Supplementary Material to Private Protocols for U-Statistics in the Local Model and Beyond

February 28, 2020

A Details and Proofs for Generic LDP Protocol

We start by introducing some notations. We denote by k the number of bins of the quantization and by $\pi : \mathcal{X} \to [k]$ the quantization scheme such that for any data point $x \in \mathcal{X}$, $\pi(x)$ denotes the quantized version of x (i.e., its image under π). Let e_i denote the vector of length k with a one in the i-th position and 0 elsewhere. A kernel function f_A on the quantized domain $[k]$ is fully described by a matrix $A \in \mathbb{R}^{k\times k}$ such that $f_A(i,j) = A_{i,j} = e_i^T A e_j$. We denote by $U_{A,\pi} = \mathbb{E}_{\mu\times\mu}[A_{\pi(x),\pi(y)})]$ the quantized analogue to the quantity U_f.

The proposed protocol, described in Algorithm 1, applies generalized randomized response on data quantized with π and uses this to compute an unbiased estimate of a quantized U-statistic. The choice of the quantized kernel A will be discussed below. Crucially, there are two sources of error in this protocol. More precisely, the mean squared error of the estimate $\widehat{U}_{f,n}$ returned by Algorithm 1 can be bounded as follows:

$$\texttt{MSE}(\widehat{U}_{f,n}) \leq (U_f - U_{A,\pi})^2 + \mathbb{E}[(U_{A,\pi} - \widehat{U}_{f,n})^2]. \tag{1}$$

The first term corresponds to the error due to quantization, while the second one is the estimation error due to randomization needed to satisfy local differential privacy. The latter will increase with k, thereby constraining k to remain reasonably small. We will thus need to rely on assumptions on either the kernel or the data distribution to be able to control the error due to quantization.

In line with the error decomposition in (1), we conduct our analysis by considering the effect of sampling and randomization together. Therefore, we will not provide a direct bound on the error between our estimate and the U-statistic of the sample, but directly with respect to the population quantity U_f. We now show how to control the two sources of error, which are easily combined to yield Theorem 2.

A.1 Bounding the Error of Randomized Response on Discrete Domain

In this part, we bound the second term in (1): we consider that the data is discrete ($\mathcal{X} = [k]$) and derive error bounds for the estimate $\widehat{U}_{f,n}$ with respect to $U_{A,\pi}$ for a given kernel function

$\tilde{f}(i,j) = A_{i,j}$. We propose to use the generalized randomized response mechanism as our local randomizer $\mathcal{R}$. We introduce some notations. Let β be the probability of $\mathcal{R}$ selecting a response uniformly at random, i.e. let $\mathbb{P}(\mathcal{R}(x) = y) = \beta/k + (1-\beta)\chi_{x=y}$, let b be the vector of length k with every entry β/k. For convenience, we denote the data sample by $x_1, \dots, x_n \in [k]$. Note that these data points are drawn i.i.d. from a distribution D over $[k]$ (which follows from μ and π) such that $\mathbb{P}(x_i = j) = D_j$.

With these notations, we write the expected value of the discretized kernel computed directly on the randomized data points:

$$\begin{aligned}\mathbb{E}[(e_{\mathcal{R}(x_1)})^T A e_{\mathcal{R}(x_2)}] &= \mathbb{E}(((1-\beta)e_{x_1} + b)^T A((1-\beta)e_{x_2} + b)) \\ &= \mathbb{E}((1-\beta)^2 e_{x_1}^T A e_{x_2} + (1-\beta)(e_{x_1} + e_{x_2})^T A b + b^T A b).\end{aligned}$$

This is a biased estimator of $f_A(x_1, x_2) = e_{x_1} A e_{x_2}$ due to the effect of the randomization. We correct for this by adding terms and scaling, leading to the estimator used in Algorithm 1:

$$\widehat{f}_A(\mathcal{R}(x_1), \mathcal{R}(x_2)) = (1-\beta)^{-2}(e_{\mathcal{R}(x_1)} - b)^T A(e_{\mathcal{R}(x_2)} - b).$$

This is an unbiased estimator of the population U-statistic, as for fixed x_1 and x_2 it is an unbiased estimator of $f_A(x_1, x_2)$. Averaging over all pairs of randomized inputs, we get the proposed estimator:

$$\widehat{U}_{f,n} = \binom{n}{2}^{-1} \sum_{1 \le i < j \le n} \widehat{f}_A(\mathcal{R}(x_i), \mathcal{R}(x_j)),$$

which is itself a U-statistic on the randomized sample. As this estimator is unbiased, its mean squared error is equal to its variance, for which the following lemma gives an exact expression.

Lemma 1. *The variance of $\widehat{U}_{f,n}$ is given by*

$$\binom{n}{2}^{-1} \left(\frac{2n-3}{(1-\beta)^2} \mathrm{Var}(e_{(\mathcal{R}(x_1))}^T AD) + \frac{1}{(1-\beta)^4} \mathbb{E}(\mathrm{Var}((e_{\mathcal{R}(x_1)} - b) A e_{\mathcal{R}(x_2)} \mid \mathcal{R}(x_1))) \right)$$

Proof. $\widehat{U}_{f,n}$ is a U-statistic, hence its variance is given by (3) where $\zeta_1 = \mathrm{Var}(\mathbb{E}(\widehat{f}_A(\mathcal{R}(x_1), \mathcal{R}(x_2)) \mid \mathcal{R}(x_1)))$ and $\zeta_2 = \mathrm{Var}(\widehat{f}_A(\mathcal{R}(x_1), \mathcal{R}(x_2)))$. We first simplify ζ_1:

$$\begin{aligned}\zeta_1 &= (1-\beta)^{-4} \mathrm{Var}(\mathbb{E}((e_{\mathcal{R}(x_1)} - b)^T A(e_{\mathcal{R}(x_2)} - b) \mid \mathcal{R}(x_1))) \\ &= (1-\beta)^{-4} \mathrm{Var}((e_{\mathcal{R}(x_1)} - b)^T A((1-\beta)D)) \\ &= (1-\beta)^{-2} \mathrm{Var}((e_{\mathcal{R}(x_1)} - b)^T AD) \\ &= (1-\beta)^{-2} \mathrm{Var}((e_{\mathcal{R}(x_1)})^T AD).\end{aligned}$$

Similarly for ζ_2, we have:

$$\begin{aligned}\zeta_2 &= \mathrm{Var}(\mathbb{E}(\widehat{f}(\mathcal{R}(x_1), \mathcal{R}(x_2)) \mid \mathcal{R}(x_1))) + \mathbb{E}(\mathrm{Var}(\widehat{f}(\mathcal{R}(x_1), \mathcal{R}(x_2)) \mid \mathcal{R}(x_1))) \\ &= \zeta_1 + (1-\beta)^{-4} \mathbb{E}(\mathrm{Var}((e_{\mathcal{R}(x_1)} - b)^T A(e_{\mathcal{R}(x_2)} - b) \mid \mathcal{R}(x_1))) \\ &= \zeta_1 + (1-\beta)^{-4} \mathbb{E}(\mathrm{Var}((e_{\mathcal{R}(x_1)} - b)^T A e_{\mathcal{R}(x_2)} \mid \mathcal{R}(x_1))).\end{aligned}$$

Substituting the values of ζ_1 and ζ_2 in the variance expression gives the result. ■

Assuming a uniform bound on the values of f allows a clear and simple bound on the variance.

Corollary 1. *If $f(x, x') \in [0,1]$ for all x, x', then*

$$\mathrm{Var}(\widehat{U}_{f,n}) \leq \frac{1}{n(1-\beta)^2} + \frac{(1+\beta)^2}{2n(n-1)(1-\beta)^4}.$$

Proof. Under the boundedness of f, the random variable $(e^{\mathcal{R}(x_1)})^T AD$ takes values in $[0,1]$ and so has variance at most $1/4$, whilst the random variable $(e^{\mathcal{R}(x_1)} - B)^T A e^{\mathcal{R}(x_2)}$ takes values in $[-\beta, 1]$ and so has variance at most $(1+\beta)^2/4$. Substituting these into Lemma 1 gives the result. ■

To achieve local differential privacy with parameter ϵ, β should be taken to be $k/(k+e^\epsilon - 1)$. This leads directly to the following result.

Corollary 2 (Variance under randomized reponse)**.** *Let $\mathcal{X} = [k]$ and assume f takes values in $[0,1]$. We have:*

$$\mathrm{Var}(\widehat{U}_{f,n}) \approx \frac{(1+k/\epsilon)^2}{n} + \frac{(1+k/\epsilon)^4}{2n^2} \approx \frac{k^2}{n\epsilon^2},$$

where the approximation holds for small ϵ and $n \gg k^2/\epsilon^2$.

