n/d})$ worse (multiplicatively) due to robustness (c.f. Corollary B.2); while worse in some regimes, in sufficiently high dimensions ($d = \Omega(n)$) this means no (asymptotic) cost of robustness.

[5] It is not as clear that the ℓ_1-sensitivity of the eigenvalues would be as affected by the different assumption.

Algorithm 3: Non-Euclidean DP-FTRL

Input: Datasets $\mathbf{x}_1, \dots, \mathbf{x}_T$ arriving in a stream in arbitrary order, domain $\Theta \subset \mathbb{R}^p$, step-size $\eta > 0$, noise scale $\sigma > 0$, ℓ_2-sensitivity $\Delta_2 > 0$, regularizer $\phi : \Theta \mapsto \mathbb{R}$

$\mathbf{g}_1 \leftarrow \mathbf{0}_p$

$\mathcal{T} \leftarrow$ `InitializeTree`(T, σ^2, Δ_2) `// start tree aggregation [41, Section B.1]`

for $t = 1, \dots, T$ **do**

  $\theta_t \leftarrow \arg\min_{\theta \in \Theta} \phi(\theta) + \eta \langle \mathbf{g}_t, \theta \rangle$

  suffer $\ell(\theta_t; \mathbf{x}_t)$

  $\mathcal{T} \leftarrow$ `AddToTree`$(\mathcal{T}, t, \nabla_\theta \ell(\theta_t; \mathbf{x}_t))$ `// add gradient to tree [41, Section B.1]`

  $\mathbf{g}_{t+1} \leftarrow$ `GetSum`$(\mathcal{T}, t)$ `// estimate` $\sum_{s=1}^{t} \nabla_\theta \ell(\theta_s; \mathbf{x}_s)$ `[41, Section B.1]`

5.3 Data release

As with quantiles, a natural approach to making data release robust is to mix the initialization with the default uniform distribution, achieving a tunable tradeoff. In the following result we specify the number of steps based on the the worst-case guarantees for a prediction-free algorithm and obtain a favorable tradeoff that allows for very small values of λ for high consistency while still maintaining robustness due the latter's $\log \frac{d}{\lambda}$ dependence.

Corollary 5.4. *For $d \geqslant 2$ and any $\mathbf{w} \in \triangle_d$, running Algorithm 2 with $m = \sqrt[3]{\frac{\varepsilon^2 n^2 \log d}{2 \log^4 |Q|}}$ and initialization $\mathbf{w}^{(\lambda)} = (1-\lambda)\mathbf{w} + \lambda \mathbf{1}_d/d$ is ε-DP, $\tilde{\mathcal{O}}\left((1 + \log^{4/3}|Q|) \sqrt[3]{\frac{n}{\varepsilon^2 \log d}} \log \frac{d}{\lambda}\right)$-robust, and $\tilde{\mathcal{O}}\left(\lambda(1 + \log^{4/3}|Q|) \sqrt[3]{\frac{n \log^2 d}{\varepsilon^2}}\right)$-consistent w.h.p., where $\tilde{\mathcal{O}}$ hides poly-log terms in $\varepsilon, n, \log d, \log |Q|$.*

Proof. If $\mathbf{w} = \frac{\mathbf{x}}{n}$ then we have $D_{KL}(\frac{\mathbf{x}}{n} || \mathbf{w}^{(\lambda)}) \leqslant (1-\lambda) D_{KL}(\frac{\mathbf{x}}{n} || \mathbf{w}) + \lambda D_{KL}(\frac{\mathbf{x}}{n} || \mathbf{w} \leqslant \lambda \log d$ by joint convexity of D_{KL}. On the other hand $D_{KL}(\frac{\mathbf{x}}{n} || \mathbf{w}^{(\lambda)}) \leqslant \langle \frac{\mathbf{x}}{n}, \log \frac{d\mathbf{x}}{\lambda n} \rangle \leqslant \log \frac{d}{\lambda}$. Substituting into Lemma 4.2 and simplifying yields the result. □

6 Learning predictions, privately

Our last objective will be to *learn* predictions that do well according to the quality metrics we have defined, which themselves control the utility loss of running the DP algorithms. Past work, e.g. the public-private framework [10, 11, 53], has often focused on domain adaptation-type learning where we adapt a public source to private target. We avoid assuming access to large quantities of i.i.d. public data and instead assume numerous tasks that can have sensitive data and may be adversarially generated. As discussed before, this is the online setting where we see loss functions defined by a sequence of datasets $\mathbf{x}_1, \dots, \mathbf{x}_T$ and aim to compete with best fixed prediction in-hindsight. Note such a guarantee can also be converted into excess risk bounds (c.f. Appendix D.1).

6.1 Non-Euclidean DP-FTRL

Because the optimization domain is not well-described by the ℓ_2-ball, we are able to obtain significant savings in dependence on the dimension and in some cases even in the number of instances T by extending the DP-FTRL algorithm of [41] to use non-Euclidean regularizers, as in Algorithm 4. For this we prove the following regret guarantee:

Algorithm 4: Non-Euclidean DP-FTRL. For the `InitializeTree`, `AddToTree`, and `GetSum` subroutines see Kairouz et al. [41, Section B.1].

Input: Datasets $\mathbf{x}_1, \dots, \mathbf{x}_T$ arriving in a stream in arbitrary order, domain $\Theta \subset \mathbb{R}^p$, step-size $\eta > 0$, noise scale $\sigma > 0$, ℓ_2-sensitivity $\Delta_2 > 0$, regularizer $\phi : \Theta \mapsto \mathbb{R}$

$\mathbf{g}_1 \leftarrow \mathbf{0}_p$

$\mathcal{T} \leftarrow$`InitializeTree`(T, σ^2, Δ_2) `// start tree aggregation`

for $t = 1, \dots, T$ **do**

  $\theta_t \leftarrow \arg\min_{\theta \in \Theta} \phi(\theta) + \eta \langle \mathbf{g}_t, \theta \rangle$

  suffer $\ell_{\mathbf{x}_t}(\theta_t)$

  $\mathcal{T} \leftarrow$`AddToTree`$(\mathcal{T}, t, \nabla_\theta \ell_{\mathbf{x}_t}(\theta_t))$ `// add gradient to tree`

  $\mathbf{g}_{t+1} \leftarrow$`GetSum`$(\mathcal{T}, t)$ `// estimate` $\sum_{s=1}^t \nabla_\theta \ell_{\mathbf{x}_s}(\theta_s)$

Theorem 6.1. *Let $\theta_1, \dots, \theta_T$ be the outputs of Algorithm 4 using a regularizer $\phi : \Theta \mapsto \mathbb{R}$ that is strongly-convex w.r.t. $\|\cdot\|$. Suppose $\forall\ t \in [T]$ that $\ell_{\mathbf{x}_t}(\cdot)$ is L-Lipschitz w.r.t. $\|\cdot\|$ and its gradient has ℓ_2-sensitivity Δ_2. Then w.p. $\geqslant 1 - \beta'$ we have $\forall\ \theta^* \in \Theta$ that*

$$\sum_{t=1}^T \ell(\theta_t; \mathbf{x}_t) - \ell(\theta^*; \mathbf{x}_t) \leqslant \frac{\phi(\theta^*) - \phi(\theta_1)}{\eta} + \eta L \left(L + \left(G + C\sqrt{2 \log \frac{T}{\beta'}} \right) \sigma \Delta_2 \sqrt{\lceil \log_2 T \rceil} \right) T \quad (10)$$

where $G = \mathbb{E}_{\mathbf{z} \sim \mathcal{N}(\mathbf{0}_p, \mathbf{I}_p)} \sup_{\|\mathbf{y}\| \leqslant 1} \langle \mathbf{z}, \mathbf{y} \rangle = \mathbb{E}_{\mathbf{z} \sim \mathcal{N}(\mathbf{0}_p, 1)} \|\mathbf{z}\|_*$ is the Gaussian width of the unit $\|\cdot\|$-ball and C is the Lipschitz constant of $\|\cdot\|_*$ w.r.t. $\|\cdot\|_2$. Furthermore, for any $\varepsilon' \leqslant 2 \log \frac{1}{\delta'}$, setting $\sigma = \frac{1}{\varepsilon'} \sqrt{2 \lceil \log_2 T \rceil \log \frac{1}{\delta'}}$ makes the algorithm (ε', δ')-DP.

Proof. The privacy guarantee follows from past results for tree aggregation [41, 69]. For all $t \in [T]$ we use the shorthand $\nabla_t = \nabla_\theta \ell_{\mathbf{x}_t}(\theta_t)$; we can then define $\tilde{\theta}_t = \arg\min_{\theta \in \Theta} \phi(\theta) + \eta \sum_{s=1}^t \langle \nabla_s, \theta \rangle$ and $\mathbf{b}_t = \mathbf{g}_t - \sum_{s=1}^t \nabla_s$. Then

$$\begin{aligned}
\sum_{t=1}^T \ell_{\mathbf{x}_t}(\theta_t) - \ell_{\mathbf{x}_t}(\theta^*) &\leqslant \sum_{t=1}^T \langle \nabla_t, \theta_t - \theta^* \rangle \\
&= \sum_{t=1}^T \langle \nabla_t, \tilde{\theta}_t - \theta^* \rangle + \sum_{t=1}^T \langle \nabla_t, \theta_t - \tilde{\theta}_t \rangle \\
&\leqslant \frac{\phi(\theta^*) - \phi(\theta_1)}{\eta} + \eta \sum_{t=1}^T \|\nabla_t\|_*^2 + \sum_{t=1}^T \|\nabla_t\|_* \|\tilde{\theta}_t - \theta_t\| \\
&\leqslant \frac{\phi(\theta^*) - \phi(\theta_1)}{\eta} + \eta L \left(LT + \sum_{t=1}^T \|\mathbf{b}_t\|_* \right)
\end{aligned} \quad (11)$$

where the first inequality follows from the standard linear approximation in online convex optimization [72], the second by the regret guarantee for online mirror descent [68, Theorem 2.15], and the last by applying McMahan [57, Lemma 7] with $\phi_1(\cdot) = \phi(\cdot) + \eta \sum_{s=1}^t \langle \nabla_s, \cdot \rangle$, $\psi(\cdot) = \eta \langle \mathbf{b}_t, \cdot \rangle$, and $\phi_2(\cdot) = \phi(\cdot) + \eta \langle \mathbf{g}_t, \cdot \rangle$, yielding $\|\tilde{\theta}_t - \theta_t\| \leqslant \eta \|\mathbf{b}_t\|_*\ \forall\ t \in [T]$. The final guarantee follows by observing that the tree aggregation protocol adds noise $\mathbf{b}_t \sim \mathcal{N}(\mathbf{0}_p, \sigma^2 \Delta_2^2 \lceil \log_2 t \rceil)$ to each prefix sum and applying the Gaussian concentration of Lipschitz functions [13, Theorem 5.6]. □

The above proof of this result follows that of the Euclidean case, which can be recovered by setting $G = \mathcal{O}(\sqrt{d})$, $C = 1$, and $\Delta_2 = \mathcal{O}(L)$.[6] In addition to the Lipschitz constants L, a key term that can lead to improvement is the Gaussian width G of the unit $\|\cdot\|$-ball, which for the Euclidean case is $\mathcal{O}(\sqrt{d})$ but e.g. for $\|\cdot\| = \|\cdot\|_1$ is $\mathcal{O}(\sqrt{\log d})$. Note that a related dependence on the Laplace width of Θ appears in Agarwal and Singh [2, Theorem 3.1], although their guarantee only holds for linear losses and is not obviously extendable. Thus Theorem 6.1 may be of independent interest for DP online learning.

6.2 Learning priors for one or more quantiles

We now turn to learning priors $\mu_t = \big(\mu_{t[1]}, \cdots, \mu_{t[m]}\big)$ to privately estimate m quantiles $q_1, \ldots, q_m$ on each of a sequence of T datasets $\mathbf{x}_t$. We will aim to set $\mu_1, \ldots, \mu_T$ s.t. if at each time t we run Algorithm 5 with privacy $\varepsilon > 0$ then the guarantees given by Lemmas A.1 and A.2 will be asymptotically at least as good as those of the best set of measures in $\mathcal{F}^m$, where $\mathcal{F}$ is some class of measures on the finite interval (a, b). The latter we will assume to be known and bounded. Note that in this section almost all single-quantile results follow from setting $m = 1$, so we study it jointly with learning for multiple quantiles.

Ignoring constants, the loss functions implied by our prediction-dependent upper bounds for multiple-quantiles are the following negative log-harmonic sums of prior-EM inner-products:

$$U^{(\varepsilon)}_{\mathbf{x}_t}(\mu) = \log \sum_{i=1}^{m} \frac{1}{\Psi^{(q_i,\varepsilon_i)}_{\mathbf{x}_t}(\mu_{[i]})} = \log \sum_{i=1}^{m} \frac{1}{\int_a^b \exp(-\varepsilon_i \operatorname{Gap}_{q_i}(\mathbf{x}_t, o)/2)\mu_{[i]}(o)do} \tag{12}$$

We focus on minimizing regret $\max_{\mu\in\mathcal{F}^m} \sum_{t=1}^{T} U^{(\varepsilon)}_{\mathbf{x}_t}(\mu_t) - U^{(\varepsilon)}_{\mathbf{x}_t}(\mu)$ over these losses for priors $\mu_{[i]}$ in a class $\mathcal{F}_{V,d}$ of probability measures that are piecewise V-Lipschitz over each of d intervals uniformly partitioning $[a, b)$. This is chosen because it covers the class $\mathcal{F}_{V,1}$ of V-Lipschitz measures and the class of $\mathcal{F}_{0,d}$ of discrete measures that are constant on each of the d intervals. The latter can be parameterized by $\mathbf{W} \in \triangle_d^m$, so that the losses have the form $U^{(\varepsilon)}_{\mathbf{x}_t}(\mu_{\mathbf{W}}) = \log \sum_{i=1}^{m} \langle \mathbf{s}_{t,i}, \mathbf{W}_{[i]}]\rangle^{-1}$ for $\mathbf{s}_{t,i} \in \mathbb{R}^d_{\geqslant 0}$. This can be seen by setting $\mathbf{s}_{t,i[j]} = \frac{d}{b-a} \int_{a+\frac{b-a}{d}(j-1)}^{a+\frac{b-a}{d}j} \exp(-\varepsilon_i \operatorname{Gap}_{q_i}(\mathbf{x}_t, o)/2)do$ and $\mu_{\mathbf{W}_{[i]}}(o) = \frac{d}{b-a}\mathbf{W}_{[i,j]}$ over the interval $\big[a + \frac{b-a}{d}(j-1), a + \frac{b-a}{d}j\big)$. Finally, for $\lambda \in [0, 1]$ we also let $\mathcal{F}^{(\lambda)} = \{(1-\lambda)\mu + \frac{\lambda}{b-a} : \mu \in \mathcal{F}\}$ denote the class of mixtures of measures $\mu \in \mathcal{F}$ with the uniform measure.

As detailed in Appendix D.2, losses of the form $-\log\langle \mathbf{s}_t, \cdot\rangle$, i.e. those above when $m = 1$, have been studied in (non-private) online learning [7, 37]. However, specialized approaches, e.g. those taking advantage exp-concavity, are not obviously implementable via prefix sums of gradients, the standard approach to private online learning [2, 41, 69]. Still, we can at least use the fact that we are optimizing over a product of simplices to improve the dimension-dependence by applying Non-Euclidean DP-FTRL with entropic regularizer $\phi(\mathbf{W}) = m\langle \mathbf{W}, \log \mathbf{W}\rangle$, which yields an m-way exponentiated gradient (EG) update [46]. To apply its guarantee for the problem of learning priors for quantile estimation, we need to bound the sensitivity of the gradients $\nabla_{\mathbf{W}} U^{(\varepsilon)}_{\mathbf{x}_t}(\mu_{\mathbf{W}})$ to changes in the underlying datasets $\mathbf{x}_t$. This is often done via a bound on the gradient norm, which in our case is unbounded near the boundary of the simplex. We thus restrict to γ-robust priors for some $\gamma \in (0, 1]$ by constraining $\mathbf{W} \in \triangle_d^m$ to have entries lower bounded by γ/d—a domain where $\|\nabla_{\mathbf{W}} U^{(\varepsilon)}_{\mathbf{x}_t}(\mu_{\mathbf{W}})\|_1 \leqslant d/\gamma$ (c.f. Lemma D.1)—and bounding the resulting approximation error; we are not aware of even a non-private approach that avoids this except by taking advantage of exp-concavity [37].

[6] As of this writing, the most recent arXiv version of Kairouz et al. [41, Theorem C.1] has a typo leading to missing a Lipschitz constant in the bound, confirmed via correspondence with the authors.

We thus have a bound of $2d/\gamma$ on the ℓ_2-sensitivity. However, this may be too loose since it allows for changing the entire dataset $\mathbf{x}_t$, whereas we are only interested in changing one entry. Indeed, for small ε we can obtain a tighter bound:

Lemma 6.1. *The ℓ_2-sensitivity of $\nabla_{\mathbf{w}} U^{(\varepsilon)}_{\mathbf{x}_t}(\mu_{\mathbf{w}})$ is $\frac{d}{\gamma}\min\{2, e^{\tilde{\varepsilon}_m}-1\}$, where $\tilde{\varepsilon}_m = (1+1_{m>1})\max_i \varepsilon_i$.*

Proof for $m=1$; c.f. Appendix D.2.1. Let $\tilde{\mathbf{x}}_t$ be a neighboring dataset of $\mathbf{x}_t$ and let $U^{(\varepsilon)}_{\tilde{\mathbf{x}}_t}(\mu_{\mathbf{W}}) = -\log\langle\tilde{\mathbf{s}}_t, \mathbf{w}\rangle$ be the corresponding loss. Note that $\max_{o\in[a,b]} |\operatorname{Gap}_q(\mathbf{x}_t, o) - \operatorname{Gap}_q(\tilde{\mathbf{x}}_t, o)| \leqslant 1$ so

$$\tilde{\mathbf{s}}_{t[j]} = \int_{a+\frac{b-a}{d}(j-1)}^{a+\frac{b-a}{d}j} \exp\left(-\frac{\varepsilon}{2}\operatorname{Gap}_q(\tilde{\mathbf{x}}_t, o)\right) do \in e^{\pm\frac{\varepsilon}{2}} \int_{a+\frac{b-a}{d}(j-1)}^{a+\frac{b-a}{d}j} \exp\left(-\frac{\varepsilon}{2}\operatorname{Gap}_q(\mathbf{x}_t, o)\right) do = e^{\pm\frac{\varepsilon}{2}}\mathbf{s}_{t[j]} \tag{13}$$

Therefore since $m=1$ we denote $\mathbf{w} = \mathbf{W}_{[1]}$, $\mathbf{s}_t = \mathbf{s}_{t,1}$, and $\tilde{\mathbf{s}}_t = \tilde{\mathbf{s}}_{t,1}$ and have

$$\begin{aligned}
\|\nabla_{\mathbf{w}} U^{(\varepsilon)}_{\mathbf{x}_t}(\mu_{\mathbf{w}}) - \nabla_{\mathbf{w}} U^{(\varepsilon)}_{\tilde{\mathbf{x}}_t}(\mu_{\mathbf{w}})\|_2 &= \sqrt{\sum_{j=1}^d \left(\frac{\mathbf{s}_{t[j]}}{\langle\mathbf{s}_t, \mathbf{w}\rangle} - \frac{\tilde{\mathbf{s}}_{t[j]}}{\langle\tilde{\mathbf{s}}_t, \mathbf{w}\rangle}\right)^2} \\
&= \sqrt{\sum_{j=1}^d \frac{\mathbf{s}^2_{t[j]}}{\langle\mathbf{s}_t, \mathbf{w}\rangle^2}\left(1 - \frac{\tilde{\mathbf{s}}_{t[j]}\langle\mathbf{s}_t, \mathbf{w}\rangle}{\mathbf{s}_{t[j]}\langle\tilde{\mathbf{s}}_t, \mathbf{w}\rangle}\right)^2} \\
&\leqslant \|\nabla_{\mathbf{w}} U^{(\varepsilon)}_{\mathbf{x}_t}(\mu_{\mathbf{w}})\|_1 \max_j |1-\kappa_j|
\end{aligned} \tag{14}$$

where $\kappa_j = \frac{\tilde{\mathbf{s}}_{t[j]}\langle\mathbf{s}_t,\mathbf{w}\rangle}{\mathbf{s}_{t[j]}\langle\tilde{\mathbf{s}}_t,\mathbf{w}\rangle} \in \frac{\mathbf{s}_{t[j]}\exp(\pm\frac{\varepsilon}{2})\langle\mathbf{s}_t,\mathbf{w}\rangle}{\mathbf{s}_{t[j]}\langle\mathbf{s}_t,\mathbf{w}\rangle\exp(\pm\frac{\varepsilon}{2})} \in \exp(\pm\varepsilon)$ by Equation 13. The result follows by taking the minimum with the bound on the Euclidean norm of the gradient (Lemma D.1). □

Since $e^{\varepsilon}-1 \leqslant 2\varepsilon$ for $\varepsilon \in (0, 1.25]$, for small ε this allows us to add less noise in DP-FTRL. With this sensitivity bound, we apply Algorithm 4 using the entropic regularizer to obtain the following result (c.f. Appendix D.2.2):

Theorem 6.2. *For $d \geqslant 2, \gamma \in (0, 1/2]$ if we run Algorithm 4 on $U^{(\varepsilon)}_{\mathbf{x}_t}(\mu_{\mathbf{W}}) = \log\sum_{i=1}^m \frac{1}{\Psi^{(q_i,\varepsilon_i)}_{\mathbf{x}_t}(\mu_{\mathbf{W}})}$ over γ-robust priors with step-size $\eta = \frac{\gamma m}{d}\sqrt{\frac{\log(d)/T}{1+\left(2\sqrt{\log(md)}+\sqrt{2\log\frac{T}{\beta'}}\right)\sigma\sqrt{\log\lceil\log_2 T\rceil}\min\{1,\tilde{\varepsilon}_m\}}}$ and regularizer $\phi(\mathbf{W}) = m\langle\mathbf{W}, \log\mathbf{W}\rangle$ then for any $V \geqslant 0$, $\lambda \in [0,1]$, and $\beta' \in (0,1]$ we will have regret*

$$\begin{aligned}
&\max_{\mu_{[i]}\in\mathcal{F}^{(\lambda)}_{V,d}} \sum_{t=1}^T U^{(\varepsilon)}_{\mathbf{x}_t}(\mu_{\mathbf{W}_t}) - U^{(\varepsilon)}_{\mathbf{x}_t}(\mu) \\
&\qquad\leqslant \frac{VmT}{\gamma d\bar{\psi}}(b-a)^3 + 2\max\{\gamma-\lambda, 0\}T\log 2 \\
&\qquad\quad + \frac{2md}{\gamma}\sqrt{\left(1 + \left(4\sqrt{\log(md)} + 2\sqrt{2\log\frac{T}{\beta'}}\right)\sigma\sqrt{\lceil\log_2 T\rceil}\min\{1,\tilde{\varepsilon}_m\}\right)T\log d}
\end{aligned} \tag{15}$$

w.p. $\geqslant 1-\beta'$, where $\bar{\psi}$ is the harmonic mean of $\psi_{\mathbf{x}_t} = \min_k \mathbf{x}_{t[k+1]} - \mathbf{x}_{t[k]}$ and $\tilde{\varepsilon}_m = (1+1_{m>1})\max_i\varepsilon_i$. For any $\varepsilon' \leqslant 2\log\frac{1}{\delta'}$ setting $\sigma = \frac{1}{\varepsilon'}\sqrt{2\lceil\log_2 T\rceil\log\frac{1}{\delta'}}$ makes this procedure (ε', δ')-DP.

Note that in the case of $V > 0$ or $\lambda = 0$ we will need to set $d = \omega_T(1)$ or $\gamma = o_T(1)$ in order to obtain sublinear regret. Thus for these more difficult classes our extension of DP-FTRL to non-Euclidean regularizers yields improved rates, as in the Euclidean case the first term has an extra $\sqrt[4]{d}$-factor. The following provides some specific upper bounds derived from Theorem 6.2:

Corollary 6.1. *For each of the following classes of priors there exist settings of d (where needed) and $\gamma > 0$ in Theorem 6.2 that guarantee obtain the following regret w.p. $\geqslant 1 - \beta'$:*

1. *λ-robust and discrete $\mu_{[i]} \in \mathcal{F}_{0,d}^{(\lambda)}$:* $\tilde{\mathcal{O}}\left(\frac{dm}{\lambda}\sqrt{\left(1 + \frac{\min\{1,\tilde{\varepsilon}_m\}}{\varepsilon'}\right)T}\right)$
2. *λ-robust and V-Lipschitz $\mu_{[i]} \in \mathcal{F}_{V,1}^{(\lambda)}$:* $\tilde{\mathcal{O}}\left(\frac{m}{\lambda}\sqrt{\frac{V}{\psi}}\sqrt[4]{\left(1 + \frac{\min\{1,\tilde{\varepsilon}_m\}}{\varepsilon'}\right)T^3}\right)$
3. *discrete $\mu_{[i]} \in \mathcal{F}_{0,d}$:* $\tilde{\mathcal{O}}\left(\sqrt{dm}\sqrt[4]{\left(1 + \frac{\min\{1,\tilde{\varepsilon}_m\}}{\varepsilon'}\right)T^3}\right)$
4. *V-Lipschitz $\mu_{[i]} \in \mathcal{F}_{V,1}$:* $\tilde{\mathcal{O}}\left(\sqrt{m}\sqrt[4]{\frac{V}{\psi}}\sqrt[8]{\left(1 + \frac{\min\{1,\tilde{\varepsilon}_m\}}{\varepsilon'}\right)T^7}\right)$

Thus competing with λ-robust priors with discrete PDFs enjoys the fastest regret rate of $\tilde{\mathcal{O}}(\sqrt{T})$, while either removing robustness or competing with any V-Lipschitz prior has regret $\tilde{\mathcal{O}}(T^{3/4})$, and doing both has regret $\tilde{\mathcal{O}}(T^{7/8})$. When comparing to Lipschitz priors we also incur a dependence on the inverse of minimum datapoint separation, which may be small. A notable aspect of all the bounds is that the regret *improves* with small ε due to the sensitivity analysis in Lemma 6.1; indeed for $\varepsilon = \mathcal{O}(\varepsilon')$ the regret bound only has a $\mathcal{O}(\log \frac{1}{\delta'})$-dependence on the privacy guarantee. Finally, for λ-robust priors we can also apply the $\log \frac{b-a}{\lambda\psi}$-boundedness of $-\log \Psi_{\mathbf{x}}^{(q,\varepsilon)}(\mu)$ and standard online-to-batch conversion (e.g. Cesa-Bianchi et al. [17, Proposition 1] to obtain the following sample complexity guarantee:

Corollary 6.2. *For any $\alpha > 0$ and distribution $\mathcal{D}$ over finite datasets $\mathbf{x}$ of ψ-separated points from (a, b), if we run the algorithm in Theorem 6.2 on $T = \Omega\left(\frac{\log \frac{1}{\beta'}}{\alpha^2}\left(\frac{d^2m^2}{\lambda^2}\left(1 + \frac{\min\{1,\tilde{\varepsilon}_m\}}{\varepsilon'}\right) + \log^2 \frac{1}{\lambda\psi}\right)\right)$ i.i.d. samples from $\mathcal{D}$ then w.p. $\geqslant 1-\beta'$ the average $\hat{\mathbf{W}} = \frac{1}{T}\sum_{t=1}^T \mathbf{W}_t$ of the resulting iterates satisfies $\mathbb{E}_{\mathbf{x}\sim\mathcal{D}} \log \sum_{i=1}^m \frac{1}{\Psi_{\mathbf{x}}^{(q_i,\varepsilon_i)}(\mu_{\hat{\mathbf{W}}_{[i]}})} \leqslant \min_{\mu_{[i]}\in\mathcal{F}_{0,d}^{(\lambda)}} \mathbb{E}_{\mathbf{x}\sim\mathcal{D}} \log \sum_{i=1}^m \frac{1}{\Psi_{\mathbf{x}}^{(q_i,\varepsilon_i)}(\mu_{[i]})} + \alpha$. For α-suboptimality w.r.t. $\mu_{[i]} \in \mathcal{F}_{V,1}^{(\lambda)}$ the sample complexity is $\Omega\left(\frac{\log \frac{1}{\beta'}}{\alpha^2}\left(\frac{V^2m^2}{\lambda^4\psi^2\alpha^2}\left(1 + \frac{\min\{1,\tilde{\varepsilon}_m\}}{\varepsilon'}\right) + \log^2 \frac{1}{\lambda\psi}\right)\right)$.*

6.3 Learning to estimate covariance matrices

We next study how to learn prediction matrices for DP covariance estimation by targeting the trace distance between them and the ground truth. This is a more straightforward learning task, with Lipschitz losses over a finite-dimensional domain. Indeed, we could apply standard DP-FTRL and obtain regret $\tilde{\mathcal{O}}(\sqrt{(1 + d/\varepsilon')dT})$ w.r.t. any symmetric matrix $\mathbf{W}$ because the losses $U_{\mathbf{X}_t}(\mathbf{W}) = \|\mathbf{X}_t\mathbf{X}_t^T/|\mathbf{X}_t| - \mathbf{W}\|_{\mathrm{Tr}}$ are $\sqrt{d}$-Lipschitz w.r.t. the Frobenius norm. However, we can reduce the dependence on the dimension by a $\sqrt{d}$-factor by combining our non-Euclidean DP-FTRL algorithm with the well-known matrix-learning technique of using Schatten p-norm regularization [29]:

