

Private Multi-Group Aggregation

Carolina Naim[✉], *Graduate Student Member, IEEE*, Rafael G. L. D’Oliveira[✉], *Member, IEEE*, and Salim El Rouayheb, *Senior Member, IEEE*

Abstract—We study the differentially private multi-group aggregation (PMGA) problem. This setting involves a single server and n users. Each user belongs to one of k distinct groups and holds a discrete value. The goal is to design schemes that allow the server to find the aggregate (sum) of the values in each group (with high accuracy) under communication and local differential privacy constraints. The privacy constraint guarantees that the user’s group remains private. This is motivated by applications where a user’s group can reveal sensitive information, such as his religious and political beliefs, health condition, or race. We propose a novel scheme, dubbed Query and Aggregate (Q&A) for PMGA. The novelty of Q&A is that it is an interactive aggregation scheme. In Q&A, each user is assigned a random query matrix, to which he sends the server an answer based on his group and value. We characterize the Q&A scheme’s performance in terms of accuracy (MSE), privacy, and communication. We compare Q&A to the Randomized Group (RG) scheme, which is non-interactive and adapts existing randomized response schemes to the PMGA setting. We observe that typically Q&A outperforms RG, in terms of privacy vs. utility, in the high privacy regime.

Index Terms—Differential privacy, data privacy, estimation.

I. INTRODUCTION

WE CONSIDER the problem of distributed aggregation in which a centralized server wishes to compute the aggregate (sum) of the data (values) held by several users. Privacy is a significant concern since participants have to share their data, which can be personal and sensitive. This has motivated works on private and secure distributed aggregation in many applications such as medical studies [1] or, more recently, federated learning [2]–[5].

In this work, we focus on the setting depicted in Figure 1, in which users belong to different groups. The server wants to find the aggregate for each group separately. As opposed to finding the aggregate over the whole population, as in

Manuscript received June 1, 2021; revised November 1, 2021; accepted December 21, 2021. Date of publication January 12, 2022; date of current version February 17, 2022. The work of Carolina Naim and Salim El Rouayheb was supported in part by NSF under Grant CCF 1817635 and Grant CNS 1801630. The work of Rafael G. L. D’Oliveira was supported by the FinTech@CSAIL Research Initiative. An earlier version of this paper was presented in part at the IEEE International Symposium on Information Theory 2021 [DOI: 10.1109/ISIT45174.2021.9518142]. (Corresponding author: Carolina Naim.)

Carolina Naim and Salim El Rouayheb are with the Department of Electrical and Computer Engineering, Rutgers, The State University of New Jersey, New Brunswick, NJ 08854 USA (e-mail: carolina.naim@rutgers.edu; salim.elrouayheb@rutgers.edu).

Rafael G. L. D’Oliveira is with the Research Laboratory of Electronics (RLE), Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: rafaeld@mit.edu).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/JSAC.2022.3142357>.

Digital Object Identifier 10.1109/JSAC.2022.3142357

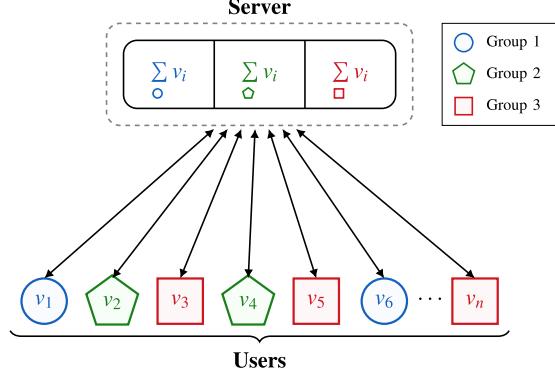


Fig. 1. An instance of the private multi-group aggregation problem with n users. Each user i , for $i \in \{1, \dots, n\}$, has a scalar value, v_i , and belongs to one of the $k = 3$ distinct groups. The server’s goal is to estimate the sum of the values in each of the groups.

typical distributed aggregation problems, *e.g.*, [2], [6]. The users’ groups can be based, for example, on their political views, immigration status, health condition, or race, to name a few. This raises additional privacy concerns since participating users may be rightfully wary of revealing their group.

Consider, for example, medical or clinical trials conducted to determine how having a certain illness, say diabetes, affects the efficacy of a new vaccine. A volunteer may want to contribute his vaccine test results, but does not want to reveal his medical condition (diabetes), *i.e.*, the group he belongs to. Another application is population polling during elections, where pollsters want to estimate how different groups of the population vote. Such groups could depend on race, gender, age, or income bracket. The poll participants want to indicate which political candidate they will vote for while keeping their group private.