The above result shows that for fixed ϵ the error incurred by this estimator is within a constant factor of the error due to the finite sample setting. As expected, k should be reasonably small for the protocol to yield any utility.

A.2 Bounding the Error of Quantization

We now study the effect of quantization, which is needed to control the error due to privacy when the domain is continuous or has large cardinality. Recall that we quantize $\mathcal{X}$ using a projection $\pi : \mathcal{X} \to [k]$ (assumed to be simple rounding for simplicity), and our goal is to approximate $U_f = \mathbb{E}_{\mu \times \mu}(f(x,y))$ by $U_{A,\pi} = \mathbb{E}_{\mu\times\mu}(A_{\pi(x),\pi(y)})$, which we can privately estimate using the results of Section A.1.

The error incurred by the quantization can be written as follows:

$$\begin{aligned}(U_f - U_{A,\pi})^2 &\leq \int\int (f(x,y) - A_{\pi(x),\pi(y)})^2 d\mu(y) d\mu(x) \\ &= \sum_{i,j=1}^{k} \int_{\pi^{-1}(i)} \int_{\pi^{-1}(j)} (f(x,y) - A_{i,j})^2 d\mu(y) d\mu(x). \qquad (2)\end{aligned}$$

To bound this quantization error, we need additional assumptions. We consider two options, each suggesting a different choice for the quantized kernel $A_{i,j}$. We first consider a Lipschitz assumption on the original kernel function, for which the preferred quantized kernel minimizes the worst-case bound on (2). Then, we consider a smoothness assumption on the data distribution, leading to a quantized kernel that attempts to minimize the average-case error. We stress the fact that in some cases these quantized statistics will match or at least be very close, meaning that the particular choice of quantized kernel will not be crucial.

A.2.1 Assumption of Lipschitz Kernel Function

Our first assumption is motived by the fact that for all data distributions μ, the quantization error (2) can be bounded by

$$\sum_{i,j=1}^{k} \mu(\pi^{-1}(i))\mu(\pi^{-1}(j)) \max_{\substack{x\in\pi^{-1}(i)\\ y\in\pi^{-1}(j)}} (f(x,y) - A_{i,j})^2. \tag{3}$$

This bound is minimized by choosing the quantized kernel to be

$$A_{i,j}^{Mid} = \frac{1}{2} \max_{\substack{x\in\pi^{-1}(i)\\ y\in\pi^{-1}(j)}} f(x,y) + \frac{1}{2} \min_{\substack{x\in\pi^{-1}(i)\\ y\in\pi^{-1}(j)}} f(x,y), \tag{4}$$

which we will call the midpoint kernel. With this kernel we can define

$$\Delta_{i,j} = \frac{1}{4}\Big(\max_{x\in\pi^{-1}(i), y\in\pi^{-1}(j)} f(x,y) - \min_{x\in\pi^{-1}(i), y\in\pi^{-1}(j)} f(x,y)\Big)^2,$$

which allows us to write the bound in equation 3 as

$$\Delta = \sum_{i,j=1}^{k} \mu(\pi^{-1}(i))\mu(\pi^{-1}(j))\Delta_{i,j}.$$

Note that this is itself a discrete U-statistic over the population. The error can now be bounded through a bound on Δ. A natural way to achieve this is to uniformly bound $\Delta_{i,j}$, which can be done by assuming that the kernel function f is Lipschitz. This allows to control the error within each bin.

Lemma 2 (Quantization error for Lipschitz kernel functions)**.** *Let $\mathcal{X} = [0,1]$ and assume $f : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is L_f-Lipschitz in each input. Let the set of bins be $\{(2l-1)/2k : l \in [k]\}$ and let the quantization scheme π perform simple rounding of inputs (affecting them to the nearest bin). Then we have $(U_f - U_{A,\pi})^2 \leq L_f^2/2k^2$.*

Proof. By the Lipschitz property of f, we have that $|f(x,y) - f(x',y')| \leq L_f(|x-x'|+|y-y'|)$ for all x, x', y, y'. Since the diameter of each bin is equal to $1/k$, we have $\Delta_{i,j} \leq L_f^2/2k^2$ for all $i, j \in [k]$ and the lemma follows. ■

As desired, the quantization error decreases with k. Note that the Lipschitz assumption is met in the important case of the Gini mean difference, while it does not hold for AUC and Kendall's tau. Bounding Δ is not the right approach for such kernels: indeed, for AUC, $\Delta_{i,i} = 1/2$ and so for data distributions μ with $\mu(\pi^{-1}(i)) = 1$ for some i, the quantization error $\Delta \geq 1/2$. In the next section, we consider generic kernel functions under a smoothness assumption on the data distributions.

Remark 1 (Empirical Estimation of Δ)**.** *The quantization error Δ is a discrete U-statistic which can be estimated from the data collected to estimate U. This provides a good empirical estimate of Δ after the fact. However, as the data has to be collected before the estimate can be made it provides no guidance in choosing π (this might be addressed by a multi-round protocol, which we leave for future work). The empirical assessment of Δ may provide a tighter bound on the actual error than can be ascertained by the worst-case Lipschitz assumption.*

A.2.2 Assumption of Smooth Data Distribution

We now consider a smoothness assumption on the data distribution μ. Specifically, we assume that the density $d\mu/d\lambda$ with respect to a measure λ (which varies little on $\pi^{-1}(i)$ for all i) is C-Lipschitz.

In this case, a more sensible choice of quantized kernel is given by

$$A_{i,j}^{Avg} = \frac{1}{\lambda(\pi^{-1}(i))\lambda(\pi^{-1}(j))} \int_{\pi^{-1}(i)} \int_{\pi^{-1}(j)} f(x,y) d\lambda(y) d\lambda(x), \tag{5}$$

which we call the average kernel as the value of $A_{i,j}^{Avg}$ corresponds to the (normalized) expectation of $f(x,y)$, with respect to λ, over points x and y that are mapped to bin i and j respectively.

Under our smoothness assumption, the quantization error (2) can be bounded as follows.

Lemma 3 (Quantization error for smooth distributions)**.** *Let $\mathcal{X} = [0,1]$ and $f(x,y) \in [0,1]$ for all x, y.*[1] *Assume that $d\mu/d\lambda$ is L_μ-Lipschitz. Then we have $(U_f - U_{A,\pi})^2 \leq 4L_\mu^2 D^2(1+L_\mu^2 D^2)$, where D is the maximum diameter of the quantization bins.*

Proof. For notational convenience, let us denote $\bar{\mu}_i := \mu(\pi^{-1}(i))$ and $\bar{\lambda}_i := \lambda(\pi^{-1}(i))$ for each i. The absolute quantization error with quantized kernel (5) is given by:

$$\begin{aligned}
&\left| \sum_{i,j=1}^{k} \int_{\pi^{-1}(i)} \int_{\pi^{-1}(j)} f(x,y) - A_{i,j}^{Avg} d\mu(y) d\mu(x) \right| \\
\leq &\sum_{i,j=1}^{k} \left| \int_{\pi^{-1}(i)} \int_{\pi^{-1}(j)} f(x,y) - A_{i,j}^{Avg} d\mu(y) d\mu(x) \right| \\
= &\sum_{i,j=1}^{k} \left| \int_{\pi^{-1}(i)} \int_{\pi^{-1}(j)} f(x,y) d\mu(y) d\mu(x) - \bar{\mu}_i \bar{\mu}_j A_{i,j}^{Avg} \right| \qquad (6)
\end{aligned}$$

Note that

$$\int_{\pi^{-1}(i)} \int_{\pi^{-1}(j)} f(x,y) d\mu(y) d\mu(x) = \int_{\pi^{-1}(i)} \int_{\pi^{-1}(j)} f(x,y) \frac{d\mu(y)}{d\lambda(y)} \frac{d\mu(x)}{d\lambda(x)} d\lambda(y) d\lambda(x),$$

and

$$\bar{\mu}_i \bar{\mu}_j A_{i,j}^{Avg} = \frac{\bar{\mu}_i \bar{\mu}_j}{\bar{\lambda}_i \bar{\lambda}_j} \int_{\pi^{-1}(i)} \int_{\pi^{-1}(j)} f(x,y) d\lambda(y) d\lambda(x).$$

[1] Similar arguments can be made in more general metric spaces.