Theorem 6.3. *Let $\mathbf{X}_1, \dots, \mathbf{X}_T$ be a sequence of datasets with d-dimensional columns bounded by 1 in the ℓ_2-norm. If we run Algorithm 4 on losses $U_{\mathbf{X}_t}(\mathbf{W}) = \|\mathbf{X}_t\mathbf{X}_t^T/|\mathbf{X}_t| - \mathbf{W}\|_{\mathrm{Tr}}$ with step-size $\eta = \sqrt{\frac{6\log(d)/T}{1+\left(\sqrt{d}+\sqrt{2\log\frac{T}{\beta'}}\right)\sigma\sqrt{d\lceil\log_2 T\rceil}}}$ and regularizer $\phi(\cdot) = \frac{3}{2}\log d\|\|\cdot\|\|_p^2$ then we will have regret*

$$\max_{\mathbf{W}\in\mathbb{R}^{d\times d}} \sum_{t=1}^{T} U_{\mathbf{X}_t}(\mathbf{W}_t) - U_{\mathbf{X}_t}(\mathbf{W}) \leqslant \mathcal{O}\left(\sqrt{\left(1 + \left(\sqrt{d} + \sqrt{\log\frac{T}{\beta'}}\right)\sigma\sqrt{d\lceil\log_2 T\rceil}\right) T\log d}\right) \tag{16}$$

w.p. $\geqslant 1-\beta'$. For any $\varepsilon' \leqslant 2\log\frac{1}{\delta'}$ setting $\sigma = \frac{1}{\varepsilon'}\sqrt{2\lceil\log_2 T\rceil\log\frac{1}{\delta'}}$ makes this procedure (ε', δ')-DP. Furthermore, suppose the datasets are drawn i.i.d. from some distribution $\mathcal{D}$. If we run the same algorithm and return the average prediction $\hat{\mathbf{W}} = \frac{1}{T}\sum_{t=1}^{T}\mathbf{W}_t$ then $T = \tilde{\Omega}\left(\frac{1+d/\varepsilon'}{\alpha^2}\log\frac{1}{\beta'}\right)$ samples suffice to guarantee that w.p. $1-\beta'$

$$\mathbb{E}_{\mathbf{X}\sim\mathcal{D}}\|\mathbf{X}\mathbf{X}^T/|\mathbf{X}| - \hat{\mathbf{W}}\|_{\mathrm{Tr}} \leqslant \min_{\mathbf{W}} \mathbb{E}_{\mathbf{X}\sim\mathcal{D}}\|\mathbf{X}\mathbf{X}^T/|\mathbf{X}| - \mathbf{W}\|_{\mathrm{Tr}} + \alpha \tag{17}$$

Proof. The loss functions $\|\mathbf{X}_t\mathbf{X}_t^T/|\mathbf{X}_t| - \mathbf{W}\|_{\mathrm{Tr}}$ have gradients $-\mathbf{U}_t\mathbf{S}_t\mathbf{U}_t^T$, where $\mathbf{U}_t$ is the matrix of eigenvectors of $\mathbf{X}_t\mathbf{X}_t^T/|\mathbf{X}_t| - \mathbf{W}$ and $\mathbf{S}_t$ is the diagonal matrix of the signs of its eigenvalues; the losses are thus $\sqrt{d}$-Lipschitz w.r.t. the Frobenius norm and 1-Lipschitz w.r.t. the trace norm. Note that these gradients can be computed in polynomial time via eigendecomposition and used in DP-FTRL with the Schatten-p norm regularizer $\frac{3}{2}\log d\|\|\cdot\|\|_p^2$ for $p = 1 + 1/\log d$, which is strongly-convex w.r.t. the trace norm $\|\cdot\|_{\mathrm{Tr}}$ [29]. Since the Gaussian width of the (symmetric) trace ball is $\mathcal{O}(\sqrt{d})$ [1] and the spectral norm is 1-Lipschitz w.r.t. the Frobenius norm, applying Theorem 6.1 yields the bound

$$\frac{3\log d\|\|\mathbf{W}\|\|_p^2}{2\eta} + \eta\left(1 + \left(\mathcal{O}(\sqrt{d}) + \sqrt{2\log\frac{T}{\beta'}}\right)\sigma\sqrt{d\lceil\log_2 T\rceil}\right)T \tag{18}$$

For any optimal $\mathbf{W}$ we have

$$\begin{aligned}\|\|\mathbf{W}\|\|_p \leqslant \|\mathbf{W}\|_{\mathrm{Tr}} &\leqslant \frac{1}{T}\sum_{t=1}^{T}\|\mathbf{W} - \mathbf{X}_t\mathbf{X}_t^T/|\mathbf{X}_t|\|_{\mathrm{Tr}} + \|\mathbf{X}_t\mathbf{X}_t^T/|\mathbf{X}_t|\|_{\mathrm{Tr}} \\ &\leqslant \frac{2}{T}\sum_{t=1}^{T}\mathrm{Tr}(\mathbf{X}_t\mathbf{X}_t^T)/|\mathbf{X}_t| \leqslant 2\end{aligned} \tag{19}$$

so the regret follows by substituting η. The sample complexity result follows from online-to-batch conversion (c.f. Appendix D.1). □

Thus prediction matrices for covariance estimation are efficiently and privately learnable, in both the online and distributional settings. Moreover, for both our extension to non-Euclidean DP-FTRL is critical for obtaining a weaker dependence on the dimension. One limitation of the analysis is that, unlike for quantiles, we did not conduct a refined analysis by studying how swapping single columns of $\mathbf{X}_t$ rather than the entire dataset affects the gradient of $U_{\mathbf{X}_t}$. It is not immediately clear that an improvement is possible, with the difficulty being the gradient's dependence on the signs of the eigenvalues.

6.4 Learning the initialization and number of iterations for data release

Finally, we learn to initialize `MWEM`-based data release. Here we are faced with optimizing

$$U_{\mathbf{x}_t}(\mathbf{w}, m) = \frac{8n_t}{m}D_{KL}\left(\frac{\mathbf{x}_t}{n_t}\middle\|\mathbf{w}\right) + \frac{16m^2}{\varepsilon^2 n + t}\left(3\log\frac{2m}{\beta} + 2\log^2|Q|\right)^2 \tag{20}$$

Notably, unlike the past learning settings, this function is parameterized by both a prediction $\mathbf{w}$ and the number of steps m, which we will also set online. The reason for this is that the optimal step-size depends on the similarity between instances: if for the optimal $\mathbf{w}$ the measure $D_{KL}(\mathbf{x}_t/n_t||\mathbf{w})$ is small for most datasets $\mathbf{x}_t$ then it is better to set a small m above, whereas if it is usually large it should be counter-acted with a larger m. Our goal will thus be to set $\mathbf{w}_t$ and m_t together in an online fashion so as to *simultaneously* compete with the optimal λ-robust $\mathbf{w} \in \triangle_d$ for some $\lambda > 0$ and the optimal number of steps $m > 0$. To do so we will run DP-FTRL with the entropic regularizer, i.e. private exponentiated gradient (EG), to set both the initialization from the simplex $\triangle_d$ and to set the number of iterations m_t at step t by sampling from a categorical distribution. This has the following regret guarantee (c.f. Appendix D.4):

Theorem 6.4. *Let $\mathbf{x}_1, \dots, \mathbf{x}_T \in \mathbb{Z}^d_{\geqslant 0}$ be a sequence of datasets with $n_t = \|\mathbf{x}_t\|_1$ entries each, let $N = \max_t n_t$, and consider any $\gamma \in (0, 1]$ and $\lambda \in [0, 1]$. Then there exists $M \in \mathbb{Z}_{>0}$ and $\eta_\theta, \sigma_\theta, \eta_{\mathbf{w}}, \sigma_{\mathbf{w}} > 0$ s.t. running DP-FTRL with regularizer $\phi(\mathbf{w}) = \langle \mathbf{w}, \log \mathbf{w} \rangle$, step-size $\eta_{\mathbf{w}}$, and noise $\sigma_{\mathbf{w}}$ on the losses $n_t D_{KL}(\frac{\mathbf{x}_t}{n_t}||\mathbf{w})$ over the domain $\mathbf{w}_{[i]} \geqslant \gamma/d$ to set $\mathbf{w}_t$ and simultaneously running DP-FTRL with regularizer $\phi(\theta) = \langle \theta, \log \theta \rangle$, step-size η_θ, and noise σ_θ on the losses $\mathbb{E}_{m\sim\theta} U_{\mathbf{x}_t}(\mathbf{w}_t, m)$ over the domain $\triangle_M$ and setting m_t using the categorical distribution defined by θ_t over the M-simplex such that the entire scheme is (ε', δ')-DP and w.h.p. has regret*

$$\tilde{\mathcal{O}}\left(\left(\frac{N^{\frac{4}{3}}}{\min\{1, \varepsilon^2\}} + \frac{N^{\frac{2}{3}}/\sqrt{\varepsilon'}}{\min\{1, \varepsilon\}} + \frac{dN}{\gamma} + \frac{d}{\gamma}\sqrt{\frac{N}{\varepsilon'}}\right)\sqrt{T} + \max\{\gamma - \lambda, 0\} NT\right) \tag{21}$$

w.r.t. any $m > 0$ and $\mathbf{w} \in \triangle_d$ satisfying $\mathbf{w}_{[i]} \geqslant \frac{\lambda}{d}$. For $\lambda > 0$ setting $\gamma = \lambda$ yields regret $\tilde{\mathcal{O}}\left(\frac{d}{\lambda \min\{1,\varepsilon^2\}} N^{\frac{4}{3}} \sqrt{T/\varepsilon'}\right)$; for $\lambda = 0$ and $T \geqslant d^2$, setting $\gamma = \frac{\sqrt{d}}{\sqrt[4]{T}}$ has regret $\tilde{\mathcal{O}}\left(\frac{N^{\frac{4}{3}}}{\min\{1,\varepsilon^2\}} \sqrt{d/\varepsilon'} T^{\frac{3}{4}}\right)$.

As in quantile learning, we suffer a strong dependence on the dimension here, and the rate is worse if we try to compete the non-robust initializations ($\lambda = 0$). It thus remains an open question whether either a better learning result or upper bound is possible. Nevertheless, to interpret this guarantee, note that for $H_\lambda = \min_{\mathbf{w}_{[i]} \geqslant \lambda/d} \left(\sum_{t=1}^T n_t D_{KL}(\mathbf{x}_t/n_t||\mathbf{w})\right) / \left(\sum_{t=1}^T n_t\right)$ and if $n_t = n \;\forall\; t$ then we have that the optimum-in-hindsight for the average upper bound is

$$\begin{aligned} \min_{m>0, \mathbf{w}_{[i]} \geqslant \lambda/d} \frac{1}{T} \sum_{t=1}^T U_{\mathbf{x}_t}(\mathbf{w}, m) &= \tilde{\mathcal{O}}\left(\frac{\log^{\frac{4}{3}} |Q|}{\varepsilon^{\frac{2}{3}}} \frac{\left(\frac{1}{T}\sum_{t=1}^T n_t\right)^{\frac{2}{3}}}{\left(T/\sum_{t=1}^T \frac{1}{n_t}\right)^{\frac{1}{3}}} H_\lambda^{\frac{2}{3}}\right) \\ &= \tilde{\mathcal{O}}\left(\sqrt[3]{\frac{n \log^4 |Q|}{\varepsilon^2}} H_\lambda^2\right) \end{aligned} \tag{22}$$

Since H_λ approximates the entropy of the aggregate distribution across instances $\mathbf{x}_1, \dots, \mathbf{x}_T$—indeed for $\lambda = 0$ it is exactly the entropy of the average distribution $\left(\sum_{t=1}^T \mathbf{x}_t\right) / \sum_{t=1}^T n_t$—the regret guarantee shows that we will do well asymptotically if the entropy is small. Note that being able to choose m in addition to $\mathbf{w}$ is crucial to adapting to this entropy, and is closely related to the problem of choosing the step-size in meta-learning, where similar aggregate measures appear as forms of task-similarity [8, 43].

7 Applications

Having derived prediction-dependent performance bounds for three DP tasks and analyzed their robustness and learnability, we now investigate how these algorithms might be deployed in practice. We focus on the problem of multiple quantile release and consider the two motivating settings from the introduction: public-private transfer and sequential release. While we make direct use of the robust mixing scheme devised in Section 5, our learnability analysis in Section 6 yielded unwieldy discretization-based algorithms due to the focus on approximating very general priors. This generality seems unnecessary, as we might reasonably expect simple, unimodal distributions to be good priors for quantiles.

We thus consider instead the problem of optimizing the performance bounds $U_{\mathbf{x}} = -\log \Psi_{\mathbf{x}}$ for multiple quantile release across classes of **location-scale** priors, which for some measure $f : \mathbb{R} \mapsto \mathbb{R}_{\geqslant 0}$ have the form $\mu_{\nu,\sigma}(x) = \frac{1}{\sigma} f\left(\frac{x-\nu}{\sigma}\right)$ for $\nu \in \mathbb{R}$ and $\sigma > 0$. Such families allows us to model both the location of a quantile using $\nu = \langle \mathbf{w}, \mathbf{f} \rangle$—where $\mathbf{w} \in \mathbb{R}^d$ is a linear model from public features $\mathbf{f} \in \mathbb{R}^d$ about the dataset—and our uncertainty about it using σ, all while staying in reasonable dimensions. Note that in this section we target only the ε-independent bound $U_{\mathbf{x}}$, as $U_{\mathbf{x}}^{(\varepsilon)}$ does not yield a convex objective; furthermore, while we mainly discuss the single-quantile bound $U_{\mathbf{x}}^{(q)}$ for simplicity, the general results (c.f. Section E) extend naturally to the case of $m > 1$ because it is the log-sum-exp of the former.

7.1 Convexity vs. robustness of location-scale models

We must first determine which location-scale family to use, as this include Gaussians with mean ν and variance σ^2, Laplace with mean ν and scale σ, Cauchy with location ν and scale σ, and more. To make this decision, we consider two desiderata: (1) the prior should be robust in the way the Cauchy is robust, i.e. being wrong about the data location should not harm us too much, and (2) it should be easy to learn the parameters ν and σ, e.g. by optimizing $U_{\mathbf{x}}^{(q)}(\mu_{\nu,\sigma})$.

While not necessary, one way of ensuring (2) is convexity of $U_{\mathbf{x}}^{(q)}$, which we focus on as it enables efficient algorithms. Here we make use of a connection between these upper bounds and the likelihood of **censored regression** [62], which for noise $\xi_i \in \mathbb{R}$ models a relationship between features $\mathbf{f}_i \in \mathbb{R}^d$ and a variable $y_i = \langle \mathbf{w}, \mathbf{f}_i \rangle + \xi_i$ when information about y_i is only provided in terms of an interval $[a_i, b_i)$ containing it (e.g. an individual's income bracket, not their exact income). If ξ_i is from a location-scale distribution with $\nu = 0$ the log-likelihood given datapoints $(a_i, b_i, \mathbf{f}_i)$ is

$$\mathcal{L}_{\{a_i,b_i,\mathbf{f}_i\}_{i=1}^n}(\mathbf{w}, \sigma) = \sum_{i=1}^{n} \log \int_{a_i}^{b_i} \frac{1}{\sigma} f\left(\frac{y - \langle \mathbf{w}, \mathbf{f}_i \rangle}{\sigma}\right) dy \tag{23}$$

Observe that for $a = \mathbf{x}_{[\lfloor qn \rfloor]}$ and $b = \mathbf{x}_{[\lfloor qn \rfloor]+1}$ we have

$$U_{\mathbf{x}}^{(q)}(\mu_{\langle \mathbf{w},\mathbf{f} \rangle,\sigma}) = -\log \mu_{\langle \mathbf{w},\mathbf{f} \rangle,\sigma}((a, b]) = -\log \int_a^b \frac{1}{\sigma} f\left(\frac{o - \langle \mathbf{w}, \mathbf{f} \rangle}{\sigma}\right) do \tag{24}$$

which is the negative of $\mathcal{L}_{a,b,\mathbf{f}}(\mathbf{w}, \sigma)$. We thus adopt the reparameterization of Burridge [15], who showed that (23) is concave w.r.t. $(\mathbf{v}, \phi) = (\frac{\mathbf{w}}{\sigma}, \frac{1}{\sigma})$ whenever f is **log-concave**, a property satisfied by the Gaussian and Laplace families but not the Cauchy. Therefore, for such f we have that $\ell_{\mathbf{x}}^{(q)}(\langle \mathbf{v}, \mathbf{f} \rangle, \phi) = U_{\mathbf{x}}^{(q)}(\mu_{\frac{\langle \mathbf{v},\mathbf{f} \rangle}{\phi}, \frac{1}{\phi}})$ is convex w.r.t. $(\mathbf{v}, \phi)$.

Unfortunately, we show that no log-concave f is robust, in the sense that for any $R > 0$ there exists a dataset of points in the interval $(\theta \pm R)^n$ s.t. $U_{\mathbf{x}}^{(q)}(\mu_{\theta,1}) = \Omega(R)$ (rather than $\mathcal{O}(\log(1 + R^2))$ as shown for the Cauchy family in Corollary 4.1). On the other hand, log-concave location-scale families are the only ones for which $U_{\mathbf{x}}^{(q)}$ is convex, both for the original parameterization and that of Burridge [15]. We record these facts in the following theorem:

Theorem 7.1 (c.f. Thm. E.1)**.** *Let $\mu_{\nu,\sigma}$ be a location-scale family associated with a continuous measure $f : \mathbb{R} \mapsto \mathbb{R}_{\geqslant 0}$.*

1. *If f is log-concave then $\exists\ a, b > 0$ s.t. for any $R > 0$, $\psi \in (0, \frac{R}{2n}]$, $q \geqslant \frac{1}{n}$, and $\theta \in \mathbb{R}$ there exists $\mathbf{x} \in (\theta \pm R)^n$ with $\min_i \mathbf{x}_{[i+1]} - \mathbf{x}_{[i]} = \psi$ s.t. $U_{\mathbf{x}}^{(q)}(\mu_{\theta,1}) = aR + \log \frac{b}{\psi}$.*

2. *If f is not log-concave then there exists $\mathbf{x} \in \mathbb{R}^n$ with $\min_i \mathbf{x}_{[i+1]} - \mathbf{x}_{[i]} > 0$ s.t. $U_{\mathbf{x}}^{(q)}(\mu_{\theta,1})$ is non-convex in θ.*

Note the latter dataset is not degenerate: for f strictly log-convex over $[a, b]$, any $\mathbf{x}$ whose optimal interval has length $< \frac{b-a}{2}$ has non-convex $U_{\mathbf{x}}^{(q)}(\mu_{\theta,1}) = -\log \Psi_{\mathbf{x}}^{(q)}(\mu_{\theta,1})$. We must thus choose between having a robust location-scale family like the Cauchy or an easy-to-optimize log-concave one. As we can ensure robustness of the learned prior *post-hoc* using the approach of Section 5, we choose the latter. Specifically, we use the Laplace prior, as it is in some sense the most robust log-concave distribution (it has loss $\Theta(R)$ if $\mathbf{x} \in (\theta \pm R)^n$, whereas e.g. the Gaussian has loss $\Theta(R^2)$) and because it yields a numerically stable closed-form expression (99) for $\ell_{\mathbf{x}}^{(q)}(\theta, \phi)$ (unlike e.g. the Gaussian).

7.2 Augmenting quantile release using public data

We turn to two applications that depend on optimizing upper bounds $\ell_{\mathbf{x}}^{(q)}(\theta, \phi)$ on the performance of quantile release using the Laplace prior with scale $\frac{1}{\phi}$ and location $\frac{\theta}{\phi}$. While our final objective is small Gap_q, we will mainly discuss optimizing $\ell_{\mathbf{x}}^{(q)} = U_{\mathbf{x}}^{(q)}$, or its expectation if $\mathbf{x}$ is drawn from some distribution. In the former case this directly bounds (w.h.p.) the cost of multiple quantile release, while a bound on $\mathbb{E}_{\mathbf{x}} U_{\mathbf{x}}$ can bound $\mathbb{E}\,\mathrm{Gap}_{\max}$ by setting β appropriately. For example, using $\beta = \frac{2\pi^2}{\varepsilon n} \exp(2\sqrt{\log(2)\log(m+1)})$ in Theorem 4.3 implies $\mathrm{Gap}_{\max}$ has expectation at most

$$\mathcal{O}\left(\exp\left(2\sqrt{\log(2)\log(m+1)}\right)\frac{\log(\varepsilon m n) + \mathbb{E}_{\mathbf{x}} U_{\mathbf{x}}}{\varepsilon}\right) \tag{25}$$

Our first application is the frequently studied setting where we have a large public dataset $\mathbf{x}' \in \mathbb{R}^N$ and want to use it to improve the release of statistics of a smaller private dataset $\mathbf{x} \in \mathbb{R}^n$. To apply our quantile release method, we must use $\mathbf{x}'$ to construct a prior μ' for each that makes $U_{\mathbf{x}}^{(q)}(\mu')$ small. If the entries of $\mathbf{x}$ and $\mathbf{x}'$ are sampled i.i.d. from similar distributions $\mathcal{D}$ and $\mathcal{D}'$, respectively, the convexity of $U_{\mathbf{x}}^{(q)}$ suggests using stochastic optimization find a prior μ that approximately minimizes the expectation $\mathbb{E}_{\mathbf{z} \sim \mathcal{D}'^n} U_{\mathbf{z}}(\mu)$ using samples of size n drawn from $\mathbf{x}'$. We provide a guarantee for a variant of this generic approach that runs online gradient descent (OGD) with separate learning rates for θ and ϕ on samples drawn without replacement from $\mathbf{x}'$:

Theorem 7.2 (c.f. Thm. E.2)**.** *If $\mathcal{D}$ and $\mathcal{D}'$ have bounded densities with bounded support then there exists an algorithm optimizing $U_{\mathbf{x}'_t}$ over T datasets $\mathbf{x}'_t$ of size n drawn from $\mathbf{x}' \in \mathbb{R}^N$ without replacement that runs in time $\mathcal{O}(mN)$ and returns a set μ' of m Laplace priors s.t. w.h.p.*

$$\mathbb{E}_{\mathbf{x} \sim \mathcal{D}^n} U_{\mathbf{x}}(\mu') \leqslant \min_{\mu \in \mathrm{Lap}^m_{B,\sigma_{\min},\sigma_{\max}}} \mathbb{E}_{\mathbf{x} \sim \mathcal{D}^n} U_{\mathbf{x}}(\mu) + \tilde{\mathcal{O}}\left(\mathrm{TV}_q(\mathcal{D}, \mathcal{D}') + \sqrt{\frac{mn}{N}}\right) \tag{26}$$

where $\mathrm{Lap}_{B,\sigma_{\min},\sigma_{\max}}$ is the set of Laplace priors with locations in $[\pm B]$ and scales in $[\sigma_{\min}, \sigma_{\max}]$ and $\mathrm{TV}_q(\mathcal{D}, \mathcal{D}')$ is the total variation distance between the joint distributions of the order statistics $\left\{(\mathbf{x}_{[\lfloor q_i n \rfloor]}, \mathbf{x}_{[\lfloor q_i n \rfloor + 1]})\right\}_{i=1}^m$ for $\mathbf{x} \sim \mathcal{D}^n$ and $\left\{(\mathbf{x}'_{[\lfloor q_i n \rfloor]}, \mathbf{x}'_{[\lfloor q_i n \rfloor + 1]})\right\}_{i=1}^m$ for $\mathbf{x}' \sim \mathcal{D}'^n$.

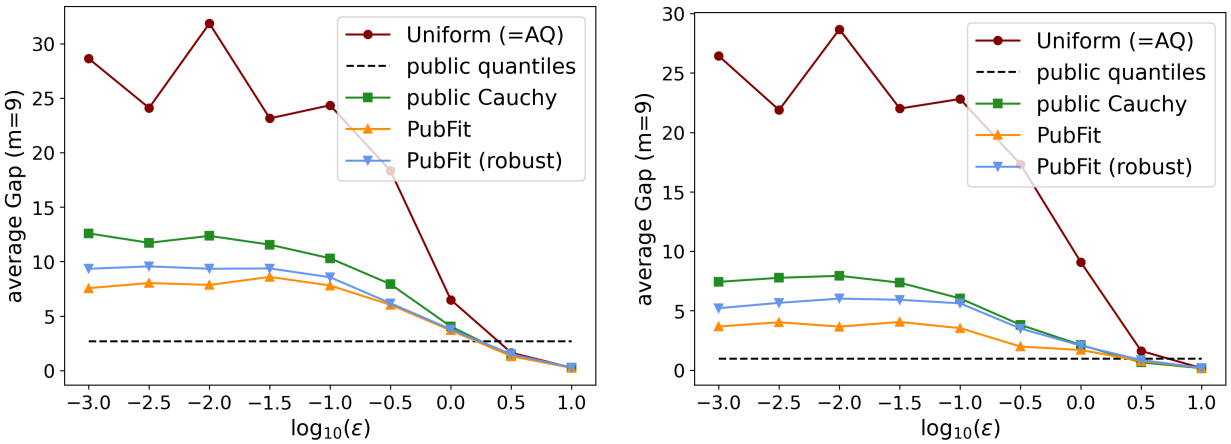

Figure 2: Public-private release of nine quantiles using one hundred samples from the Adult age (left) and hours (right) datasets. The public data is the Adult training set while private data is test.

For $N \gg mn$, the suboptimality of μ' for the upper bound $U_{\mathbf{x}}$ will depend on the statistical distance between the quantile intervals of $\mathcal{D}$ and $\mathcal{D}'$: even if $\mathcal{D}$ and $\mathcal{D}'$ are dissimilar, similar order statistic distributions will ensure good performance. Note, as in Section 5, we can hedge against large $\mathrm{TV}_q(\mathcal{D}, \mathcal{D}')$ by mixing the output μ' with a robust prior.

We evaluate this approach, which we call *Public Fit* or `PubFit`, on Adult [47] and Goodreads [70], both used previously for DP quantiles [33, 42]. Because our guarantees improve with different step-sizes for θ and ϕ, we use COCOB [60]—an OGD variant that provably sets per-coordinate step-sizes without the need for tuning—as `PubFit`'s stochastic solver. We also test a robust version where its output is mixed with a half-Cauchy distribution, and three baselines: the Uniform prior, just using the quantiles of the public data (`public quantiles`), and using the public quantiles to set the location parameters of m Cauchy priors (`public Cauchy`).

Adult tests the $\mathcal{D} = \mathcal{D}'$ case, with its "train" set the public dataset and a hundred samples from "test" as private. Figure 2 shows that `public quantiles` does best at small ε, as is expected with no distribution shift, but it cannot adapt to the empirical distribution of a small number of private points, and so is worse at $\varepsilon > 1$. Among the rest, `PubFit` is most similar to `public-quantiles` at small ε but still does well at large ε.

We use the Goodreads "History" and "Poetry" genres to evaluate under distribution shift by fitting on all but a small fraction of data from the former and releasing quantiles of samples from varying mixtures of the two datasets. As expected, the performance of `public quantiles` deteriorates with more samples from "Poetry." For book ratings, `PubFit` is best among the remaining methods, but without much change with distribution shift, possibly due to an incomplete fit of the data. For page counts, the `PubFit` methods and `public Cauchy` both do as well as `public-quantiles` when most data is from "History," but `PubFit (robust)` deteriorates least—and much less than regular `PubFit`—as the distribution shifts. This highlights the importance of robustness analysis, and suggest the former as a good method to start with, as it takes advantage of similar public and private distributions (Fig. 2) while never doing much worse than the default method (Uniform) when the the distributions are dissimilar (Fig. 3).

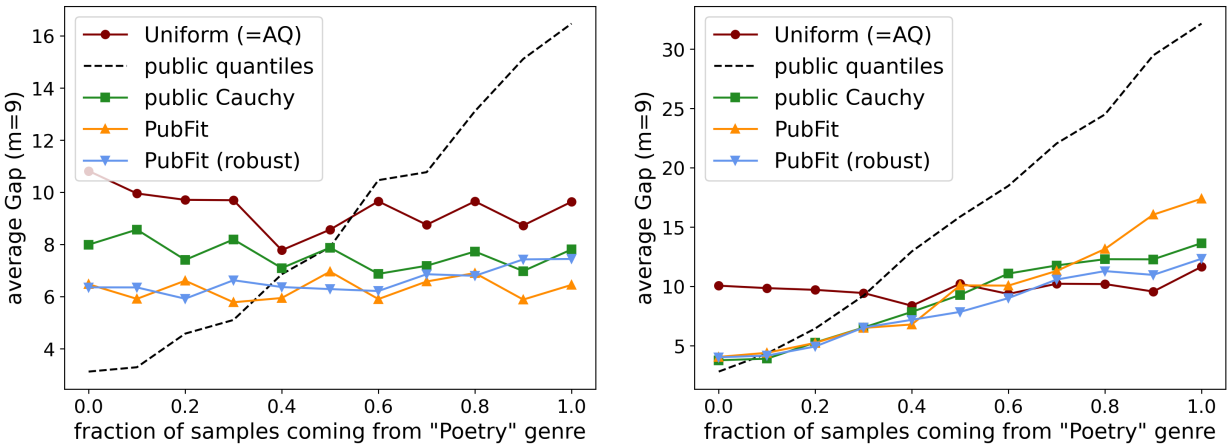

Figure 3: Public-private release of nine quantiles on one hundred samples from the Goodreads rating (left) and page count (right) datasets, with $\varepsilon = 1$. The public data is the "History" genre while private data is sampled from a mixture of it and "Poetry."

7.3 Sequentially setting priors using past sensitive data

Our second application is sequential release, which we do not believe has been studied, but arises naturally if e.g. we wish to release daily statistics from a continuous stream of data. Here we have a sequence of datasets $\mathbf{x}_1, \dots, \mathbf{x}_T$, each with associated *public* features $\mathbf{f}_1, \dots, \mathbf{f}_T \in \mathbb{R}^d$ (e.g. day of the week), and we wish to minimize the average maximum gap $\frac{1}{T}\sum_{t=1}^{T} \max_i \mathrm{Gap}_{q_i}(\mathbf{x}_t, o_{t,i})$, whose expectation can be bounded (25) in terms of $\frac{1}{T}\sum_{t=1}^{T} U_{\mathbf{x}_t}$. For simplicity, we assume individuals do not occur in multiple datasets $\mathbf{x}_t$, e.g. we are releasing the median age of new users of a service. Note the natural way to avoid this assumption is to compose the privacy budgets at each time; empirically our methods are especially useful in the low privacy regime this entails.