Motivated by these examples, we present the problem of Private Multi-Group Aggregation (PMGA), where local differential privacy [7], [8] guarantees are given over a user’s group. We are interested in schemes that scale well with the number of groups since more groups allow the server more refined statistics about the population. Our main objective is to design schemes with low communication costs per user, as users can have limited bandwidth. In particular, we focus on schemes that offer communication costs that are constant or at most logarithmic in the number of groups. Moreover, we study the trade-offs they offer between privacy (measured using local differential privacy) and accuracy, *i.e.*, the aggregate estimator’s mean square error.

A. Related Work

The classical setup for secure and private aggregation in the literature does not distinguish among groups, and the

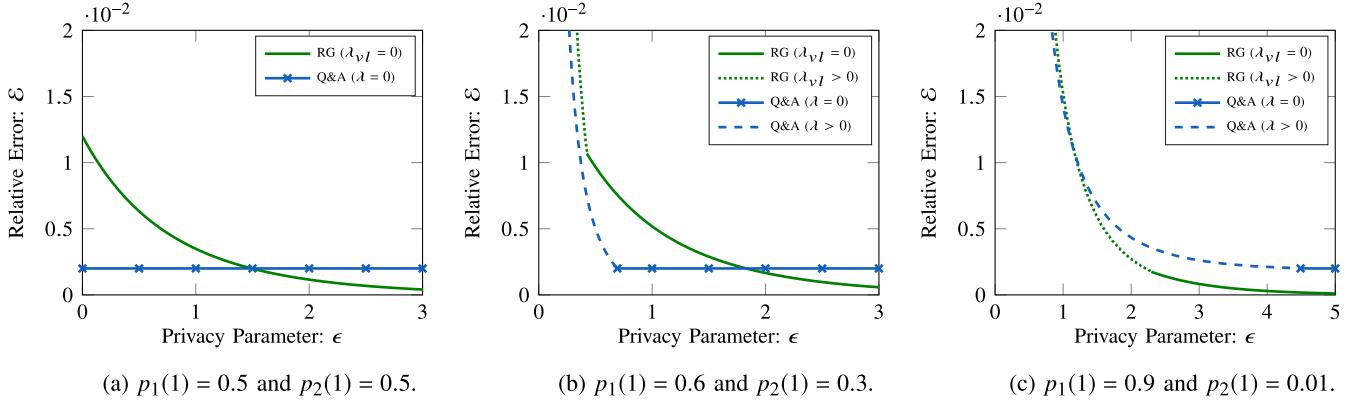


Fig. 2. Comparison of the Q&A and RG schemes for $k = 2$ groups, binary alphabet, *i.e.*, $v \in \{-1, 1\}$, and fixed total communication cost equal to 500 bits, *i.e.*, 500 bits communicated by all the users to the server. The Q&A scheme requires less communication cost per user compared to the RG scheme; therefore, for fixed total communication cost, the Q&A scheme has more users. The subfigures (a), (b) and (c) illustrate accuracy vs. privacy of the Q&A and RG schemes for different user's value distributions, $p_1(1)$ and $p_2(1)$. The dashed (or dotted) curves represent the performance of the schemes with an additional layer of privacy that hides the user's values, *i.e.*, $\lambda > 0$ for the Q&A scheme, and $\lambda_{vl} > 0$ for the RG scheme. We present a more detailed comparison in Section VI.

privacy guarantees are on the users' data (values). Differentially private schemes and bounds for private aggregation were studied in [6], [9]–[12]. In [2], secure aggregation based on information-theoretic (secret sharing) and cryptographic techniques was developed for applications to federated learning (FL) [13]. Secure aggregation algorithms for FL with improved communication and computation overhead were proposed in [14], [15], and with robustness against adversarial users in [16]. These schemes have a per-user communication cost that grows with the number of users.

Although in this paper we focus on estimating the sum, other works have focused on different estimation problems. For instance, distributed empirical mean estimation under communication constraints has been looked at in [17], [18]. Beyond estimating the mean, discrete distribution estimation under communication constraints has been studied in [19], [20], and under privacy constraints in [21]–[26]. Moreover, heavy hitters (most frequent items) estimation has been studied in [27], [28] under privacy and communication constraints. Recent work in [29] devises schemes that achieve optimal privacy and communication for mean and frequency estimation.

