A Distributed Frequency Estimation

In this section, we consider the frequency estimation problem for federated analytics. Recall that for the frequency estimation task, each client's private data $x_i \in \{0,1\}^d$ satisfies $\|x_i\|_0 = 1$, and the goal is to estimate $\pi := \frac{1}{n}\sum_i x_i$ by minimizing the ℓ_2 (or ℓ_1, ℓ_∞) error $\mathbb{E}\left[\|\pi - \hat{\pi}(Y^n)\|_2^2\right]$ subject to communication and (ε, δ)-DP constraints. When the context is clear, we sometimes use x_i to denote, by abuse of notation, the index of the item, i.e., $x_i \in [d]$.

To fully make use of the ℓ_0 structure of the problem, a standard technique is applying a Hadamard transform to convert the ℓ_0 geometry to an ℓ_∞ one and then leveraging the recursive structure of Hadamard matrices to efficiently compress local messages.

Specifically, for a given b-bit constraint, we partition each local item x_i into 2^{b-1} chunks $x_i^{(1)}, ..., x_i^{(2^b-1)} \in \{0,1\}^B$, where $B := d/2^{b-1}$ and $x_i^{(j)} = x_i[B\cdot(j-1) : B\cdot j - 1]$. Note that since x_i is one-hot, only one chunk of $x_i^{(j)}$ is non-zero. Then, client i performs the following Hadamard transform for each chunk: $y_i^{(\ell)} = H_B \cdot x_i^{(\ell)}$, where H_B is defined recursively as follows:

$$H_{2^n} = \frac{1}{\sqrt{2}}\begin{bmatrix} H_{2^{n-1}}, & H_{2^{n-1}} \\ H_{2^{n-1}}, & -H_{2^{n-1}} \end{bmatrix}, \text{ and } H_0 = [1].$$

Each client then generates a sampling vector $Z_{ij} \overset{\text{i.i.d.}}{\sim} \mathsf{Bern}\left(\frac{1}{B}\right)$ via shared randomness that is also known by the server, and commits $(y_i^{(1)}(j), ..., y_i^{(2^{b-1})}(j))$ as its local report. Since $(y_i^{(1)}(j), ..., y_i^{(2^{b-1})}(j))$ only contains a single non-zero entry that can be $\frac{1}{\sqrt{B}}$ or $-\frac{1}{\sqrt{B}}$, the local report can be represented in b bits ($b-1$ bits for the location of the non-zero entry and 1 bit for its sign).

From the local reports, the server can compute an unbiased estimator by summing them together (with proper normalization) and performing an inverse Hadamard transform. Moreover, with an adequate injection of Gaussian noise, the frequency estimator satisfies (ε, δ)-DP.

The idea has been used in previous literature under local DP [19, 6, 3, 32], but in order to obtain the order-optimal trade-off under *central*-DP, one has to combine Hadamard transform with a random subsampling step and incorporate the privacy amplification due to random compression in the analysis. In Algorithm 3, we provide a summary of the resultant scheme which builds on the Recursive Hadamard Response (RHR) mechanism from [32], which was originally designed for communication-efficient frequency estimation under *local* DP.

In the following theorem, we control the ℓ_∞ error of Algorithm 3.

Theorem A.1. *Let $\hat{\pi}(x^n)$ be the output of Algorithm 3. Then it holds that for all $j \in [d]$,*

$$\mathbb{E}\left[|\pi(j) - \hat{\pi}(j)|\right] \leq \sqrt{\frac{\sum_i \mathbb{1}_{\{x_i \in [B\cdot(j-1):B\cdot j-1]\}}}{n^2} + \frac{\sigma^2}{B}}, \tag{7}$$

and the ℓ_2^2 and ℓ_1 errors are bounded by

$$\mathbb{E}\left[\|\pi - \hat{\pi}\|_2^2\right] \leq \frac{B}{n} + \frac{d\sigma^2}{B}, \textit{ and} \tag{8}$$

$$\mathbb{E}\left[\|\pi - \hat{\pi}\|_1\right] \leq \sqrt{\frac{dB}{n} + \frac{d^2\sigma^2}{B}}. \tag{9}$$

Theorem A.2. *For any $\varepsilon, \delta > 0$, Algorithm 3 is (ε, δ)-DP, if*

$$\sigma^2 \geq O\left(\frac{B^2\log(B/\delta)}{n^2} + \frac{B(\log(1/\delta)+\varepsilon)\log(B/\delta)}{n^2\varepsilon^2}\right).$$

By combining Theorem A.1 and Theorem A.2, we conclude that Algorithm 3 achieves (ε, δ)-DP with ℓ_2^2 error

$$\begin{aligned} & O\left(\frac{B}{n} + \frac{dB\log(B/\delta)}{n^2} + \frac{d(\log(1/\delta)+\varepsilon)\log(B/\delta)}{n^2\varepsilon^2}\right) \\ & = O\left(\frac{d}{n2^b} + \frac{d^2\log(d/\delta)}{n^2 2^b} + \frac{d(\log(1/\delta)+\varepsilon)\log(d/\delta)}{n^2\varepsilon^2}\right). \end{aligned}$$