Our analysis suggests that we can apply online learning here, e.g. doing the following at each t starting with a prior μ_1:

1. release o_t using the prior μ_t and suffer $\mathrm{Gap}_q(\mathbf{x}_t, o_t)$
2. update to μ_{t+1} using online learning on the loss $\ell^{(q)}_{\mathbf{x}_t}$

Because $\ell^{(q)}_{\mathbf{x}_t}(\theta, \phi) = U^{(q)}_{\mathbf{x}_t}(\mu_{\frac{\theta}{\phi}, \frac{1}{\phi}})$ is convex for Laplace priors, online convex optimization (OCO) [68] lets us compete with the best prior in hindsight according to the upper bounds $U^{(q)}_{\mathbf{x}_t}(\mu_t)$, or with the best linear map $\mathbf{w}$ to locations $\langle \mathbf{w}, \mathbf{f}_t \rangle$. We can again hedge against poor predictions by mixing with a constant robust distribution.

However, we face the difficulty that online learning on losses $\ell^{(q)}_{\mathbf{x}_t}$ leaks information about $\mathbf{x}_t$. There are two natural solutions. One is to use part of the budget $\varepsilon' < \varepsilon$ on a DP online learner [39, 69] and hope that the reduction in budget allocated to quantile release is made up for by the improved priors. Alternatively, we can replace ℓ with a *proxy* loss $\hat{\ell}$ that does not depend on the data and optimize it using regular OCO. The first can be done with provable guarantees by applying DP-FTRL [41], again using two different step-sizes:

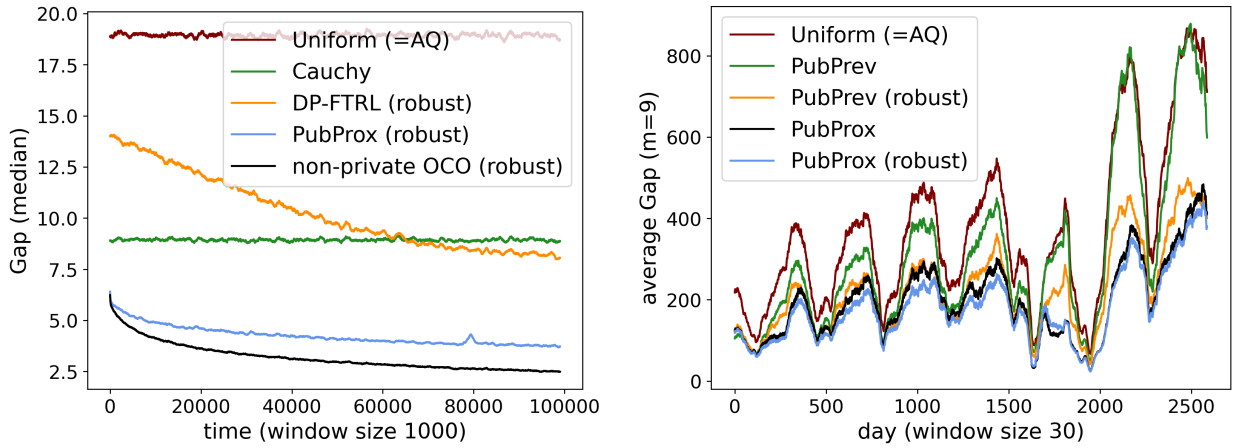

Figure 4: Comparison of sequential release over time on Synthetic (left, $\log_{10}\varepsilon = -1/2$) and CitiBike (right, $\log_{10}\varepsilon = -2$) tasks.

Theorem 7.3 (c.f. Thm. E.3)**.** *Consider a sequence of datasets $\mathbf{x}_t \in [\pm B]^{n_t}$ with bounded features $\mathbf{f}_t$ and suppose we set Laplace priors $\mu_{t,i} = \mu_{\frac{\langle \mathbf{v}_{t,i}, \mathbf{f}_t\rangle}{\phi_{t,i}}, \frac{1}{\phi_{t,i}}}$ via two DP-FTRL algorithms applied separately to the variables $\mathbf{v}_i$ and ϕ_i of the losses $\ell_{\mathbf{x}_t}(\langle \mathbf{v}_i, \mathbf{f}_t\rangle, \phi_i)$ with budgets $\frac{\varepsilon'}{2}$, with respective step-sizes $\tilde{\Theta}\left(\sqrt{\frac{\varepsilon'}{\sigma_{\min}^2 T}\sqrt{\frac{m}{d}}}\right)$ and $\tilde{\Theta}\left(\sqrt{\frac{\varepsilon'\sqrt{m}}{\sigma_{\min}^2\sigma_{\max}^2 T}}\right)$. This is (ε', δ')-DP and w.h.p. has regret*

$$\frac{1}{T}\sum_{t=1}^{T} U_{\mathbf{x}_t}(\mu_t) - \min_{\substack{\mathbf{w}_i \in [\pm B]^d \\ \sigma_i \in [\sigma_{\min}, \sigma_{\max}]}} \frac{1}{T}\sum_{t=1}^{T} U_{\mathbf{x}_t}(\mu_{\langle \mathbf{w}_i, \mathbf{f}_t\rangle, \sigma_i}) = \tilde{\mathcal{O}}\left(\frac{d^{\frac{3}{4}} + \sigma_{\max}}{\sigma_{\min}}\sqrt{\frac{m}{\varepsilon' T}\sqrt{m \log\frac{2}{\delta'}}}\right) \tag{27}$$

Thus we can do as well as any sequence of Laplace priors μ_t with locations determined by a fixed linear map from $\mathbf{f}_t$, up to a term that decreases at rate $\tilde{\mathcal{O}}(\frac{1}{\sqrt{T}})$. Furthermore, running quantile release with budget $\varepsilon - \varepsilon'$ ensures (ε, δ')-DP for each dataset $\mathbf{x}_t$. Note that using different step-sizes allows us to separate the difficulty of learning a d-dimensional linear map from the difficulty of learning a scale parameter of magnitude at most $\sigma_{\max}$.

Unfortunately, DP-FTRL is too noisy to learn competitive priors, except with a lot of stationary data (c.f. Fig. 4 (left)). One issue is that its DP guarantee is too strong, as it it allows swapping out the entire dataset $\mathbf{x}_t$ rather than a single entry. It is unclear if a better sensitivity is possible for $U_{\mathbf{x}_t}$, as changing an entry can flip the sign of the gradient while preserving magnitude. We show (c.f. Lem. 6.1) that it is possible for the ε-dependent bound $U_{\mathbf{x}_t}^{(\varepsilon)}$ over piecewise-constant priors—remarkably sensitivity *decreases* with ε—but that upper bound is non-convex for location-scale families, which are preferable for model learning.

Our second solution involves recognizing that $U_{\mathbf{x}_t}^{(q)}$ depends only on the optimal interval $[\mathbf{x}_{t[\lfloor qn \rfloor]}, \mathbf{x}_{t[\lfloor qn \rfloor + 1]})$, whose location and size we have (public) estimates for: the former via the quantile estimate o_t and the size is lower-bounded by the underlying data discretization, which we have access to in-practice (e.g. age is reported in years, bicycle trip length in seconds). We use this information to construct *proxy* losses $\hat{\ell}_{o_t}^{(q)}(\langle \mathbf{v}, \mathbf{f}_t\rangle, \phi)$, which do not depend on $\mathbf{x}_t$ and so be learned with (standard) OCO. As our DP-FTRL analysis again showed the importance of different step-sizes, we again use the COCOB optimizer here.

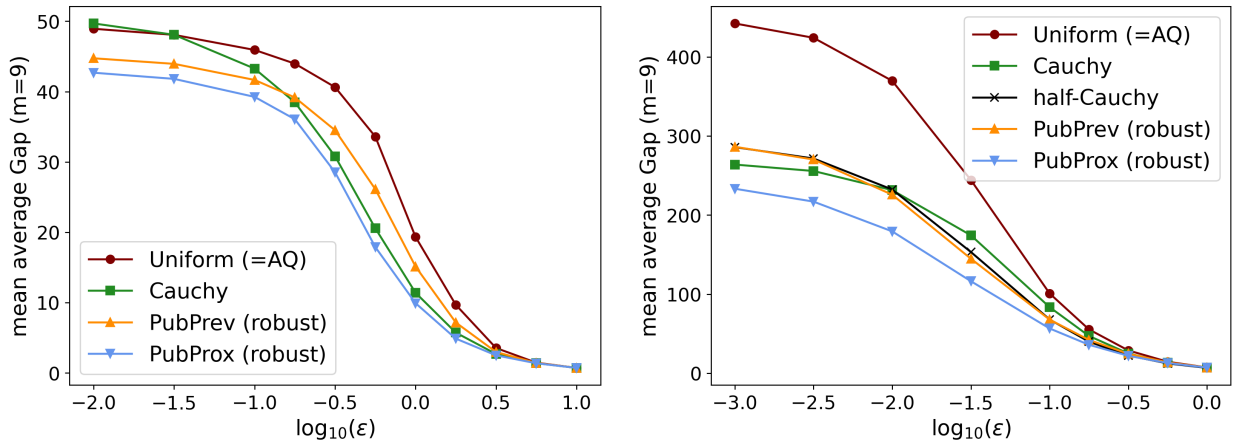

Figure 5: Time-averaged performance of the sequential release of nine quantiles on the Synthetic (left) and CitiBike (right) tasks.

We evaluate sequential release on three online tasks, each consisting of a sequence of datasets needing quantiles:

1. Synthetic: each dataset is generated such that the quantiles are fixed linear functions of a random Gaussian feature vector, plus noise.

2. CitiBike: the data are the lengths of a day's bicycle trips, with the date and NYC weather information features.

3. BBC: the data are the Flesch readability scores of the comments on a headline posted to Reddit's `worldnews` forum, with date and headline text information features.

In addition to the proxy approach, which we call `PubProx`, we evaluate static priors—the uniform, Cauchy, and half-Cauchy (if nonnegative)—and an approach we call `PubPrev`, which uses a Laplace prior centered around the previous step's released quantile. Note that using the Uniform is equivalent to `ApproximateQuantiles` (AQ). For both `PubProx` and `PubPrev` we ensure robustness by mixing with a Cauchy (or half-Cauchy, if nonnegative) distribution with coefficient 0.1; this nearly always improves performance for these methods, likely by ensuring their training data is not too noisy. To see its effectiveness, note how in Figure 4 (right) both augmented methods are almost always better when made robust, especially `PubPrev`; in fact, non-robust `PubPrev` is unable to do better than Uniform after around day 1600, when the start of the COVID-19 pandemic significantly affects bicycle trips.

Our main comparisons is time-aggregated performance as a function of ε (c.f. Figs. 5 and 6). All except perhaps Synthetic demonstrate significant improvement by our methods over the Uniform (AQ) baseline, especially at small ε. On Synthetic and CitiBike, both tasks with features for which a linear model should provide some benefit, we see in Figure 5 that `PubProx` is indeed the best across all except perhaps the lowest privacy settings. For BBC, Figure 6 reveals a large difference between mean and median performance (note the difference in y-axis scales), with `PubProx` doing best for the typical headline but the Cauchy doing better on-average due to better performance on headlines with many comments. The result suggests that in highly noisy settings, the learning-based scheme should help, but it might not overcome the robustness of a static Cauchy prior in-expectation.

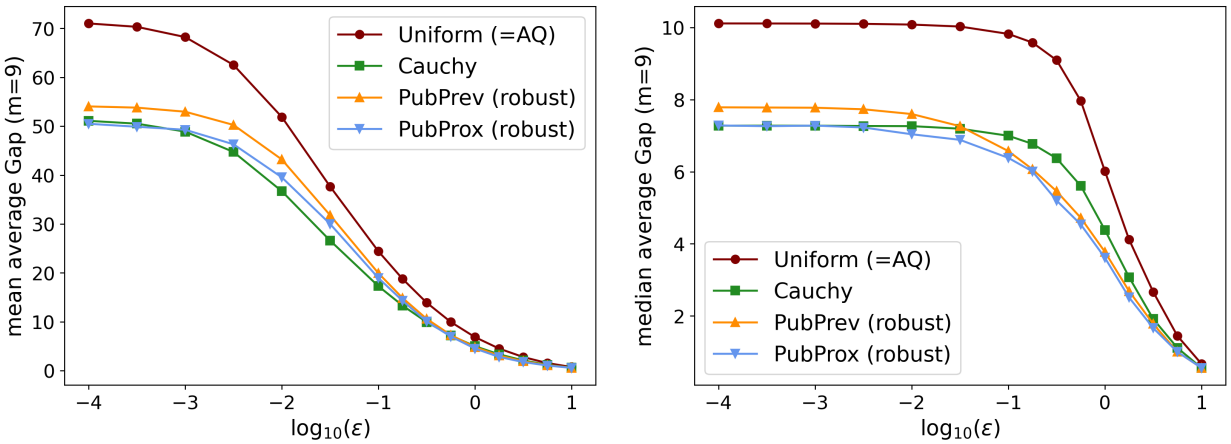

Figure 6: Time-aggregated mean (left) and median (right) performance of sequential release of nine quantiles on the BBC task.

Overall, the results demonstrate the strength of the Cauchy and half-Cauchy priors, both as unbounded substitutes for the Uniform and as a means of robustifying learning-augmented algorithms. They also demonstrate the utility of our upper bound in providing an objective for learning, albeit using proxy data rather the DP online learning: `PubProx` usually does better than `PubPrev` despite using the same information. Overall, `PubProx` performs the best at most privacy levels in all evaluation settings (Synthetic, CitiBike, and BBC) except when the mean is used as the metric for BBC (Fig. 6, left), where it does almost as well as the best. Narrowing the performance gap with non-private OCO (c.f. Fig. 4 (left), where we run COCOB directly on $\ell_{\mathbf{x}_t}^{(q)}$)—remains an important research direction.

8 Conclusion

Our work introduces the framework of private algorithms with private predictions, an application of the algorithms with predictions setup to DP methods. We provide extensive evidence of its utility as a way of integrating external information into privacy-preserving algorithms. In particular, we show how it informs the design of methods that are robust to poor predictions and of learning algorithms for obtaining good predictions from data. Finally, we demonstrate how combining optimization with learning-augmented private algorithms can be used to significantly improve the quality of released statistics in-practice. As a result, we believe these methods hold great promise for reducing error while preserving privacy on practical, real-world problems.

Beyond the current work, we believe this way of studying DP methods is highly applicable and will see a great deal of future work in finding new applications for incorporating predictions or improving the approaches described here. By conducting a fine-grained analysis of DP algorithms beyond their default parameterizations, it also is highly likely to lead to significant contributions even in the prediction-free setting, as exemplified by our guarantees for unbounded-domain quantile release and improved bounds for trace-sensitive covariance estimation. Some specific areas to explore include other forms of iterative data analysis beyond `MWEM` [34, 35] and other important dataset statistics [12]. Another important direction is that of learning: can DP online convex optimization be made useful for the purpose of learning predictions, or can guarantees be shown for our alternative of using public proxies?

References

[1] Rafał Latała, Ramon van Handel, and Pierre Youssef. The dimension-free structure of nonhomogeneous random matrices. *Inventiones Mathematicae*, 214:1031–1080, 2018.

[2] Naman Agarwal and Karan Singh. The price of differential privacy for online learning. In *Proceedings of the 34th International Conference on Machine Learning*, 2017.

[3] Ehsan Amid, Arun Ganesh, Rajiv Mathews, Swaroop Ramaswamy, Shuang Song, Thomas Steinke, Vinith M. Suriyakumar, Om Thakkar, and Abhradeep Thakurta. Public data-assisted mirror descent for private model training. In *Proceedings of the 39th International Conference on Machine Learning*, 2022.

[4] Kareem Amin, Travis Dick, Alex Kulesza, Andrés Muñoz Medina, and Sergei Vassilvitskii. Differentially private covariance estimation. In *Advances in Neural Information Processing Systems*, 2019.

[5] Keerti Anand, Rong Ge, and Debmalya Panigrahi. Customizing ML predictions for online algorithms. In *Proceedings of the 37th International Conference on Machine Learning*, 2020.

[6] Galen Andrew, Om Thakkar, H. Brendan McMahan, and Swaroop Ramaswamy. Differentially private learning with adaptive clipping. In *Advances in Neural Information Processing Systems*, 2021.

[7] Maria-Florina Balcan, Mikhail Khodak, Dravyansh Sharma, and Ameet Talwalkar. Learning-to-learn non-convex piecewise-Lipschitz functions. In *Advances in Neural Information Processing Systems*, 2021.

[8] Maria-Florina Balcan, Keegan Harris, Mikhail Khodak, and Zhiwei Steven Wu. Meta-learning adversarial bandits. arXiv, 2022.

[9] Etienne Bamas, Andreas Maggiori, and Ola Svensson. The primal-dual method for learning augmented algorithms. In *Advances in Neural Information Processing Systems*, 2020.

[10] Raef Bassily, Mehryar Mohri, and Ananda Theertha Suresh. Private domain adaptation from a public source. arXiv, 2022.

[11] Alex Bie, Gautam Kamath, and Vikrant Singhal. Private estimation with public data. In *Advances in Neural Information Processing Systems*, 2022.

[12] Sourav Biswas, Yihe Dong, Gautam Kamath, and Jonathan Ullman. CoinPress: Practical private mean and covariance estimation. In *Advances in Neural Information Processing Systems*, 2020.

[13] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Caledon Press, 2012.

[14] Mark Bun and Thomas Steinke. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Proceedings, Part I, of the 14th International Conference on Theory of Cryptography*, 2016.

[15] J. Burridge. A note on maximum likelihood estimation for regression models using grouped data. *Journal of the Royal Statistical Society. Series B (Methodological)*, 43(1):41–45, 1981.

[16] Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

[17] Nicolò Cesa-Bianchi, Alex Conconi, and Claudio Gentile. On the generalization ability of on-line learning algorithms. *IEEE Transactions on Information Theory*, 50(9):2050–2057, 2004.

[18] Kamalika Chaudhuri and Staal A. Vinterbo. A stability-based validation procedure for differentially private machine learning. In *Advances in Neural Information Processing Systems*, 2013.

[19] Justin Y. Chen, Sandeep Silwal, Ali Vakilian, and Fred Zhang. Faster fundamental graph algorithms via learned predictions. In *Proceedings of the 40th International Conference on Machine Learning*, 2022.

[20] Nicholas Christianson, Junxuan Shen, and Adam Wierman. Optimal robustness-consistency tradeoffs for learning-augmented metrical task systems. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics*, 2023.

[21] Thomas M. Cover. Universal portfolios. *Mathematical Finance*, 1:1–29, 1991.

[22] Madeleine Cule and Richard Samworth. Theoretical properties of the log-concave maximum likelihood estimator of a multidimensional density. *Electronic Journal of Statistics*, 4:254–270, 2010.

[23] Herbert Aron David and Haikady Navada Nagaraja. *Order Statistics*. John Wiley & Sons, Inc., 2003.

[24] Ilias Diakonikolas, Vasilis Kontonis, Christos Tzamos, Ali Vakilian, and Nikos Zarifis. Learning online algorithms with distributional advice. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

[25] Christos Dimitrakakis, Blaine Nelson, Zuhe Zhang, Aikaterini Mitrokotsa, and Benjamin I. P. Rubinstein. Differential privacy for bayesian inference through posterior sampling, 2017.

[26] Michael Dinitz, Sungjin Im, Thomas Lavastida, Benjamin Moseley, and Sergei Vassilvitskii. Faster matchings via learned duals. In *Advances in Neural Information Processing Systems*, 2021.

[27] Wei Dong, Yuting Liang, and Ke Yi. Differentially private covariance revisited. In *Advances in Neural Information Processing Systems*, 2022.

[28] Elbert Du, Franklyn Wang, and Michael Mitzenmacher. Putting the "learning" into learning-augmented algorithms for frequency estimation. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

[29] John Duchi, Shai Shalev-Shwartz, Yoram Singer, and Ambuj Tewari. Composite objective mirror descent. In *Proceedings of the 23rd Annual Conference on Learning Theory*, 2010.

[30] Paul Dütting, Silvio Lattanzi, Renato Paes Leme, and Sergei Vassilvitskii. Secretaries with advice. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, 2021.

[31] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.

[32] Joseph Geumlek, Shuang Song, and Kamalika Chaudhuri. Rényi differential privacy mechanisms for posterior sampling. In *Advances in Neural Information Processing Systems*, 2017.

[33] Jennifer Gillenwater, Matthew Joseph, and Alex Kulesza. Differentially private quantiles. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

[34] Anupam Gupta, Aaron Roth, and John Ullman. Iterative constructions and private data release. In *Theory of Cryptography Conference*, 2012.

[35] Moritz Hardt and Guy Rothblum. A multiplicative weights mechanism for privacy-preserving data analysis. In *51st Annual IEEE Symposium on Foundations of Computer Science*, 2010.

[36] Moritz Hardt, Katrina Ligett, and Frank McSherry. A simple and practical algorithm for differentially private data release. In *Advances in Neural Information Processing Systems*, 2012.

[37] Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69:169–192, 2007.

[38] Piotr Indyk, Frederik Mallmann-Trenn, Slobodan Mitrović, and Ronitt Rubinfeld. Online page migration with ML advice. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics*, 2022.

[39] Prateek Jain, Pravesh Kothari, and Abhradeep Thakurta. Differentially private online learning. In *Proceedings of the 25th Annual Conference on Learning Theory*, 2012.

[40] Zhihao Jiang, Debmalya Panigrahi, and Kevin Sun. Online algorithms for weighted paging with predictions. In *Proceedings of the 47th International Colloquium on Automata, Languages, and Programming*, 2020.

[41] Peter Kairouz, Brendan McMahan, Shuang Song, Om Thakkar, Abhradeep Thakurta, and Zheng Xu. Practical and private (deep) learning without sampling or shuffling. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

[42] Haim Kaplan, Shachar Schnapp, and Uri Stemmer. Differentially private approximate quantiles. In *Proceedings of the 39th International Conference on Machine Learning*, 2022.

[43] Mikhail Khodak, Maria-Florina Balcan, and Ameet Talwalkar. Adaptive gradient-based meta-learning methods. In *Advances in Neural Information Processing Systems*, 2019.

[44] Mikhail Khodak, Maria-Florina Balcan, Ameet Talwalkar, and Sergei Vassilvitskii. Learning predictions for algorithms with predictions. In *Advances in Neural Information Processing Systems*, 2022.

[45] Mikhail Khodak, Kareem Amin, Travis Dick, and Sergei Vassilvitskii. Learning-augmented private algorithms for multiple quantile release. In *Proceedings of the 40th International Conference on Machine Learning*, 2023.

[46] Jyrki Kivinen and Manfred K. Warmuth. Exponentiated gradient versus gradient descent for linear predictors. *Information and Computation*, 132:1–63, 1997.

[47] Ron Kohavi. 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*, 1996.

[48] Tim Kraska, Alex Beutel, Ed H. Chi, Jeffrey Dean, and Neoklis Polyzotis. The case for learned index structures. In *Proceedings of the 2018 International Conference on Management of Data*, 2018.

[49] Ravi Kumar, Manish Purohit, and Zoya Svitkina. Improving online algorithms via ML predictions. In *Advances in Neural Information Processing Systems*, 2018.

[50] Silvio Lattanzi, Thomas Lavastida, Benjamin Moseley, and Sergei Vassilvitskii. Online scheduling via learned weights. In *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms*, 2020.

[51] Tian Li, Manzil Zaheer, Sashank Reddi, and Virginia Smith. Private adaptive optimization with side information. In *Proceedings of the 39th International Conference on Machine Learning*, 2022.

[52] Alexander Lindermayr and Nicole Megow. Permutation predictions for non-clairvoyant scheduling. In *Proceedings of the 34th ACM Symposium on Parallelism in Algorithms and Architectures*, 2022.

[53] Terrance Liu, Giuseppe Vietri, Thomas Steinke, Jonathan Ullman, and Zhiwei Steven Wu. Leveraging public data for practical private query release. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

[54] Zhenhua Liu, Yuan Chen, Cullen Bash, Adam Wierman, Daniel Gmach, Zhikui Wang, Manish Marwah, and Chris Hyser. Renewable and cooling aware workload management for sustainable data centers. In *ACM SIGMETRICS Performance Evaluation Review*, 2012.

[55] Edward Loper and Steven Bird. NLTK: The natural language toolkit. arXiv, 2002.

[56] Thodoris Lykouris and Sergei Vassilvitskii. Competitive caching with machine learned advice. *Journal of the ACM*, 68(4), 2021.

[57] H. Brendan McMahan. A survey of algorithms and analysis for adaptive online learning. *Journal of Machine Learning Research*, 18, 2017.

[58] Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, 2007.

[59] Michael Mitzenmacher and Sergei Vassilvitskii. Algorithms with predictions. In Tim Roughgarden, editor, *Beyond the Worst-Case Analysis of Algorithms*. Cambridge University Press, Cambridge, UK, 2021.

[60] Francesco Orabona and Tatiana Tomassi. Training deep networks without learning rates through coin betting. In *Advances in Neural Information Processing Systems*, 2017.

[61] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, 2014.

[62] John W. Pratt. Concavity of the log likelihood. *Journal of the American Statistical Association*, 76(373):103–106, 1981.

[63] Dhruv Rohatgi. Near-optimal bounds for online caching with machine learned advice. In *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms*, 2020.

[64] Timothy Roughgarden. *Beyond Worst-Case Analysis of Algorithms*. Cambridge University Press, 2020.

[65] Shinsaku Sakaue and Taihei Oki. Discrete-convex-analysis-based framework for warm-starting algorithms with predictions. In *Advances in Neural Information Processing Systems*, 2022.

[66] Ziv Scully, Isaac Grosof, and Michael Mitzenmacher. Uniform bounds for scheduling with job size estimates. In *Proceedings of the 13th Innovations in Theoretical Computer Science Conference*, 2022.

[67] Jeremy Seeman, Aleksandra Slavkovic, and Matthew Reimherr. Private posterior inference consistent with public information: A case study in small area estimation from synthetic census data. In *Proceedings of the International Conference on Privacy in Statistical Databases*, 2020.

[68] Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2011.

[69] Adam Smith and Abhradeep Thakurta. (Nearly) optimal algorithms for private online learning in full-information and bandit settings. In *Advances in Neural Information Processing Systems*, 2013.

[70] Mengting Wan and Julian J. McAuley. Item recommendation on monotonic behavior chains. In *Proceedings of the 12th ACM Conference on Recommender Systems*, 2018.

[71] Chenkai Yu, Guanya Shi, Soon-Ju Chung, Yisong Yue, and Adam Wierman. Competitive control with delayed imperfect information. In *Proceedings of the American Control Conference*, 2022.

[72] Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th International Conference on Machine Learning*, 2003.

A Quantile release

A.1 Section 4.1 details

The base measure μ of DP mechanisms such as the exponential is the starting point of many approaches to incorporating external information, especially ones focused on Bayesian posterior sampling [25, 32, 67]; while it is also our approach to single-quantile estimation with predictions, a key difference here is the focus on utility guarantees depending on both the prediction and instance, which is missing from this past work. In the quantile problem, given a quantile q and a sorted dataset $\mathbf{x} \in \mathbb{R}^n$ of n distinct points, the goal is to release a number o that upper bounds exactly $\lfloor qn \rfloor$ of the entries. A natural error metric, $\operatorname{Gap}_q(\mathbf{x}, o)$, is the number of entries between the released number o and $\lfloor qn \rfloor$, and we can show that prediction-dependent bound using astraightforward application of EM with utility $-\operatorname{Gap}_q$:

Lemma A.1. *Releasing $o \in \mathbb{R}$ w.p. $\propto \exp(-\varepsilon \operatorname{Gap}_q(\mathbf{x}, o)/2)\mu(o)$ is ε-DP, and w.p. $1-\beta$*

$$\operatorname{Gap}_q(\mathbf{x}, o) \leqslant \frac{2}{\varepsilon}\left(\log\frac{1}{\beta} - \log \Psi_{\mathbf{x}}^{(q,\varepsilon)}(\mu)\right) \leqslant \frac{2}{\varepsilon}\left(\log\frac{1}{\beta} - \log \Psi_{\mathbf{x}}^{(q)}(\mu)\right) \tag{28}$$

where $\Psi_{\mathbf{x}}^{(q,\varepsilon)}(\mu) = \sum_{i=0}^{n} \exp(-\varepsilon \operatorname{Gap}_q(\mathbf{x}, I_i)/2)\mu(I_i) = \int \exp(-\varepsilon \operatorname{Gap}_q(\mathbf{x}, o)/2)\mu(o)do$ is the inner product between μ and the exponential score while $\Psi_{\mathbf{x}}^{(q)}(\mu) = \mu(I_{\lfloor qn \rfloor})$ is the measure of the optimal interval (note $\max_k u_q(\mathbf{x}, I_k) = -\operatorname{Gap}_q(\mathbf{x}, I_{\lfloor qn \rfloor}) = 0$ and so $\Psi_{\mathbf{x}}^{(q)}(\mu) \leqslant \Psi_{\mathbf{x}}^{(q,\varepsilon)}(\mu)\ \forall\ \varepsilon > 0$).