Another related problem is federated submodel learning [30]–[32]. In this setting, one or multiple servers hold various submodels (vectors) and each user wants to train (update) a private subset of these submodels. The notion of submodels in this setting is similar to the notion of groups in our problem; however, a user's update depends on the submodels at the server in addition to his data. The proposed solutions in [31], [32] use information-theoretic private information retrieval (PIR) to privately download and update the submodels. Thus, they require multiple servers, and the communication cost per user is linear in the number of submodels (groups). Moreover, in [30] differentially private techniques were used to allow a user to download the required submodels, and update it using secure aggregation. The resulting scheme has a communication cost per user that grows with the total number of users.

B. Contributions

We introduce the problem of private multi-group aggregation (Figure 1), where n users communicate with a central server. Each user holds a value and belongs to a private group. The goal is for the server to accurately compute the sum of values per group while keeping the user's group private. The notion of privacy we use is local differential privacy.

Our main contribution is a novel scheme for PMGA that we call the Query and Aggregate (Q&A) scheme that provides local differential privacy guarantees on the users' groups. The Q&A scheme is interactive in that the user is assigned a query matrix and sends the server an answer based on his group and value. This allows to shift the bulk of the total communication cost to the query stage (server-to-user) which can be done offline since it does not depend on a user's group and value. Thus, the online user-to-server communication cost does not depend on the number of groups and users, as typically occurs in secure aggregation problems, *e.g.*, [2]. In Theorem 1, we characterize the performance of the Q&A scheme in terms of privacy, communication cost, and accuracy.

We compare Q&A to a non-interactive scheme which we call the Randomized Group (RG) scheme. RG is an adaptation of standard randomized response [33] schemes from the literature and consists of each user reporting a noisy version of his group and value to the server. For a fixed total communication cost, we observe that in general Q&A offers better accuracy in high privacy regimes (small ϵ), as illustrated in Figure 2.

C. Paper Organization

The rest of the paper is organized as follows. In Section II, we describe the formulation of the Private Multi-Group Aggregation problem. In Section III, we present our main results, which consist of the Query and Aggregate (Q&A) scheme and its performance (Theorem 1) compared to our proposed Randomized Group (RG) scheme. We present the details of the Q&A scheme in Section IV, and those of the RG scheme

in Section V. We compare the two schemes in Section VI. Finally, we conclude and give future directions in Section VII.

D. Notation

We represent random variables by upper case letters, *e.g.*, X , realizations of these random variables by lower case letters, *e.g.*, x , and the alphabets of the random variables by calligraphic letters, *e.g.*, \mathcal{X} . We use $\log(x) = \log_2(x)$ and $\ln(x) = \log_e(x)$. Also, for any positive integer n , we denote $[n] := \{1, \dots, n\}$. Moreover, we use a colon to refer to whole rows or columns in a matrix or vector. For instance, $A(:, i) = (A(1, i), A(2, i), \dots, A(n, i))^\top$ is the i^{th} column of A . The L_2 -norm of a vector \mathbf{x} is denoted by $\|\mathbf{x}\|$.

II. PROBLEM FORMULATION

We consider the setting depicted in Figure 1 in which there are n users, indexed from 1 to n , and a single server. The users can communicate with the server but not among each other. Each user $i \in [n] := \{1, \dots, n\}$ belongs to one of k groups, indexed from 1 to k . Moreover, user $i \in [n]$ holds a value $v_i \in \mathcal{V} := \{\pm 1, \dots, \pm m\}$. We assume that the server knows each user's index but does not know his value or group. We assume that the users are not adversarial and faithfully participate in the scheme.

We denote by G_i the random variable representing the group that user i belongs to. We assume that the G_i , for all $i \in [n]$, are identical and independent random variables from the alphabet $\mathcal{G} = \{1, \dots, k\}$, where $k \geq 2$. The probability that any user $i \in [n]$ belongs to group $g \in \mathcal{G}$ is denoted by $\theta_g := \Pr(G_i = g)$. We denote by $\theta := (\theta_1, \dots, \theta_k)$ the realization of the random vector Θ .