Algorithm 3 Subsampled Recursive Hadamard Response

```
Input: user data $x_1, ..., x_n \in \{0,1\}^d$ (where $d$ is a power of two), DP parameters $(\varepsilon, \delta)$, communication budget $b$.
Output: frequency estimate $\hat{\pi}$
Set $B \coloneqq d/2^{b-1}$ and partition each one-hot vector $x_i$ into $2^{b-1}$ chunks: $x_i^{(1)}, ..., x_i^{(2^b-1)} \in \{0,1\}^B$.
for user $i \in [n]$ do
  Compute the Hadamard transform of each chunk: $y_i^{(\ell)} = H_B \cdot x_i^{(\ell)}$.
  for coordinate $j \in [B]$ do
    Draw $Z_{i,j} \overset{\text{i.i.d.}}{\sim} \mathsf{Bern}\left(\frac{1}{B}\right)$
    if $Z_{i,j} = 1$ then
      Send $(y_i^{(1)}(j), ..., y_i^{(2^{b-1})}(j))$ to the server.
    end if
  end for
end for
Server computes the average: $\forall \ell \in [2^{b-1}], j \in [B]$,
  $$\hat{y}^{(\ell)}(j) \coloneqq \frac{B}{n} \sum_{i: Z_{ij}=1} y_i^{(\ell)}(j) + N(0, \sigma^2),$$
where $\sigma^2$ is computed according to Theorem A.2.
Server performs the inverse Hadamard transform $\hat{\pi}^{(\ell)} = H_B \cdot \hat{y}^{(\ell)}$, for $\ell = 1, ..., B$.
Return: $\hat{\pi} = \left( \left(\hat{\pi}^{(1)}\right)^\intercal, ..., \left(\hat{\pi}^{(2^{b-1})}\right)^\intercal \right)$.
```

Notice that when $n = \tilde{\Omega}(d)$, the error can be simplified to

$$O\left(\frac{d}{n2^b} + \frac{d(\log(1/\delta) + \varepsilon)\log(d/\delta)}{n^2\varepsilon^2} \right),$$

which matches the order-optimal estimation error (up to a $\log d$ factor) subject to a b-bit constraint [54, 3, 2] and (ε, δ)-DP constraint [15, 7].

B Proof of Theorem 4.1

It is trivial to see that the average communication cost is $d \cdot \gamma = b$ bits. To compute the ℓ_2^2 estimation error, observe that

$$
\begin{aligned}
&\mathbb{E}\left[\|\hat{\mu}_{x^n} - \mu_{x^n}\|_2^2\right] \\
&= \sum_{j=1}^{d} \mathbb{E}\left[\left(\frac{1}{n\gamma}\sum_i x_i(j)\cdot Z_{i,j} + N(0,\sigma^2) - \frac{1}{n}\sum_i x_i(j)\right)^2\right] \\
&= \sum_{j=1}^{d} \frac{1}{n^2}\mathbb{E}\left[\left(\frac{1}{\gamma}\sum_i x_i(j)\cdot Z_{i,j} - \sum_i x_i(j)\right)^2\right] + d\sigma^2 \\
&= \sum_{j=1}^{d} \frac{1}{n^2}\mathbb{E}\left[\left(\frac{1}{\gamma}\sum_i x_i(j)\cdot Z_{i,j}\right)^2\right] - \frac{1}{n^2}\left(\sum_i x_i(j)\right)^2 + d\sigma^2 \\
&= \sum_{j=1}^{d} \frac{1}{n^2}\mathbb{E}\left[\frac{1}{\gamma^2}\sum_i x_i^2(j)\cdot Z_{i,j}^2 + \frac{1}{\gamma^2}\sum_{i\neq i'} x_i(j)x_{i'}(j)Z_{i,j}Z_{i',j}\right] - \frac{1}{n^2}\left(\sum_i x_i(j)\right)^2 + d\sigma^2 \\
&= \sum_{j=1}^{d} \frac{1}{n^2}\left(\frac{1}{\gamma}\sum_i x_i^2(j) + \sum_{i\neq i'} x_i(j)x_{i'}(j)\right) - \frac{1}{n^2}\left(\sum_i x_i(j)\right)^2 + d\sigma^2 \\
&= \sum_{j=1}^{d} \frac{1}{n^2}\left(\frac{1}{\gamma} - 1\right)\left(\sum_i x_i^2(j)\right) + d\sigma^2 \\
&\le \frac{dc^2}{n\gamma} + d\sigma^2,
\end{aligned}
$$

which yields the inequality of (2). Next, we analyze the privacy of Algorithm 1. We first the following two lemmas for subsampling and the Gaussian mechanism:

Lemma B.1 ([65, 81])**.** *If $\mathcal{M}$ is (ε,δ)-DP, then $\mathcal{M}'$ that applies $\mathcal{M} \circ$ PoissonSample satisfies (ε',δ')-DP with $\varepsilon' = \log\left(1+\gamma\left(e^{\varepsilon}-1\right)\right)$ and $\delta' = \gamma\delta$.*

Lemma B.2 ([15])**.** *For any $\varepsilon, \delta \in (0,1)$, the Gaussian output perturbation mechanism with $\sigma^2 := \frac{\Delta^2 2\log(1.25/\delta)}{\varepsilon^2}$ satisfies (ε,δ)-DP, where Δ is the ℓ_2 sensitivity of the target function.*