Proof. ε-DP follows from u_q having sensitivity one and the guarantee of EM with base measure μ [58, Theorem 6]. For the error, since we sample an interval I_k and then sample $o \in I_k$ we have

$$\begin{aligned}\Pr\{\operatorname{Gap}_q(\mathbf{x}, o) \geqslant \gamma\} = \Pr\{u_q(\mathbf{x}, I_k) \leqslant -\gamma\} &= \sum_{j=0}^{n} \Pr\{k = j\} 1_{u_q(\mathbf{x}, I_j) \leqslant -\gamma} \\ &\leqslant \sum_{j=0}^{n} \frac{\exp(-\frac{\varepsilon\gamma}{2})\mu(I_j)}{\sum_{i=0}^{n} \exp(\frac{\varepsilon}{2} u_q(\mathbf{x}, I_i))\mu(I_i)} \leqslant \frac{\exp(-\frac{\varepsilon\gamma}{2})}{\Psi_{\mathbf{x}}^{(q,\varepsilon)}(\mu)}\end{aligned} \tag{29}$$

The result follows by substituting β for the failure probability and solving for γ. □

We can also analyze the error metrics in this bound for specific measures μ. In particular, if the points are in a bounded interval (a, b) and we use the uniform measure $\mu(o) = 1_{o \in (a,b)}/(b-a)$ then $\Psi_{\mathbf{x}}^{(q,\varepsilon)}(\mu) \geqslant \frac{\psi_{\mathbf{x}}}{b-a}$, where $\psi_{\mathbf{x}} = \min_k \mathbf{x}_{[k+1]} - \mathbf{x}_{[k]}$, and we exactly recover the standard bound of $\frac{2}{\varepsilon}\log\frac{b-a}{\beta\psi_{\mathbf{x}}}$, e.g. the one in Kaplan et al. [42, Lemma A.1] (indeed their analysis implicitly uses this measure). However, our approach also allows us to remove the boundedness assumption, which itself can be viewed as a type of prediction, as one needs external information to assume that the data, or at least the quantile, lies within the interval (a, b). Taking this view, we can use the prediction to set the location $\nu \in \mathbb{R}$ and scale $\sigma > 0$ of a Cauchy prior $\mu_{\nu,\sigma}(o) = \sigma/(\pi(\sigma^2 + (o-\nu)^2))$ without committing to (a, b) actually containing the data. Since we know that the optimal interval $(\mathbf{x}_{[\lfloor qn \rfloor]}, \mathbf{x}_{[\lfloor qn \rfloor+1]}]$ is a subset of $(\frac{a+b}{2} \pm R)$ for some $R > 0$, setting $\nu = \frac{a+b}{2}$ and $\sigma = \frac{b-a}{2}$ yields

$$\Psi_{\mathbf{x}}^{(q)}(\mu_{\nu,\sigma}) \geqslant \frac{\sigma}{\pi}\frac{\mathbf{x}_{[\lfloor qn \rfloor+1]} - \mathbf{x}_{[\lfloor qn \rfloor]}}{\sigma^2 + \max_{k \in \{\lfloor qn \rfloor, \lfloor qn \rfloor+1\}}(\nu - \mathbf{x}_{[k]})^2} \geqslant \frac{\sigma}{\pi}\min_k \frac{\mathbf{x}_{[k+1]} - \mathbf{x}_{[k]}}{\sigma^2 + R^2} \geqslant \frac{2(b-a)\psi_{\mathbf{x}}/\pi}{(b-a)^2 + 4R^2} \tag{30}$$

If $R = \frac{b-a}{2}$, i.e. we get the interval containing the data correct, then substituting the above into Lemma A.1 recovers the guarantee of the uniform prior up to an additive factor $\frac{2}{\varepsilon}\log\pi$. However,

whereas for the uniform prior we have no performance guarantees if the interval is incorrect, using the Cauchy prior the performance degrades gracefully as the error (R) grows. While this first result can be viewed as designing a better prediction-free algorithm, it can also be viewed as making more robust use of the external information about the interval containing the data.

A.1.1 Multiple quantile release using multiple priors

To estimate $m > 1$ quantiles $q_1, \dots, q_m$ at once, we adapt the recursive approach of [42], whose method `ApproximateQuantiles` implicitly constructs a binary tree with a quantile q_i at each node and uses the exponential mechanism to compute the quantile $\tilde{q}_i = (q_i - \underline{q}_i)/(\overline{q}_i - \underline{q}_i)$ of the dataset $\hat{\mathbf{x}}_i$ of points in the original dataset $\mathbf{x}$ restricted to the interval $(\hat{a}_i, \hat{b}_i)$; here $\underline{q}_i < q_i$ and $\overline{q}_i > q_i$ are quantiles appearing earlier in the tree whose respective estimates $\hat{a}_i$ and $\hat{b}_i$ determine the sub-interval (if there is no earlier quantile on the left and/or right of q_i we use $\underline{q}_i = 0, \hat{a}_i = a$ and/or $\overline{q}_i = 1, \hat{b}_i = b$). Because each datapoint only participates in $\mathcal{O}(\log_2 m)$ exponential mechanisms, the approach is able to run each mechanism with budget $\Omega(\varepsilon/\log_2 m)$ and thus only suffer error logarithmic in the number of quantiles m, a significant improvement upon running one EM with budget ε/m on the entire dataset for each quantile, which has error $\mathcal{O}(m)$ in the number of quantiles.

We can apply prior-dependent guarantees to `ApproximateQuantiles`—pseudocode for a generalized version of which is provided in Algorithm 5—by recognizing that implicitly the method assigns a uniform prior μ_i to each quantile q_i and then running EM with the *conditional* prior $\hat{\mu}_i$ restricted to the interval $[\hat{a}_i, \hat{b}_i]$ determined by earlier quantiles in the binary tree. An extension of the argument in Equation 29 (c.f. Lemma A.2) then yields a bound on the error of the estimate o_i returned for quantile q_i in terms of the prior-EM inner-product computed with this conditional prior $\hat{\mu}_i$ over the subset $\hat{\mathbf{x}}_i$:

$$\Pr\{\mathrm{Gap}_{q_i}(\mathbf{x}, o_i) \geqslant \gamma\} \leqslant \frac{\exp\left(\frac{\varepsilon_i}{2}(\hat{\gamma}_i - \gamma)\right)}{\Psi^{(\tilde{q}_i, \varepsilon_i)}_{\hat{\mathbf{x}}_i}(\hat{\mu}_i)} \qquad \text{for} \qquad \hat{\gamma}_i = (1 - \tilde{q}_i)\,\mathrm{Gap}_{\underline{q}_i}(\mathbf{x}, \hat{a}_i) + \tilde{q}_i\,\mathrm{Gap}_{\overline{q}_i}(\mathbf{x}, \hat{b}_i) \quad (31)$$

Note that the error is offset by a weighted combination $\hat{\gamma}_i$ of the errors of the estimates of quantiles earlier in the tree. Controlling this error allows us to bound the maximum error of any quantile via the harmonic mean of the inner products between the exponential scores and conditional priors:

Lemma A.2. *Algorithm 5 with $K = 2$ and $\varepsilon_i = \varepsilon/\lceil \log_2 m \rceil$ $\forall$ i is ε-DP and w.p. $\geqslant 1 - \beta$ has*

$$\max_i \mathrm{Gap}_{q_i}(\mathbf{x}, o_i) \leqslant \frac{2}{\varepsilon}\lceil \log_2 m \rceil^2 \log \frac{m}{\beta \hat{\Psi}^{(\varepsilon)}_{\mathbf{x}}} \qquad \textit{for} \qquad \hat{\Psi}^{(\varepsilon)}_{\mathbf{x}} = \left(\sum_{i=1}^{m} \frac{1/m}{\Psi^{(\tilde{q}_i, \varepsilon_i)}_{\hat{\mathbf{x}}_i}(\hat{\mu}_i)} \right)^{-1} \quad (32)$$

Proof. The privacy guarantee follows as in [42, Lemma 3.1]. Setting the above probability bound (31) to $\frac{\beta \hat{\Psi}^{(\varepsilon)}_{\mathbf{x}}}{m \Psi^{(\varepsilon_i)}_{\tilde{q}_i}(\hat{\mathbf{x}}_i, \hat{\mu}_i)}$ for each i we have w.p. $\geqslant 1 - \beta$ that $\mathrm{Gap}_{q_i}(\mathbf{x}, o_i) \leqslant \frac{2}{\bar{\varepsilon}} \log \frac{m}{\beta \hat{\Psi}^{(\varepsilon)}_{\mathbf{x}}} + \hat{\gamma}_i$ $\forall$ i. Now let k_i be the depth of quantile q_i in the tree. If $k_i = 1$ then i is the root node so $\hat{\gamma}_i = 0$ and we have $\mathrm{Gap}_{q_i}(\mathbf{x}, o_i) \leqslant \frac{2}{\bar{\varepsilon}} \log \frac{m}{\beta \hat{\Psi}^{(\varepsilon)}_{\mathbf{x}}}$. To make an inductive argument, we assume $\mathrm{Gap}_{q_i}(\mathbf{x}, o_i) \leqslant \frac{2k}{\bar{\varepsilon}} \log \frac{m}{\beta \hat{\Psi}^{(\varepsilon)}}$ $\forall$ i s.t. $k_i \leqslant k$, and so for any i s.t. $k_i = k + 1$ we have that

$$\mathrm{Gap}_{q_i}(\mathbf{x}, o_i) \leqslant \frac{2}{\bar{\varepsilon}} \log \frac{m}{\beta \hat{\Psi}^{(\varepsilon)}_{\mathbf{x}}} + (1 - \tilde{q}_i)\,\mathrm{Gap}_{\underline{q}_i}(\mathbf{x}, \hat{a}_i) + \tilde{q}_i\,\mathrm{Gap}_{\overline{q}_i}(\mathbf{x}, \hat{b}_i) \leqslant \frac{2(k+1)}{\bar{\varepsilon}} \log \frac{m}{\beta \hat{\Psi}^{(\varepsilon)}_{\mathbf{x}}} \quad (33)$$

Thus $\mathrm{Gap}_{q_i}(\mathbf{x}, o_i) \leqslant \frac{2k_i}{\bar{\varepsilon}} \log \frac{m}{\beta \hat{\Psi}^{(\varepsilon)}_{\mathbf{x}}}$ $\forall$ i, so using $k_i \leqslant \lceil \log_2 m \rceil$ and $\bar{\varepsilon} = \frac{\varepsilon}{\lceil \log_2 m \rceil}$ yields the result. □

Setting $\hat{\mu}_i$ to be uniform on $[\hat{a}_i, \hat{b}_i]$ exactly recovers both the algorithm and guarantee of [42, Theorem 3.3]. As before, we can also extend the algorithm to the infinite interval:

Corollary A.1. *If all priors are Cauchy with location $\frac{a+b}{2}$ and scale $\frac{b-a}{2}$ and the data lies in the interval $(\frac{a+b}{2} \pm R)$ then w.p. $\geqslant 1-\beta$ the maximum error is at most $\frac{2}{\varepsilon}\lceil\log_2 m\rceil^2 \log\left(\pi m \frac{b-a+\frac{4R^2}{b-a}}{2\beta\psi_{\mathbf{x}}}\right)$.*

However, while this demonstrates the usefulness of Lemma A.2 for obtaining robust priors on infinite intervals, the associated prediction measure $\hat{\Psi}_{\mathbf{x}}^{(\varepsilon)}$ is imperfect because it is non-deterministic: its value depends on the random execution of the algorithm, specifically on the data subsets $\hat{\mathbf{x}}_i$ and priors $\hat{\mu}_i$, which for i not at the root of the tree are affected by the DP mechanisms of i's ancestor nodes. In addition to not being given fully specified by the prediction and data, this makes $\hat{\Psi}^{(\varepsilon)}$ difficult to use as an objective for learning. A natural more desirable prediction metric is the harmonic mean of the inner products between the exponential scores and *original* priors μ_i over the *original* dataset $\mathbf{x}$, i.e. the direct generalization of our approach for single quantiles.

Unfortunately, the conditional restriction of μ_i to the interval $[\hat{a}_i, \hat{b}_i]$ removes the influence of probabilities assigned to intervals between points *not* in this interval. To solve this, we propose a different *edge*-restriction of μ_i that assigns probabilities $\mu_i((-\infty, \hat{a}_i))$ and $\mu_i((\hat{b}_i, \infty))$ of being outside the interval $[\hat{a}_i, \hat{b}_i]$ to atoms on its edges $\hat{a}_i$ and $\hat{b}_i$, respectively. Despite not using any information from points outside $\hat{\mathbf{x}}_i$, this approach puts probabilities assigned to intervals outside $[\hat{a}_i, \hat{b}_i]$ to the edge closest to them, allowing us to extend the previous probability bound (31) to depend on the original prior-EM inner-product (c.f. Lemma A.5):

$$\Pr\{\mathrm{Gap}_{q_i}(\mathbf{x}, o_i) \geqslant \gamma\} \leqslant \exp(\varepsilon(\hat{\gamma}_i - \gamma/2))/\Psi_{\mathbf{x}}^{(q_i,\varepsilon_i)}(\mu_i) \tag{34}$$

However, the stronger dependence of this bound on errors $\hat{\gamma}_i$ earlier in the tree lead to an $\tilde{\mathcal{O}}(\phi^{\log_2 m}) = \mathcal{O}(m^{0.7})$ dependence on m, where $\phi = \frac{1+\sqrt{5}}{2}$ is the golden ratio:

Theorem A.1. *If the quantiles are uniform negative powers of two then Algorithm 5 with $K = 2$, edge-based prior adaptation, and $\varepsilon_i = \varepsilon/\lceil\log_2(m+1)\rceil\ \forall\ i$ is ε-DP and w.p. $\geqslant 1-\beta$ has*

$$\max_i \mathrm{Gap}_{q_i}(\mathbf{x}, o_i) \leqslant \frac{2}{\varepsilon}\phi^{\log_2(m+1)}\lceil\log_2(m+1)\rceil \log\frac{m}{\beta\Psi_{\mathbf{x}}^{(\varepsilon)}} \qquad \textit{for} \qquad \Psi_{\mathbf{x}}^{(\varepsilon)} = \left(\sum_{i=1}^{m} \frac{1/m}{\Psi_{\mathbf{x}}^{(q_i,\varepsilon_i)}(\mu_i)}\right)^{-1} \tag{35}$$

Proof. Since $\tilde{q}_i = 1/2\ \forall\ i$, setting the new probability bound equal to $\frac{\beta\Psi_{\mathbf{x}}^{(\varepsilon)}}{m\Psi_{\mathbf{x}}^{(q_i\varepsilon_i)}(\mu_i)}$ yields that w.p. $\geqslant 1-\beta$

$$\mathrm{Gap}_{q_i}(\mathbf{x}, o_i) \leqslant \frac{2}{\bar{\varepsilon}}\log\frac{m}{\beta\Psi_{\mathbf{x}}^{(\varepsilon)}} + 2\hat{\gamma}_i = \frac{2}{\bar{\varepsilon}}\log\frac{m}{\beta\Psi_{\mathbf{x}}^{(\varepsilon)}} + \mathrm{Gap}_{\underline{q}_i}(\mathbf{x}, \hat{a}_i) + \mathrm{Gap}_{\overline{q}_i}(\mathbf{x}, \hat{b}_i)\ \forall\ i \tag{36}$$

If for each $k \leqslant \lceil\log_2 m\rceil$ we define E_k to be the maximum error of any quantile of at most depth k in the tree then since one of $\underline{q}_i$ and $\overline{q}_i$ is at depth at least one less than q_i and the other is at depth at least two less than q_i we have $E_k \leqslant \frac{2A_k}{\bar{\varepsilon}}\log\frac{m}{\beta\Psi_{\mathbf{x}}^{(\varepsilon)}}$ for recurrent relation $A_k = 1 + A_{k-1} + A_{k-2}$ with $A_0 = 0$ and $A_1 = 1$. Since $A_k = F_{k+1} - 1$ for Fibonacci sequence $F_j = \frac{\phi^j - (1-\phi)^j}{\sqrt{5}}$, we have

$$\max_i \mathrm{Gap}_{q_i}(\mathbf{x}, o_i) = \max_k E_k \leqslant \frac{2\phi^{\lceil\log_2(m+1)\rceil+1}}{\bar{\varepsilon}\sqrt{5}}\log\frac{m}{\beta\Psi_{\mathbf{x}}^{(\varepsilon)}} = \frac{2\phi^{\lceil\log_2(m+1)\rceil+1}}{\varepsilon\sqrt{5}}\lceil\log_2(m+1)\rceil\log\frac{m}{\beta\Psi_{\mathbf{x}}^{(\varepsilon)}} \tag{37}$$

□

Thus while we have obtained a performance guarantee depending only on the prediction and the data via the harmonic mean $\Psi_{\mathbf{x}}^{(\varepsilon)}$ of the true prior-EM inner-products, the dependence on m is now polynomial. Note that it is still sublinear, which means it is better than the naive baseline of running m independent exponential mechanisms. Still, we can do much better—in-fact asymptotically better than any power of m—by recognizing that the main issue is the compounding error induced by successive errors to the boundaries of sub-intervals. We can reduce this by reducing the depth of the tree using a K-ary rather than binary tree and instead paying $K-1$ times the privacy budget at each depth in order to naively release values for $K-1$ quantiles. This can introduce out-of-order quantiles, but by Lemma A.6 swapping any two out-of-order quantiles does not increase the maximum error and so this issue can be solved by sorting the $K-1$ quantiles before using them to split the data. We thus have the following prediction-dependent performance bound for multiple quantiles:

Theorem A.2. *If we run Algorithm 5 with* $K = \lceil \exp(\sqrt{\log 2 \log(m+1)}) \rceil$, *edge-based adaptation, and* $\varepsilon_i = \frac{\bar{\varepsilon}}{k_i^p}$ *for some power* $p > 1$, k_i *the depth of* q_i *in the* K*-ary tree, and* $\bar{\varepsilon} = \frac{\varepsilon}{K-1}\left(\sum_{k=1}^{\lceil \log_K(m+1)\rceil} \frac{1}{k^p}\right)^{-1}$, *then the result satisfies* ε*-DP and w.p.* $\geqslant 1-\beta$ *we have*

$$\max_i \operatorname{Gap}_{q_i}(\mathbf{x}, o_i) \leqslant \frac{2\pi^2}{\varepsilon} \exp\left(2\sqrt{\log(2)\log(m+1)}\right) \log \frac{m}{\beta \Psi_{\mathbf{x}}^{(\varepsilon)}} \tag{38}$$

if $p = 2$ *and more generally* $\max_i \operatorname{Gap}_{q_i}(\mathbf{x}, o_i) \leqslant \frac{c_p}{\varepsilon} \exp\left(2\sqrt{\log(2)\log(m+1)}\right) \log \frac{m}{\beta \Psi_{\mathbf{x}}^{(\varepsilon)}}$, *where* c_p *depends only on* p.

Proof. The privacy guarantee follows as in [42, Lemma 3.1] except before each split we compute $K-1$ quantiles with $K-1$ times less budget. As in the previous proof, we have w.p. $\geqslant 1-\beta$ that

$$\operatorname{Gap}_{q_i}(\mathbf{x}, o_i) \leqslant \frac{2}{\varepsilon_i} \log \frac{m}{\beta \Psi_{\mathbf{x}}^{(\varepsilon)}} + 2\hat{\gamma}_i = \frac{2k_i^2}{\bar{\varepsilon}} \log \frac{m}{\beta \Psi_{\mathbf{x}}^{(\varepsilon)}} + 2(1-\tilde{q}_i) \operatorname{Gap}_{\underline{q}_i}(\mathbf{x}, \hat{a}_i) + 2\tilde{q}_i \operatorname{Gap}_{\overline{q}_i}(\mathbf{x}, \hat{b}_i) \;\forall\; i \tag{39}$$

If for each $k \leqslant \lceil \log_K(m+1) \rceil$ we define E_k to be the maximum error of any quantile of at most depth k in the tree then since both $\underline{q}_i$ and $\overline{q}_i$ are at depth at least one less than q_i we have $E_k \leqslant \frac{2A_k}{\bar{\varepsilon}} \log \frac{m}{\beta \Psi_{\mathbf{x}}^{(\varepsilon)}}$, where $A_k = k^p + 2A_{k-1}$ and $A_1 = 1$. For the case of $p = 2$, $A_k \leqslant 6 \cdot 2^k$ and $1/\bar{\varepsilon} = \frac{K-1}{\varepsilon} \sum_{k=1}^{\lceil \log_K(m+1)\rceil} \frac{1}{k^2} \leqslant \frac{\pi^2}{6\varepsilon}(K-1)$ so we have that

$$\max_i \operatorname{Gap}_{q_i}(\mathbf{x}, o_i) = \max_k E_k \leqslant \frac{12}{\bar{\varepsilon}} 2^{\lceil \log_K(m+1)\rceil} \log \frac{m}{\beta \Psi_{\mathbf{x}}^{(\varepsilon)}} \leqslant \frac{2\pi^2}{\varepsilon}(K-1) 2^{\lceil \log_K(m+1)\rceil} \log \frac{m}{\beta \Psi_{\mathbf{x}}^{(\varepsilon)}} \tag{40}$$

Substituting $K = \lceil \exp(\sqrt{\log 2 \log(m+1)}) \rceil$ and simplifying yields the result. For $p > 1$, $A_k \leqslant 2^{k-2}\left(2 + \Phi\left(\frac{1}{2}, -p, 2\right)\right)$, where Φ is the Lerch transcendent, and $1/\bar{\varepsilon} \leqslant \frac{K-1}{\varepsilon}\zeta(p)$, where ζ is the Riemann zeta function. Therefore

$$\begin{aligned} \max_i \operatorname{Gap}_{q_i}(\mathbf{x}, o_i) = \max_k E_k &\leqslant \frac{2^{\lceil \log_K(m+1)\rceil}}{2\bar{\varepsilon}} \left(2 + \Phi\left(\frac{1}{2}, -p, 2\right)\right) \log \frac{m}{\beta \Psi_{\mathbf{x}}^{(\varepsilon)}} \\ &\leqslant \frac{c_p}{\varepsilon}(K-1) 2^{\lceil \log_K(m+1)\rceil} \log \frac{m}{\beta \Psi_{\mathbf{x}}^{(\varepsilon)}} \end{aligned} \tag{41}$$

for $c_p = \left(1 + \Phi\left(\frac{1}{2}, -p, 2\right)/2\right)\zeta(p)$. □

Similarly to Theorem A.1, the proof establishes a recurrence relationship between the maximum errors at each depth. Note that in addition to the K-ary tree this bound uses depth-dependent budgeting to remove a $\mathcal{O}(\log_2 m)$-factor; the constant depending upon the parameter $p > 1$ of the latter has a minimum of roughly 8.42 at $p \approx 1.6$. As discussed before, the new dependence $\tilde{\mathcal{O}}\left(\exp\left(2\sqrt{\log(2)\log(m+1)}\right)\right)$ on m is sub-polynomial, i.e $o(m^\alpha)\ \forall\ \alpha > 0$. While it is also super-polylogarithmic, its shape for any practical value of m is roughly $\mathcal{O}(\log_2^2 m)$, making the result of interest as a justification for the negative log-inner-product performance metric.

A.1.2 Experimental details

For the experiments in Section 4.1, specifically Figure 3, we evaluate three variants of the algorithm on data drawn from a standard Gaussian distribution and from the Adult "age" dataset [47]. In both cases we use 1000 samples and run each experiment 40 times, reporting the average performance. As we do for all datasets, we use reasonable guesses of mean, scale, and bounds on each dataset to set priors. As in this section we report the Uniform, we need to specify its range; for Gaussian we use $[-10, 10]$, while for "age" we use $[10, 120]$.

The original AQ algorithm of Kaplan et al. [42] is now fully specified. We test two variants of our K-ary modification: one with edge-based adaptation, and the other using the original conditional adaptation. For both cases we set K as a function of m according to the formula in Theorem 4.3, and we set the power p of the depth-dependent budget discounting to 1.5, which is close to the theoretically optimal value of around 1.6 (c.f. Thm A.2).

A.2 Additional proofs

Lemma A.3. *In Algorithm 5, for any $i \in [m]$ and $\hat{\gamma}_i = (1-\tilde{q}_i)\operatorname{Gap}_{\underline{q}_i}(\mathbf{x}, \hat{a}_i) + \tilde{q}_i \operatorname{Gap}_{\overline{q}_i}(\mathbf{x}, \hat{b}_i)$ we have*

1. $\operatorname{Gap}_{\tilde{q}_i}(\hat{\mathbf{x}}_i, o) \leqslant \operatorname{Gap}_{q_i}(\mathbf{x}, o) + \hat{\gamma}_i\ \forall\ o \in \mathbb{R}$
2. $\operatorname{Gap}_{q_i}(\mathbf{x}, o) \leqslant \operatorname{Gap}_{\tilde{q}_i}(\hat{\mathbf{x}}_i, o) + \hat{\gamma}_i\ \forall\ o \in [\hat{a}_i, \hat{b}_i]$

Proof. For $o \in [\hat{a}_i, \hat{b}_i]$ we apply the triangle inequality twice to get

$$\begin{aligned}
\operatorname{Gap}_{\tilde{q}_i}(\hat{\mathbf{x}}_i, o) &= |\max_{\hat{\mathbf{x}}_{[j]} < o} j - \lfloor \tilde{q}_i \hat{n}_i \rfloor| \\
&= |\max_{\hat{\mathbf{x}}_{[j]} < o} j + \max_{\mathbf{x}_{[j]} < \hat{a}_i} j - \lfloor q_i n \rfloor + \lfloor q_i n \rfloor - \max_{\mathbf{x}_{[j]} < \hat{a}_i} j - \lfloor \tilde{q}_i \hat{n}_i \rfloor| \\
&\leqslant \operatorname{Gap}_{q_i}(\mathbf{x}, o) + \left| \lfloor \tilde{q}_i (\lfloor \overline{q}_i n \rfloor - \lfloor \underline{q}_i n \rfloor) \rfloor + \lfloor \underline{q}_i n \rfloor - \max_{\mathbf{x}_{[j]} < \hat{a}_i} j - \lfloor \tilde{q}_i (\max_{\mathbf{x}_{[j]} < \hat{b}_i} j - \max_{\mathbf{x}_{[j]} < \hat{a}_i} j) \rfloor \right| \\
&\leqslant \operatorname{Gap}_{q_i}(\mathbf{x}, o) + (1-\tilde{q}_i)\operatorname{Gap}_{\underline{q}_i}(\mathbf{x}, \hat{a}_i) + \tilde{q}_i \operatorname{Gap}_{\overline{q}_i}(\mathbf{x}, \hat{b}_i)
\end{aligned} \tag{42}$$

and again to get

$$\begin{aligned}
\operatorname{Gap}_{q_i}(\mathbf{x}, o) &= |\max_{\mathbf{x}_{[j]} < o} j - \lfloor q_i n \rfloor| \\
&= |\max_{\hat{\mathbf{x}}_{[j]} < o} j + \max_{\mathbf{x}_{[j]} < \hat{a}_i} j - \lfloor \tilde{q}_i \hat{n}_i \rfloor + \lfloor \tilde{q}_i \hat{n}_i \rfloor - \lfloor q_i n \rfloor| \\
&\leqslant \operatorname{Gap}_{\tilde{q}_i}(\hat{\mathbf{x}}_i, o) + \left| \max_{\mathbf{x}_{[j]} < \hat{a}_i} j - \lfloor \tilde{q}_i (\max_{\mathbf{x}_{[j]} < \hat{b}_i} j + \max_{\mathbf{x}_{[j]} < \hat{a}_i} j) \rfloor - \lfloor \tilde{q}_i (\lfloor \overline{q}_i n \rfloor - \lfloor \underline{q}_i n \rfloor) \rfloor - \lfloor \underline{q}_i n \rfloor \right| \\
&\leqslant \operatorname{Gap}_{\tilde{q}_i}(\hat{\mathbf{x}}_i, o) + (1-\tilde{q}_i)\operatorname{Gap}_{\underline{q}_i}(\mathbf{x}, \hat{a}_i) + \tilde{q}_i \operatorname{Gap}_{\overline{q}_i}(\mathbf{x}, \hat{b}_i)
\end{aligned} \tag{43}$$

For $o < \hat{a}_i$ we use the fact that $\max_{\mathbf{x}_{[j]}<o} j \leqslant \max_{\mathbf{x}_{[j]}<\hat{a}_i} j$ and the triangle inequality to get

$$\begin{aligned}
\mathrm{Gap}_{\tilde{q}_i}(\hat{\mathbf{x}}_i, o) &= \lfloor \tilde{q}_i \hat{n}_i \rfloor \\
&= \lfloor \tilde{q}_i (\max_{\mathbf{x}_{[j]}<\hat{b}_i} j - \max_{\mathbf{x}_{[j]}<\hat{a}_i} j) \rfloor \\
&\leqslant \lfloor \tilde{q}_i \max_{\mathbf{x}_{[j]}<\hat{b}_i} j \rfloor + \lfloor (1-\tilde{q}_i) \max_{\mathbf{x}_{[j]}<\hat{a}_i} j \rfloor - \max_{\mathbf{x}_{[j]}<o} j \\
&= \lfloor \tilde{q}_i \max_{\mathbf{x}_{[j]}<\hat{b}_i} j \rfloor + \lfloor (1-\tilde{q}_i) \max_{\mathbf{x}_{[j]}<\hat{a}_i} j \rfloor - \max_{\mathbf{x}_{[j]}<o} j + \lfloor q_i n \rfloor | - \lfloor \tilde{q}_i (\lfloor \overline{q}_i n \rfloor - \lfloor \underline{q}_i n \rfloor) \rfloor - \lfloor \underline{q}_i n \rfloor \\
&\leqslant \mathrm{Gap}_{q_i}(\mathbf{x}, o) + (1-\tilde{q}_i) \mathrm{Gap}_{\underline{q}_i}(\mathbf{x}, \hat{a}_i) + \tilde{q}_i \mathrm{Gap}_{\overline{q}_i}(\mathbf{x}, \hat{b}_i)
\end{aligned} \tag{44}$$

For $o > \hat{b}_i$ we use the fact that $\max_{\mathbf{x}_{[j]}<\hat{b}_i} j \leqslant \max_{\mathbf{x}_{[j]}<o} j$ and the triangle inequality to get

$$\begin{aligned}
\mathrm{Gap}_{\tilde{q}_i}(\hat{\mathbf{x}}_i, o) &= \lfloor (1-\tilde{q}_i) \hat{n}_i \rfloor \\
&= \lfloor (1-\tilde{q}_i) (\max_{\mathbf{x}_{[j]}<\hat{b}_i} j - \max_{\mathbf{x}_{[j]}<\hat{a}_i} j) \rfloor \\
&\leqslant \max_{\mathbf{x}_{[j]}<o} j - \lfloor \tilde{q}_i \max_{\mathbf{x}_{[j]}<\hat{b}_i} j - \lfloor (1-\tilde{q}_i) \max_{\mathbf{x}_{[j]}<\hat{a}_i} j \\
&= \max_{\mathbf{x}_{[j]}<o} j - \lfloor \tilde{q}_i \max_{\mathbf{x}_{[j]}<\hat{b}_i} j - \lfloor (1-\tilde{q}_i) \max_{\mathbf{x}_{[j]}<\hat{a}_i} j - \lfloor q_i n \rfloor + \lfloor \tilde{q}_i (\lfloor \overline{q}_i n \rfloor - \lfloor \underline{q}_i n \rfloor) \rfloor + \lfloor \underline{q}_i n \rfloor \\
&\leqslant \mathrm{Gap}_{q_i}(\mathbf{x}, o) + (1-\tilde{q}_i) \mathrm{Gap}_{\underline{q}_i}(\mathbf{x}, \hat{a}_i) + \tilde{q}_i \mathrm{Gap}_{\overline{q}_i}(\mathbf{x}, \hat{b}_i)
\end{aligned} \tag{45}$$