Plugging these equations into (6) we get:

$$
\begin{aligned}
&\left|\int_{\pi^{-1}(i)}\int_{\pi^{-1}(j)} f(x,y)d\mu(y)d\mu(x) - \bar{\mu_i}\bar{\mu_j}A_{i,j}^{Avg}\right| \\
=&\left|\int_{\pi^{-1}(i)}\int_{\pi^{-1}(j)} f(x,y)\Big(\frac{d\mu(y)}{d\lambda(y)}\frac{d\mu(x)}{d\lambda(x)} - \frac{\bar{\mu_i}\bar{\mu_j}}{\bar{\lambda_i}\bar{\lambda_j}}\Big)d\lambda(y)d\lambda(x)\right| \\
\leq&\int_{\pi^{-1}(i)}\int_{\pi^{-1}(j)} \left|\frac{d\mu(y)}{d\lambda(y)}\frac{d\mu(x)}{d\lambda(x)} - \frac{\bar{\mu_i}\bar{\mu_j}}{\bar{\lambda_i}\bar{\lambda_j}}\right| d\lambda(y)d\lambda(x) && (7) \\
\leq&\int_{\pi^{-1}(i)}\int_{\pi^{-1}(j)} L_\mu D\Big(\max_{z\in\pi^{-1}(i)} d\mu(z)/d\lambda(z) + \max_{w\in\pi^{-1}(j)} d\mu(w)d\lambda(w)\Big)d\lambda(y)d\lambda(x) && (8) \\
\leq&\int_{\pi^{-1}(i)}\int_{\pi^{-1}(j)} (L_\mu D)d\lambda(y)d\mu(x) + \int_{\pi^{-1}(i)}\int_{\pi^{-1}(j)} (L_\mu D)d\mu(y)d\lambda(x) \\
&+\int_{\pi^{-1}(i)}\int_{\pi^{-1}(j)} (L_\mu D)(2L_\mu D)d\lambda(y)d\lambda(x). && (9)
\end{aligned}
$$

Summing over all i,j and taking the square finally gives the result:

$$
\begin{aligned}
&\Big(\int\int f(x,y)d\mu(y)d\mu(x) - \sum \mu(\pi^{-1}(i))\mu(\pi^{-1}(j))A_{i,j}^{Avg}\Big)^2 \\
\leq& 4L_\mu^2 D^2 + 4L_\mu^4 D^4.
\end{aligned}
$$

■

The diameter of quantization bins is typically of order $1/k$, hence the quantization error is of order $1/k^2$. In practice, λ can simply be taken to be Lebesgue measure, hence computing (5) amounts to averaging the kernel function over all possible points $(x,y) \in \mathcal{X}$ that fall in the bins (i,j), and can be easily approximated by Monte Carlo sampling when one does not have a closed form expression for the integral.

B Details and Proofs for AUC Protocol

B.1 Proof of Theorem 3

We define $R^m = \{p \in \{0,1\}^m : \forall p' \preceq p, \tilde{h}^-_{p'}\tilde{h}^+_{p'} > \tau\}$ as the set of nodes recursed on at level m. Similarly, and for $m > 0$, let $A^m = R^{m-1} \cdot \{0,1\}$ be the active nodes at level m, i.e. those to be either recursed on or discarded. Then, the set of discarded nodes at level m is defined as $D^m = A^m \setminus R^m$. Our algorithm has two main sources of error: (i) the one incurred on by discarded nodes, i.e. nodes in $\bigcup_{i\in[\alpha]} D^m$ for whose intervals the algorithm uses a rough estimate, and (ii) the error in the estimating the contribution to the UAUC of the recursed nodes, i.e. nodes in $\bigcup_{i\in[\alpha]} R^m$.

The threshold τ is carefully chosen according to the error of the estimator $\hat{h}$ to balance these two errors. In this way we translate error bounds for $\hat{h}^+_p, \hat{h}^-_p$ into error bounds for $\widehat{\texttt{UAUC}}$. Our proof starts by bounding the expected size of R^m.

Lemma 4. *Consider the instantiation of Equation 10 with a frequency oracle for estimating* $h^\pm$ *satisfying* $\forall p \in \{0,1\}^{\le\alpha} : \big(\mathbb{E}(\hat{h}_p^\pm) = h_p^\pm, \mathbb{E}((\hat{h}_p^\pm - h_p^\pm)^2) \le v^\pm\big)$*, with* $v^\pm = Cn^\pm\alpha$*, i.e. the estimate is unbiased and has uniformly bounded* $\texttt{MSE}$*. If* $a > 1$*, then for all* $m \in [\alpha]$*,*

$$\mathbb{E}(|R^m|) \le \frac{\sqrt{n^+} + \sqrt{n^-}}{2(\sqrt{a}-1)\sqrt{C\alpha}} \le \frac{1}{\sqrt{a}-1}\sqrt{\frac{n}{2C\alpha}}$$

Proof. Let $\hat{n}^\pm = \sum_{p\in A^m} \max(\hat{h}_p^\pm, 0)$, the sum of the positive estimated counts of active nodes at level m.

Note that if $p \in R^m$ then $\tilde{h}_p^+\tilde{h}_p^- \ge a\sqrt{v^+v^-}$. In this case either, $\hat{h}_p^\pm = \tilde{h}_p^\pm$ and thus $\hat{h}_p^+\hat{h}_p^- \ge a\sqrt{v^+v^-}$, $\hat{h}_p^- \ne \tilde{h}_p^-$ and thus $\hat{h}_p^+ > 2\sqrt{av^+}$, or $\hat{h}_p^- \ne \tilde{h}_p^-$ and thus $\hat{h}_p^- > 2\sqrt{av^-}$. In any of these cases $\hat{h}_p^+/\sqrt{av^+} + \hat{h}_p^-/\sqrt{av^-} \ge 2$.

Therefore

$$2|R^m| \le \sum_{p\in R^m} \frac{\hat{h}_p^+}{\sqrt{av^+}} + \frac{\hat{h}_p^-}{\sqrt{av^-}} \le \frac{\hat{n}^+}{\sqrt{av^+}} + \frac{\hat{n}^-}{\sqrt{av^-}}$$

and thus

$$\mathbb{E}(|R^m|) \le \frac{\mathbb{E}(\hat{n}^+)}{2\sqrt{av^+}} + \frac{\mathbb{E}(\hat{n}^-)}{2\sqrt{av^-}}.$$

We bound $\mathbb{E}(\hat{n}^\pm)$ as follows

$$\begin{aligned}\mathbb{E}(\hat{n}^\pm) = &\sum_{p\in A^m} \mathbb{E}(\max(\hat{h}_p^\pm, 0)) \le n^\pm + \sum_{p\in A^m} \mathbb{E}(\max(e_p^\pm, 0)) \\ \le & n^\pm + \mathbb{E}(|A^m|) \max_{p\in A^m} \mathbb{E}(\max(e_p^\pm, 0)) \le n^\pm + \mathbb{E}(|R^{m-1}|) \max_{p\in A^m} \mathbb{E}(|e_p^\pm|) \\ \le & n^\pm + \mathbb{E}(|R^{m-1}|) \max_{p\in A^m} \sqrt{\mathbb{E}(|e_p^\pm|^2)} \le n^\pm + \mathbb{E}(|R^{m-1}|)\sqrt{v_\pm}\end{aligned}$$

We can now use this to bound the expression for $\mathbb{E}(|R^m|)$.

$$\mathbb{E}(|R^m|) \le \frac{n^+}{2\sqrt{av^+}} + \frac{n^-}{2\sqrt{av^-}} + \frac{\mathbb{E}(|R^{m-1}|)}{\sqrt{a}} \tag{10}$$

We now need a bound on $\mathbb{E}(|R^{m-1}|)$ so we will proceed by induction.

Let $B = \frac{\sqrt{n^+}+\sqrt{n^-}}{2(\sqrt{a}-1)\sqrt{C\alpha}}$. We take $\mathbb{E}(|R^{m-1}|) \le B$ as the induction hypothesis, and $\mathbb{E}(|R^0|) = 1 \le B$ as the base case.

The expression on the right hand side of inequality 10 is a monotonically increasing function of $\mathbb{E}(|R^{m-1}|)$ and has a fixed point

$$\frac{\frac{n^+}{2\sqrt{av^+}} + \frac{n^-}{2\sqrt{av^-}}}{1 - \frac{1}{\sqrt{a}}} = \frac{\frac{n^+}{2\sqrt{v^+}} + \frac{n^-}{2\sqrt{v^-}}}{\sqrt{a}-1} = \frac{\sqrt{\frac{n^+}{C\alpha}} + \sqrt{\frac{n^-}{C\alpha}}}{2(\sqrt{a}-1)} = \frac{\sqrt{n^+}+\sqrt{n^-}}{2(\sqrt{a}-1)\sqrt{C\alpha}} = B.$$

Thus we can conclude that

$$\mathbb{E}(|R^m|) \le B \tag{11}$$

completing the induction and thus (11) holds for all m.