Each user i in group g holds an independent random scalar value V_i drawn from the alphabet $\mathcal{V} := \{\pm 1, \dots, \pm m\}$ according to the distribution $p_g(v) := \Pr(V_i = v | G_i = g)$. The values of the users in the same group are independent and identically distributed. We represent the users' value distributions by a $k \times 2m$ matrix

$$p := \begin{bmatrix} p_1(-m) & \dots & p_1(m) \\ \vdots & \ddots & \vdots \\ p_k(-m) & \dots & p_k(m) \end{bmatrix}.$$

The matrix p is unknown, to both the users and the server, and is assumed to be the realization of a random matrix P . Given their group $g \in \mathcal{G}$, for all $v \in \mathcal{V}$, the users behave identically, *i.e.*, $p_g(v) = \Pr(V_i = v | G_i = g)$ for any $i \in [n]$.

User i knows the realizations of the random variables G_i and V_i representing his group and value. However, the distribution of the random variables P and Θ , and their realizations, are not necessarily known neither to the server nor to the user.

The goal is to design a scheme that allows the server to compute an estimate of the sum of values per group, *i.e.*, to estimate the aggregate vector $\mathbf{S} \in \mathbb{Z}^k$ with

$$\mathbf{S}(g) = \sum_{i \in [n]: g_i=g} v_i, \quad \text{for all } g \in \mathcal{G}.$$

We consider schemes where each user i can be assigned a query $q_i \in \mathcal{Q}$, which is also known to the server. In response

to the query, the user sends the server an answer $a_i \in \mathcal{A}$. Upon receiving the answers from all users, the server finds an estimate $\hat{\mathbf{S}}$ of \mathbf{S} . We characterize the efficiency of a scheme according to (i) communication, (ii) accuracy, and (iii) privacy.

(i) **Communication:** We characterize the communication cost by the number of bits communicated between the server and the user. We look at the communication cost from two vantage points: (i) user-centric, that measures the communication per user, *i.e.*, the number of bits communicated between a user and the server; and (ii) server-centric, that measures the total communication the server receives from all the users. We refer to the latter as the *total communication cost*.

(ii) **Accuracy:** We use the relative mean square error to measure the accuracy of a scheme π . The risk of the estimator $\hat{\mathbf{S}}_\pi$ is

$$\mathcal{E}_\pi := \frac{1}{n^2} \text{MSE}(\hat{\mathbf{S}}_\pi), \quad (1)$$

where $\text{MSE}(\hat{\mathbf{S}}_\pi) = \mathbb{E} \left[\|\hat{\mathbf{S}}_\pi - \mathbf{S}\|^2 | P = p, \Theta = \theta \right]$. For ease of notation, the conditioning on P and Θ is implicit in the rest of the paper. The relative mean square error captures the accuracy of our estimate relative to the expected true aggregate \mathbf{S} . Since $\mathbb{E} [\|\mathbf{S}\|^2]$ grows as $\mathcal{O}(n^2)$, we normalize by n^2 .

(iii) **Privacy:** We keep a user's group private. We use local differential privacy [7], [8] as our measure of privacy for a user's group. Since a user's value and group can be correlated, it is sometimes necessary (depending on the required privacy parameter) to also hide a user's value in addition to his group. To that end, a user's answer to the server is the output of a randomized mechanism $\mathcal{M} : \mathcal{G} \times \mathcal{V} \times \mathcal{Q} \rightarrow \mathcal{A}$ that outputs a user's answer a belonging to an alphabet \mathcal{A} based on his group, value, query and local randomness.

Definition 1: Let ϵ be a positive real number, and \mathcal{M} be a randomized mechanism. We say \mathcal{M} is ϵ -locally differentially private with respect to the group if for any $g, g' \in \mathcal{G}$, $q \in \mathcal{Q}$, and $a \in \mathcal{A}$,

$$\begin{aligned} \Pr(\mathcal{M}(G, V, Q) = a | G = g, Q = q, P = p, \Theta = \theta) \\ \leq e^\epsilon \Pr(\mathcal{M}(G, V, Q) = a | G = g', \\ Q = q, P = p, \Theta = \theta), \end{aligned} \quad (2)$$

where the probability is taken over the randomness of the mechanism and the random variable V .

The probabilities in the local differential privacy definition are taken given the realizations of the random variables P and Θ . Even though the server does not necessarily know these realizations, the privacy definition above assumes this knowledge. This is needed because, with enough answers collected from users, the server might infer information about the distributions of P and Θ .

We note that if a randomized mechanism is ϵ_0 locally differentially private, then it is also locally differentially private for all $\epsilon > \epsilon_0$ motivating the following definition.