□

Lemma A.4. *For any $\gamma > 0$ the estimate o_i of the quantile q_i by Algorithm 5 satisfies*

$$Pr\{\mathrm{Gap}_{q_i}(\mathbf{x}, o_i) \geqslant \gamma\} \leqslant \frac{\exp\left(\varepsilon_i(\hat{\gamma}_i - \gamma)/2\right)}{\Psi_{\hat{\mathbf{x}}_i}^{(\tilde{q}_i, \varepsilon_i)}(\hat{\mu}_i)} \tag{46}$$

Proof. We use k_i to denote the interval $\hat{I}_k^{(j)}$ sampled at index i in the algorithm and note that o_i corresponds to the released number o at that index. Since $o_i \in [\hat{a}_i, \hat{b}_i]$, applying Lemma A.3 yields

$$\begin{aligned}
\Pr\{\mathrm{Gap}_{q_i}(\mathbf{x}, o_i) \geqslant \gamma\} &= \sum_{j=0}^{\hat{n}_i} \Pr\{k_i = j\} 1_{\mathrm{Gap}_{q_i}(\mathbf{x}, \hat{I}_j^{(i)}) \geqslant \gamma} \\
&= \sum_{j=0}^{n_i} \frac{\exp(-\varepsilon \mathrm{Gap}_{\tilde{q}_i}(\hat{\mathbf{x}}_i, \hat{I}_j^{(i)})/2) \hat{\mu}_i(\hat{I}_j^{(i)}) 1_{\mathrm{Gap}_{q_i}(\mathbf{x}, \hat{I}_j^{(i)}) \geqslant \gamma}}{\sum_{l=0}^{\hat{n}_i} \exp(\varepsilon u_{\tilde{q}_i}(\hat{\mathbf{x}}_i, \hat{I}_l^{(i)})/2) \hat{\mu}_i(\hat{I}_l)} \\
&\leqslant \frac{\exp(\varepsilon \hat{\gamma}_i/2)}{\Psi_{\hat{\mathbf{x}}_i}^{(\tilde{q}_i, \varepsilon_i)}(\hat{\mu}_i)} \sum_{j=0}^{n_i} \exp(-\varepsilon \mathrm{Gap}_{q_i}(\mathbf{x}, \hat{I}_j^{(i)})/2) \hat{\mu}_i(\hat{I}_j^{(i)}) 1_{\mathrm{Gap}_{q_i}(\mathbf{x}, \hat{I}_j^{(i)}) \geqslant \gamma} \\
&\leqslant \frac{\exp(\varepsilon(\hat{\gamma}_i - \gamma)/2)}{\Psi_{\hat{\mathbf{x}}_i}^{(\tilde{q}_i, \varepsilon_i)}(\hat{\mu}_i)}
\end{aligned} \tag{47}$$

□

Lemma A.5. *For any $\gamma > 0$ the estimate o_i of the quantile q_i by Algorithm 5 with edge-based prior adaptation satisfies*

$$\Pr\{\mathrm{Gap}_{q_i}(\mathbf{x}, o_i) \geqslant \gamma\} \leqslant \frac{\exp(\varepsilon(\hat{\gamma}_i - \gamma/2))}{\Psi_{\mathbf{x}}^{(q_i, \varepsilon_i)}(\mu_i)} \tag{48}$$

Proof. Applying Lemma A.3 yields the following lower bound on $\Psi_{\tilde{q}_i}^{(\varepsilon_i)}(\hat{\mathbf{x}}_i, \hat{\mu}_i)$:

$$\begin{aligned}
\sum_{l=0}^{\hat{n}_i} \exp(\varepsilon u_{\tilde{q}_i}(\hat{\mathbf{x}}_i, \hat{I}_l^{(i)})/2)\hat{\mu}_i(\hat{I}_l^{(i)}) &= \exp(\varepsilon u_{\tilde{q}_i}(\hat{\mathbf{x}}_i, \hat{I}_0^{(i)})/2)\mu_i((-\infty, \hat{a}_i]) + \exp(\varepsilon u_{\tilde{q}_i}(\hat{\mathbf{x}}_i, \hat{I}_{\hat{n}_i}^{(i)})/2)\mu_i([\hat{b}_i, \infty)) \\
&\quad + \sum_{l=0}^{\hat{n}_i} \exp(\varepsilon u_{\tilde{q}_i}(\hat{\mathbf{x}}_i, \hat{I}_l^{(i)})/2)\mu_i(\hat{I}_l) \\
&= \sum_{l=0}^{\max_{\mathbf{x}_{[j]}<\hat{a}_i} j} \exp(-\varepsilon \operatorname{Gap}_{\tilde{q}_i}(\hat{\mathbf{x}}_i, I_l \cap (-\infty, \hat{a}_i])/2)\mu_i(I_l \cap (-\infty, \hat{a}_i]) \\
&\quad + \sum_{l=\max_{\mathbf{x}_{[j]}<\hat{b}_i} j}^{n} \exp(-\varepsilon \operatorname{Gap}_{\tilde{q}_i}(\hat{\mathbf{x}}_i, I_l \cap [\hat{b}_i, \infty))/2)\mu_i(I_l \cap [\hat{b}_i, \infty)) \\
&\quad + \sum_{l=\max_{\mathbf{x}_{[j]}<\hat{a}_i} j}^{\max_{\mathbf{x}_{[j]}<\hat{b}_i} j} \exp(-\varepsilon \operatorname{Gap}_{\tilde{q}_i}(\hat{\mathbf{x}}_i, I_l \cap [\hat{a}_i, \hat{b}_i])\mu_i(I_l \cap [\hat{a}_i, \hat{b}_i]) \\
&\geqslant \Psi_{\mathbf{x}}^{(q_i, \varepsilon_i)}(\mu_i) \exp(-\varepsilon\hat{\gamma}_i/2)
\end{aligned} \tag{49}$$

Substituting into Lemma A.2 yields the result. □

Lemma A.6. *Suppose $q_0 < q_1$ are two quantiles and $o_0 > o_1$. Then*

$$\max_{i=0,1} \operatorname{Gap}_{q_i}(\mathbf{x}, o_i) \geqslant \max_{i=0,1} \operatorname{Gap}_{q_i}(\mathbf{x}, o_{1-i}) \tag{50}$$

Proof. We consider four cases. If $\lfloor q_0|\mathbf{x}| \rfloor \leqslant \max_{\mathbf{x}_{[j]}<o_1} j$ and $\lfloor q_1|X| \rfloor \leqslant \max_{\mathbf{x}_{[j]}<o_0} j$ then

$$\lfloor q_0|\mathbf{x}| \rfloor \leqslant \min\{\lfloor q_1|\mathbf{x}| \rfloor, \max_{\mathbf{x}_{[j]}<o_1} j\} \leqslant \max\{\lfloor q_1|\mathbf{x}| \rfloor, \max_{\mathbf{x}_{[j]}<o_1} j\} \leqslant \max_{\mathbf{x}_{[j]}<o_0} j \tag{51}$$

and so

$$\max_{i=0,1} \operatorname{Gap}_{q_i}(\mathbf{x}, o_i) = \max_{\mathbf{x}_{[j]}<o_0} j - \lfloor q_0|\mathbf{x}| \rfloor \geqslant \max_{i=0,1} \operatorname{Gap}_{q_i}(X, o_{i-1}) \tag{52}$$

If $\lfloor q_0|X| \rfloor \leqslant \max_{\mathbf{x}_{[j]}<o_1} j$ and $\lfloor q_1|\mathbf{x}| \rfloor > \max_{\mathbf{x}_{[j]}<o_0} j$ then

$$\lfloor q_0|\mathbf{x}| \rfloor \leqslant \max_{\mathbf{x}_{[j]}<o_1} j \leqslant \max_{\mathbf{x}_{[j]}<o_0} j < \lfloor q_1|\mathbf{x}| \rfloor \tag{53}$$

and so both improve after swapping. If $\lfloor q_0|\mathbf{x}| \rfloor > \max_{\mathbf{x}_{[j]}<o_1} j$ and $\lfloor q_1|\mathbf{x}| \rfloor > \max_{\mathbf{x}_{[j]}<o_0} j$ then

$$\max_{\mathbf{x}_{[j]}<o_1} j \leqslant \min\{\lfloor q_0|\mathbf{x}| \rfloor, \max_{\mathbf{x}_{[j]}<o_0} j\} \leqslant \max\{\lfloor q_0|\mathbf{x}| \rfloor, \max_{\mathbf{x}_{[j]}<o_0} j\} \leqslant \lfloor q_1|\mathbf{x}| \rfloor \tag{54}$$

and so

$$\max_{i=0,1} \operatorname{Gap}_{q_i}(\mathbf{x}, o_i) = \max_{\mathbf{x}_{[j]}<o_1} j - \lfloor q_1|\mathbf{x}| \rfloor \geqslant \max_{i=0,1} \operatorname{Gap}_{q_i}(\mathbf{x}, o_{i-1}) \tag{55}$$

Finally, if $\lfloor q_0|\mathbf{x}| \rfloor > \max_{\mathbf{x}_{[j]}<o_1} j$ and $\lfloor q_1|\mathbf{x}| \rfloor \leqslant \max_{\mathbf{x}_{[j]}<o_0} j$ then

$$\max_{\mathbf{x}_{[j]}<o_1} j < \lfloor q_0|\mathbf{x}| \rfloor \leqslant \lfloor q_1|\mathbf{x}| \rfloor \leqslant \max_{\mathbf{x}_{[j]}<o_0} j \tag{56}$$

so swapping will make the new largest error for each quantile at most as large as the other quantile's current error. □

Algorithm 5: `ApproximateQuantiles` with predictions

Input: sorted unrepeated data $\mathbf{x} \in (a,b)^n$, ordered quantiles $q_1, \dots, q_m \in (0,1)$,
priors $\mu_1, \dots, \mu_m : \mathbb{R} \mapsto \mathbb{R}_{\geqslant 0}$, prior adaptation rule $r \in$ {`conditional`,`edge`},
privacy parameters $\varepsilon_1, \dots, \varepsilon_m > 0$, branching factor $K \geqslant 2$

`// runs single-quantile algorithm on datapoints` $\hat{\mathbf{x}}$

Method `quantile`($\hat{\mathbf{x}}$, q, ε, μ)**:**

  Output: $o \in (a,b)$ w.p. $\propto \exp(-\varepsilon \operatorname{Gap}_q(\hat{\mathbf{x}}, o)/2)\mu(o)$

Method `recurse`($\mathbf{j}$, $\underline{q}$, $\overline{q}$, $\hat{a}$, $\hat{b}$)**:**

  `// determines` $K-1$ `indices` $\mathbf{i}$ `whose quantiles to compute at this node`

  if $|\mathbf{j}| \geqslant K$ **then**

    $\mathbf{i} \leftarrow \left(\mathbf{j}_{[\lceil |\mathbf{j}|/K \rceil]}, \cdots, \mathbf{j}_{[\lceil (K-1)|\mathbf{j}|/K \rceil]}\right)$

  else

    $\mathbf{i} \leftarrow \mathbf{j}$

  `// restricts dataset to the interval` $(\hat{a}, \hat{b})$

  $\underline{k}_{\mathbf{i}} \leftarrow \min_{\mathbf{x}_{[k]} > \hat{a}} k$

  $\overline{k}_{\mathbf{i}} \leftarrow \max_{\mathbf{x}_{[k]} < \hat{b}} k$

  $\hat{\mathbf{x}}_{\mathbf{i}} \leftarrow \left(\mathbf{x}_{[\underline{k}_{\mathbf{i}}]}, \cdots, \mathbf{x}_{[\overline{k}_{\mathbf{i}}]}\right)$

  `// sets relative quantiles` $\tilde{q}_i$ `and restricts priors to the interval` $[\hat{a}, \hat{b}]$

  for $j = 1, \dots, |\mathbf{i}|$ **do**

    $\tilde{q}_{\mathbf{i}_{[j]}} \leftarrow (q_{\mathbf{i}_{[j]}} - \underline{q})/(\overline{q} - \underline{q})$

    if $r =$ `conditional` **then**

      $\hat{\mu}_{\mathbf{i}_{[j]}}(o) \leftarrow \frac{\mu_{\mathbf{i}_{[j]}}(o)}{\mu_{\mathbf{i}_{[j]}}([\hat{a},\hat{b}])} 1_{o \in [\hat{a},\hat{b}]}$

    else

      $\hat{\mu}_{\mathbf{i}_{[j]}}(o) \leftarrow \mu_{\mathbf{i}_{[j]}}(o) 1_{o \in (\hat{a},\hat{b})} + \mu_{\mathbf{i}_{[j]}}((-\infty, \hat{a}])\delta(o - \hat{a}) + \mu_{\mathbf{i}_{[j]}}([\hat{b}, \infty))\delta(o - \hat{b})$

  `// computes` $K-1$ `quantiles` $\mathbf{o_i}$ `and sorts the results`

  $\mathbf{o_i} \leftarrow \left(\texttt{quantile}(\hat{\mathbf{x}}_{\mathbf{i}}, \tilde{q}_{\mathbf{i}_{[1]}}, \varepsilon_{\mathbf{i}_{[1]}}/|\mathbf{i}|, \hat{\mu}_{\mathbf{i}_{[1]}}) \ , \cdots \ , \texttt{quantile}(\hat{\mathbf{x}}_{\mathbf{i}}, \tilde{q}_{\mathbf{i}_{[|\mathbf{i}|]}}, \varepsilon_{\mathbf{i}_{[|\mathbf{i}|]}}/|\mathbf{i}|, \hat{\mu}_{\mathbf{i}_{[|\mathbf{i}|]}})\right)$

  $\mathbf{o_i} \leftarrow$`sort(`$\mathbf{o_i}$`)`

  `// recursively computes remaining indices on the` K `intervals induced by` $\mathbf{o_i}$

  if $|\mathbf{j}| < K$ **then**

    $\mathbf{o} \leftarrow \mathbf{o}_i$

  else

    $\mathbf{o} \leftarrow$ `concat(recurse(`$\left(\mathbf{j}_{[1]}, \cdots, \mathbf{j}_{[\lceil |\mathbf{j}|/K \rceil - 1]}\right), \underline{q}, q_{\mathbf{i}_{[1]}}, \hat{a}, \mathbf{o}_{[1]}$`),`$\left(\mathbf{o}_{[1]}\right)$`)`

    for $j = 2, \dots, |\mathbf{i}|$ **do**

      $\mathbf{o} \leftarrow$`concat(`$\mathbf{o}$, `recurse(`$\left(\mathbf{j}_{[\lceil (j-1)|\mathbf{j}|/K \rceil + 1]}, \cdots, \mathbf{j}_{[\lceil j|\mathbf{j}|/K \rceil - 1]}\right), q_{\mathbf{i}_{[j-1]}}, q_{\mathbf{i}_{[j]}}, \mathbf{o}_{[j-1]}, \mathbf{o}_{[j]}$`))`

      $\mathbf{o} \leftarrow$`concat(`$\mathbf{o}$, $\left(\mathbf{o}_{[j]}\right)$`)`

    $\mathbf{o} \leftarrow$ `concat(`$\mathbf{o}$, `recurse(`$\left(\mathbf{j}_{[\lceil (K-1)|\mathbf{j}|/K \rceil + 1]}, \cdots, \mathbf{j}_{[|\mathbf{j}|]}\right), q_{\mathbf{i}_{[K-1]}}, \overline{q}, \mathbf{o}_{[K-1]}, \hat{b}$`))`

  Output: $\mathbf{o}$

Output: `recurse(`$\left(1, \cdots, m\right), 0, 1, -\infty, \infty$`)`

B Covariance estimation

B.1 Section 4.2 details

B.1.1 Sensitivity results

Lemma B.1. *The eigenvalues Λ of $\mathbf{X}\mathbf{X}^T/n - \mathbf{W} = \mathbf{U}\Lambda\mathbf{U}^T$ for $\mathbf{X} \in \mathbb{R}^{d\times n}$ with 1-bounded columns and symmetric $\mathbf{W} \in \mathbb{R}^{d\times d}$ has ℓ_1-sensitivity $2/n$, as does its trace norm $\|\Lambda\|_1 = \|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_{\mathrm{Tr}}$.*

Proof. Consider two datasets $\mathbf{X}, \tilde{\mathbf{X}} \in \mathbb{R}^{d\times n}$ that share the same first $n-1$ columns $\mathbf{Z} \in \mathbb{R}^{d\times n-1}$ but have different respective last columns $\mathbf{x}, \tilde{\mathbf{x}} \in \mathbb{R}^d$. For any vector $\mathbf{v} \in \mathbb{R}^d$ we have

$$\mathbf{v}^T(\mathbf{X}\mathbf{X}^T/n - \mathbf{W})\mathbf{v} = \mathbf{v}^T\mathbf{Z}\mathbf{Z}^T\mathbf{v}/n + \mathbf{v}^T\mathbf{x}\mathbf{x}^T\mathbf{v}/n - \mathbf{v}^T\mathbf{W}\mathbf{v} \geqslant \mathbf{v}^T(\mathbf{Z}\mathbf{Z}^T/n - \mathbf{W})\mathbf{v} \tag{57}$$

so for $\hat{\mathbf{U}}\hat{\Lambda}\hat{\mathbf{U}}^T = \mathbf{Z}\mathbf{Z}^T/n - \mathbf{W}$ we have

$$\Lambda_{[i]} \geqslant \hat{\Lambda}_{[i]} \ \forall \ i \in [d] \tag{58}$$

Thus

$$\|\Lambda - \hat{\Lambda}\|_1 = \mathrm{Tr}(\mathbf{X}\mathbf{X}^T/n - \mathbf{W}) - \mathrm{Tr}(\mathbf{Z}\mathbf{Z}^T/n - \mathbf{W}) = \mathrm{Tr}(\mathbf{x}\mathbf{x}^T/n) \leqslant 1/n \tag{59}$$

The same argument holds when replacing $\mathbf{X}$ by $\tilde{\mathbf{X}}$, so the result for the eigenvalues follows by the triangle inequality.

For the trace norm we have that

$$\begin{aligned}\left|\|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_{\mathrm{Tr}} - \|\mathbf{Z}\mathbf{Z}^T/n - \mathbf{W}\|_{\mathrm{Tr}}\right| = \left|\sum_{i=1}^d |\Lambda_{[i]}| - |\hat{\Lambda}_{[i]}|\right| &\leqslant \sum_{i=1}^d ||\Lambda_{[i]}| - |\hat{\Lambda}_{[i]}|| \\ &\leqslant \sum_{i=1}^d |\Lambda_{[i]} - \hat{\Lambda}_{[i]}| \leqslant 1/n\end{aligned} \tag{60}$$

where the second inequality holds trivially when $\Lambda_{[i]}$ and $\hat{\Lambda}_{[i]}$ have the same sign and otherwise the latter is negative (58) so we have $||\Lambda_{[i]}| - |\tilde{\Lambda}_{[i]}|| = |\Lambda_{[i]} + \tilde{\Lambda}_{[i]}| \leqslant |\Lambda_{[i]} - \tilde{\Lambda}_{[i]}|$, and the third is by Equation 59. This also holds when replacing $\mathbf{X}$ by $\tilde{\mathbf{X}}$, so the result follows by the triangle inequality. □

Claim B.1. *If $a_k, b_k \geqslant c_k \ \forall \ k \in [d]$ then $\sum_{k=1}^d (a_k - b_k)^2 \leqslant \left(\sum_{k=1}^d a_k - c_k\right)^2 + \left(\sum_{k=1}^d b_k - c_k\right)^2$.*

Proof. Note that this result is an easy corollary of [27, Fact 1], but for completeness:

$$\begin{aligned}&\left(\sum_{k=1}^d a_k - c_k\right)^2 + \left(\sum_{k=1}^d b_k - c_k\right)^2 \\ &\quad= \sum_{k=1}^d (a_k - c_k)^2 + (a_k - c_k)\sum_{j\neq k} a_j - c_j + \sum_{k=1}^d (b_k - c_k)^2 + (b_k - c_k)\sum_{j\neq k} b_j - c_j \\ &\quad\geqslant \sum_{k=1}^d a_k^2 - 2a_kc_k + c_k^2 + b_k^2 - 2b_kc_k + c_k^2 \\ &\quad= \sum_{k=1}^d (a_k - b_k)^2 + 2a_kb_k - 2a_kc_k - 2b_kc_k + 2c_k^2 \quad \geqslant \quad \sum_{k=1}^d (a_k - b_k)^2\end{aligned} \tag{61}$$

where the first inequality follows because $a_k - c_k, b_k - c_k \geqslant 0 \ \forall \ k$ and the second because the convex function $a_kb_k - a_kc_k - 2b_kc_k + c_k^2$ attains its minimum over $c_k \in (-\infty, \min\{a_k, b_k\}]$ at $\min\{a_k, b_k\}$. □

Lemma B.2. *The ℓ_2-sensitivities of $\mathbf{X}\mathbf{X}^T/n - \mathbf{W}$ and its eigenvalues Λ are both $\sqrt{2}/n$.*

Proof. The first result follows directly by [12, Lemma 3.2] For the second, consider two datasets $\mathbf{X}, \tilde{\mathbf{X}} \in \mathbb{R}^{d\times n}$ that share the same first $n-1$ columns $\mathbf{Z} \in \mathbb{R}^{d\times n-1}$ but have different respective last columns $\mathbf{x}, \tilde{\mathbf{x}} \in \mathbb{R}^d$. Applying Claim B.1, Equation 58, and Equation 59 yields

$$\|\Lambda - \tilde{\Lambda}\|_F^2 = \sum_{k=1}^{d} (\Lambda_{[i]} - \tilde{\Lambda}_{[i]})^2 \leqslant \left(\sum_{k=1}^{d} \Lambda_{[i]} - \hat{\Lambda}_{[i]}\right)^2 + \left(\sum_{k=1}^{d} \tilde{\Lambda}_{[i]} - \hat{\Lambda}_{[i]}\right)^2 \leqslant \frac{1}{n^2} + \frac{1}{n^2} = \frac{2}{n^2} \tag{62}$$

□

B.1.2 Proof of Lemma 4.1

Proof. Let $\mathbf{C} = \mathbf{X}\mathbf{X}^T/n$. Then by triangle inequality and the orthonormality of $\tilde{\mathbf{U}}$ we have

$$\begin{aligned}
\|\mathbf{C} - \mathbf{W} - \tilde{\mathbf{U}}\hat{\Lambda}\tilde{\mathbf{U}}^T\|_F &= \|\mathbf{C} - \mathbf{W} - \tilde{\mathbf{U}}\hat{\Lambda}\tilde{\mathbf{U}}^T\|_F \\
&= \|\mathbf{C} - \mathbf{W} - \tilde{\mathbf{U}}\Lambda\tilde{\mathbf{U}}^T - \tilde{\mathbf{U}}(\hat{\Lambda} - \Lambda)\tilde{\mathbf{U}}^T\|_F \\
&\leqslant \|\mathbf{C} - \mathbf{W} - \tilde{\mathbf{U}}\Lambda\tilde{\mathbf{U}}^T\|_F + \|\tilde{\mathbf{U}}(\hat{\Lambda} - \Lambda)\tilde{\mathbf{U}}^T\|_F \\
&\leqslant \sqrt{\|\mathbf{C} - \mathbf{W}\|_F^2 - 2\operatorname{Tr}((\mathbf{C} - \mathbf{W})(\tilde{\mathbf{U}}\Lambda\tilde{\mathbf{U}}^T)) + \|\tilde{\mathbf{U}}\Lambda\tilde{\mathbf{U}}^T\|_F^2} + \|\mathbf{z}\|_2 \\
&= \sqrt{2\sum_{j=1}^{d} \Lambda_{[j]}^2 - 2\operatorname{Tr}(\Lambda\tilde{\mathbf{U}}^T(\mathbf{C} - \mathbf{W})\tilde{\mathbf{U}})} + \|\mathbf{z}\|_2 \\
&= \sqrt{2\sum_{j=1}^{d} \Lambda_{[j]}^2 - 2\sum_{j=1}^{d} \Lambda_{[j]}\tilde{\mathbf{U}}_{[j]}^T(\mathbf{C} - \mathbf{W})\tilde{\mathbf{U}}_{[j]}} + \|\mathbf{z}\|_2 \\
&= \sqrt{2\sum_{j=1}^{d} \Lambda_{[j]}(\Lambda_{[j]} - \tilde{\mathbf{U}}_{[j]}^T(\mathbf{C} - \mathbf{W})\tilde{\mathbf{U}}_{[j]})} + \|\mathbf{z}\|_2
\end{aligned} \tag{63}$$

For $j \in [d]$ s.t. $\Lambda_{[j]} > 0$ let $\mathbf{u}_j = \arg\max_{\|\mathbf{u}\|_2=1, \tilde{\mathbf{U}}_{[j:d]}\mathbf{u}=\mathbf{0}} \mathbf{u}(\mathbf{C} - \mathbf{W})\mathbf{u}$. By the Courant-Fischer-Weyl min-max principle we have that

$$\Lambda_{[j]} = \min_{\mathbf{V}\in\mathbb{R}^{(d-j+1)\times d}} \max_{\|\mathbf{u}\|_2=1, \mathbf{V}\mathbf{u}=\mathbf{0}} \mathbf{u}^T(\mathbf{C} - \mathbf{W})\mathbf{u} \leqslant \max_{\|\mathbf{u}\|_2=1, \tilde{\mathbf{U}}_{[j:d]}\mathbf{u}=\mathbf{0}} \mathbf{u}^T(\mathbf{C} - \mathbf{W})\mathbf{u} = \mathbf{u}_j(\mathbf{C} - \mathbf{W})\mathbf{u}_j \tag{64}$$

Therefore

$$\begin{aligned}
\tilde{\mathbf{U}}_{[j]}^T(\mathbf{X}\mathbf{X}^T - \mathbf{W})\tilde{\mathbf{U}}_{[j]} &= \tilde{\mathbf{U}}_{[j]}^T(\mathbf{X}\mathbf{X}^T - \mathbf{W} + \mathbf{Z})\tilde{\mathbf{U}}_{[j]} - \tilde{\mathbf{U}}_{[j]}^T\mathbf{Z}\tilde{\mathbf{U}}_{[j]} \\
&\geqslant \tilde{\mathbf{U}}_{[j]}^T(\mathbf{X}\mathbf{X}^T - \mathbf{W} + \mathbf{Z})\tilde{\mathbf{U}}_{[j]} - |||\mathbf{Z}|||_\infty \\
&\geqslant \mathbf{u}_j(\mathbf{C} - \mathbf{W} + \mathbf{Z})\mathbf{u}_j - |||\mathbf{Z}|||_\infty \\
&\geqslant \mathbf{u}_j(\mathbf{C} - \mathbf{W})\mathbf{u}_j - 2|||\mathbf{Z}|||_\infty \geqslant \Lambda_{[j]} - 2|||\mathbf{Z}|||_\infty
\end{aligned} \tag{65}$$

Similarly, for $j \in [d]$ s.t. $\Lambda_{[j]} < 0$ let $\mathbf{u}_j = \arg\min_{\|\mathbf{u}\|_2=1, \tilde{\mathbf{U}}_{[1:j]}\mathbf{u}=\mathbf{0}} \mathbf{u}(\mathbf{C} - \mathbf{W})\mathbf{u}$. By the Courant-Fischer-Weyl min-max principle we have that

$$\Lambda_{[j]} = \max_{\mathbf{V}\in\mathbb{R}^{j\times d}} \min_{\|\mathbf{u}\|_2=1, \mathbf{V}\mathbf{u}=\mathbf{0}} \mathbf{u}^T(\mathbf{C} - \mathbf{W})\mathbf{u} \geqslant \min_{\|\mathbf{u}\|_2=1, \tilde{\mathbf{U}}_{[1:j]}\mathbf{u}=\mathbf{0}} \mathbf{u}^T(\mathbf{C} - \mathbf{W})\mathbf{u} = \mathbf{u}_j(\mathbf{C} - \mathbf{W})\mathbf{u}_j \tag{66}$$

Algorithm 6: `SeparateCov` with predictions (zCDP)

Input: data $\mathbf{X} \in \mathbb{R}^{d\times n}$, symmetric prediction matrix $\mathbf{W} \in \mathbb{R}^{d\times d}$, privacy parameter $\rho > 0$
$\mathbf{U}\Lambda\mathbf{U}^T \leftarrow \mathbf{X}\mathbf{X}^T/n - \mathbf{W}$
$\hat{\Lambda} \leftarrow \Lambda + \mathrm{diag}(\mathbf{z})$ where $\mathbf{z} \sim \mathcal{N}(\mathbf{0}_d, \frac{2}{\rho n^2})$ `// add prediction noise to error eigenvalues`
$\tilde{\mathbf{C}} \leftarrow \mathbf{X}\mathbf{X}^T/n + \mathbf{Z}$ for $\mathbf{Z}_{[i,j]} = \mathbf{Z}_{[j,i]} \sim \mathcal{N}\left(0, \frac{2}{\rho n^2}\right)$
$\tilde{\mathbf{U}}\hat{\Lambda}\tilde{\mathbf{U}}^T \leftarrow \tilde{\mathbf{C}} - \mathbf{W}$ `// get eigenvectors of noised prediction error`
Output: $\hat{\mathbf{C}} = \tilde{\mathbf{U}}\hat{\Lambda}\tilde{\mathbf{U}}^T + \mathbf{W}$ `// combine to estimate` $\mathbf{X}\mathbf{X}^T/n - \mathbf{W}$`, then add` $\mathbf{W}$