Finally we note that

$$\sqrt{n^+} + \sqrt{n^-} \leq \sqrt{2n}$$

and so

$$\frac{\sqrt{n^+} + \sqrt{n^-}}{2(\sqrt{a}-1)\sqrt{C\alpha}} \leq \frac{1}{\sqrt{a}-1}\sqrt{\frac{n}{2C\alpha}}$$

completing the proof. ■

We are now ready to prove Theorem 3.

Proof of Theorem 3. The estimation error $E_p = \widehat{\texttt{UAUC}}(\hat{h}_p^+, \hat{h}_p^-) - \texttt{UAUC}(h_p^+, h_p^-)$ at a given node p can be written recursively as follows:

$$E_p = \begin{cases} \frac{1}{2}(\hat{h}_{p\cdot 1}^+ + \hat{h}_{p\cdot 0}^+)(\hat{h}_{p\cdot 1}^- + \hat{h}_{p\cdot 0}^-) - \texttt{UAUC}(h_p^+, h_p^-) & \text{if } p \in D \\ 0 & \text{if } p \text{ is a leaf} \\ \hat{h}_{p\cdot 1}^+ \hat{h}_{p\cdot 0}^- - h_{p\cdot 1}^+ h_{p\cdot 0}^- + E_{p\cdot 0} + E_{p\cdot 1} & \text{if } p \in R \end{cases}$$

We will consider the error E_λ in two parts. Firstly, there is the contribution E^D from those prefixes $p \in D$ which we define by setting

$$E_m^D = \sum_{p \in D_{m-1}} \frac{1}{2}(\hat{h}_{p\cdot 1}^+ + \hat{h}_{p\cdot 0}^+)(\hat{h}_{p\cdot 1}^- + \hat{h}_{p\cdot 0}^-) - \texttt{UAUC}(h_p^+, h_p^-)$$

and $E^D = \sum_{m \in [\alpha]} E_m^D$. Secondly, there is the contribution from the prefixes $p \in R$ excluding their recursive subcalls which we define by setting

$$E_m^R = \sum_{p \in R_{m-1}} \hat{h}_{p\cdot 1}^+ \hat{h}_{p\cdot 0}^- - h_{p\cdot 1}^+ h_{p\cdot 0}^-$$

and $E^R = \sum_{m \in [\alpha]} E_m^R$. In bounding both of these we will make use of conditioning on $\mathcal{F}_m = (\hat{h}_p^-, \hat{h}_p^+)_{p \in \{0,1\}^{\leq m}}$ i.e. the answers of the frequency oracles for layers up to m.

We start by bounding E^R. For any $m \in [\alpha]$, we first show that E_m^R is a martingale difference sequence i.e.

$$\begin{aligned} \mathbb{E}(E_m^R | \mathcal{F}_{m-1}) &= \mathbb{E}(\sum_{p \in R^{m-1}} (\hat{h}_{p\cdot 1}^+ \hat{h}_{p\cdot 0}^- - h_{p\cdot 1}^+ h_{p\cdot 0}^-) | \mathcal{F}_{m-1}) \\ &= \sum_{p \in R^{m-1}} \mathbb{E}((h_{p\cdot 1}^+ + e_{p\cdot 1}^+)(h_{p\cdot 0}^- + e_{p\cdot 0}^-) - h_{p\cdot 1}^+ h_{p\cdot 0}^-) \\ &= \sum_{p \in R^{m-1}} \mathbb{E}(h_{p\cdot 1}^+ e_{p\cdot 0}^- + e_{p\cdot 0}^+ h_{p\cdot 0}^- + e_{p\cdot 1}^+ e_{p\cdot 0}^-) \\ &= 0 \end{aligned}$$

where the final equality holds because $\mathbb{E}(e_p^\pm) = 0$ for all p and $e_{p\cdot 1}^+$ and $e_{p\cdot 0}^-$ are independent. From this we can conclude that for $m' > m$

$$\mathbb{E}(E_m^R E_{m'}^R) = \mathbb{E}(\mathbb{E}(E_m^R E_{m'}^R | \mathcal{F}_{m'-1}))) = \mathbb{E}(E_m^R \mathbb{E}(E_{m'}^R | \mathcal{F}_{m'-1})) = \mathbb{E}(0) = 0$$

and thus

$$\mathbb{E}(E^{R^2}) = \mathbb{E}(\sum_{m\in[\alpha]}\sum_{m'\in[\alpha]} E^R_m E^R_{m'}) = \sum_{m\in[\alpha]} \mathbb{E}(E^{R^2}_m) = \sum_{m\in[\alpha]} \mathbb{E}(\mathbb{E}(E^{R^2}_m|\mathcal{F}_{m-1})). \tag{12}$$

Next we shall bound $\mathbb{E}(E^{R^2}_m|\mathcal{F}_{m-1})$. We start by writing out

$$\mathbb{E}(E^{R^2}_m|\mathcal{F}_{m-1}) = \mathbb{E}((\sum_{p\in R^{m-1}} (\hat{h}^+_{p\cdot 1}\hat{h}^-_{p\cdot 0} - h^+_{p\cdot 1}h^-_{p\cdot 0}))^2|\mathcal{F}_{m-1}).$$

By Equation 12 this becomes

$$\mathbb{E}(E^{R^2}_m|\mathcal{F}_{m-1}) = \sum_{p\in R^{m-1}} \mathbb{E}((\hat{h}^+_{p\cdot 1}\hat{h}^-_{p\cdot 0} - h^+_{p\cdot 1}h^-_{p\cdot 0})^2|\mathcal{F}_{m-1}).$$

After expanding the above and removing all the terms that are zero, because they are the expected value of the product of $e^{\pm}_{p\cdot i}$ with something independent of it, we are left with

$$\begin{aligned}
\mathbb{E}(E^{R^2}_m|\mathcal{F}_{m-1}) &= \sum_{p\in R^{m-1}} \mathbb{E}({h^+_{p\cdot 1}}^2 {e^-_{p\cdot 0}}^2 + {e^+_{p\cdot 0}}^2 {h^-_{p\cdot 0}}^2 + {e^+_{p\cdot 1}}^2 {e^-_{p\cdot 0}}^2) \\
&\le \sum_{p\in R^{m-1}} (v^- {h^+_{p\cdot 1}}^2 + v^+ {h^-_{p\cdot 0}}^2) + |R^m|v^+v^- \\
&\le v^- {n^+}^2 + v^+ {n^-}^2 + |R^m|v^+v^-.
\end{aligned}$$

Subbing this into (12) and using Lemma 4 gives

$$\begin{aligned}
\mathbb{E}(E^{R^2}) &\le \alpha \max_m \mathbb{E}(v^- {n^+}^2 + v^+ {n^-}^2 + |R^m|v^+v^-) \\
&= \alpha(v^- {n^+}^2 + v^+ {n^-}^2 + \max_m \mathbb{E}(|R^m|)v^+v^-) \\
&\le n^+n^- C\alpha^2(n^+ + n^- + \frac{C\alpha}{\sqrt{a}-1}\sqrt{\frac{n}{2C\alpha}}) \\
&\le n^+n^- C\alpha^2(n + \frac{\sqrt{C\alpha n}}{\sqrt{2}(\sqrt{a}-1)}) \\
&=: B^R
\end{aligned}$$

To bound E^D, first define $E^F_m = \sum_{p\in D_{m-1}} \frac{1}{2}(\hat{h}^+_{p\cdot 1} + \hat{h}^+_{p\cdot 0})(\hat{h}^-_{p\cdot 1} + \hat{h}^-_{p\cdot 0}) - \frac{1}{2}h^+_p h^-_p$ and $E^G_m = \sum_{p\in D_m} \frac{1}{2}h^+_p h^-_p - \mathtt{UAUC}(h^+_p, h^-_p)$. We refer to the leaves in $[0..2^\alpha - 1]$ *covered* by a path p as $\mathcal{I}(p) = \{i \in [0..d-1] : p \preceq b_i\}$. Now note that

$$E^D = \sum_{p\in D} \frac{1}{2}(\hat{h}^+_{p\cdot 1} + \hat{h}^+_{p\cdot 0})(\hat{h}^-_{p\cdot 1} + \hat{h}^-_{p\cdot 0}) - \sum_{i\in\mathcal{I}(p)} h^+_{b_i} \sum_{j\in\mathcal{I}(p), j<i} h^-_{b_j} = \sum_{m\in[\alpha]} E^F_m + \sum_{m\in[\alpha]} E^G_m.$$