Now, we use the above two lemmas to analyze the per-coordinate privacy leakage of Algorithm 1. For simplicity, we analyze the sum of $x_i(j)$'s instead (and normalized it in the last step). Let $S_j(x^n) := \sum_{i=1}^{n}(x_i(j))$, then clearly the sensitivity of $S_j(x^n)$ is c, so Lemma B.2 implies $S_j(x^n) + N(0,\sigma_1^2)$ satisfies (ε_1,δ_1)-DP if we set $\sigma_1^2 = \frac{2c^2\log(1.25/\delta_1)}{\varepsilon_1^2}$ (assuming $\varepsilon_1 < 1$). Next, if applying subsampling before computing the sum, i.e.,

$$
S_j \circ \mathsf{PoissonSample}_\gamma(x^n) := \sum_{i=1}^{n} x_i(j) Z_{i,j},
$$

where $Z_{i,j} \overset{\text{i.i.d.}}{\sim} \mathsf{Bern}(1/\gamma)$ as defined in Algorithm 1, then by Lemma B.1,

$$
S_j \circ \mathsf{PoissonSample}_\gamma(x^n) + N(0,\sigma_1^2)
$$

satisfies (ε_2,δ_2)-DP with $\varepsilon_2 := \log\left(1+\gamma\left(e^{\varepsilon_1}-1\right)\right) = C_1\gamma\varepsilon_1$ (since we assume $\epsilon_1 < 1$) and $\delta_2 := \gamma\delta_1$. Equivalently, we have

$$
\begin{cases}
\varepsilon_1 = \tilde{C}_1 \frac{1}{\gamma}\varepsilon_2 \\
\delta_1 = \frac{1}{\gamma}\delta_2.
\end{cases}
\tag{10}
$$

Now, since we have established the per-coordinate privacy leakage, we apply the following composition theorem to account for the total privacy budgets.

Theorem B.3. *For any $\varepsilon > 0$, $\delta \in [0,1]$ and $\tilde{\delta} \in (0,1]$, the class of (ε, δ)-DP mechanisms satisfies $(\tilde{\varepsilon}_{\tilde{\delta}}, d\delta + \tilde{\delta})$-DP under d-fold adaptive composition, for*

$$\tilde{\varepsilon}_{\tilde{\delta}} = d\varepsilon\left(e^{\varepsilon} - 1\right) + \varepsilon\sqrt{2d\log(1/\tilde{\delta})}.$$

According Theorem B.3, Algorithm 1 satisfies (ε, δ)-DP for

$$\varepsilon = d\varepsilon_2(e^{\varepsilon_2} - 1) + \varepsilon_2\sqrt{2d\log(1/\tilde{\delta})}, \tag{11}$$

and $\delta = d\delta_2 + \tilde{\delta}$ (where $\tilde{\delta}$ is a free parameter that we can optimize).

Consequently, for a pre-specified (total) privacy budget (ε, δ), we set parameters as follows. Let $\tilde{\delta} = \frac{\delta}{2}$ and $\delta_1 = \frac{1}{\gamma}\delta_2 = \frac{1}{2d\gamma}\delta$. Let $\varepsilon_2 \le 1$ so that $e^{\varepsilon}_2 - 1 \le 2\varepsilon_2$ holds. Then (11) implies Algorithm 1 is

$$\varepsilon = 2d\varepsilon_2^2 + \varepsilon_2\sqrt{2d\log(1/\tilde{\delta})} \ge d\varepsilon_2(e^{\varepsilon_2} - 1) + \varepsilon_2\sqrt{2d\log(1/\tilde{\delta})}.$$

Solving the above quadratic (in-)equality for ε_2, it yields that

$$\varepsilon_2 = \min\left(1, \frac{-\sqrt{2d\log(2/\delta)} + \sqrt{2d\log(2/\delta) + 8\varepsilon d}}{4d}\right) = O\left(\min\left(1, \frac{\varepsilon}{\sqrt{d\left(\log(1/\delta) + \varepsilon\right)}}\right)\right).$$

Consequently, we set $\varepsilon_1 = \frac{\tilde{C}_1}{\gamma}\varepsilon_2 = O\left(\min\left(1, \frac{\varepsilon}{\gamma\sqrt{d(\log(1/\delta)+\varepsilon)}}\right)\right)$ (note that we require $\varepsilon_1 = O(1)$ so that (10) holds).

Plug in $(\varepsilon_1, \delta_1)$ into σ_1^2, we have

$$\sigma_1^2 := \frac{2c^2\log(1.25/\delta_1)}{\varepsilon_1^2} = \Omega\left(\max\left(c^2\log(d/\delta), \frac{\gamma^2c^2d(\log(1/\delta) + \varepsilon)\log(d/\delta)}{\varepsilon^2}\right)\right).$$

Finally, as we are interested in estimating the (subsampled) mean instead of the sum, we will normalize the private sum by

$$\hat{\mu}_j(x^n) = \frac{1}{n\gamma}\left(S_j \circ \mathsf{PoissonSample}_\gamma(x^n) + N(0, \sigma_1^2)\right) = \frac{1}{n\gamma}S_j \circ \mathsf{PoissonSample}_\gamma(x^n) + N(0, \sigma^2),$$

where

$$\sigma^2 = O\left(\max\left(\frac{c^2\log(d/\delta)}{n^2\gamma^2}, \frac{c^2d(\log(1/\delta) + \varepsilon)\log(d/\delta)}{n^2\varepsilon^2}\right)\right).$$

Plugging in σ^2 above and $\gamma = b/d$ yields the desired accuracy in Theorem 4.1. ☐

Since we will reuse the above result, we summarize it into the following lemma:

Lemma B.4. *Let $f_i : \mathbb{R}^{d\times m} \mapsto \mathbb{R}^D$ for $i = 1, ..., B$ be n functions with sensitivity bounded by Δ (where the number of inputs m can be a random variable). Then*

$$\left(f_1 \circ \mathsf{PoissonSample}_\gamma(x^n) + N(0, \sigma^2), ..., f_B \circ \mathsf{PoissonSample}_\gamma(x^n) + N(0, \sigma^2)\right)$$

satisfies (ε, δ)-DP, if

$$\sigma^2 \ge O\left(\max\left(\Delta^2\log(B/\delta), \frac{\gamma^2\Delta^2B(\log(1/\delta) + \varepsilon)\log(B/\delta)}{\varepsilon^2}\right)\right).$$

C Omitted details of dimension-free communication cost

C.1 Proof of Theorem 4.4

To prove Theorem 4.4, it suffices to prove the following ℓ_∞ version:

Theorem C.1. *Let* $x_1, ..., x_n \in \{-c, c\}^d$, $d' = \min\left(nb, \frac{n^2\varepsilon^2}{(\log(1/\delta)+\varepsilon)\log(d/\delta)}\right)$*, and*

$$\sigma^2 = O\left(\frac{c^2\log(1/\delta)}{n^2\gamma^2} + \frac{c^2 d'(\log(d'/\delta)+\varepsilon)\log(d'/\delta)}{n^2\varepsilon^2}\right). \tag{12}$$

Then Algorithm 2 is (ε, δ)*-DP and yields an unbiased estimator on* μ*. In addition, the (average) per-client communication cost is* $\gamma d' = b$ *bits, and the* ℓ_2^2 *estimation error is at most*

$$O\left(c^2 d^2 \log\left(\frac{d}{\delta}\right)\max\left(\frac{1}{nb}, \frac{(\log(1/\delta)+\varepsilon)}{n^2\varepsilon^2}\right)\right). \tag{13}$$

With a slight abuse of notation, we let $\mu_{\mathcal{J}} \in \mathbb{R}^d$ be such that

$$\mu_{\mathcal{J}}(j) = \begin{cases} 0, & \text{if} j \notin \mathcal{J} \\ \frac{d\mu_j}{d'}, & \text{else.} \end{cases}$$

Note that $\mu_{\mathcal{J}}$ is an unbiased estimate of μ if $\mathcal{J}$ is selected uniformly at random. Then the ℓ_2^2 error can be controlled by

$$\begin{aligned}
\mathbb{E}\left[\|\mu - \hat{\mu}\|_2^2\right] &\overset{\text{(a)}}{=} \mathbb{E}\left[\|\mu - \mu_{\mathcal{J}}\|_2^2\right] + \mathbb{E}\left[\|\mu_{\mathcal{J}} - \hat{\mu}\|_2^2\right] \\
&\overset{\text{(b)}}{\leq} \mathbb{E}\left[\|\mu - \mu_{\mathcal{J}}\|_2^2\right] + \frac{d^2}{d'^2} O\left(\max\left(\frac{d'^2c^2}{nb}, \frac{d'^3c^2\log(d/\delta)}{n^2b^2}, \frac{c^2d'^2(\log(1/\delta)+\varepsilon)\log(d/\delta)}{n^2\varepsilon^2}\right)\right) \\
&= \mathbb{E}\left[\|\mu - \mu_{\mathcal{J}}\|_2^2\right] + O\left(\max\left(\frac{d^2c^2}{nb}, \frac{d^2d'c^2\log(d/\delta)}{n^2b^2}, \frac{c^2d^2(\log(1/\delta)+\varepsilon)\log(d/\delta)}{n^2\varepsilon^2}\right)\right) \\
&\overset{\text{(c)}}{\leq} \frac{d^2c^2}{d'} + O\left(\max\left(\frac{d^2c^2}{nb}, \frac{d^2d'c^2\log(d/\delta)}{n^2b^2}, \frac{c^2d^2(\log(1/\delta)+\varepsilon)\log(d/\delta)}{n^2\varepsilon^2}\right)\right),
\end{aligned}$$

where (a) holds since $\mu_{\mathcal{J}}$ is an unbiased estimate of μ and conditioned on $\mathcal{J}$, $\hat{\mu}$ is an unbiased estimate of $\mu_{\mathcal{J}}$; (b) follows from Theorem 4.1; (c) holds due to the following fact:

$$\mathbb{E}\left[\|\mu - \mu_{\mathcal{J}}\|_2^2\right] \leq \sum_{j\in\mathcal{J}} \mu_{\mathcal{J}}(j)^2 + \sum_{j\in[d]} \mu_j^2 \leq \frac{d^2c^2}{d'} + dc^2 \leq \frac{2d^2c^2}{d'}.$$

Therefore, by setting $d' = \min\left(nb, \frac{n^2\varepsilon^2}{(\log(1/\delta)+\varepsilon)\log(d/\delta)}\right)$ we ensure the first term in (c) is always smaller than the second term, and the second term can be simplified as follows:

$$\begin{aligned}
&O\left(c^2d^2\max\left(\frac{1}{nb}, \frac{d'\log(d/\delta)}{n^2b^2}, \frac{(\log(1/\delta)+\varepsilon)\log(d/\delta)}{n^2\varepsilon^2}\right)\right) \\
&\leq O\left(c^2d^2\max\left(\frac{1}{nb}, \frac{nb\log(d/\delta)}{n^2b^2}, \frac{(\log(1/\delta)+\varepsilon)\log(d/\delta)}{n^2\varepsilon^2}\right)\right) \\
&\leq O\left(c^2d^2\log(d/\delta)\max\left(\frac{1}{nb}, \frac{(\log(1/\delta)+\varepsilon)}{n^2\varepsilon^2}\right)\right).
\end{aligned}$$

Finally, applying the same trick of Kashin's representation, we can transform the ℓ_∞ geometry to ℓ_2 (similar to Corollary 4.3), hence proving Theorem 4.4. □

D Proof of Theorem A.1

Let $\pi := \frac{1}{n}\sum_i x_i$ and $\pi^{(\ell)}$ be defined in the same way as $x_i^{(\ell)}$ for $\ell \in [B]$. Then our goal is to bound $\left|\pi^{(\ell)}(j) - \hat{\pi}^{(\ell)}(j)\right|$, for all $\ell \in [2^{b-1}]$ and $j \in [B]$.