Therefore

$$\begin{aligned}
\tilde{\mathbf{U}}_{[j]}^T(\mathbf{C}-\mathbf{W})\tilde{\mathbf{U}}_{[j]} &= \tilde{\mathbf{U}}_{[j]}^T(\mathbf{C}-\mathbf{W}+\mathbf{Z})\tilde{\mathbf{U}}_{[j]} - \tilde{\mathbf{U}}_{[j]}^T\mathbf{Z}\tilde{\mathbf{U}}_{[j]} \\
&\leqslant \tilde{\mathbf{U}}_{[j]}^T(\mathbf{C}-\mathbf{W}+\mathbf{Z})\tilde{\mathbf{U}}_{[j]} + |||\mathbf{Z}|||_\infty \\
&\leqslant \mathbf{u}_j(\mathbf{C}-\mathbf{W}+\mathbf{Z})\mathbf{u}_j + |||\mathbf{Z}|||_\infty \\
&\leqslant \mathbf{u}_j(\mathbf{C}-\mathbf{W})\mathbf{u}_j + 2|||\mathbf{Z}|||_\infty \leqslant \Lambda_{[j]} + 2|||\mathbf{Z}|||_\infty
\end{aligned} \tag{67}$$

Substituting the bounds (65) and (67) in for the appropriate j terms in the summation in Equation 63 yields $\|\mathbf{C}-\mathbf{W}-\tilde{\mathbf{U}}\hat{\Lambda}\tilde{\mathbf{U}}^T\|_F \leqslant 2\|\mathbf{z}\|_2 + 2\sqrt{|||\mathbf{Z}|||_\infty \sum_{j=1}^d |\Lambda_{[j]}|} = 2\left(\|\mathbf{z}\|_2 + \sqrt{|||\mathbf{Z}|||_\infty \|\mathbf{C}-\mathbf{W}\|_{\mathrm{Tr}}}\right)$. □

B.2 zCDP guarantees for `SeparateCov` with predictions

Definition B.1 ([14])**.** *Algorithm $\mathcal{A}$ is ρ-zCDP if $D_\alpha(\mathcal{A}(\mathbf{X})||\mathcal{A}(\tilde{\mathbf{X}})) \leqslant \rho\alpha \;\forall\; \alpha > 1$ whenever $\mathbf{X}$ and $\tilde{\mathbf{X}}$ differ in a single element, where D_α is the α-Rényi divergence.*

Theorem B.1 ([14])**.** *If a query $q : \mathcal{X} \mapsto \mathbb{R}^d$ has ℓ_2-**sensitivity** $\max_{\mathbf{X}\sim\tilde{\mathbf{X}}} \|q(\mathbf{X}) - q(\tilde{\mathbf{X}})\|_2^2 \leqslant \Delta$ then the* **Gaussian mechanism***, which releases $q(\mathbf{X}) + \mathbf{z}$ for $\mathbf{z} \sim \mathcal{N}(\mathbf{0}_d, \frac{\Delta^2}{2\rho})$, is ρ-zCDP.*

Theorem B.2. *If $\mathbf{X}$ has columns bounded by 1 in ℓ_2-norm then Algorithm 6 is ρ-zCDP and w.p. $\geqslant 1-\beta$*

$$\|\hat{\mathbf{C}} - \mathbf{X}\mathbf{X}^T/n\|_F^2 \leqslant \tilde{\mathcal{O}}\left(\frac{d}{n^2\rho} + \frac{\|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_{\mathrm{Tr}}}{n}\sqrt{\frac{d}{\rho}}\right) \tag{68}$$

Proof. The privacy guarantee follows from the composition of two Gaussian mechanisms with the sensitivities of Lemma B.2. The utility guarantee is due to substituting Gaussian concentration from [27, Lemmas 6 & 7] into Lemma 4.1. □

Corollary B.1. *If $\mathbf{X}$ has columns bounded by 1 in ℓ_2-norm then Algorithm 6 with $\mathbf{W} = \mathbf{0}_{d\times d}$ returns w.p. $\geqslant 1-\beta$ an estimate $\hat{\mathbf{C}} \in \mathbb{R}^{d\times d}$ satisfying*

$$\|\hat{\mathbf{C}} - \mathbf{X}\mathbf{X}^T/n\|_F^2 \leqslant \tilde{\mathcal{O}}\left(\frac{d}{n^2\rho} + \min_{c\in\mathbb{R}} \frac{\|\mathbf{X}\mathbf{X}^T/n - c\mathbf{I}_d\|_{\mathrm{Tr}}}{n}\sqrt{\frac{d}{\rho}}\right) \tag{69}$$

Corollary B.2. *Pick $\lambda \in (0,1)$ and run Algorithm 6 with privacy $(1-\lambda)\rho$ and symmetric prediction matrix $\mathbf{W}$ if $\|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_{\mathrm{Tr}} + z \leqslant \|\mathbf{X}\mathbf{X}^T\|_{\mathrm{Tr}}/n$ and $\mathbf{0}_{d\times d}$, where $z \sim \mathcal{N}(0, \frac{1}{\lambda\rho n})$. This procedure is ρ-zCDP, $\tilde{\mathcal{O}}\left(\frac{\sqrt{d}}{n\sqrt{\rho}}\left(\frac{\sqrt{d}/n+1/\sqrt{n}}{\sqrt{\rho}} + \|\mathbf{X}\mathbf{X}^T/n\|_{\mathrm{Tr}}\right)\right)$-robust and $\tilde{\mathcal{O}}\left(\frac{\sqrt{d}}{n\sqrt{\rho}}\left(\frac{\sqrt{d}/n+1/\sqrt{n}}{\sqrt{\rho}}\right)\right)$-consistent.*

Proof. By Lemma B.2 the difference $\|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_{\mathrm{Tr}} - \|\mathbf{X}\mathbf{X}^T/n\|_{\mathrm{Tr}}$ has sensitivity $2/\sqrt{n}$, so the comparison of $\|\mathbf{X}\mathbf{X}^T - \mathbf{W}\|_{\mathrm{Tr}} + z$ and $\|\mathbf{X}\mathbf{X}^T/n\|_{\mathrm{Tr}}$ is equivalent to using the Gaussian mechanism with $\lambda\rho$-zCDP to estimate this difference and then taking the sign. Composing this with the privacy guarantee of Theorem B.2 yields ρ-zCDP. Since $\Pr\{|z| \geqslant 2\sqrt{\frac{1}{\lambda\rho n}\log\frac{2}{\beta}}\} \leqslant \beta/2$, the matrix $\mathbf{W}_z \in \{\mathbf{W}, \mathbf{0}_{d\times d}\}$ passed to Algorithm 6 satisfies $\|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}_z\|_{\mathrm{Tr}} \leqslant \min\{\|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_{\mathrm{Tr}}, \|\mathbf{X}\mathbf{X}^T/n\|_{\mathrm{Tr}} + 2\sqrt{\frac{1}{\lambda\rho n}\log\frac{2}{\beta}}$ w.p. $\geqslant 1 - \beta/2$. Applying the utility guarantee of Theorem B.2 w.p. $1 - \beta/2$ for constant $\lambda \in (0, 1)$ yields the result. □

B.3 `IterativeEigenvectorSampling` with predictions

Lemma B.3. *Given a dataset $\mathbf{X} \in \mathbb{R}^{d\times n}$ with 1-bounded columns, any orthonormal basis $\mathbf{P} \in \mathbb{R}^{k\times d}$, and a symmetric matrix $\mathbf{W} \in \mathbb{R}^{d\times d}$ the queries $\mathbf{u}^T\mathbf{P}\{\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\}_+\mathbf{P}^T\mathbf{u}$ and $\mathbf{u}^T\mathbf{P}\{\mathbf{M} - \mathbf{X}\mathbf{X}^T/n\}\mathbf{P}^T\mathbf{u}$—where $\{\mathbf{A}\}_+$ denotes taking only the components of $\mathbf{A}$ with positive eigenvalues—have sensitivity $2/n$.*

Proof. Consider two datasets $\mathbf{X}, \tilde{\mathbf{X}} \in \mathbb{R}^{d\times n}$ that share the same first $n-1$ columns $\mathbf{Z} \in \mathbb{R}^{d\times n-1}$ but have different respective last columns $\mathbf{x}, \tilde{\mathbf{x}} \in \mathbb{R}^d$. Let $\mathbf{P}_+$ and $\mathbf{Q}_+ \in \mathbb{R}^{d\times d}$ be projection matrices removing the negative components of $\mathbf{X}\mathbf{X}^T - \mathbf{W}$ and $\mathbf{Z}\mathbf{Z}^T - \mathbf{W}$, respectively. Then we have

$$\begin{aligned}
&\|\{\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\}^+ - \{\mathbf{Z}\mathbf{Z}^T/n - \mathbf{W}\}^+\|_2 \\
&\quad= \|\mathbf{P}_+(\mathbf{X}\mathbf{X}^T/n - \mathbf{W})\mathbf{P}_+ - \mathbf{Q}_+(\mathbf{Z}\mathbf{Z}^T/n - \mathbf{W})\mathbf{Q}_+\|_2 \\
&\quad= \max_{\|\mathbf{v}\|_2=1} \mathbf{v}^T\mathbf{P}_+(\mathbf{Z}\mathbf{Z}^T/n - \mathbf{W})\mathbf{P}_+\mathbf{v} + \mathbf{v}^T\mathbf{P}_+\mathbf{x}\mathbf{x}^T\mathbf{P}_+\mathbf{v}/n - \mathbf{v}^T\mathbf{Q}_+(\mathbf{Z}\mathbf{Z}^T/n - \mathbf{W})\mathbf{Q}_+\mathbf{v} \\
&\quad\leqslant 1/n + \max_{\|\mathbf{v}\|_2=1} \mathbf{v}^T\mathbf{Q}_+(\mathbf{Z}\mathbf{Z}^T/n - \mathbf{W})\mathbf{Q}_+\mathbf{v} - \mathbf{v}^T\mathbf{Q}_+(\mathbf{Z}\mathbf{Z}^T/n - \mathbf{W})\mathbf{Q}_+\mathbf{v}/n = 1/n
\end{aligned} \tag{70}$$

where the second equality follows by (57) and the definition of the spectral norm. The same argument holds when replacing $\mathbf{X}$ by $\tilde{\mathbf{X}}$, so we can bound the sensitivity by the triangle inequality:

$$\begin{aligned}
&|\mathbf{u}^T\mathbf{P}\{\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\}_+\mathbf{P}^T\mathbf{u} - \mathbf{u}^T\mathbf{P}\{\tilde{\mathbf{X}}\tilde{\mathbf{X}}^T/n - \mathbf{W}\}_+\mathbf{P}^T\mathbf{u}| \\
&\qquad\leqslant \|\mathbf{P}(\{\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\}^+ - \{\tilde{\mathbf{X}}\tilde{\mathbf{X}}^T/n - \mathbf{W}\}^+)\mathbf{P}^T\|_2 \\
&\qquad\leqslant \|\{\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\}^+ - \{\tilde{\mathbf{X}}\tilde{\mathbf{X}}^T/n - \mathbf{W}\}^+\|_2 \leqslant 2/n
\end{aligned} \tag{71}$$

Similarly, for $\mathbf{P}_-$ and $\mathbf{Q}_- \in \mathbb{R}^{d\times d}$ the projection matrices removing the negative components of $\mathbf{X}\mathbf{X}^T - \mathbf{W}$ and $\mathbf{Z}\mathbf{Z}^T - \mathbf{W}$, respectively, we have

$$\begin{aligned}
&\|\{\mathbf{W} - \mathbf{X}\mathbf{X}^T/n\}^+ - \{\mathbf{W} - \mathbf{Z}\mathbf{Z}^T/n\}^+\|_2 \\
&\quad= \|\mathbf{P}_-(\mathbf{W} - \mathbf{X}\mathbf{X}^T/n)\mathbf{P}_- - \mathbf{Q}_-(\mathbf{W} - \mathbf{Z}\mathbf{Z}^T/n)\mathbf{Q}_-\|_2 \\
&\quad= \max_{\|\mathbf{v}\|_2=1} \mathbf{v}^T\mathbf{Q}_-(\mathbf{W} - \mathbf{X}\mathbf{X}^T/n)\mathbf{Q}_-\mathbf{v} + \mathbf{v}^T\mathbf{Q}_-\mathbf{x}\mathbf{x}^T\mathbf{Q}_-\mathbf{v}/n - \mathbf{v}^T\mathbf{P}_-(\mathbf{W} - \mathbf{X}\mathbf{X}^T/n)\mathbf{P}_-\mathbf{v} \\
&\quad\leqslant 1/n + \max_{\|\mathbf{v}\|_2=1} \mathbf{v}^T\mathbf{P}_-(\mathbf{W} - \mathbf{X}\mathbf{X}^T/n)\mathbf{P}_-\mathbf{v} - \mathbf{v}^T\mathbf{P}_-(\mathbf{W} - \mathbf{X}\mathbf{X}^T/n - \mathbf{W})\mathbf{P}_-\mathbf{v} = 1/n
\end{aligned} \tag{72}$$

where the second equality follows by (57) and the definition of the spectral norm. The same argument holds when replacing $\mathbf{X}$ by $\tilde{\mathbf{X}}$, so as before we can obtain the sensitivity via the triangle inequality. □

Algorithm 7: `IterativeEigenvectorSampling` with predictions

Input: data matrix $\mathbf{X} \in \mathbb{R}^{d \times n}$, symmetric prediction $\mathbf{W} \in \mathbb{R}^{d \times d}$, privacy parameters $\varepsilon_0^{(\pm 1)}, \dots, \varepsilon_d^{(\pm 1)}$

initialize $\hat{\mathbf{C}} \leftarrow \mathbf{W}$

for $s, \mathbf{C} \in ((1, \{\mathbf{X}\mathbf{X}^T - \mathbf{W}\}_+), (-1, \{\mathbf{W} - \mathbf{X}\mathbf{X}^T\}_+))$ **do**

  `// run IterativeEigenvectorSampling [4] on` $\mathbf{C}$`, add` $s\times$ `the result to` $\hat{\mathbf{C}}$

  initialize $\mathbf{C}_1 \leftarrow \mathbf{C}$ and $\mathbf{P}_1 \leftarrow \mathbf{I}_d$

  $\mathbf{U}\Lambda\mathbf{U}^T \leftarrow \mathbf{C}$ `// get eigenvalues of` $\mathbf{C}$

  for *i=1,...,d* **do**

    $\lambda_i^{(s)} \leftarrow \Lambda_{[i]} + \mathrm{Lap}(2/\varepsilon_0^{(s)})$

    $\hat{\theta}_i^{(s)} \leftarrow \mathbf{P}_i^T \hat{\mathbf{u}}_i^{(s)}$ for $\hat{\mathbf{u}}_i^{(s)}$ sampled w.p. $\propto f_{\mathbf{C}_i}(\mathbf{u}) = \exp\left(\frac{\varepsilon_i^{(s)}}{4} \mathbf{u}^T \mathbf{C}_i \mathbf{u}\right)$

    set $\mathbf{P}_{i+1} \in \mathbb{R}^{(d-i)\times d}$ to be an orthonormal basis orthogonal to $\hat{\theta}_1^{(s)}, \dots, \hat{\theta}_i^{(s)}$

    $\mathbf{C}_{i+1} \leftarrow \mathbf{P}_{i+1} \mathbf{C} \mathbf{P}_{i+1}^T \in \mathbb{R}^{(d-i)\times(d-i)}$

  $\hat{\mathbf{C}} \leftarrow \hat{\mathbf{C}} + s \sum_{i=1}^d \hat{\lambda}_i^{(s)} \hat{\theta}_i^{(s)} \hat{\theta}_i^{(s)^T}$

Output: $\hat{\mathbf{C}}$

Theorem B.3. *Algorithm 7 preserves* $\left(\sum_{s\in\{\pm1\}} \sum_{i=0}^d \varepsilon_i^{(s)}\right)$*-DP and the output* $\hat{\mathbf{C}}$ *satisfies w.p.* $\geqslant 1-\beta$

$$\|\mathbf{X}\mathbf{X}^T/n - \hat{\mathbf{C}}\|_F^2 \leqslant \tilde{\mathcal{O}}\left(\frac{d}{n}\left(\sum_{s\in\{\pm1\}} \frac{1}{\varepsilon_0^{(s)^2} n} + \sum_{i=1}^d \frac{|\Lambda_{[i]}|}{\varepsilon_i^{(\mathbf{S}_{[i]})}}\right)\right) \tag{73}$$

where $\Lambda_{[i]}$ *is the matrix of eigenvalues of* $\mathbf{X}\mathbf{X}^T/n - \mathbf{W}$*,* $\mathbf{S}_{[i]}$ *is the matrix of its signs, and* $\tilde{\mathcal{O}}$ *hides logarithmic factors in* d*,* $\|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_2$*, and* β*.*

Proof. The privacy result follows from Lemma B.3 applied to Algorithm 7's release of $\lambda_i^{(s)}$ and $\hat{\mathbf{u}}_i^{(s)}$ using the Laplace and Exponential mechanisms, respectively, followed by basic composition. For utility, since $\mathbf{X}\mathbf{X}^T/n - \hat{\mathbf{C}} = \mathbf{X}\mathbf{X}^T - \mathbf{W} - \hat{\mathbf{C}} + \mathbf{W} = \{\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\}_+ - \{\hat{\mathbf{C}} - \mathbf{W}\}_+ - \{\mathbf{W} - \mathbf{X}\mathbf{X}^T/n\}_+ + \{\mathbf{W} - \hat{\mathbf{C}}\}_+$ we have by the triangle inequality, the fact that $(a+b)^2 \leqslant 2(a^2+b^2)\ \forall\ a, b \in \mathbb{R}$, and the utility guarantee (squared and normalized by n) of `IterativEigenvectorSampling` [4, Theorem 1] applied to $\{\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\}_+$ and $\{\mathbf{W} - \mathbf{X}\mathbf{X}^T/n\}_+$ that

$$\begin{aligned}
\|\mathbf{X}\mathbf{X}^T/n - \hat{\mathbf{C}}\|_F^2 &\leqslant 2\|\{\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\}_+ - \{\hat{\mathbf{C}} - \mathbf{W}\}_+\|_F^2 + 2\|\{\mathbf{W} - \mathbf{X}\mathbf{X}^T/n\}_+ - \{\mathbf{W} - \hat{\mathbf{C}}\}_+\|_F^2 \\
&\leqslant \tilde{\mathcal{O}}\left(d\left(\frac{1}{(\varepsilon_0^{(1)} n)^2} + \sum_{i=1}^d \frac{\max\{\Lambda_{[i]}, 0\}}{\varepsilon_i^{(1)} n} + \frac{1}{(\varepsilon_0^{(-1)} n)^2} + \sum_{i=1}^d \frac{\max\{-\Lambda_{[i]}, 0\}}{\varepsilon_i^{(-1)} n}\right)\right) \\
&= \tilde{\mathcal{O}}\left(d\left(\sum_{s\in\{\pm1\}} \frac{1}{(\varepsilon_0^{(s)} n)^2} + \sum_{i=1}^d \frac{|\Lambda_{[i]}|}{\varepsilon_i^{(S_{[i]})} n}\right)\right)
\end{aligned} \tag{74}$$

□

Corollary B.3. *Suppose for $s \in \{\pm 1\}$ we set $\varepsilon_0^{(s)} = \varepsilon/4$ and $\varepsilon_i^{(s)} = \frac{\varepsilon}{4d}\ \forall\ i \in [d]$, where $\varepsilon > 0$ is the overall privacy budget. Then w.p. $1-\beta$ the output $\hat{\mathbf{C}}$ of Algorithm 7 satisfies*

$$\|\mathbf{X}\mathbf{X}^T/n - \hat{\mathbf{C}}\|_F^2 \leqslant \tilde{\mathcal{O}}\left(\frac{d}{\varepsilon n}\left(\frac{1}{\varepsilon n} + d\|\mathbf{X}\mathbf{X}^T/n - \mathbf{W}\|_{\mathrm{Tr}}\right)\right) \tag{75}$$

C Data release

Proof of Lemma 4.2. Follow the proof of the original [36] but replace Fact A.3 by $\Psi_0 \leqslant D_{KL}(\frac{\mathbf{x}}{n}||\mathbf{w})$, upper bound the square of the result by twice the sum of the squares of the two terms, and obtain guarantees w.p. $\geqslant 1-\beta$ by solving $\frac{2m}{|Q|^c} = \beta$ for c and substituting the solution into the bound. □

D Online learning

D.1 Online-to-batch conversion

Theorem D.1. *Suppose an online algorithm sees a sequence $\ell(\cdot;\mathbf{X}_1),\dots,\ell(\cdot;\mathbf{X}_T) : \Theta \mapsto [0,B]$ of convex losses whose data $\mathbf{X}_1,\dots,\mathbf{X}_T$ are drawn i.i.d. from some distribution $\mathcal{D}$, and let $\theta_1,\dots,\theta_T$ be its predictions. If $\max_{\theta\in\Theta}\sum_{t=1}^T \ell(\theta_t;\mathbf{X}_t) - \ell(\theta;\mathbf{X}_t) \leqslant R_T$, $\hat{\theta} = \frac{1}{T}\sum_{t=1}^T \theta_t$, and $T = \Omega\left(T_\alpha + \frac{B^2}{\alpha^2}\log\frac{1}{\beta'}\right)$ for $T_\alpha = \min_{2R_T\leqslant T\alpha} T$, then w.p. $\geqslant 1-\beta'$*

$$\mathbb{E}_{\mathbf{X}\sim\mathcal{D}}\ell(\hat{\theta};\mathbf{X}) \leqslant \min_{\theta\in\Theta}\mathbb{E}_{\mathbf{X}\sim\mathcal{D}}\ell(\theta;\mathbf{X}) + \alpha \tag{76}$$

Proof. This is a formalization of a standard procedure; we follow the argument in [44, Lemma A.1]. Applying Jensen's inequality, [17, Proposition 1], the assumption that regret is $\leqslant R_T$, and Hoeffding's inequality yields

$$\begin{aligned}
\mathbb{E}_{\mathbf{X}\sim\mathcal{D}}\ell(\hat{\theta};\mathbf{X}) \leqslant \frac{1}{T}\sum_{t=1}^T \mathbb{E}_{\mathbf{X}\sim\mathcal{D}}\ell(\theta_t;\mathbf{X}) &\leqslant \frac{1}{T}\sum_{t=1}^T \ell(\theta_t;\mathbf{X}_t) + B\sqrt{\frac{2}{T}\log\frac{2}{\beta'}} \\
&\leqslant \min_{\theta\in\Theta}\frac{1}{T}\sum_{t=1}^T \ell(\theta;\mathbf{X}_t) + \frac{R_T}{T} + B\sqrt{\frac{2}{T}\log\frac{2}{\beta'}} \\
&\leqslant \min_{\theta\in\Theta}\mathbb{E}_{\mathbf{X}\sim\mathcal{D}}\ell(\theta;\mathbf{X}) + \frac{R_T}{T} + 2B\sqrt{\frac{2}{T}\log\frac{2}{\beta'}}
\end{aligned} \tag{77}$$

w.p. $\geqslant 1-\beta'$. Substituting the lower bound on T yields the result. □

D.2 Negative log-inner-product losses

For functions of the form $f_t(\mu) = -\log\int_a^b s_t(o)\mu(o)do$, [7] showed $\tilde{\mathcal{O}}(T^{3/4})$ regret for the case $s_t(o) \in \{0,1\}\ \forall\ o \in [a,b]$ using a variant of exponentiated gradient with a dynamic discretization. Notably their algorithm can be extended to (non-privately) learn $-\log\Psi_q^*(\mathbf{x}_t,\mu)$, since s_t in this case is one on the optimal interval and zero elsewhere. However, the changing discretization and dependence of the analysis on the range of s_t suggests it may be difficult to privatize their approach. The discretized form $-\log\langle\mathbf{s}_t,\mathbf{w}\rangle$ is more heavily studied, arising in portfolio management [21]. It enjoys the exp-concavity property, leading to $\mathcal{O}(d\log T)$ regret using the EWOO method [37]. However, EWOO requires maintaining and sampling from a distribution defined by a product of inner products, which is inefficient and similarly difficult to privatize. Other algorithms, e.g. adaptive

FTAL [37], also attain logarithmic regret for exp-concave functions, but the only private variant we know of is non-adaptive and only guarantees $\mathcal{O}(\sqrt{T})$-regret for non-strongly-convex losses [69]. The adaptivity, which is itself data-dependent, seems critical for taking advantage of exp-concavity.

Lemma D.1. *If* $f_t(\mu_{\mathbf{W}}) = -\log \sum_{i=1}^m \frac{1/m}{\langle \mathbf{s}_{t,i}, \mathbf{W}_{[i]} \rangle}$ *for* $\mathbf{s}_{t,i} \in \mathbb{R}^d_{\geqslant 0}$ *then* $\|\nabla_{\mathbf{W}} f_t(\mu_{\mathbf{W}})\|_1 \leqslant d/\gamma \;\forall\; \mathbf{W} \in \triangle_d^m$ *s.t.* $\mathbf{W}_{[i,j]} \geqslant \gamma/d \;\forall\; i,j$ *for some* $\gamma \in (0,1]$.

Proof.

$$\begin{aligned}\|\nabla_{\mathbf{W}} f_t(\mu_{\mathbf{W}})\|_1 = \sum_{i=1}^m \|\nabla_{\mathbf{W}_{[i]}} f_t(\mu_{\mathbf{W}})\|_1 &= \left(\sum_{i=1}^m \frac{1}{\langle \mathbf{s}_{t,i}, \mathbf{W}_{[i]} \rangle}\right)^{-1} \sum_{i=1}^m \sum_{j=1}^d \frac{\mathbf{s}_{t,i[j]}}{\langle \mathbf{s}_{t,i}, \mathbf{W}_{[i]} \rangle^2} \\ &\leqslant \left(\sum_{i=1}^m \frac{1}{\langle \mathbf{s}_{t,i}, \mathbf{W}_{[i]} \rangle}\right)^{-1} \sum_{i=1}^m \frac{1}{\langle \mathbf{s}_{t,i}, \mathbf{W}_{[i]} \odot \mathbf{W}_{[i]} \rangle} \leqslant d/\gamma\end{aligned} \tag{78}$$

where the first inequality follows by Sedrakyan's inequality and the second by $\mathbf{W}_{[i,j]} \geqslant \gamma/d$. □

D.2.1 Proof of Lemma 6.1 for $m > 1$

Proof. Let $\tilde{\mathbf{x}}_t$ be a neighboring dataset of $\mathbf{x}_t$ constructed by adding or removing a single element, and let $\tilde{U}_t$ be the corresponding loss function. We note that changing from $\mathbf{x}_t$ to $\tilde{\mathbf{x}}_t$ changes the value of $\operatorname{Gap}_{q_i}(\mathbf{x}_t, o)$ at any point $o \in [a,b]$ by at most ± 1 and so the value of the exponential score at any point $o \in [a,b]$ is changed by at most a multiplicative factor $\exp(-\varepsilon_i/2)$ in either direction. Therefore

$$\begin{aligned}\tilde{\mathbf{s}}_{t,i[j]} &= \int_{a+\frac{b-a}{d}(j-1)}^{a+\frac{b-a}{d}j} \exp(-\varepsilon_i \operatorname{Gap}_{q_i}(\tilde{\mathbf{x}}_t, o)/2) do \\ &\in \exp(\pm\varepsilon_i/2) \int_{a+\frac{b-a}{d}(j-1)}^{a+\frac{b-a}{d}j} \exp(-\varepsilon_i \operatorname{Gap}_{q_i}(\mathbf{x}_t, o)/2) do = \exp(\pm\varepsilon_i/2)\mathbf{s}_{t,i[j]}\end{aligned} \tag{79}$$

where $\pm$ indicates the interval between values.

$$\begin{aligned}&\|\nabla_{\mathbf{W}} U_t(\mathbf{W}) - \nabla_{\mathbf{W}} \tilde{U}_t(\mathbf{W})\|_F \\ &= \sqrt{\sum_{i=1}^m \sum_{j=1}^d \left(\left(\sum_{i'=1}^m \frac{1}{\langle \mathbf{s}_{t,i'}, \mathbf{W}_{[i']} \rangle}\right)^{-1} \frac{\mathbf{s}_{t,i[j]}}{\langle \mathbf{s}_{t,i}, \mathbf{W}_{[i]} \rangle^2} - \left(\sum_{i'=1}^m \frac{1}{\langle \tilde{\mathbf{s}}_{t,i'}, \mathbf{W}_{[i']} \rangle}\right)^{-1} \frac{\tilde{\mathbf{s}}_{t,i[j]}}{\langle \tilde{\mathbf{s}}_{t,i}, \mathbf{W}_{[i]} \rangle^2} \right)^2} \\ &= \left(\sum_{i'=1}^m \frac{1}{\langle \mathbf{s}_{t,i'}, \mathbf{W}_{[i']} \rangle}\right)^{-1} \sqrt{\sum_{i=1}^m \sum_{j=1}^d \left(\frac{\mathbf{s}_{t,i[j]}}{\langle \mathbf{s}_{t,i}, \mathbf{W}_{[i]} \rangle^2} - \frac{\tilde{\mathbf{s}}_{t,i[j]}}{\langle \tilde{\mathbf{s}}_{t,i}, \mathbf{W}_{[i]} \rangle^2} \frac{\sum_{i'=1}^m \frac{1}{\langle \mathbf{s}_{t,i'}, \mathbf{W}_{[i']} \rangle}}{\sum_{i'=1}^m \frac{1}{\langle \tilde{\mathbf{s}}_{t,i'}, \mathbf{W}_{[i']} \rangle}} \right)^2} \\ &= \left(\sum_{i'=1}^m \frac{1}{\langle \mathbf{s}_{t,i'}, \mathbf{W}_{[i']} \rangle}\right)^{-1} \sqrt{\sum_{i=1}^m \sum_{j=1}^d \frac{\mathbf{s}_{t,i[j]}^2}{\langle \mathbf{W}_{t,i}, \mathbf{W}_{[i]} \rangle^4} \left(1 - \frac{\langle \mathbf{g}_{t,i}, \mathbf{x}_{[i]} \rangle^2}{\langle \tilde{\mathbf{s}}_{t,i}, \mathbf{W}_{[i]} \rangle^2} \frac{\sum_{i'=1}^m \frac{\tilde{\mathbf{s}}_{t,i[j]}}{\langle \mathbf{s}_{t,i'}, \mathbf{W}_{[i']} \rangle}}{\sum_{i'=1}^m \frac{\mathbf{s}_{t,i[j]}}{\langle \tilde{\mathbf{s}}_{t,i'}, \mathbf{W}_{[i']} \rangle}} \right)^2} \\ &\leqslant \left(\sum_{i'=1}^m \frac{1}{\langle \mathbf{s}_{t,i'}, \mathbf{W}_{[i']} \rangle}\right)^{-1} \sum_{i=1}^m \sum_{j=1}^d \frac{\mathbf{s}_{t,i[j]}}{\langle \mathbf{s}_{t,i}, \mathbf{W}_{[i]} \rangle^2} |1 - \kappa_{i,j}| \leqslant \frac{d}{\gamma} \max_{i,j} |1 - \kappa_{i,j}|\end{aligned} \tag{80}$$

where we have

$$\kappa_{i,j} = \frac{\langle \mathbf{s}_{t,i}, \mathbf{W}_{[i]} \rangle^2}{\langle \tilde{\mathbf{s}}_{t,i}, \mathbf{x}_{[i]} \rangle^2} \frac{\sum_{i'=1}^{m} \frac{\tilde{\mathbf{s}}_{t,i[j]}}{\langle \mathbf{s}_{t,i'}, \mathbf{W}_{[i']} \rangle}}{\sum_{i'=1}^{m} \frac{\mathbf{s}_{t,i[j]}}{\langle \tilde{\mathbf{s}}_{t,i'}, \mathbf{W}_{[i']} \rangle}} \in \frac{\langle \mathbf{s}_{t,i}, \mathbf{W}_{[i]} \rangle^2}{\langle \mathbf{s}_{t,i}, \mathbf{W}_{[i]} \rangle^2 \exp(\pm\varepsilon_i)} \frac{\sum_{i'=1}^{m} \frac{\mathbf{s}_{t,i[j]} \exp(\pm \frac{\varepsilon_{i'}}{2})}{\langle \mathbf{s}_{t,i'}, \mathbf{W}_{[i']} \rangle}}{\sum_{i'=1}^{m} \frac{\mathbf{s}_{t,i[j]}}{\langle \mathbf{s}_{t,i'}, \mathbf{W}_{[i']} \rangle \exp(\pm \frac{\varepsilon_{i'}}{2})}} = \exp(\pm 2 \max_i \varepsilon_i) \tag{81}$$