We now bound E^F_m and E^G_m separately. For a leaf node s, let us denote by $v(s)$ the *unique* node in D that is a prefix of s. We then have:

$$\begin{aligned}\mathbb{E}((\sum_{m\in[\alpha]} E^G_m)^2) &= \mathbb{E}((\sum_{p\in D} \frac{1}{2} h^+_p h^-_p - \texttt{UAUC}(h^+_p, h^-_p))^2) \\ &\leq \mathbb{E}((\sum_{p\in D} \frac{1}{2} h^+_p h^-_p)^2) = \frac{1}{4}\mathbb{E}((\sum_{p\in D}\sum_{i\in\mathcal{I}(p)} h^+_{b_i} \sum_{j\in\mathcal{I}(p), j<i} h^-_{b_j})^2) \\ &\leq \frac{n^{+2}}{4} \max_{s\in[0..d-1]} \mathbb{E}((h^-_{v(s)})^2).\end{aligned}$$

We can then bound

$$\begin{aligned}\mathbb{E}({h^-_{v(s)}}^2) = \sum_{p\preceq p(s)} \mathbb{E}({h^-_p}^2 \mathbb{I}_{p=v(s)}) &= \sum_{p\preceq p(s)} \mathbb{E}((\hat{h}^-_p - e^-_p)^2 \mathbb{I}_{p=v(s)}) \\ \leq &\sum_{p\preceq s} \mathbb{E}((2\sqrt{av^-} - e^-_p)^2 \mathbb{I}_{p=v(s)}) \leq \sum_{p\preceq s} \mathbb{E}(4av^- - 4\sqrt{av^-} e^-_p + {e^-_p}^2) \\ \leq &(4a+1)\alpha v^-.\end{aligned}$$

Thus

$$\mathbb{E}({E^G}^2) \leq n^{+2}(a+1/4)\alpha v^- \leq C(a+1/4)n^- n^{+2}\alpha^2.$$

Furthermore by symmetry between $-$ and $+$

$$\mathbb{E}({E^G}^2) \leq C(a+1/4)n^- n^+ \min(n^-, n^+)\alpha^2 =: B^G.$$

Secondly we bound $\sum_{m\in[\alpha]} E^F_m$. Note that E^F_{m-1} is a function of $\mathcal{F}_{m-1}$ and $\mathbb{E}(E^F_m|\mathcal{F}_{m-1}) = 0$ so

$$\begin{aligned}\mathbb{E}((\sum_m E^F_m)^2) &= \mathbb{E}(\sum_m {E^F_m}^2) = \sum_m \mathbb{E}({E^F_m}^2) = \sum_m \mathbb{E}(\mathbb{E}({E^F_m}^2|\mathcal{F}_{m-1})) \\ &\leq \sum_m \mathbb{E}(\mathbb{E}((\sum_{p\in D_{m-1}} \frac{1}{2}(\hat{h}^+_{p\cdot 1} + \hat{h}^+_{p\cdot 0})(\hat{h}^-_{p\cdot 1} + \hat{h}^-_{p\cdot 0}) - \frac{1}{2} h^+_p h^-_p)^2|\mathcal{F}_{m-1}))\end{aligned}$$

Similarly to the bound on E^R, we now apply the pairwise independence property and note that $\hat{h}^\pm_{p\cdot 1} + \hat{h}^\pm_{p\cdot 0}$ is an unbiased estimator of $h^\pm_p$ with variance bounded by $2v^\pm$. This results in

$$\begin{aligned}\mathbb{E}((\sum_m E^F_m)^2) &\leq \sum_m \mathbb{E}(\sum_{p\in A_{m-1}} \mathbb{I}_{\tilde{h}^+_p \tilde{h}^-_p < \tau} \mathbb{E}({h^+_p}^2 v^-/2 + v^+ {h^-_p}^2/2 + v^+ v^- |\mathcal{F}_{m-1})) \\ &\leq \sum_m v^- \mathbb{E}(\sum_{p\in A_{m-1}} \mathbb{I}_{\tilde{h}^+_p \tilde{h}^-_p < \tau} {h^+_p}^2)/2 + v^+ \mathbb{E}(\sum_{p\in A_{m-1}} \mathbb{I}_{\tilde{h}^+_p \tilde{h}^-_p < \tau} {h^-_p}^2)/2 + v^+ v^- \mathbb{E}(|A_{m-1}|).\end{aligned}$$

Noting that $\mathbb{E}(\mathbb{I}_{\tilde{h}^+_p \tilde{h}^-_p < \tau} {h^+_p}^2) \leq \min({h^+_p}^2, \frac{{h^+_p}^2 v^+}{(h^+_p - \sqrt{v^+})^2}) \leq 4v^+$ and that $\mathbb{E}(|A_{m-1}|) = 2\mathbb{E}(|R_{m-2}|) \leq \frac{1}{\sqrt{a}-1}\sqrt{\frac{n}{2C\alpha}}$ gives

$$\mathbb{E}((\sum_m E^F_m)^2) \leq \sum_m E(|A_{m-1}|) 5v^+ v^- \leq \frac{5\sqrt{2n}C^{1.5}\alpha^{2.5} n^+ n^-}{\sqrt{a}-1} := B^F.$$

By the Cauchy-Schwarz inequality we can conclude that,

$$\mathbb{E}(E^{D^2}) \leq 2(B^G + B^F).$$

Finally applying Cauchy-Schwarz again gives

$$\begin{aligned}\mathbb{E}(E_\lambda^2) &= \mathbb{E}((E^R + E^G + E^F)^2)\\ &\leq 2B^R + 4B^G + 4B^F\\ &= Cn^-n^+\alpha^2(2n + (4a+1)\min(n^-, n^+) + \frac{21\sqrt{2nC\alpha}}{\sqrt{a}-1})\end{aligned}$$

■

Remark 2. *The use of Cauchy-Schwarz to combine the separate errors in this proof is optimized for simplicity rather than minimizing the constants. At the expense of making the bound substantially more complicated a more precise analysis would reduce the bound. Gaining up to a factor of two in the case of very large n and min(n^-, n^+) small compared to n.*

The value of a in Theorem 3 can be chosen to minimize the error by taking it to solve $\sqrt{a}(\sqrt{a}-1)^2 = 21\sqrt{2nc\alpha}/(8\min(n^-, n^+))$ which is approximately

$$a = (1 + \sqrt{21/8}(2Cn\alpha/\min(n^-, n^+)^2)^{\frac{1}{4}})^2.$$

This leads to the following corollary.

Corollary 3. *Let $n_{\min} = \min(n^-, n^+)$ and $a = (1 + \sqrt{21/8}(2Cn\alpha/n_{\min}^2)^{\frac{1}{4}})^2 = 1 + o(1)$ then*

$$\mathtt{MSE}(\widehat{\mathtt{AUC}}) \leq \frac{C}{n_{\min}(n - n_{\min})}\alpha^2(2n + (4a+1)n_{\min} + 14(Cnn_{\min}^2\alpha)^{\frac{1}{4}}) = O(\alpha^2/n_{\min}).$$

Remark 3. *For fixed α this is of the same order as the sampling error incurred in non-private AUC.*

Algorithm variant. An alternative algorithm assigns a value of zero to edges that it discards. For this algorithm a similar theorem holds by the same argument (actually a slightly simpler argument) the resulting error bound is

$$\mathtt{MSE}(\widehat{\mathtt{UAUC}}) = Cn^-n^+\alpha^2(2n + (8a+2)\min(n^-, n^+) + \frac{\sqrt{2C\alpha n}}{(\sqrt{a}-1)}).$$

Note that the second term which is of leading order for $\min(n^-, n^+)$ a fixed fraction of n is twice as large however the final term which is lower order is twenty-one times smaller. This lower order term might not be negligible in practice and so this algorithm should be considered. The corresponding choice of a and bound on the final error is given by the following result.

Corollary 4. *Let $n_{\min} = \min(n^-, n^+)$ and $a = (1 + \sqrt{\frac{\sqrt{2C\alpha n}}{16n_{\min}}})^2 = 1 + o(1)$ then*

$$\mathtt{MSE}(\widehat{\mathtt{AUC}}) \leq \frac{C}{n_{\min}(n - n_{\min})}\alpha^2((8a+2)n_{\min} + 2n + (512C\alpha nn_{\min}^2)^{\frac{1}{4}}) = O(\alpha^2/n_{\min})$$

Algorithm 1: Local Randomizer

```
Public Parameters: Domain size $2^l$, privacy budget $\epsilon$.
Input: Private index $q$
Output: A single bit $z$ submitted to the server
$j \leftarrow [0..2^l - 1]$    ▷ Selected uniformly at random
$y := \frac{1}{\sqrt{2^l}}(-1)^{\langle j,q \rangle}$    ▷ $y$ is $M_{j,x^l}$, where $M \in \{-1,1\}^{2^l \times 2^l}$ is a Hadamard matrix
$z := \begin{cases} y & \text{with probability } \frac{e^\epsilon}{1+e^\epsilon} \\ \neg y & \text{otherwise} \end{cases}$    ▷ Submit randomized response on $y$
Send $j, z$ to the Aggregator
```

B.2 Instantiating the Private Hierarchical Histogram $\hat{h}$

Theorem 3 does not yield a complete algorithm as it only states that, if we had a differentially private algorithm for computing estimates of a hierarchical histogram that satisfy the conditions of Theorem 3, then we could solve AUC with the stated accuracy. In this section we instantiate such algorithm and show that, besides the required error guarantees, our proposal also has other nice properties, namely (i) it is one round, (ii) each user sends a single bit, and (iii) it is sublinear in d processing space at the server.