To this end, let $y^{(\ell)} := H_B \cdot \pi^{(\ell)}$ (so it holds that $\pi^{(\ell)} = \frac{1}{B}H_B \cdot y^{(\ell)}$). Then we have

$$\begin{aligned}
\mathbb{E}\left[\left|\pi^{(\ell)}(j) - \hat{\pi}^{(\ell)}(j)\right|\right] &\overset{\text{(a)}}{\leq} \sqrt{\mathbb{E}\left[\left(\pi^{(\ell)}(j) - \hat{\pi}^{(\ell)}(j)\right)^2\right]} \\
&= \sqrt{\mathbb{E}\left[\left(\frac{1}{B}H_B \cdot \left(y^{(\ell)} - \hat{y}^{(\ell)}\right)(j)\right)^2\right]}.
\end{aligned} \tag{14}$$

Next, observe that due to the subsampling step, for all $\ell \in [2^{b-1}]$ and $j \in [B]$,

$$\hat{y}^{(\ell)}(j) = \frac{B}{n}\sum_{i=1}^{n}\langle (H_B)_j, x_i^{(\ell)}\rangle \cdot Z_{ij} + N(0, \sigma^2),$$

where recall that $Z_{ij} \overset{\text{i.i.d.}}{\sim} \mathsf{Ber}(1/B)$. Therefore, $\hat{y}^{(\ell)}(j)$ is an unbiased estimator of $y^{(\ell)}(j)$. In addition, since we choose Z_{ij} independently in Algorithm 3, $\hat{y}^{(\ell)}(j)$'s are independent for different j's, so we have

$$\begin{aligned}
\mathbb{E}\left[\left(\hat{y}^{(\ell)}(j) - y^{(\ell)}(j)\right)^2\right] &= \mathsf{Var}\left(\hat{y}^{(\ell)}(j)\right)\\
&= \sigma^2 + \frac{B^2}{n^2}\sum_{i=1}^{n}\langle (H_B)_j, x_i^{(\ell)}\rangle^2 \mathsf{Var}\left(Z_{ij}\right)\\
&\leq \sigma^2 + \frac{B}{n^2}\sum_{i=1}^{n}\langle (H_B)_j, x_i^{(\ell)}\rangle^2\\
&= \sigma^2 + \frac{B}{n^2}\underbrace{\sum_{i=1}^{n}\mathbb{1}_{\{x_i \in \ell\text{-th chunk}\}}}_{:=C_\ell}, \qquad (15)
\end{aligned}$$

and for all $j \neq j'$

$$\mathbb{E}\left[\left(\hat{y}^{(\ell)}(j) - y^{(\ell)}(j)\right)\cdot\left(\hat{y}^{(\ell)}(j') - y^{(\ell)}(j')\right)\right] = 0. \qquad (16)$$

Therefore, we continue bounding (14) as follows:

$$\begin{aligned}
\sqrt{\mathbb{E}\left[\left(\frac{1}{B}H_B \cdot \left(y^{(\ell)} - \hat{y}^{(\ell)}\right)(j)\right)^2\right]} &= \sqrt{\frac{1}{B^2}\mathbb{E}\left[\langle (H_B)_j, \left(\hat{y}^{(\ell)} - y^{(\ell)}\right)\rangle^2\right]}\\
&= \sqrt{\frac{1}{B^2}\mathbb{E}\left[\left(\sum_{k=1}^{B}(H_B)_{jk}\cdot\left(\hat{y}^{(\ell)}(k) - y^{(\ell)}(k)\right)\right)^2\right]}\\
&\overset{\text{(a)}}{=} \sqrt{\frac{1}{B^2}\mathbb{E}\left[\sum_{k=1}^{B}\left(\hat{y}^{(\ell)}(k) - y^{(\ell)}(k)\right)^2\right]}\\
&\overset{\text{(b)}}{=} \sqrt{\frac{C_\ell}{n^2} + \frac{\sigma^2}{B}}\\
&\overset{\text{(c)}}{\leq} \sqrt{\frac{1}{n} + \frac{\sigma^2}{B}},
\end{aligned}$$

where (a) holds since each entry of H_B takes value in $\{-1, 1\}$ and by (16), (b) holds due to (15), and (c) holds because $C_\ell \leq n$ for all ℓ.