Substituting into the previous inequality and taking the minimum with the ℓ_1 bound on the gradient of the losses from Lemma D.1 yields the result. □

D.2.2 Proof of Theorem 6.2

Proof. For set of γ-robust priors ρ s.t. $\rho_{[i]} = \min\{1-\gamma+\lambda, 1\}\mu_{[i]} + \frac{\max\{\gamma-\lambda,0\}}{b-a}$ and $\mathbf{W} \in \triangle_d^m$ s.t. $\mathbf{W}_{[i,j]} = \frac{b-a}{d} \int_{a+\frac{b-a}{d}(j-1)}^{a+\frac{b-a}{d}j} \rho_{[i]}(o) do$ we can divide the regret into three components:

$$\sum_{t=1}^{T} U_{\mathbf{x}_t}^{(\varepsilon)}(\mu_{\mathbf{W}_t}) - U_{\mathbf{x}_t}^{(\varepsilon)}(\mu) = \sum_{t=1}^{T} U_{\mathbf{x}_t}^{(\varepsilon)}(\mu_{\mathbf{W}_t}) - U_{\mathbf{x}_t}^{(\varepsilon)}(\mu_{\mathbf{W}}) + \sum_{t=1}^{T} U_{\mathbf{x}_t}^{(\varepsilon)}(\mu_{\mathbf{W}}) - U_{\mathbf{x}_t}^{(\varepsilon)}(\rho) + \sum_{t=1}^{T} U_{\mathbf{x}_t}^{(\varepsilon)}(\rho) - U_{\mathbf{x}_t}^{(\varepsilon)}(\mu) \tag{82}$$

The first summation is the regret of DP-FTRL with regularizer ϕ, which is strongly convex w.r.t. $\|\cdot\|_1$. The Gaussian width of its unit ball is $2\sqrt{\log(md)}$, by Lemma D.1 the losses are $\frac{d}{\gamma}$-Lipschitz w.r.t. $\|\cdot\|_1$, and by Lemma 6.1 the ℓ_2-sensitivity is $\Delta_2 = \frac{d}{\gamma}\min\{2, e^{\tilde{\varepsilon}_m} - 1\} \leqslant \frac{2d}{\gamma}\min\{1, \tilde{\varepsilon}_m\}$, so applying Theorem 6.1 yields the bound $\frac{m^2 \log d}{\eta} + \frac{\eta d^2 T}{\gamma^2}\left(1 + \left(4\sqrt{\log d} + 2\sqrt{2\log\frac{T}{\beta'}}\right)\sigma\sqrt{\lceil \log_2 T \rceil}\min\{1, \varepsilon\}\right)$. The second summation is a sum over the errors due to discretization, where we have

$$\begin{aligned}
\sum_{t=1}^{T} U_{\mathbf{x}_t}^{(\varepsilon)}(\mu_{\mathbf{W}}) - U_{\mathbf{x}_t}^{(\varepsilon)}(\rho) &= \sum_{t=1}^{T} \log \sum_{i=1}^{m} \langle \mathbf{s}_{t,i}, \mathbf{W}_{[i]} \rangle^{-1} - \log \sum_{i=1}^{m} \frac{1}{\int_a^b \exp(-\varepsilon_i \operatorname{Gap}_{q_i}(\mathbf{x}_t, o)/2)\rho_{[i]}(o) do} \\
&\leqslant \sum_{t=1}^{T} \sum_{i=1}^{m} \frac{\int_a^b \exp(-\frac{\varepsilon_i}{2}\operatorname{Gap}_{q_i}(\mathbf{x}_t, o))\rho_{[i]}(o) do - \langle \mathbf{s}_{t,i}, \mathbf{W}_{[i]} \rangle}{\langle \mathbf{s}_{t,i}, \mathbf{W}_{[i]} \rangle} \\
&\leqslant \sum_{t=1}^{T} \sum_{i=1}^{m} \frac{\sum_{j=1}^{d} \int_{a+\frac{b-a}{d}(j-1)}^{a+\frac{b-a}{d}j} \exp(-\frac{\varepsilon_i}{2}\operatorname{Gap}_{q_i}(\mathbf{x}_t, o))(\rho_{[i]}(o) - \mu_{\mathbf{W}_{[i]}}(o)) do}{\gamma\psi_{\mathbf{x}_t}/(b-a)} \\
&\leqslant \sum_{t=1}^{T} \sum_{i=1}^{m} \frac{\sum_{j=1}^{d} \int_{a+\frac{b-a}{d}(j-1)}^{a+\frac{b-a}{d}j} |\rho_{[i]}(o) - \rho_{[i]}(o_{i,j})| do}{\gamma\psi_{\mathbf{x}_t}/(b-a)} \leqslant \frac{VmT}{\gamma d\bar{\psi}}(b-a)^3
\end{aligned} \tag{83}$$

where the first inequality follows by concavity, the second by using the definition of $\mathbf{W}$ to see that $\langle \mathbf{s}_{t,i}, \mathbf{W}_{[i]} \rangle = \int_a^b \exp(-\frac{\varepsilon_i}{2}\operatorname{Gap}_{q_i}(\mathbf{x}_t, o))\mu_{\mathbf{W}_{[i]}}(o) do \geqslant \frac{\gamma\psi_{\mathbf{x}_t}}{b-a}$, the third by Hölder's inequality and the mean value theorem for some $o_{i,j} \in (a + \frac{b-a}{d}(j-1), a + \frac{b-a}{d}j)$, and the fourth by the Lipschitzness of $\rho_{[i]} \in \mathcal{F}_{V,d}^{(\gamma)}$. The third summation is a sum over the errors due to γ-robustness, with the result following by $U_{\mathbf{x}_t}^{(\varepsilon)}(\rho) - U_{\mathbf{x}_t}^{(\varepsilon)}(\mu) \leqslant U_{\mathbf{x}_t}^{(\varepsilon)}(\mu) - \log(1-\max\{\gamma-\lambda,0\}) - U_{\mathbf{x}_t}^{(\varepsilon)}(\mu) \leqslant 2\max\{\gamma-\lambda,0\}\log 2$. □

D.2.3 Settings of γ and d for Corollary 6.1

1. λ-robust and discrete $\mu_{[i]} \in \mathcal{F}_{0,d}^{(\lambda)}$: $\gamma = \lambda$
2. λ-robust and V-Lipschitz $\mu_{[i]} \in \mathcal{F}_{V,1}^{(\lambda)}$: $\gamma = \lambda$ and $d = \left\lceil \sqrt{\frac{V(b-a)^3}{\bar{\psi}} \sqrt{\left(1 + \frac{\min\{1, \tilde{\varepsilon}_m\}}{\varepsilon'}\right) T}} \right\rceil$
3. discrete $\mu_{[i]} \in \mathcal{F}_{0,d}$: $\gamma = \sqrt{md} \sqrt[4]{\frac{1+\min\{1,\tilde{\varepsilon}_m\}/\varepsilon'}{T}}$
4. V-Lipschitz $\mu_{[i]} \in \mathcal{F}_{V,1}$: $\gamma = \sqrt{m} \sqrt[4]{\frac{V(b-a)^3}{\bar{\psi}}} \sqrt[8]{\frac{1+\min\{1,\tilde{\varepsilon}_m\}/\varepsilon'}{T}}$ and $d = \left\lceil \sqrt{\frac{V(b-a)^3}{\bar{\psi}} \sqrt{\left(1 + \frac{\min\{1, \tilde{\varepsilon}_m\}}{\varepsilon'}\right) T}} \right\rceil$

D.3 Data release

Lemma D.2. *For $\mathbf{w} \in \triangle_d$ s.t. $\mathbf{w}_{[i]} \geqslant \gamma/d \; \forall \; i \in [n]$ the gradient $\nabla_{\mathbf{w}} D_{KL}(\frac{\mathbf{x}}{n} || \mathbf{w})$ of the KL divergence w.r.t. its second argument is bounded in ℓ_∞-norm by d/γ and has ℓ_2-sensitivity $\frac{d\sqrt{2}}{\gamma n}$.*

Proof. We have $\nabla_{\mathbf{w}} D_{KL}(\mathbf{x}/n||\mathbf{w}) = -\nabla_{\mathbf{w}} \langle \mathbf{x}/n, \log \mathbf{w} \rangle = \frac{\mathbf{x}}{n\mathbf{w}}$ so since $\mathbf{w}_{[i]} \geqslant \gamma/d$ we have that $\|\nabla_{\mathbf{w}} D_{KL}(\mathbf{x}/n||\mathbf{w})\|_\infty = \left\|\frac{\mathbf{x}}{n\mathbf{w}}\right\|_\infty \leqslant \frac{d \max_i \mathbf{x}_{[i]}}{\gamma n} \leqslant d/\gamma$ Furthermore, for neighboring datasets $\mathbf{x}$ and $\tilde{\mathbf{x}}$ we have

$$\|\nabla_{\mathbf{w}} D_{KL}(\mathbf{x}/n||\mathbf{w}) - \nabla_{\mathbf{w}} D_{KL}(\tilde{\mathbf{x}}/n||\mathbf{w})\|_2 = \left\| \frac{\mathbf{x}}{n\mathbf{w}} - \frac{\tilde{\mathbf{x}}}{n\mathbf{w}} \right\|_2 \leqslant \frac{d\sqrt{2}}{\gamma n} \tag{84}$$

□

Lemma D.3. *For $\mathbf{w} \in \triangle_d$ s.t. $\mathbf{w}_{[i]} \geqslant \gamma/d \; \forall \; i \in [n]$ the gradient $\nabla_\theta \mathbb{E}_{m\sim\theta} U_t(\mathbf{w}, m)$ has ℓ_2-sensitivity at most $7\pi \log \frac{d}{\gamma}$.*

Proof. For any $\mathbf{x}_t$ and neighboring $\tilde{\mathbf{x}}_t$ that replaces one element we have

$$|D_{KL}(\mathbf{x}_t/n_t||\mathbf{w}) - D_{KL}(\tilde{\mathbf{x}}_t/n_t||\mathbf{w})| = |\langle \mathbf{x}_t - \tilde{\mathbf{x}}_t, \log \mathbf{w} \rangle|/n_t \leqslant \frac{2}{n_t} \log \frac{d}{\gamma} \tag{85}$$

Therefore

$$\begin{aligned} \|\nabla_\theta \mathbb{E}_{m\sim\theta} U_t(\mathbf{w}, m) - \nabla_\theta \mathbb{E}_{m\sim\theta} \tilde{U}_t(\mathbf{w}, m)\|_2 &\leqslant 8 n_t \sqrt{\sum_{m=1}^{\infty} \left(\frac{D_{KL}(\mathbf{x}_t/n_t||\mathbf{w}) - D_{KL}(\tilde{\mathbf{x}}_t/n_t||\mathbf{w})}{m} \right)^2} \\ &\leqslant 7\pi \log \frac{d}{\gamma} \end{aligned} \tag{86}$$

□

D.4 Proof of Theorem 6.4

Proof. Let $M = \left\lceil \sqrt[3]{\frac{\varepsilon^2 N^2 \log \frac{d}{\gamma}}{16 \log^2 \frac{2|Q|}{\beta}}} \right\rceil$ and note that the m minimizing $\sum_{t=1}^{T} U_t(\mathbf{w}_t, m)$ is in $[M]$ and also

$$\begin{aligned}\max_{t,m} U_t(\mathbf{w}_t, m) &\leqslant 8N \log \frac{d}{\gamma} + 54 \log^{\frac{2}{3}} \frac{d}{\gamma} ((N^2/\varepsilon)^{\frac{2}{3}} + 1/\varepsilon^2) \left(\log^2 \frac{\varepsilon N \log \frac{d}{\gamma}}{\beta} + \log^4 |Q| \right) \\ &= \tilde{\mathcal{O}} \left(\frac{N^{\frac{4}{3}} \log^4 \frac{|Q|}{\beta}}{\min\{1, \varepsilon^2\}} \log \frac{d}{\gamma} \right)\end{aligned} \tag{87}$$

Letting $A = \mathcal{O}\left(\log \frac{d}{\gamma}\right)$ and $B = \tilde{\mathcal{O}}\left(\frac{N^{\frac{4}{3}} \log^4 \frac{|Q|}{\beta}}{\min\{1, \varepsilon^2\}} \log \frac{d}{\gamma} \right)$ we have

$$\begin{aligned}\sum_{t=1}^{T} U_t(\mathbf{w}_t, m_t) &\leqslant \frac{\log M}{\eta_\theta} + \eta_\theta \left(B \left(B + \left(2\sqrt{\log M} + \sqrt{2 \log \frac{3T}{\beta'}} \right) \right) \sigma_\theta A \sqrt{\lceil \log_2 T \rceil} \right) T \\ &\quad + B \sqrt{\frac{T}{2} \log \frac{3}{\beta'}} + \min_{\theta \in \Theta} \sum_{t=1}^{T} U_t(\mathbf{w}_t, \theta) \\ &\leqslant \frac{\log M}{\eta_\theta} + \eta_\theta \left(B \left(B + \left(2\sqrt{\log M} + \sqrt{2 \log \frac{3T}{\beta'}} \right) \right) \sigma_\theta A \sqrt{\lceil \log_2 T \rceil} \right) T \\ &\quad + B \sqrt{\frac{T}{2} \log \frac{3}{\beta'}} + \min_{m \in [M]} \sum_{t=1}^{T} \frac{8 n_t}{m} D_{KL} \left(\frac{\mathbf{x}_t}{n_t} \middle\| \mathbf{w}_t \right) + \frac{16 m^2}{\varepsilon^2 n_t} \left(3 \log \frac{2m}{\beta} + 2 \log^2 |Q| \right)^2 \\ &\leqslant \frac{\log M}{\eta_\theta} + \eta_\theta \left(B \left(B + \left(2\sqrt{\log M} + \sqrt{2 \log \frac{3T}{\beta'}} \right) \right) \sigma_\theta A \sqrt{\lceil \log_2 T \rceil} \right) T \\ &\quad + 8 \left(\frac{\log d}{\eta_{\mathbf{w}}} + \eta_{\mathbf{w}} \frac{Nd}{\gamma} \left(\frac{Nd}{\gamma} + \left(2\sqrt{\log d} + \sqrt{2 \log \frac{3T}{\beta'}} \right) \sigma_{\mathbf{w}} \frac{d\sqrt{2}}{\gamma} \sqrt{\lceil \log_2 T \rceil} \right) T \right) \\ &\quad + B \sqrt{\frac{T}{2} \log \frac{3}{\beta'}} + \min_{m > 0, \mathbf{w}_{[i]} \geqslant \gamma/d} \sum_{t=1}^{T} U_t(\mathbf{w}, m) \\ &\leqslant \tilde{\mathcal{O}} \left(\sqrt{(B + A\sigma_\theta) BT} \right) + \tilde{\mathcal{O}} \left(\frac{d}{\gamma} \sqrt{(N + \sigma_{\mathbf{w}}) NT} \right) + B \sqrt{\frac{T}{2} \log \frac{3}{\beta'}} \\ &\quad + \tilde{\mathcal{O}} \left(\max\{\gamma - \lambda, 0\} NT \right) + \min_{m > 0, \mathbf{w}_{[i]} \geqslant \lambda/d} \sum_{t=1}^{T} U_t(\mathbf{w}, m) \\ &= \tilde{\mathcal{O}} \left(\left(\frac{N^{\frac{4}{3}}}{\min\{1, \varepsilon^2\}} + \frac{N^{\frac{2}{3}}/\sqrt{\varepsilon'}}{\min\{1, \varepsilon\}} + \frac{dN}{\gamma} + \frac{d}{\gamma} \sqrt{\frac{N}{\varepsilon'}} \right) \sqrt{T} + \max\{\gamma - \lambda, 0\} NT \right) \\ &\quad + \min_{m > 0, \mathbf{w}_{[i]} \geqslant \lambda/d} \sum_{t=1}^{T} U_t(\mathbf{w}, m)\end{aligned} \tag{88}$$

where the first inequality follows by the regret of DP-FTRL w.r.t. θ together with [16, Lemma 4.1], the second by noting the definition of U_t and restricting to integer m, the third by the guarantee of DP-FTRL w.r.t. $\mathbf{w}$, and the fourth by joint-convexity of D_{KL} and simplifying terms. □

E Section 7 details

E.1 Location-scale families

A location-scale model is a distribution parameterized by a location $\nu \in \mathbb{R}$ and scale $\sigma \in \mathbb{R}_{\geqslant 0}$ whose density has the form $\mu_{\nu,\sigma}(x) = \frac{1}{\sigma} f\left(\frac{x-\nu}{\sigma}\right)$ for some centered probability measure $f : \mathbb{R} \mapsto \mathbb{R}_{\geqslant 0}$.

E.1.1 Impossibility of simultaneous robustness and convexity

Theorem E.1. *Let $f : \mathbb{R} \mapsto \mathbb{R}_{\geqslant 0}$ be a centered probability measure and for each $\theta \in \Theta$ define $\mu_\theta(x) = f(x - \theta)$.*

1. *If f is continuous then $U_{\mathbf{x}}(\mu_\theta)$ is convex in θ for all sorted dataset $\mathbf{x} \in \mathbb{R}^n$ if and only if f is log-concave.*
2. *There exist constants $a, b > 0$ s.t. for any $r > 0$, $\psi \in (0, \frac{R}{2n}]$, $q \geqslant \frac{1}{n}$, and $\theta \in \mathbb{R}$ there exists a sorted dataset $\mathbf{x} \in (\theta \pm R)^n$ with $\min_{i\in[n-1]} \mathbf{x}_{[i+1]} - \mathbf{x}_{[i]} = \psi$ s.t. $U^{(q)}_{\mathbf{x}}(\mu_\theta) = aR + \log \frac{b}{\psi}$.*

Proof. For the first direction of the first result, consider any $\theta, \theta' \in \mathbb{R}$ and $\lambda \in [0, 1]$. We have that

$$U^{(q)}_{*x}(\mu_{\lambda\theta+(1-\lambda)\theta'}) - \left(\lambda U^{(q)}_{\mathbf{x}}(\mu_\theta) - (1-\lambda)\log U^{(q)}_{\mathbf{x}}(\mu_{\theta'}\right) = \log \frac{\Psi^{(q)}_{\mathbf{x}}(\mu_\theta)^\lambda \Psi^{(q)}_{\mathbf{x}}(\mu_{\theta'})^{1-\lambda}}{\Psi^{(q)}_{\mathbf{x}}\mu_{\lambda\theta+(1-\lambda)\theta'})} \tag{89}$$

so it suffices to show that $\Psi_{\mathbf{x}}(q)(\mu_{\lambda\theta+(1-\lambda)\theta'}) \geqslant \Psi^{(q)}_{\mathbf{x}}(\mu_\theta)^\lambda \Psi^{(q)}_{\mathbf{x}}(\mu_{\theta'})^{1-\lambda}$. By the log-concavity of f we have

$$\mu_{\lambda\theta+(1-\lambda)\theta'}(\lambda x + (1-\lambda)y) = f(\lambda(x-\theta) + (1-\lambda)(y-\theta')) \geqslant f(x-\theta)^\lambda f(y-\theta')^{1-\lambda} = \mu_\theta(x)^\lambda \mu_{\theta'}(y)^{1-\lambda} \tag{90}$$

for all $x, y \in \mathbb{R}$. Therefore by the Prékopa-Leindler inequality we have that

$$\begin{aligned}\Psi^{(q)}_{\mathbf{x}}(\mu_{\lambda\theta+(1-\lambda)\theta'}) = \int_{\mathbf{x}_{[\lfloor qn\rfloor]}}^{\mathbf{x}_{[\lfloor qn\rfloor+1]}} \mu_{\lambda\theta+(1-\lambda)\theta'}(x)dx &\geqslant \left(\int_{\mathbf{x}_{[\lfloor qn\rfloor]}}^{\mathbf{x}_{[\lfloor qn\rfloor+1]}} \mu_\theta(x)dx\right)^\lambda \left(\int_{\mathbf{x}_{[\lfloor qn\rfloor]}}^{\mathbf{x}_{[\lfloor qn\rfloor+1]}} \mu_{\theta'}(x)dx\right)^{1-\lambda} \\ &= \Psi_{\mathbf{x}}(q)(\mu_\theta)^\lambda \Psi^{(q)}_{\mathbf{x}}(\mu_{\theta'})^{1-\lambda}\end{aligned} \tag{91}$$

For the second direction, by assumption $\exists\ a < c, b > c$ s.t. $\sqrt{f(x)f(y)} > f(\frac{x+y}{2})\ \forall\ x, y \in [a, b]$, i.e. f is strictly log-convex on $[a, b]$. Let $\mathbf{x} \in \mathbb{R}^n$ be any dataset s.t. $\mathbf{x}_{[\lfloor qn\rfloor+1]} - \mathbf{x}_{[\lfloor qn\rfloor]} \leqslant \frac{b-a}{2}$ and set $\theta = \mathbf{x}_{[\lfloor qn\rfloor]} - a$, $\theta' = \mathbf{x}_{[\lfloor qn\rfloor]} - \frac{a+b}{2}$. Then we have

$$\begin{aligned}\sqrt{\int_{\mathbf{x}_{[\lfloor qn\rfloor]}}^{\mathbf{x}_{[\lfloor qn\rfloor+1]}} \mu_\theta(x)dx \int_{\mathbf{x}_{[\lfloor qn\rfloor]}}^{\mathbf{x}_{[\lfloor qn\rfloor+1]}} \mu_{\theta'}(x)dx} &= \sqrt{\int_{\mathbf{x}_{[\lfloor qn\rfloor]}}^{\mathbf{x}_{[\lfloor qn\rfloor+1]}} \sqrt{\mu_\theta(x)}^2 dx \int_{\mathbf{x}_{[\lfloor qn\rfloor]}}^{\mathbf{x}_{[\lfloor qn\rfloor+1]}} \sqrt{\mu_{\theta'}(x)}^2 dx} \\ &\geqslant \int_{\mathbf{x}_{[\lfloor qn\rfloor]}}^{\mathbf{x}_{[\lfloor qn\rfloor+1]}} \sqrt{\mu_\theta(x)\mu_{\theta'}(x)}dx \\ &= \int_{\mathbf{x}_{[\lfloor qn\rfloor]}}^{\mathbf{x}_{[\lfloor qn\rfloor+1]}} \sqrt{f(x-\theta)f(x-\theta')}dx \\ &> \int_{\mathbf{x}_{[\lfloor qn\rfloor]}}^{\mathbf{x}_{[\lfloor qn\rfloor+1]}} f\left(x - \frac{\theta+\theta'}{2}\right)dx = \int_{\mathbf{x}_{[\lfloor qn\rfloor]}}^{\mathbf{x}_{[\lfloor qn\rfloor+1]}} \mu_{\frac{\theta+\theta'}{2}}(x)dx\end{aligned} \tag{92}$$

where the first inequality is Hölder's and the second is due to the strict log-convexity of f on $[a, b]$. Taking the logarithm of both sides followed by their negatives completes the proof.

Finally, for the second result, since f is centered and log-concave, by Cule and Samworth [22, Lemma 1] there exist constants $C, c > 0$ s.t. $\mu_\theta(x) \leqslant C \exp(-c|x - \theta|) \; \forall \; \theta \in \mathbb{R}$. Let

$$\mathbf{x} = \begin{pmatrix} \theta + R - n\psi & \theta + R - (n-1)\psi & \cdots & \theta + R - 2\psi & \theta + R - \psi \end{pmatrix} \tag{93}$$

so that $|\mathbf{x}_{[\lfloor qn \rfloor]} - \theta| \geqslant |\mathbf{x}_{[1]} - \theta| = R - n\psi \geqslant \frac{R}{2}$. Then

$$\Psi_{\mathbf{x}}^{(q)}(\mu_\theta) = \int_{\mathbf{x}_{[\lfloor qn \rfloor]}}^{\mathbf{x}_{[\lfloor qn \rfloor + 1]}} \mu_\theta(x) dx \leqslant C\psi \exp(-c|\mathbf{x}_{[\lfloor qn \rfloor]} - \theta|) \leqslant C\psi \exp(-cR/2) \tag{94}$$

so $U_{\mathbf{x}}^{(q)}(\mu) = -\log \Psi_{\mathbf{x}}^{(q)}(\mu_\theta) \geqslant \log \frac{1}{C\psi} + \frac{cR}{2}$. □

Variants of the first result have been shown in the censored regression literature [15, 62]. In fact, Burridge [15] shows convexity of $U_{\mathbf{x}}^{(q)}(\mu_{\frac{\langle \mathbf{v}, \mathbf{f} \rangle}{\phi}, \frac{1}{\phi}})$ w.r.t. $(\mathbf{v}, \phi) \in \mathbb{R}^d \times \mathbb{R}_{>0}$, i.e. simultaneous learning of a feature map and inverse scale. Convexity of $U_{\mathbf{x}} = -\log \Psi_{\mathbf{x}} = \log \sum_{i=1}^m \frac{1}{\Psi_{\mathbf{x}}^{(q_i)}} = \log \sum_{i=1}^m \exp(-\log \Psi_{\mathbf{x}}^{(q_i)})$ follows because $\log \sum_{i=1}^m e^{x_i}$ is convex and non-decreasing in each argument. Note that for the converse direction, the dataset $\mathbf{x}$ is not a degenerate case; in-fact if f is strictly log-convex over an interval $[a, b]$ then any dataset whose optimal interval has length smaller than $\frac{b-a}{2}$ will yield a non-convex $U_{\mathbf{x}}^{(q)}(\mu_\theta)$.