B.2.1 Frequency Oracle

Relevant previous work on estimating hierarchical histograms in the local model includes the work of Bassily et al. (2017). While in that work the target problem is heavy hitters, their algorithm is similar to ours, as the server retrieves the heavy hitters by exploring a hierarchical histogram. Moreover their protocol – called `TreeHist` – has the nice properties listed above, as it is one round, every user sends a single bit and requires reconstruction space sublinear in d. This satisfies the three above conditions. It is thus tempting to reuse the hierarchical histogram construction from Bassily et al. (2017). However, it does not satisfy the conditions of Theorem 3, as it is not guaranteed to be unbiased.

Alternative recent algorithms for constructing hierarchical histograms in the local model are presented in Kulkarni et al. (2019), with the motivation of answering range queries over a large domain. This proposal is much closer to what we need. However, it has some shortcomings: first, although it is one round, each user sends $O(\log(d))$ bits, and more importantly, it requires space $O(d)$ space at the server, as it reconstructs the whole hierarchical histogram. However, one can tweak the protocol from Kulkarni et al. (2019) to overcome these limitations. We shall first split the users into $\log(d)$ groups (one for each level) and then for each level we shall apply the frequency oracle. Algorithm 1 and Algorithm 2 show the local randomizer (user side) and frequency oracle (server side) for each histogram.

Let count(q) be the true count of an index q in a histogram. The following lemma is shown in Kulkarni et al. (2019).

Lemma 5. *The frequency oracle, Algorithm 2, run with n_l users is unbiased $\mathbb{E}(z_q) = count(q)$*

Algorithm 2: Frequency Oracle

```
Public Parameters: Domain size $2^l$, privacy budget $\epsilon$.
Input: The index $j_i$ and response $z_i$ of each party $i$ and an index $q$ to estimate the
       frequency of
Output: $z$ an estimated count of $q$
For all $i$, $y_i := \frac{1}{\sqrt{2^l}}(-1)^{\langle j_i, q\rangle}$    ▷ $y_i$ is $M_{j_i,q}$, where $M \in \{-1,1\}^{2^l \times 2^l}$ is a
  Hadamard matrix
$z_q := \frac{e^\epsilon+1}{e^\epsilon-1}\sum_i y_i z_i$    ▷ De-bias the sum of contributions
Return $z_q$
```

and satisfies the following bound on the MSE*.*

$$\mathbb{E}((z_q - count(q))^2) \leq \frac{4n_l e^\epsilon}{(e^\epsilon - 1)^2} \tag{13}$$

Additionally we require the following lemma on the frequency oracle satisfies condition (3) in Theorem 3 which is given by the following lemma.

Lemma 6. *For distinct $q, q' \in [0..2^l - 1]$, z_q and $z_{q'}$ are independent i.e. the responses of the oracle are pairwise independent.*

Proof. As each user is independent of every other user it suffices to show that each user's contribution to the two entries are independent. Suppose that a user has input $q'' \neq q'$, chooses index j to report and let b be a bit indicating that the user chose $z = \neg y$ in Algorithm 1. That user's contributions to the two estimates (scaled by 2^l) are $(-1)^{\langle j,q\rangle+\langle j,q''\rangle+b}$ and $(-1)^{\langle j,q'\rangle+\langle j,q''\rangle+b}$. Note that we can consider j, q, q' and q'' as elements of $\mathbb{F}_2^l$. Then $q + q''$ and $q' + q''$ are distinct and $q' + q'' \neq 0$. These two facts imply respectively that $\langle j, q' + q''\rangle$ is independent of $\langle j, q + q''\rangle$ and that $\langle j, q' + q''\rangle$ is uniformly distributed in $\mathbb{F}_2$. Thus the contributions are independent. ■

B.2.2 Splitting Strategies

We will instantiate $\hat{h}$ by running the frequency oracle above for each level of the hierarchy. The main choice remaining is how to determine which users contribute to each layer, we will consider two possibilities here. Firstly we can have everyone contribute to all layers, splitting their privacy budget. Alternatively, users can be split evenly across levels at random, each contributing to only one frequency oracle. Another possibility is to assign each user to a level independently and uniformly, this is similar to splitting them evenly though adds slightly more noise and is more complicated to analyse. In all cases, conditions 1 and 3 in Theorem 3 follow from Lemmas 5 and 6.

Splitting Privacy Budget Across Layers. In the case of everyone contributing to all layers the privacy budget can be split using either basic or advanced composition. In either case condition 4 from Theorem 3 holds as the randomness for each layer is independent.

For pure differential privacy we must use basic composition. This allows us to run each frequency oracle can be run with a privacy budget of $\tilde{\epsilon} = \epsilon/\alpha$. Lemma 5 then gives a bound of $O_\epsilon(n\alpha^2)$ on the mean squared error of each entry. While this is insufficient to establish condition 2 of Theorem 3, similar arguments can be used to prove that the algorithm built in this way achieves pure differential privacy at the cost of an α factor in the $\texttt{MSE}$.

If we instead settle for (ϵ, δ)-differential privacy, and assume for convenience that $\epsilon \leq \sqrt{\alpha}\ln(2)$, advanced composition allows each frequency oracle to be run with privacy budget $\tilde{\epsilon} = \epsilon/(\sqrt{\alpha}(1 + \sqrt{2\log(1/\delta)}))$. Condition 2 in Theorem 3 then holds for some C depending on ϵ and δ. This is the implementation and analysis that gives Theorem 3 as it is stated.

Splitting Users Across Layers. When splitting users across levels the frequency oracles can each be run with privacy budget ϵ. However, each oracle will have only n/α users and there is a subsampling error between the total sample and the input given to the frequency oracle. The squared error due to subsampling is $O(1/n)$ thus Lemma 5 provides a $O_\epsilon(n\alpha)$ bound on the $\texttt{MSE}$. This means that condition 2 of Theorem 3 holds. This would provide a version of Theorem 3 with pure differential privacy, however condition 4 from Theorem 3 fails to hold. Intuitively this is because if a user contributes to one level they can't contribute to another level. There are still two things that can be proved about this version of the algorithm.

Firstly, it is still possible to prove a result like Theorem 3, but in which the $\texttt{MSE}$ is α times bigger. The proof of this result follows the same steps as that of Theorem 3 except that the martingale difference sequences argument must be replaced by a bound not assuming pairwise independence.

A second way of viewing this algorithm is to think of each input as being drawn independently from some population distribution and then compare the output to the AUC of that distribution. That is, given a pair of distributions $\mathcal{D}^\pm$, $\mathcal{S}^\pm$ is obtained by sampling each value independently from $\mathcal{D}^\pm$. Denote $\texttt{AUC}_{\text{pop}} = \mathbb{E}_{x^+\sim\mathcal{D}^+, x^-\sim\mathcal{D}^-}\left[f(x^+, x^-)\right]$ and let $\texttt{MSE}_{\text{pop}}(\widehat{\texttt{AUC}}) = \mathbb{E}[(\texttt{AUC}_{\text{pop}} - \widehat{\texttt{AUC}})^2]$. The fact that each of the users has an independent identically distributed input means that the contribution to each layer is independent, i.e. we can recover Theorem 3 with $\texttt{MSE}$ replaced by $\texttt{MSE}_{\text{pop}}$. This alternative notion of $\texttt{MSE}_{\text{pop}}$ is the correct notion to work with if the purpose of the deployment of the algorithm is to find the AUC of the population the sample is drawn from rather than just of the sample. This is likely to be the case in many applications.

Summary. Table 1 summarizes the choices in the algorithm and analysis. The resulting orders of the $\texttt{MSE}$, corresponding to Theorem 4 are given in the final column.

C Details and Proofs for 2PC Protocol

C.1 Proof of Theorem 5 and Discussion

Proof. The ϵ-DP follows from the Laplace mechanism and the simple composition property of DP (observing that each input x_i appears in exactly P pairs in $\mathcal{P}$).

Splitting	Analysis	Error in $\widehat{\texttt{AUC}}$
Privacy budget	Basic composition	$\texttt{MSE} \leq O(\frac{\alpha^3}{n\epsilon^2})$
Privacy budget	Advanced composition	$\texttt{MSE} \leq O(\frac{\alpha^2 \log(1/\delta)}{n\epsilon^2})$
Users	w.r.t. sample	$\texttt{MSE} \leq O(\frac{\alpha^3}{n\epsilon^2})$
Users	w.r.t. population	$\texttt{MSE}_{\text{pop}} \leq O(\frac{\alpha^2}{n\epsilon^2})$

Table 1: Summary of error bounds for our AUC protocol for different splitting strategies and analysis techniques.