Finally, to bound the ℓ_2^2 error, observe that the above analysis ensures that

$$\mathbb{E}\left[\left(\pi^{(\ell)}(j) - \hat{\pi}^{(\ell)}(j)\right)^2\right] \leq \frac{C_{\ell(j)}}{n^2} + \frac{\sigma^2}{B},$$

where $\ell(j) \in [2^{b-1}]$ is the index of the chuck containing j. Therefore, summing over $j \in [d]$, we must have

$$\mathbb{E}\left[\left\|\pi^{(\ell)} - \hat{\pi}^{(\ell)}\right\|_2^2\right] \leq \sum_{j=1}^{d}\frac{C_{\ell(j)}}{n^2} + \frac{d\sigma^2}{B} = \frac{B}{n} + \frac{d\sigma^2}{B},$$

since

$$\sum_{j} C_{\ell(j)} = \sum_{\ell=1}^{2^{b-1}}\sum_{j' \in \ell\text{-th chunk}}\sum_{i=1}^{n}\mathbb{1}_{\{i \in \ell-\text{th chunk}\}} = B\sum_{\ell=1}^{2^{b-1}}\sum_{i=1}^{n}\mathbb{1}_{\{i \in \ell-\text{th chunk}\}} = B \cdot n.$$

$\square$

E Proof of Theorem A.2

Let $f_j(x^n) := (\pi^{(1)}(j), ..., \pi^{(2^{b-1})}(j))$, for $j = 1, ..., B$. Then the ℓ_2 sensitivity of f_j is $\Delta = \frac{B}{n}$. Set the sampling rate $\gamma = \frac{1}{B}$ and the proof is complete by Lemma B.4. $\square$

F Algorithm of Shuffled SQKR

Algorithm 4 Shuffled SQKR

```
Input: users' data $x_1, \ldots, x_n$, local-DP parameter $\varepsilon_0$, communication parameters $b_0, T$
Output: mean estimator $\hat{\mu}$
for round $k \in [T]$ do
  for user $i \in [n]$ do
    Sample $s(i,1), \ldots, s(i,b_0) \overset{\text{i.i.d.}}{\sim} \mathsf{Unif}[d]$
    Sample $Z \sim \mathsf{Bern}\left(\frac{e^{\varepsilon_0}}{e^{\varepsilon_0}+2^{b_0}-1}\right)$
    if Z=1 then
      Set $Y(i,1), \ldots, Y(i,b_0) \leftarrow x_i(s(i,1)), \ldots, x_i(s(i,b_0))$
    else
      Sample $Y(i,1), \ldots, Y(i,b_0) \overset{\text{i.i.d.}}{\sim} \mathsf{Unif}\{-c, c\}$
    end if
    Send $Y(i,1), \ldots, Y(i,b_0)$ and $s(i,1), \ldots, s(i,b_0)$ to shuffler
  end for
  Shuffler samples a permutation $\pi \sim \mathsf{Unif}\{f : [n] \to [n] \text{ bijective}\}$
  for $j \in [b_0]$ do
    Shuffler sends $Y(\pi(1),j), \ldots, Y(\pi(n),j)$ and $s(\pi(1),j), \ldots, s(\pi(n),j)$ to server
  end for
  $\hat{\mu}^{(k)} \leftarrow \frac{d}{nb_0} \frac{e^{\varepsilon_0}+2^{b_0}-1}{e^{\varepsilon_0}-1} \sum_{i=1}^{n} \sum_{j=1}^{b_0} Y(\pi(i),j) e_{s(\pi(i),j)}$
end for
Return $\hat{\mu} := \frac{1}{T} \sum_{k=1}^{T} \hat{\mu}^{(k)}$
```

G Proof of Theorem 5.3

Each round $x^n \mapsto \hat{\mu}^{(k)}$ of Algorithm 4 implements the private-coin SQKR scheme of [32], achieving the communication cost and error as stated in Lemma 5.2.

Lemma G.1 (SQKR [32]). *For all* $\varepsilon_0 > 0, b_0 > 0$, *the random mapping* $x_i \mapsto Y(i,1), \ldots, Y(i,b_0), s(i,1), \ldots, s(i,b_0)$ *in Algorithm 4 is* $(\varepsilon_0, 0)$*-LDP and has output that can be communicated in* $b_0 \log(d)$ *bits, and the* $\hat{\mu}^{(k)}$ *computed from* $Y(i,1), \ldots, Y(i,b_0), s(i,1), \ldots, s(i,b_0)$ *is an unbiased estimator of* μ *satisfying*

$$\max_{x^n} \mathbb{E}\left[\left\|\mu(x^n) - \hat{\mu}^{(k)}(x^n)\right\|_2^2\right] = O\left(\frac{c^2 d}{n \min(\varepsilon_0^2, \varepsilon_0, b_0)}\right). \tag{17}$$

We now characterize the error performance of Algorithm 4 for general choices of parameters that satisfy privacy and communication constraints.

Proposition G.2. *For all* $\varepsilon > 0, b > 0, n > 0$, *with any arbitrary choice of*

$$\delta_1 \in \left(e^{-n/16e}, 1\right] \tag{18}$$

$$\delta_2 \in (0, 1], \tag{19}$$

there exists a choice of parameters ε_0, b_0, T *such that Algorithm 4 is* $(\varepsilon, T\delta_1 + \delta_2)$*-DP, uses no more than* b *bits of communication, and produces* $\hat{\mu}$ *such that*

$$\max_{x^n} \mathbb{E}\left[\|\mu - \hat{\mu}\|_2^2\right] = O\left(\max\left(\frac{c^2 d \log(d) b_0}{nb}, \frac{c^2 d \log(1/\delta_1)(\log(1/\delta_2) + \varepsilon)}{n^2 \varepsilon^2}\right)\right). \tag{20}$$

Proof. For arbitrary choice of

$$b_0 < \log\left(\frac{n}{16\log(2)}\right), \tag{21}$$

it suffices to choose

$$T = \left\lfloor \frac{b}{(\log_2(d)+1)b_0} \right\rfloor \tag{22}$$

$$\varepsilon_0 = O\left(\min\left(1, \frac{\varepsilon\sqrt{n}}{\sqrt{T\log(1/\delta_1)\left(\log(1/\delta_2)+\varepsilon\right)}}\right)\right). \tag{23}$$

The fact that Algorithm 4 uses less than b bits is immediate from the choice of T.