E.1.2 The case of the Laplacian

For the Laplace prior with $a = \mathbf{x}_{[\lfloor qn \rfloor]}$ and $b = \mathbf{x}_{[\lfloor qn \rfloor + 1]}$ we have

$$\begin{aligned} &-\log \Psi_{\mathbf{x}}^{(q)}(\mu_{\frac{\theta}{\phi}, \frac{1}{\phi}}) \\ &\quad = \log 2 \\ &\qquad - \log \left(\operatorname{sign}\left(b - \frac{\theta}{\phi}\right) \left(1 - \exp\left(-\left|b - \frac{\theta}{\phi}\right| \phi\right)\right) - \operatorname{sign}\left(a - \frac{\theta}{\phi}\right) \left(1 - \exp\left(-\left|a - \frac{\theta}{\phi}\right| \phi\right)\right)\right) \end{aligned} \tag{95}$$

For $\theta < a\phi$ this simplifies to

$$\begin{aligned} \log 2 - \log\left(e^{\theta - a\phi} - e^{\theta - b\phi}\right) &= \log 2 - \log\left((e^{\frac{b-a}{2}\phi} - e^{\frac{a-b}{2}\phi}) e^{\theta - \frac{a+b}{2}\phi}\right) \\ &= \left|\theta - \frac{a+b}{2}\phi\right| - \log\left(\sinh\left(\frac{b-a}{2}\phi\right)\right) \end{aligned} \tag{96}$$

and similarly for $\theta > b\phi$ it becomes

$$\begin{aligned} \log 2 - \log\left(e^{b\phi - \theta} - e^{a\phi - \theta}\right) &= \log 2 - \log\left((e^{\frac{b-a}{2}\phi} - e^{\frac{a-b}{2}\phi}) e^{\frac{a+b}{2}\phi - \theta}\right) \\ &= \left|\frac{a+b}{2}\phi - \theta\right| - \log\left(\sinh\left(\frac{b-a}{2}\phi\right)\right) \end{aligned} \tag{97}$$

On the other hand for $\theta \in [a\phi, b\phi]$ it is

$$\begin{aligned}
\log 2 - \log\left(2 - e^{-|b\phi-\theta|} - e^{-|a\phi-\theta|}\right) &= \log 2 - \log\left(2 - e^{\theta-b\phi} - e^{a\phi-\theta}\right) \\
&= \log 2 - \log\left(e^{-\frac{b-a}{2}\phi}\left(2e^{\frac{b-a}{2}\phi} - e^{\theta-\frac{a+b}{2}\phi} - e^{\frac{a+b}{2}\phi-\theta}\right)\right) \\
&= \frac{b-a}{2}\phi + \log 2 - \log\left(2e^{\frac{b-a}{2}\phi} - e^{\theta-\frac{a+b}{2}\phi} - e^{\frac{a+b}{2}\phi-\theta}\right) \\
&= \frac{b-a}{2}\phi - \log\left(e^{\frac{b-a}{2}\phi} - \cosh\left(\theta - \frac{a+b}{2}\phi\right)\right)
\end{aligned} \tag{98}$$

Thus we have

$$U_{\mathbf{x}}^{(q)}(\mu_{\frac{\theta}{\phi},\frac{1}{\phi}}) = \begin{cases} \frac{b-a}{2}\phi - \log\left(\exp\left(\frac{b-a}{2}\phi\right) - \cosh\left(\theta - \frac{a+b}{2}\phi\right)\right) & \text{if} \quad \theta \in [a\phi, b\phi] \\ \left|\theta - \frac{a+b}{2}\phi\right| - \log\left(\sinh\left(\frac{b-a}{2}\phi\right)\right) & \text{else} \end{cases} \tag{99}$$

Suppose $\mathbf{x} \in [\pm B]^n$ and has the optimal interval has separation $\psi > 0$, $\frac{\theta}{\phi} \in [\pm B]$, and $\frac{1}{\phi} \in [\sigma_{\min}, \sigma_{\max}]$. Then $\phi \in [1/\sigma_{\max}, 1/\sigma_{\min}]$ and $\theta \in [\pm B/\sigma_{\min}]$, and so

$$U_{\mathbf{x}}^{(q)}(\mu_{\frac{\theta}{\phi},\frac{1}{\phi}}) \leqslant \frac{2B}{\sigma_{\min}} + \log\frac{2\sigma_{\max}}{\psi} \tag{100}$$

For $\theta \notin [a\phi, b\phi]$, the derivative w.r.t. θ always has magnitude 1. Within the interval, the derivative w.r.t. θ is $-\frac{\sinh(\frac{a+b}{2}\phi-\theta)}{\exp(\frac{b-a}{2}\phi)-\cosh(\theta-\frac{a+b}{2}\phi)}$, which attains its extrema at the endpoints $a\phi$ and $b\phi$, where its magnitude is also 1. Outside the interval, the derivative w.r.t. ϕ has magnitude

$$\begin{aligned}
\left|\frac{a+b}{2}\operatorname{sign}\left(\frac{a+b}{2}\phi - \theta\right) - \frac{b-a}{2}\coth\left(\frac{b-a}{2}\phi\right)\right| &\leqslant \frac{|a+b|}{2} + \frac{b-a}{2}\coth\left(\frac{b-a}{2}\phi\right) \\
&\leqslant \frac{|a+b|}{2} + \frac{b-a}{2}\left(\frac{2/\phi}{(b-a)} + 1\right) \\
&= \frac{|a+b|}{2} + \frac{b-a}{2} + \frac{1}{\phi}
\end{aligned} \tag{101}$$

while inside the interval the derivative w.r.t. ϕ is $\frac{b-a}{2} - \frac{(b-a)\exp(\frac{b-a}{2}\phi)-(a+b)\sinh(\frac{a+b}{2}\phi-\theta)}{2\left(\exp(\frac{b-a}{2}\phi)-\cosh(\frac{a+b}{2}\phi-\theta)\right)}$, which again attains its extrema at the endpoints $a\phi$ and $b\phi$, yielding magnitudes

$$\begin{aligned}
\frac{b-a}{2} + \frac{b-a}{2}\left(\coth\left(\frac{b-a}{2}\phi\right) + 1\right) + \frac{|a+b|}{2} &\leqslant \frac{b-a}{2}\left(\frac{2/\phi}{(b-a)} + 3\right) + \frac{|a+b|}{2} \\
&\leqslant \frac{1}{\phi} + \frac{3}{2}(b-a) + \frac{|a+b|}{2}
\end{aligned} \tag{102}$$

Thus we have

$$|\partial_\theta U_{\mathbf{x}}^{(q)}(\mu_{\frac{\theta}{\phi},\frac{1}{\phi}})| \leqslant 1 \qquad \text{and} \qquad |\partial_\phi U_{\mathbf{x}}^{(q)}(\mu_{\frac{\theta}{\phi},\frac{1}{\phi}})| \leqslant 4B + \sigma_{\max} \tag{103}$$

E.2 Public-private release

E.2.1 Guarantees

Theorem E.2. *Suppose for $N \geqslant n$ we have a private dataset $\mathbf{x} \sim \mathcal{D}^n$ and a public dataset $\mathbf{x}' \sim \mathcal{D}'^N$, both drawn from κ-bounded distributions over $[\pm B]$. Use i.i.d. draws from the public dataset to construct $T = \lfloor N/n \rfloor$ datasets $\mathbf{x}'_t \sim \mathcal{D}'^n$ and run online gradient descent on the resulting losses $\ell_{\mathbf{x}'_t}(\theta, \psi) = \mathrm{LSE}_i(\ell^{(q_i)}_{\mathbf{x}_t}(\theta_{[i]}, \phi_{[i]}))$ over the parameter space $\theta \in [\pm B/\sigma_{\min}]^m$ starting at $\theta = \mathbf{0}_m$ and $\phi \in [1/\sigma_{\max}, 1/\sigma_{\min}]^m$ starting at the midpoint, with stepsize $B\sqrt{\frac{m}{T}}$ for θ and $\frac{\sigma_{\max}-\sigma_{\min}}{4B+\sigma_{\max}}\sqrt{\frac{m}{T}}$ for ϕ, obtaining iterates $(\theta_1, \phi_1), \ldots, (\theta_T, \phi_T)$. Return the priors $\mu_i = \mu_{\bar{\theta}_{[i]}, \frac{1}{\bar{\phi}_{[i]}}}$ for $\bar{\theta} = \frac{1}{T}\sum_{t=1}^T \theta_t$ and $\bar{\phi} = \frac{1}{T}\sum_{t=1}^T \phi_t$ the average of these iterates. Then $\mu' = \begin{pmatrix}\mu_1 & \cdots & \mu_m\end{pmatrix}$ satisfies*

$$\begin{aligned}
&\mathbb{E}_{\mathbf{x}\sim\mathcal{D}^n} U_{\mathbf{x}}(\mu') \\
&\quad\leqslant \min_{\mu \in \mathrm{Lap}^m_{B,\sigma_{\min},\sigma_{\max}}} \mathbb{E}_{\mathbf{x}\sim\mathcal{D}^n} U_{\mathbf{x}}(\mu) + 2\left(\frac{2B}{\sigma_{\min}} + \log\frac{4\kappa m(n+1)N\sigma_{\max}}{\beta'}\right)\mathrm{TV}_q(\mathcal{D}, \mathcal{D}') \\
&\qquad + (B + 4B\sigma_{\max} + \sigma^2_{\max})\sqrt{\frac{m(n+1)}{N}} + 2\left(\frac{4B}{\sigma_{\min}} + \log\frac{4\kappa m(n+1)N\sigma_{\max}}{\beta'}\right)\sqrt{\frac{2(n+1)}{N}\log\frac{4}{\beta'}} \\
&\qquad + \frac{(n+1)\beta'}{N}\left(3 + \frac{4B}{\sigma_{\min}} + 4\log\frac{2\kappa(n+1)N\sqrt{2m\sigma_{\max}}}{\beta'}\right)
\end{aligned} \tag{104}$$

where $\mathrm{Lap}_{B,\sigma_{\min},\sigma_{\max}}$ is the set of Laplace priors with locations in $[\pm B]$ and scales in $[\sigma_{\min}, \sigma_{\max}]$.

Proof. Define ${\mathcal{D}'_\psi}^n$ to be the conditional distribution over $\mathbf{z} \sim \mathcal{D}'^n$ s.t. $\psi_{\mathbf{z}} \geqslant \psi$, with associated density $\rho'_\psi(\mathbf{z}) = \frac{\rho'(\mathbf{z})1_{\psi_{\mathbf{z}}\geqslant\psi}}{1-p'_\psi}$, where $p'_\psi = \int_{\psi_{\mathbf{z}}<\psi}\rho'(\mathbf{z}) \leqslant \kappa n^2\psi$. Then we have for any $\mu^* \in \mathrm{Lap}^m_{B,\sigma_{\min},\sigma_{\max}}$ that

$$\begin{aligned}
&\mathbb{E}_{\mathbf{z}\sim\mathcal{D}^n} U_{\mathbf{x}}(\mu') \\
&\quad= \mathbb{E}_{\mathbf{z}\sim\mathcal{D}^n} U_{\mathbf{z}}(\mu') - \mathbb{E}_{\mathbf{z}\sim\mathcal{D}'^n} U_{\mathbf{z}}(\mu') + \mathbb{E}_{\mathbf{z}\sim\mathcal{D}'^n} U_{\mathbf{z}}(\mu') - \mathbb{E}_{\mathbf{z}\sim{\mathcal{D}'_\psi}^n} U_{\mathbf{z}}(\mu') + \mathbb{E}_{\mathbf{z}\sim{\mathcal{D}'_\psi}^n} U_{\mathbf{z}}(\mu') \\
&\quad\leqslant \int U_{\mathbf{z}}(\mu')(\rho(\mathbf{z}) - \rho'(\mathbf{z})) + \int U_{\mathbf{z}}(\mu')(\rho'(\mathbf{z}) - \rho'_\psi(\mathbf{x})) + \mathbb{E}_{\mathbf{z}\sim{\mathcal{D}'_\psi}^n} U_{\mathbf{z}}(\mu^*) + \mathcal{E}_\psi \\
&\quad\leqslant \mathbb{E}_{\mathbf{z}\sim\mathcal{D}^n} U_{\mathbf{x}}(\mu^*) + \int (U_{\mathbf{z}}(\mu') + U_{\mathbf{x}}(\mu^*))|\rho(\mathbf{z}) - \rho'(\mathbf{x})| + \int (U_{\mathbf{z}}(\mu') + U_{\mathbf{z}}(\mu^*))|\rho'(\mathbf{z}) - \rho'_\psi(\mathbf{z})| + \mathcal{E}_\psi
\end{aligned} \tag{105}$$

where $\mathcal{E}_\psi$ is the error of running online gradient descent with the specified step-sizes on samples $\mathbf{z}'_t \sim {\mathcal{D}'_\psi}^n$ for $t = 1, \ldots, T$. Now if $\mathbf{z}$ has entries drawn i.i.d. from a κ-bounded distribution $\mathcal{D}^n$ (or $\mathcal{D}'^n$), then we have that

$$\int_0^\psi \rho_{\psi_{\mathbf{z}}}(y)dy = \Pr(\psi_{\mathbf{z}} \leqslant \psi : \mathbf{z} \sim \mathcal{D}^n) \leqslant n(n-1)\max_{z\in\mathbb{R}}\Pr(|z - z'| \leqslant \psi : z' \sim \mathcal{D}) \leqslant \kappa n^2\psi \tag{106}$$

where $\rho_{\psi_{\mathbf{z}}}$ is the density of $\psi_{\mathbf{z}}$ for $\mathbf{z} \sim \mathcal{D}^n$ (not to be confused with the conditional density ρ_ψ over $\mathbf{z}$); the same holds for the analog $\rho'_{\psi_{\mathbf{z}}}$ for $\mathcal{D}'^n$. Since this holds for all $\psi \geqslant 0$ and $\log\frac{1}{y}$ is monotonically decreasing on $y > 0$, this means the worst-case measure that $\rho_{\psi_{\mathbf{z}}}$ can be is constant over $[0, \psi]$ and thus $\int_0^\psi \rho_{\psi_{\mathbf{z}}}(y)\log\frac{1}{y}dy \leqslant \kappa n^2\int_0^\psi \log\frac{1}{y}dy = \kappa n^2\psi(1 + \log\frac{1}{\psi})$, and similarly for $\rho'_{\psi_{\mathbf{z}}}$. We then bound

the first integral, noting that $U_{\mathbf{z}} = \mathrm{LSE}_i(U_{\mathbf{z}}^{(q_i)}) \leqslant \max_i U_{\mathbf{z}}^{(q_i)} + \log m \leqslant \frac{2B}{\sigma_{\min}} + \log \frac{2m\sigma_{\max}}{\psi_{\mathbf{z}}}$ and that the r.v. $\psi_{\mathbf{z}}$ depends only on the joint distribution over the order statistics of $\mathcal{D}^n$ and $\mathcal{D}'^n$:

$$\begin{aligned}
&\int (U_{\mathbf{z}}(\mu') + U_{\mathbf{z}}(\mu^*))|\rho(\mathbf{z}) - \rho'(\mathbf{z})| \\
&\qquad \leqslant \int \left(\frac{2B}{\sigma_{\min}} + \log \frac{2m\sigma_{\max}}{\psi_{\mathbf{z}}}\right) |\rho(\mathbf{z}) - \rho'(\mathbf{z})| \\
&\qquad \leqslant 2\left(\frac{2B}{\sigma_{\min}} + \log \frac{2m\sigma_{\max}}{\psi}\right) \mathrm{TV}_q(\mathcal{D}, \mathcal{D}') + \int_{\psi_{\mathbf{z}} < \psi} |\rho(\mathbf{z}) - \rho'(\mathbf{z})| \log \frac{1}{\psi_{\mathbf{z}}} \\
&\qquad \leqslant 2\left(\frac{2B}{\sigma_{\min}} + \log \frac{2m\sigma_{\max}}{\psi}\right) \mathrm{TV}_q(\mathcal{D}, \mathcal{D}') + \int_0^{\psi} (\rho_{\psi_{\mathbf{z}}}(y) + \rho'_{\psi_{\mathbf{z}}}(y)) \log \frac{1}{y} dy \\
&\qquad \leqslant 2\left(\frac{2B}{\sigma_{\min}} + \log \frac{2m\sigma_{\max}}{\psi}\right) \mathrm{TV}_q(\mathcal{D}, \mathcal{D}') + 2\kappa n^2 \psi \left(1 + \log \frac{1}{\psi}\right)
\end{aligned} \tag{107}$$

For the second integral we have for $p'_\psi = \int_{\psi_{\mathbf{z}} < \psi} \rho'(\mathbf{z}) \leqslant \kappa n^2 \psi$ that

$$\begin{aligned}
&\int (U_{\mathbf{z}}(\mu') + U_{\mathbf{z}}(\mu^*))|\rho'(\mathbf{z}) - \rho'_\psi(\mathbf{z})| \\
&\qquad = \int_{\psi_{\mathbf{z}} \geqslant \psi} (U_{\mathbf{x}}(\mu') + U_{\mathbf{z}}(\mu^*)) \left|\rho'(\mathbf{z}) - \frac{\rho'(\mathbf{z})}{1 - p'_\psi}\right| + \int_{\psi_{\mathbf{z}} < \psi} (U_{\mathbf{z}}(\mu') + U_{\mathbf{z}}(\mu^*))\rho'(\mathbf{z}) \\
&\qquad = \frac{2p'_\psi}{1 - p'_\psi} \int_{\psi_{\mathbf{z}} \geqslant \psi} \left(\frac{2B}{\sigma_{\min}} + \log \frac{2m\sigma_{\max}}{\psi}\right) \rho'(\mathbf{z}) + \int_{\psi_{\mathbf{z}} < \psi} \left(\frac{2B}{\sigma_{\min}} + \log \frac{2m\sigma_{\max}}{\psi_{\mathbf{z}}}\right) \rho'(\mathbf{z}) \\
&\qquad = 2p'_\psi \left(\frac{4B}{\sigma_{\min}} + \log \frac{4m^2\sigma_{\max}^2}{\psi}\right) + \int_{\psi_{\mathbf{z}} < \psi} \rho'(\mathbf{z}) \log \frac{1}{\psi_{\mathbf{z}}} \\
&\qquad \leqslant 2\kappa n^2 \psi \left(\frac{4B}{\sigma_{\min}} + \log \frac{4m^2\sigma_{\max}^2}{\psi}\right) + \kappa n^2 \psi \left(1 + \log \frac{1}{\psi}\right)
\end{aligned} \tag{108}$$

Finally, we bound $\mathcal{E}_\psi$. By κ-boundedness of $\mathcal{D}'$, the probability that $\exists\ t \in [T]$ s.t. $\psi_{\mathbf{z}'_t} < \psi\ \forall\ t \in [T]$ is at most $\kappa n^2 T\psi$, so if we set $\psi = \frac{\beta'}{2\kappa n^2 T}$ then w.p. $\geqslant 1 - \beta'/2$ the sampling $\mathbf{z}'_t$ from $\mathbf{x}'$ as specified is equivalent to rejection sampling from ${\mathcal{D}'_\psi}^n$, on which the functions $U_{\mathbf{z}}$ are bounded by $\frac{2B}{\sigma_{\min}} + \log \frac{2m\sigma_{\max}}{\psi}$. Therefore with probability $\geqslant 1 - \beta'/2$ by Shalev-Shwartz [68, Theorem 2.21] and Theorem D.1 we have that w.p. $1 - \beta'/2$

$$\begin{aligned}
\mathcal{E}_\psi &\leqslant (B + (\sigma_{\max} - \sigma_{\min})(4B + \sigma_{\max}))\sqrt{\frac{m}{T}} + 2\left(\frac{4B}{\sigma_{\min}} + \log \frac{2m\sigma_{\max}}{\psi}\right)\sqrt{\frac{2}{T}\log \frac{4}{\beta'}} \\
&= (B + 4B\sigma_{\max} + \sigma_{\max}^2)\sqrt{\frac{m(n+1)}{N}} + 2\left(\frac{4B}{\sigma_{\min}} + \log \frac{2m\sigma_{\max}}{\psi}\right)\sqrt{\frac{2(n+1)}{N}\log \frac{4}{\beta'}}
\end{aligned} \tag{109}$$

Combining terms and substituting the selected value for ψ yields the result. □

E.2.2 Experimental details

For our public-private experiments we evaluate several methods on the Adult ("age" and "hours" categories) and Goodreads ("rating" and "page count" categories). For the former we use the train set as the public data, while for the latter we use the "History" genre as the public data and the "Poetry" genre as the private data [70]. The public data are used to fit Laplace location and scale parameters using the COCOB optimizer run until progress stops. We use the implementation here: `https://github.com/anandsaha/nips.cocob.pytorch`. All evaluations are averages of forty trials.

We use the following reasonable guesses for locations ν, scales σ, and quantile ranges $[a, b]$ for these distributions:

- age: $\nu = 40$, $\sigma = 5$, $a = 10$, $b = 120$
- hours: $\nu = 40$, $\sigma = 2$, $a = 0$, $b = 168$
- rating: $\nu = 2.5$, $\sigma = 0.5$, $a = 0$, $b = 5$
- page count: $\nu = 200$, $\sigma = 25$, $a = 0$, $b = \frac{1000}{1-q}$

Note that, here and elsewhere, using q-dependent range for b only helps the Uniform prior, which is the baseline. The scales σ are used to set the scale parameter of the Cauchy distribution for `public quantiles`—its location is fixed by the public quantiles. Meanwhile the locations ν are used to set to *scale* parameter of the half-Cauchy prior used to mix with `PubFit` for robustness (using coefficient 0.1 on the robust prior). We choose this prior because the data are all nonnegative.

E.3 Sequential release

E.3.1 Guarantees

Theorem E.3. *Consider a sequence of datasets $\mathbf{x}_t \in [\pm R]^{n_t}$ and associated feature vectors $\mathbf{f}_t \in [\pm F]^d$. Suppose we set the component priors $\mu_{t,i}$ of μ_t as the Laplace distributions $\mu_{t,i} = \mu_{\frac{\langle \mathbf{v}_{t,i}, \mathbf{f}_t \rangle}{\phi_{t,i}}, \frac{1}{\phi_{t,i}}}$, where $\mathbf{v}_{t,i} \in [\pm B/\sigma_{\min}]^d$ and $\phi_i \in [1/\sigma_{\max}, 1/\sigma_{\min}]$ are determined by separate runs of DP-FTRL with budgets $(\varepsilon'/2, \delta'/2)$ and step-sizes $\eta_1 = \frac{B}{F\sigma_{\min}} \sqrt{\frac{2m\varepsilon_1'}{\lceil \log_2(T+1) \rceil T \left(1 + \sqrt{2md \log \frac{T}{\beta'} \log \frac{1}{\delta'}}\right)}}$, and $\eta_2 = \frac{1/\sigma_{\min}}{B+\sigma_{\max}} \sqrt{\frac{m\varepsilon_2'}{2\lceil \log_2(T+1) \rceil T \left(1 + \sqrt{2m \log \frac{T}{\beta'} \log \frac{1}{\delta'}}\right)}}$. Then we have regret*

$$\begin{aligned} &\max_{\substack{\mathbf{w}_i \in [\pm B]^d \\ \sigma_i \in [\sigma_{\min}, \sigma_{\max}]}} \sum_{t=1}^{T} U_{\mathbf{x}_t}(\mu_t) - U_{\mathbf{x}_t}(\mu_{\langle \mathbf{w}_i, \mathbf{f}_t \rangle, \sigma_i}) \\ &\qquad \leqslant \frac{B(F+1) + \sigma_{\max}}{\sigma_{\min}} \sqrt{md \lceil \log_2(T+1) \rceil T \left(4 + \frac{8}{\varepsilon'} \sqrt{2md \log \frac{T}{\beta'} \log \frac{2}{\delta'}}\right)} \end{aligned} \tag{110}$$

For sufficiently small ε' (including $\varepsilon' \leqslant 1$) we can instead simplify the regret to

$$\frac{4}{\sigma_{\min}} \left(BFd^{\frac{3}{4}} + B + \sigma_{\max}\right) \sqrt{\frac{m \lceil \log_2(T+1) \rceil T}{\varepsilon'} \sqrt{2m \log \frac{T}{\beta'} \log \frac{2}{\delta'}}} \tag{111}$$

Proof. Note that

$$\sum_{j=1}^{m} \|\nabla_{\mathbf{v}_j} \operatorname{LSE}_i(\ell^{(q_i)}_{\mathbf{x}_t,\mathbf{f}_t})\|_2^2 \leqslant \|\mathbf{f}_t\|_2^2 \sum_{j=1}^{m} \left(\frac{\exp(\ell^{(q_j)}_{\mathbf{x}_t,\mathbf{f}_t})}{\sum_{i=1}^{m} \exp(\ell^{(q_i)}_{\mathbf{x}_t,\mathbf{f}_t})} \right)^2 \leqslant F^2 d \tag{112}$$

and

$$\sum_{j=1}^{m} (\partial_{\phi_j} \operatorname{LSE}_i(\ell^{(q_i)}_{\mathbf{x}_t,\mathbf{f}_t}))^2 \leqslant (4B + \sigma_{\max})^2 \sum_{j=1}^{m} \left(\frac{\exp(\ell^{(q_j)}_{\mathbf{x}_t,\mathbf{f}_t})}{\sum_{i=1}^{m} \exp(\ell^{(q_i)}_{\mathbf{x}_t,\mathbf{f}_t})} \right)^2 \leqslant (4B + \sigma_{\max})^2 \tag{113}$$

and so applying Theorem 6.1 twice with the assumed budgets and step-sizes yields

$$\begin{aligned}
&\max_{\substack{\mathbf{w}_i \in [\pm B]^d \\ \sigma_i \in [\sigma_{\min}, \sigma_{\max}]}} \sum_{t=1}^{T} U_{\mathbf{x}_t}(\mu_t) - U_{\mathbf{x}_t}(\mu_{\langle \mathbf{w}_i, \mathbf{f}_t \rangle, \sigma_i}) \\
&= \max_{\substack{\mathbf{v}_i \in [\pm \frac{B}{\sigma_{\min}}]^d \\ \phi_i \in [\frac{1}{\sigma}_{\max}, \frac{1}{\sigma}_{\min}]}} \sum_{t=1}^{T} \operatorname{LSE}_i(\ell^{(q_i)}_{\mathbf{x}_t,\mathbf{f}_t}(\mathbf{v}_{t,i}, \phi_{t,i})) - \operatorname{LSE}_i(\ell^{(q_i)}_{\mathbf{x}_t,\mathbf{f}_t}(\mathbf{v}_i, \phi_i)) \\
&\leqslant \sum_{i=1}^{m} \frac{\|\mathbf{v}_{1,i} - \mathbf{v}_i\|_2^2}{2\eta_1} + \eta_1 \lceil \log_2(T+1) \rceil T \left(1 + \frac{2}{\varepsilon'} \sqrt{2md \log \frac{T}{\beta'} \log \frac{2}{\delta'}} \right) \sum_{j=1}^{m} \|\nabla_{\mathbf{v}_j} \operatorname{LSE}_i(\ell^{(q_i)}_{\mathbf{x}_t,\mathbf{f}_t})\|_2^2 \\
&\quad + \sum_{i=1}^{m} \frac{(\phi_{1,i} - \phi_i)^2}{2\eta_2} + \eta_2 \lceil \log_2(T+1) \rceil T \left(1 + \frac{2}{\varepsilon'} \sqrt{2m \log \frac{T}{\beta'} \log \frac{2}{\delta'}} \right) \sum_{j=1}^{m} (\partial_{\phi_j} \operatorname{LSE}_i(\ell^{(q_i)}_{\mathbf{x}_t,\mathbf{f}_t}))^2 \\
&\leqslant \frac{2B^2 md}{\eta_1 \sigma_{\min}^2} + \eta_1 \lceil \log_2(T+1) \rceil T F^2 d \left(1 + \frac{2}{\varepsilon'} \sqrt{2md \log \frac{T}{\beta'} \log \frac{2}{\delta'}} \right) \\
&\quad + \frac{m}{2\eta_2 \sigma_{\min}^2} + \eta_2 \lceil \log_2(T+1) \rceil T (B + \sigma_{\max})^2 \left(1 + \frac{2}{\varepsilon'} \sqrt{2m \log \frac{T}{\beta'} \log \frac{2}{\delta'}} \right) \\
&\leqslant \frac{2BF}{\sigma_{\min}} \sqrt{2md \lceil \log_2(T+1) \rceil T \left(1 + \frac{2}{\varepsilon'} \sqrt{2md \log \frac{T}{\beta'} \log \frac{2}{\delta'}} \right)} \\
&\quad + \frac{2}{\sigma_{\min}} (B + \sigma_{\max}) \sqrt{2m \lceil \log_2(T+1) \rceil T \left(1 + \frac{2}{\varepsilon'} \sqrt{2m \log \frac{T}{\beta'} \log \frac{2}{\delta'}} \right)} \\
&\leqslant \frac{2}{\sigma_{\min}} \left(B(F+1) + \sigma_{\max} \right) \sqrt{md \lceil \log_2(T+1) \rceil T \left(1 + \frac{2}{\varepsilon'} \sqrt{2md \log \frac{T}{\beta'} \log \frac{2}{\delta'}} \right)}
\end{aligned} \tag{114}$$

□

E.3.2 Experimental details

For sequential release we consider the following tasks:

- Synthetic is a stationary dataset generation scheme in which we randomly sample a one standard Gaussian vector $\mathbf{a}$ for each feature dimension (we use ten) and another $\mathbf{b}$ of size $m+2$, which we sort. On each day t of T we sample the public feature vector $\mathbf{f}_t$, also from a standard normal, and the "ground truth" quantiles q_i on that day are then set by $\langle \mathbf{a}, \mathbf{f}_t \rangle + \mathbf{b}_{[i+1]}$. We generate the actual data by sampling from the uniform distributions on $[\langle \mathbf{a}, \mathbf{f}_t \rangle + \mathbf{b}_{[i]}, \langle \mathbf{a}, \mathbf{f}_t \rangle + \mathbf{b}_{[i+1]}]$. The number of points we sample is determined by $\lfloor 100/(m+1) \rfloor$ plus different Poisson-distributed random variable for each; in the "noiseless" setting used in Figure 4 (left) the Poisson's scale is zero, so the "ground truth" quantiles are correct for the dataset, while for Figure 5 (left) we use a Poisson with scale five. For the noiseless setting we use 100K timesteps, while for the noisy setting we use 2500.

- CitiBike consists of data downloaded from here: `https://s3.amazonaws.com/tripdata/index.html`, We take the period from September 2015 through November 2022, which is roughly 2500 days, although days with less than ten trips—seemingly data errors—are ignored. For each day we include a feature vector containing seven dimensions for the day of the week, one dimension for a sinusoidal encoding of the day of the year, and six weather features from the Central Park station downloaded from here `https://www.ncei.noaa.gov/cdo-web/`, specifically average wind speed, precipitation, snowfall, snow depth, maximum temperature, and minimum temperature. These are scaled to lie within similar ranges.

- BBC consists of Reddit's `worldnews` corpus downloaded from here: `https://zissou.infosci.cornell.edu/convokit/datasets/subreddit-corpus/corpus-zipped/`. We find all conversations corresponding to a post of a BBC article, specified by the domain `bbc.co.uk`, and collect those with at least ten comments. We compute the Flesch readability score of each comment using the package here `https://github.com/textstat/textstat`. The datasets for computing quantiles are then the collection of scores for each headline; the size is roughly 10K, corresponding to articles between 2008 and 2018. As features we combine a seven-dimensional day-of-the-week encoding, sinusoidal features for the day of the year and the time of day of the post, information about the post itself (whether it is gilded, its own Flesch score, and the number of tokens), and finally a 25-dimensional embedding of the title, set using a normalized sum of GloVe embeddings [61] of the tokens, excluding English stop-words via NLTK [55].

We again use reasonable guesses of data information to set the static priors, and to initialized the learning schemes.

- Synthetic: $\nu = 0$, $\sigma = 1$, $a = -100$, $b = 100$
- CitiBike: $\nu = 10$, $\sigma = 1$, $a = 0$, $b = 50/(1-q)$
- BBC: $\nu = 50$, $\sigma = 10$, $a = -100 - 100/(1-q)$, $b = 100 + 100q$

We use a and b for the static Uniform distributions, ν and σ for the static Cauchy distributions, in the case of nonnegative data (CitiBike) we use ν for the *scale* of the half-Cauchy distribution, and for the learning schemes we initialize their Laplace priors to be centered at ν with scale σ. We again use the COCOB optimizer for non-private and proxy learning, and for robustness we mix with the Cauchy (or half-Cauchy for nonnegative data) with coefficient 0.1 on the robust prior. For the `PubPrev` method, we set its scale using σ. For DP-FTRL, we heavily tune it to show the possibility of learning on the synthetic task; the implementation is adapted from the one here: `https://github.com/google-research/DP-FTRL`. All results are reported as averages over forty trials.