It is easy to see that $\widehat{U}_{f,n}$ is unbiased, hence we only need to bound its variance. We will separate the part due to subsampling and the part due to privacy. To this end, we decompose $\widehat{U}_{f,n}$ into a noise-free term and a noisy term:

$$\widehat{U}_{f,n} = \underbrace{\frac{2}{Pn} \sum_{(i,j)\in\mathcal{P}} f(x_i, x_j)}_{\widehat{U}_{f,n,P}} + \frac{2}{Pn} \sum_{(i,j)\in\mathcal{P}} \eta_{ij}. \tag{14}$$

The noisy term is an average of independent Laplace random variables: its variance is equal to $2P/n\epsilon^2$.

The quantity $\widehat{U}_{f,n,P}$ is known as an incomplete U-statistic, whose variance is given by Blom (1976):

$$\mathrm{Var}(\widehat{U}_{f,n,P}) = \frac{4}{(Pn)^2}\Big(\mathbb{E}[f_1(\mathcal{P})]\zeta_1 + \mathbb{E}[f_2(\mathcal{P})]\zeta_2\Big) \tag{15}$$

where $\zeta_1 = \mathrm{Var}(f(x_1, X_2) \mid x_1))$, $\zeta_2 = \mathrm{Var}(f(X_1, X_2)$, and $f_1(\mathcal{P}), f_2(\mathcal{P})$ are the number of members of $\mathcal{P} \times \mathcal{P}$ which have exactly 1 (respectively 2) indices in common.

We first consider $\mathbb{E}[f_2(\mathcal{P})]$. Recall that $\mathcal{P}$ is constructed from P permutations $\sigma_1, \ldots, \sigma_P$ of the set $\{1, \ldots, n\}$. As each index appears exactly once in each permutation, it suffices to consider the self pairs within permutations and the overlaps across pairs of permutations we get:

$$\mathbb{E}[f_2(\mathcal{P})] = \sum_{i<j} \sum_{p=1}^{P} \mathbb{E}\Big[q_{ij}^p(\mathcal{P})\big(1 + \sum_{p'\neq p} q_{ij}^{p'}(\mathcal{P})\big)\Big],$$

where $q_{ij}^p(\mathcal{P})$ is the number of pairs from the permutation p that contain $\{i, j\}$. The probability of a pair (i, j) to appear in a given permutation is $1/(n-1)$, hence using the independence between permutations we obtain:

$$\mathbb{E}[f_2(\mathcal{P})] = \frac{n(n-1)}{2} \frac{P}{n-1}\Big(1 + \frac{P-1}{n-1}\Big) = \frac{Pn}{2}\Big(1 + \frac{P-1}{n-1}\Big).$$

For $\mathbb{E}[f_1(\mathcal{P})]$, using a similar reasoning we only have to consider overlaps across each pair of permutations, in which each index pair shares exactly one index with a single pair of another permutation, except when the pair appears twice, hence:

$$\begin{aligned}\mathbb{E}[f_1(\mathcal{P})] &= \sum_{(i,j)\in\mathcal{P}} 2(P-1) - 2\sum_{i<j}\sum_{p=1}^{P}\sum_{p'\neq p}\mathbb{E}\big[q_{ij}^{p}(\mathcal{P})q_{ij}^{p'}(\mathcal{P})\big],\\ &= P(P-1)n - n(n-1)\frac{P(P-1)}{(n-1)^2}\\ &= P(P-1)n\Big(1-\frac{1}{n-1}\Big)\end{aligned}$$

Putting everything together into (15) we get the desired result. ■

Optimal value of P. The optimal value of P depends on the kernel function, the data distribution and the privacy budget. Roughly speaking, setting P larger than 1 can be beneficial when ζ_2 is large compared to $1/\epsilon^2$. On the other hand, when $\zeta_2 = 2\zeta_1$ (which is the minimum value of ζ_2, corresponding to the extreme case where the kernel can in fact be rewritten as a sum of univariate functions (Blom, 1976)), $\mathrm{Var}(\widehat{U}_{f,n})$ simplifies to $\frac{4\zeta_1}{n} + \frac{2P}{n\epsilon^2}$ and $P = 1$ is optimal. In practice and as illustrated in our experiments, P should be set to a small constant.

Optimality of subsampling schemes. The proposed subsampling strategy is simple to implement and leads to an optimal variance, up to an additive term of $\frac{2}{Pn}\frac{P-1}{n-1}(\zeta_2 - 2\zeta_1) \geq 0$, among unbiased approximations based on $Pn/2$ pairs. Note that this additive term is 0 when $P = 1$ or $\zeta_2 = 2\zeta_1$, and is in general negligible compared to the dominating terms for small enough P. Optimal variance could be achieved at the cost of a more involved sampling scheme.[2] Alternatively, sampling schemes that can be run independently by each user without global coordination (such as sampling $P/2$ other users uniformly at random) lead to a slight increase in variance as users are not guaranteed to appear evenly across the sampled pairs.

C.2 Implementing 2PC

MPC is a subfield of cryptography concerned with the general problem of computing on private distributed data in a way in which only the result of the computation is revealed to the parties, and nothing else. In this paper the number of parties is limited to 2, and the function to be computed is $\widetilde{f}(x, y)$. There are several protocols that allow to achieve this goal, with different trade-offs in terms of security, round complexity, and also differing on how the functionality $\widetilde{f}$ is represented. These alternatives include Yao's garbled circuits (Yao, 1986; Lindell and Pinkas, 2009), the GMW protocol (Goldreich et al., 1987), and the SPDZ protocol (Damgård et al., 2011), among others. As some of the functions $\widetilde{f}$ we are interested in involve comparisons (e.g., Gini mean difference and AUC), a Boolean representation is more suitable, as it will lead to a smaller circuit. Moreover, a constant round protocol is preferred in our setting, as as users might have limited connectivity. For this reason we choose garbled circuits as our protocol, for which (Evans et al., 2018) give a detailed description

[2] In addition to having each data point appear the same number of times in $\mathcal{P}$, one must ensure that no pair appears more than once.

including crucial practical optimizations. Moreover, we assume semi-honest adversaries in the sequel (see Goldreich, 2004, for a definition of this threat model).

Circuits for kernels. We illustrate the main ideas on Gini mean difference and AUC. As circuits for floating point arithmetic are large, they are usually avoided in MPC, to instead rely on fixed point encodings. Hence, we assume that the parties have agreed on a precision, and hence x, y are integers encoded in two's complement.

For Gini mean difference we need our 2PC protocol to compute $\texttt{fgini}(x, y) := |x - y|$. Let z be $x - y$, let $z_{k-1}, \ldots, z_0$ be the binary encoding of z, where the bitwidth k will be a constant such as 32 or 64 in practice, and let $s = z_{k-1} \cdots z_{k-1}$ be the sign bit of z replicated k times. Then $\texttt{fgini}(x, y)$ can be computed as $(z + s) \oplus s$ and, thanks to the free-XOR optimization of garbled circuits (see Evans et al., 2018), the garble circuit evaluation requires only a subtraction and a summation, and thus is very efficient.

For AUC we need our 2PC protocol to compute $\texttt{fauc}(x, y) := x < y$, which requires a single comparison and thus a small number of binary gates to be evaluated in a garbled circuit.