Applying Lemma G.1, by construction the mapping from each x_i to $Y(i,1),\ldots,Y(i,b_0)$ is $(\varepsilon_0, 0)$-LDP. By assumption

$$\delta_1 > e^{-n/16e}, \tag{24}$$

the inequality

$$1 < \log\left(\frac{n}{16\log(2/\delta_1)}\right) \tag{25}$$

is satisfied. Then the choice of

$$\varepsilon_0 \le 1 \tag{26}$$

also satisfies $\varepsilon_0 \le \log\left(\frac{n}{16\log(2/\delta)}\right)$, so by Lemma 5.1 the mapping $x^n \mapsto \hat{\mu}^{(k)}$ is $(\varepsilon_1, \delta_1)$-DP. where

$$\varepsilon_1 = O\left(\frac{\varepsilon_0\sqrt{\log(1/\delta_1)}}{\sqrt{n}}\right). \tag{27}$$

Since the output of Algorithm 4 is a function of $\left(\hat{\mu}^{(1)},\ldots,\hat{\mu}^{(T)}\right)$, by B.3 it suffices to have

$$\varepsilon_1 = O\left(\min\left(1, \frac{\varepsilon}{\sqrt{T(\log(1/\delta_2)+\varepsilon)}}\right)\right) \tag{28}$$

for Algorithm 4 to be $(\varepsilon, T\delta_1 + \delta_2)$-DP. The first inequality follows from the assumption of $\delta_1 > e^{-n/16e}$ and choice of $\varepsilon_0 = O(1)$, and the second from choice of

$$\varepsilon_0 = O\left(\frac{\varepsilon\sqrt{n}}{\sqrt{T\log(1/\delta_1)\left(\log(1/\delta_2)+\varepsilon\right)}}\right). \tag{29}$$

Since $\varepsilon_0 \le 1 \le b$, we have $\min(\varepsilon_0^2, \varepsilon_0, b) = \varepsilon_0^2$. Applying Lemma G.1,

$$\begin{aligned}
\max_{x^n} \mathbb{E}\left[\|\mu - \hat{\mu}\|_2^2\right] &= \frac{1}{T}\max_{x^n}\mathbb{E}\left[\left\|\mu - \hat{\mu}^{(1)}\right\|_2^2\right] && (30)\\
&= O\left(\frac{d}{Tn\varepsilon_0^2}\right) && (31)\\
&= O\left(\max\left(\frac{d}{Tn}, \frac{d\log(1/\delta_1)\left(\log(1/\delta_2)+\varepsilon\right)}{n^2\varepsilon^2}\right)\right). && (32)
\end{aligned}$$

Substituting the choice of T gives the desired result. □

To show Theorem 5.3, it suffices to choose

$$\begin{aligned}
b_0 &= 1 && (33)\\
\delta_1 &= \frac{\delta}{2T} && (34)\\
\delta_2 &= \frac{\delta}{2}, && (35)
\end{aligned}$$

which requires $n > 16e\log(2) \approx 30.14$ due to (21), and apply the previous proposition.

H Rényi-DP for Shuffled SQKR

In this section we restate some results for RDP which are useful for privacy accounting in experiments.

Following the proof of Corollary 4.3 in [44], applying Theorem 4.1 in the same paper yields the following.

Lemma H.1. *Let $\mathcal{M}_i$ be an independent $(\varepsilon_0, 0)$-LDP mechanism for each $i \in [n]$ with $\varepsilon_0 \leq 1$ and π be a random permutation of $[n]$. Then for any $\alpha < \frac{n}{16\varepsilon_0 \exp(\varepsilon_0)}$, the mechanism*

$$\mathcal{S} : (x_1, \ldots, x_n) \mapsto \left(\mathcal{M}_1\left(x_{\pi(1)}\right), \ldots, \mathcal{M}_n\left(x_{\pi(n)}\right)\right)$$

is $(\varepsilon_1(\alpha), \delta)$-RDP with

$$\varepsilon_1(\alpha) = \frac{\log\left(e^{2\alpha^2\sigma^2} + 4\delta_{\min} e^{\alpha\varepsilon_0}\right)}{\alpha - 1}, \tag{36}$$

where

$$\sigma = 8\sqrt{\frac{e^{\varepsilon_0}}{n}} \tag{37}$$

$$\delta_{\min} = e^{-\frac{n}{8(e^{\varepsilon_0}+1)}}. \tag{38}$$

For small ε_0, the result below is useful.

Lemma H.2 ([40])**.** *Under the same assumptions as Lemma H.1, $\mathcal{S}$ is $(\varepsilon(\alpha), \delta)$-RDP*

$$\varepsilon_1(\alpha) = 2\alpha e^{4\varepsilon_0}\left(e^{\varepsilon_0} - 1\right)^2 / n. \tag{39}$$

Applying Lemma G.1, by construction the mapping from each x_i to $y(i, 1), \ldots, y(i, b_0)$ is $(\varepsilon_0, 0)$-LDP. By Lemma H.1, respectively Lemma H.2, the mapping $x^n \mapsto \hat{\mu}^{(k)}$ is $(\varepsilon_1(\alpha), \alpha)$-RDP where $\varepsilon_1(\alpha)$ is given by (36), respectively (39). By composition, Algorithm 4 is $(T\varepsilon(\alpha), \alpha)$-RDP.

We can convert this bound back to (ε, δ)-DP using Proposition 12 from [30].

Proposition H.3. *For all $\delta > 0$, Algorithm 4 is (ε, δ)-DP where*

$$\varepsilon = \inf_{\alpha \in (1,\infty)} T\varepsilon_1(\alpha) + \frac{\log(1/\delta) + (\alpha - 1)\log(1 - 1/\alpha) - \log(\alpha)}{\alpha - 1}, \tag{40}$$

where

$$\varepsilon_1(\alpha) = \min\left(2\alpha e^{4\varepsilon_0}\left(e^{\varepsilon_0} - 1\right)^2 / n, \frac{\log\left(e^{2\alpha^2\sigma^2} + 4\delta_{\min} e^{\alpha\varepsilon_0}\right)}{\alpha - 1}\right) \tag{41}$$

and $\sigma, \delta_{\min}$ are given by (37), (38) respectively.

I Additional Experiments

Here experiments are done with the same setup as in Section 6, with local vectors $X_i(j) \overset{\text{i.i.d.}}{\sim} \frac{1}{\sqrt{d}}\left(2 \cdot \mathsf{Ber}(0.8) - 1\right)$. We set $\delta = 10^{-6}$.

Figure 2 illustrates separation between Algorithm 4 and LDP schemes. Algorithm 4 achieves error decreasing quadratically with n as guaranteed by Theorem 5.3. With only one round of shuffling, there is separation from the LDP scheme only when n is sufficiently large, and thus order-optimal error performance only occurs for large n (or equivalently small ε). This problem is avoided with multiple rounds of shuffling.

Figure 3 compares the performance of CSGM with and without coordinate pre-selection. In this regime coordinate pre-selection improves performance for all b. As predicted by Corollary 4.3 and Corollary 4.5, the MSE decreases with b but is effectively constant for sufficiently high b where the privacy term dominates. We can determine the communication cost needed for order-optimal central DP error performance to be the b at which the MSE is within some fixed constant factor away from the limiting value. We see that the communication cost increases with dimension d with the vanilla CSGM scheme, but a dimension-free communication cost is achieved with coordinate pre-selection.

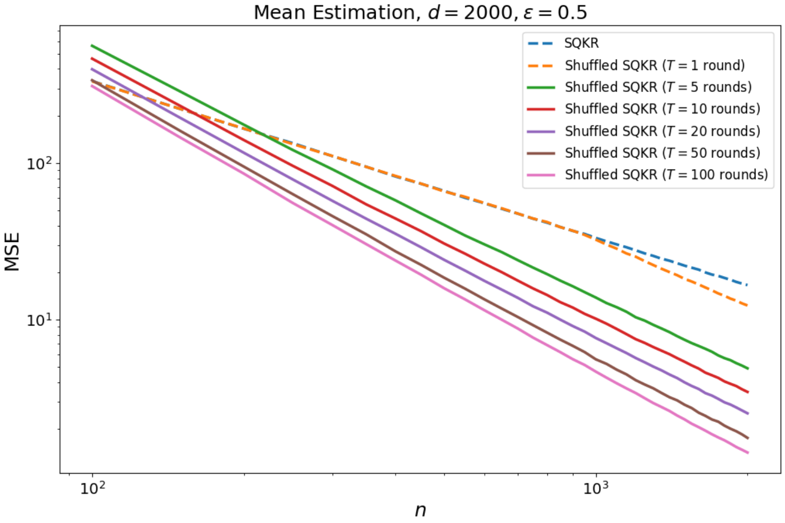

Figure 2: Comparison of MSE vs. number of clients n for LDP scheme (SQKR) and shuffled SQKR. For shuffled SQKR, we set $b_0 = 1$ and choose ε_0 using results in Section H. Communication cost is $\lceil (\log_2(2000) + 1) \rceil = 12$ bits per round.

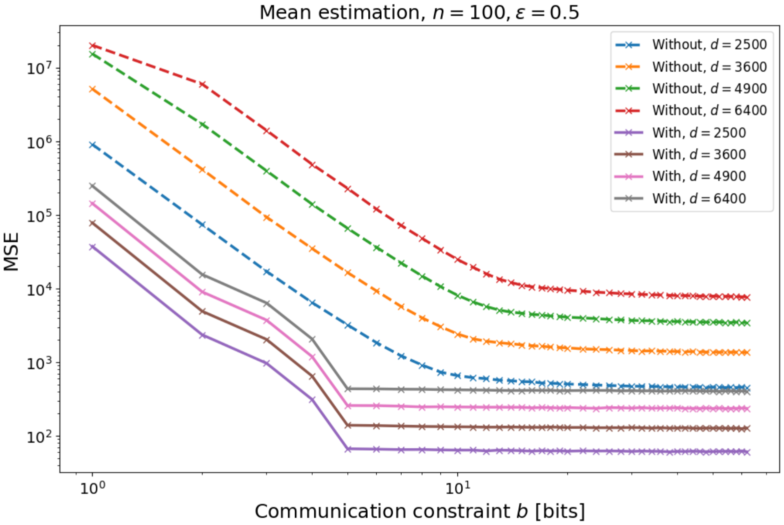

Figure 3: CSGM with and without coordinate pre-selection using $d' = 833$.