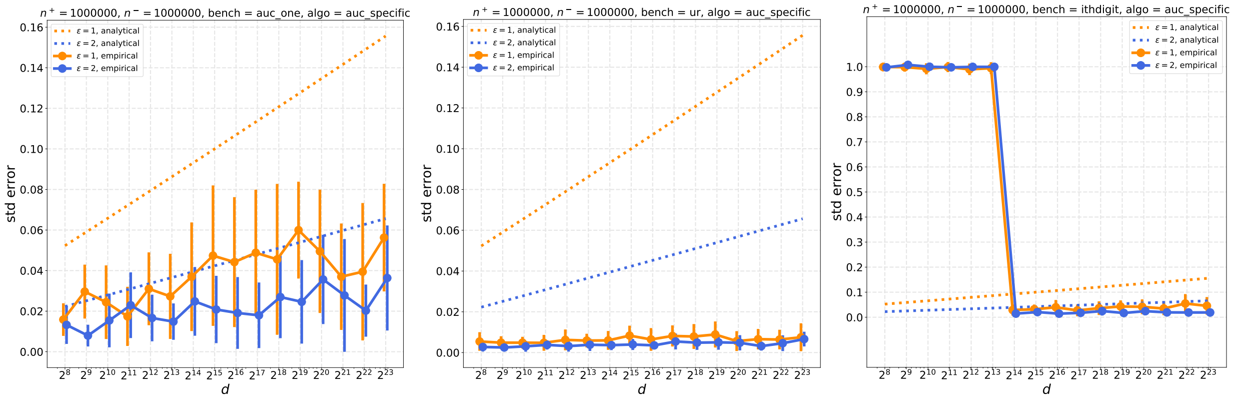
Circuits for local randomizers. The above circuits need to be extended with output perturbation corresponding to the Laplace and randomized response mechanisms discussed above. An important observation when designing efficient circuits for these tasks is the well-known fact that a random bit with bias $1/p$, for any integer p, can be generated from only two uniform random bits suffice, in expectation. Generating a uniformly random bit is easy (and extremely cheap using garbled circuits) in the semi-honest model: each party simply generates a random bit, and then inside the circuit a random bit is reconstructed as the XOR of two bits. As XORs are for free in garbled circuits this computation is very efficient. The problem of implementing differentially private mechanisms in MPC was discussed by Dwork et al. (2006), where the authors present small circuits for sampling from an exponential distribution requiring only a $log(k)$ biased random bits, which can be constructed in parallel. Recently, Champion et al. (2019) proposed optimized constructions for several well-known differentially private mechanisms (including the geometric and Laplace mechanisms), and empirically showed their concrete efficiency.

D Additional Experiments

D.1 AUC Experiments on Synthetic Data

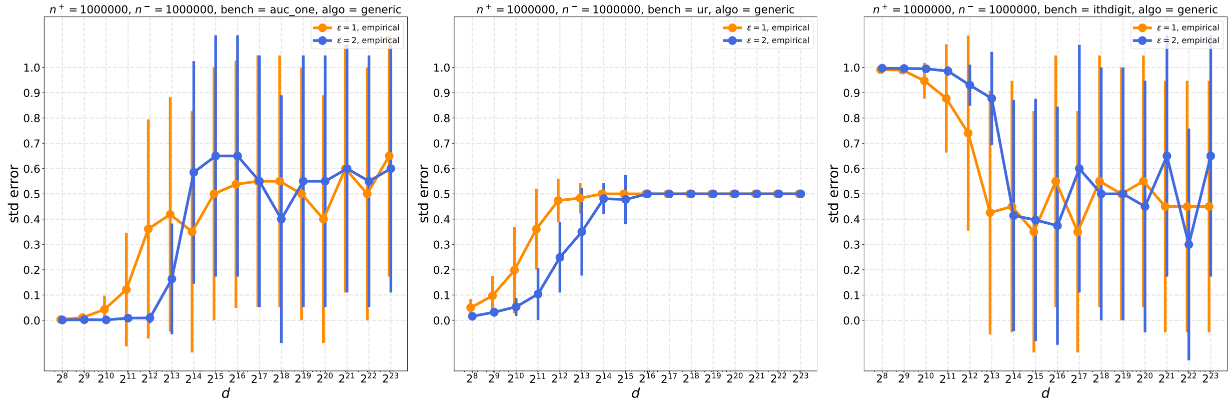
We illustrate the behavior of our AUC-specific LDP protocol of Section 4 on three synthetic datasets, and compare its performance with that of our generic LDP protocol of Section 3. For all datasets, we have 10^6 inputs in each class (positive and negative) but the score values of inputs are distributed differently:

- `auc_one` consists of two distinct inputs $(d-1, 1)$ and $(0, -1)$ each occurring 10^6 times.
- In `ur`, the score value of an input is drawn independently and uniformly from $[0..d-1]$, regardless of its class.



(a) `auc_one` (b) `ur` (c) `ithdigit`

Figure 1: Mean and std. dev. (over 20 runs) of the absolute error of our AUC-specific LDP protocol on three synthetic datasets.



(a) `auc_one` (b) `ur` (c) `ithdigit`

Figure 2: Mean and std. dev. (over 20 runs) of the absolute error of our generic LDP protocol on three synthetic datasets.

- `ithdigit` consists of two distinct inputs $(10^{-4}, 1)$ and $(0, -1)$ each occurring 10^6 times.

Figure 1 shows the error that our AUC-specific protocol incurs on the three datasets. On `auc_one`, our AUC-specific protocol incurs considerable error due to significant recursion error, E^R_m, being incurred for every level. This example illustrates that our analysis for the AUC-specific protocol is not far from being tight. On `ur`, the error is much lower. This is because (i) the algorithm does not explore any of the lower sections of the tree and so no recursion error is incurred whilst exploring it, and (ii) within intervals that are discarded the points are uniformly distributed so the estimation of the AUC within that interval as a half is effective. Both of these effects will occur approximately whenever the data is smooth so one can expect the algorithm to do better in the case of smooth data than the analytic bounds indicate. Finally, on `ithdigit`, the protocol does not learn anything when the quantization is smaller or equal to 2^{13}, which is expected as all inputs are quantized to the same bin. However, we see that the protocol achieves low error for $d \geq 2^{14}$. Importantly, in all cases

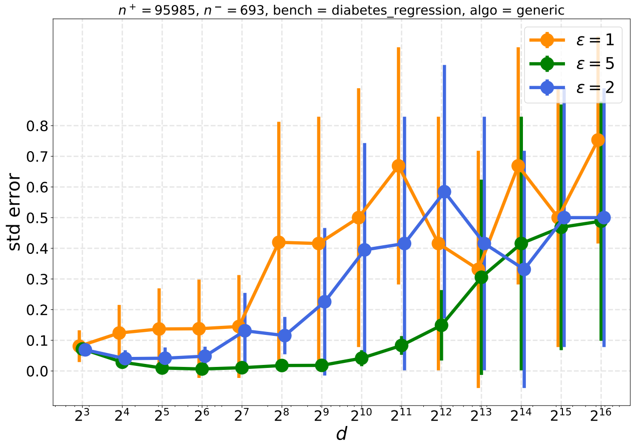


Figure 3: Mean and std. dev. (over 20 runs) of the absolute error of our generic LDP protocol on the scores of a logistic regression model trained on a Diabetes dataset.

our AUC-specific protocol scales nicely with the size of the domain. This allows to be rather agnostic about the level of quantization needed for the problem at hand, which is often not known in advance.

In contrast, we can see on Figure 2 that the generic protocol scales very poorly with the domain size due to the use of randomized response. On `auc_one` and `ur`, data can be quantized to a small domain without losing much relevant information, leading to good performance. In the case of `ithdigit` however, the generic protocol incurs very large error in all regimes: quantizing to small domain maps all inputs to the same bin, while quantizing to large domain leads to large error due to privacy.

D.2 Results of the Generic Protocol on Diabetes dataset

Figure 3 shows the results of our generic LDP protocol of Section 3 for the problem of computing the AUC on the Diabetes dataset (see Section 6 for results with the AUC-specific protocol). On this dataset, a fully trained logistic regression model yields scores of positive and negative points that are well separated. Hence, they can be quantized to a sufficiently small domain for the protocol to achieve small error.

References

Bassily, R., Nissim, K., Stemmer, U., and Thakurta, A. G. (2017). Practical locally private heavy hitters. In *NIPS*.

Blom, G. (1976). Some properties of incomplete U-statistics. *Biometrika*, 63(3):573–580.

Champion, J., Shelat, A., and Ullman, J. (2019). Securely sampling biased coins with applications to differential privacy. *IACR Cryptology ePrint Archive*, 2019:823.

Damgård, I., Pastro, V., Smart, N. P., and Zakarias, S. (2011). Multiparty computation from somewhat homomorphic encryption. *IACR Cryptology ePrint Archive*, 2011:535.

Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I., and Naor, M. (2006). Our data, ourselves: Privacy via distributed noise generation. In *EUROCRYPT*.

Evans, D., Kolesnikov, V., and Rosulek, M. (2018). A pragmatic introduction to secure multi-party computation. *Foundations and Trends in Privacy and Security*, 2(2-3):70–246.

Goldreich, O. (2004). *The Foundations of Cryptography - Volume 2, Basic Applications*. Cambridge University Press.

Goldreich, O., Micali, S., and Wigderson, A. (1987). How to play any mental game or A completeness theorem for protocols with honest majority. In *STOC*, pages 218–229. ACM.

Kulkarni, T., Cormode, G., and Srivastava, D. (2019). Answering range queries under local differential privacy. In *SIGMOD*.

Lindell, Y. and Pinkas, B. (2009). A proof of security of yao's protocol for two-party computation. *Journal of Cryptology*, 22(2):161–188.

Yao, A. C. (1986). How to generate and exchange secrets (extended abstract). In *FOCS*.