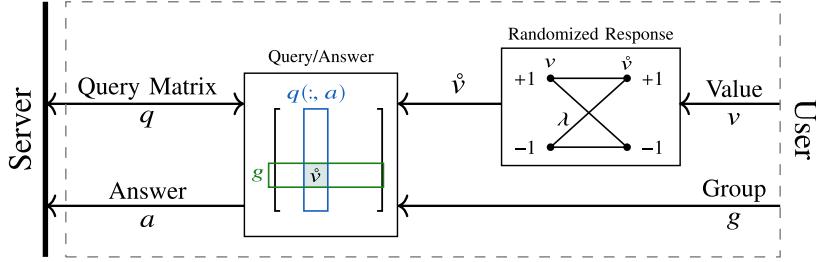


Fig. 3. A block diagram representing the Q&A scheme for a binary alphabet, $\mathcal{V} = \{-1, 1\}$. The user is assigned a query matrix q . He sends the server an answer, a , which is an index of a column of this matrix. His answer is based on his group, g , and his randomized value, \hat{v} . To randomize his value, the user applies randomized response, parameterized by λ .

Definition 2: The privacy level of a scheme (randomized mechanism) is the smallest $\epsilon > 0$ such that (2) is satisfied.

III. MAIN RESULTS

A. Query and Aggregate (Q&A) Scheme

We propose a new scheme for PMGA, which we refer to as the Query and Aggregate (Q&A) scheme. Q&A is characterized by its low communication cost per user, which is independent of the number of groups k and number of users n . It also offers an advantageous accuracy for the high privacy regime. Figure 3 summarizes this scheme, which mainly consists of two blocks:

- 1) *Query/Answer block:* The user is assigned a query matrix.¹ His answer is an index of a column of this matrix, and is determined by his value and group. This leverages the randomness in the user's value to hide his group and already provides a level of privacy over the user's group.
- 2) *Randomized Response block:* Here, the user adds noise to his value parameterized by $\lambda \geq 0$. This block is not always necessary, except for some cases, such as when the users' groups and values are highly correlated.

Theorem 1 characterizes the communication cost, privacy, and accuracy achieved by the Q&A scheme.

Theorem 1 (Q&A Scheme): Given a PMGA instance with n users, k groups, alphabet $\mathcal{V} = \{\pm 1, \dots, \pm m\}$, and the users' value distribution $p_g(v)$ for all $g \in \mathcal{G}, v \in \mathcal{V}$; the Query and Aggregate scheme (Q&A) satisfies the following properties.

- 1) The Q&A scheme has a communication cost of $\log(2m)$ bits per user.
- 2) The Q&A scheme is ϵ_{QA} -LDP with

$$e^{\epsilon_{QA}} = \max_{\substack{v, v' \in \mathcal{V}, \\ g, g' \in \mathcal{G}, g' \neq g}} \left\{ \frac{(2m(1 - \lambda) - 1)p_g(v) + \lambda}{(2m(1 - \lambda) - 1)p_{g'}(v') + \lambda} \right\}, \quad (3)$$

where the randomization parameter $\lambda \in [0, \frac{2m-1}{2m}]$.

- 3) The estimator of the Q&A scheme is unbiased and has relative mean square error

$$\mathcal{E}_{QA} = \alpha n^{-1}, \quad (4)$$

¹The assigned query matrix, q , is independent of the user's group and value, and is known to both the server and the user.

where $\alpha = \frac{2m\lambda \mathbb{E}[V_i^2]}{2m-2m\lambda-1} + \frac{(4m^2-1)(m+1)[(2m-1)(k-1)+2m\lambda]}{6(2m-2m\lambda-1)^2}$.
The relative mean square error is $\mathcal{O}\left(\frac{km^4}{n}\right)$.

We explain the Q&A scheme in more details in Section IV.

B. Randomized Group (RG) Scheme

To better understand the performance of the Q&A scheme described in Theorem 1, we compare it to the Randomized Group (RG) scheme which adds noise directly to the group. With probability λ_{gr} , the user sends the server his true group ($\log(k)$ bits) and true value or a noisy version of it ($\log(2m)$ bits). Otherwise, the user lies about his group and sends a mean zero random value that is independent of his true value.

This scheme is an adaptation of the randomized response [21], [33] method used in the differential privacy literature. In Theorem 2 in Section V we present the details and analysis of the RG scheme.

C. Comparison (Q&A vs. RG)

The Q&A scheme requires $\log(2m)$ bits per user, while the RG scheme requires $\log(2km)$ bits per user. Therefore, from a user-centric perspective, the Q&A scheme always outperforms the RG scheme in terms of communication cost. However, they achieve different error and privacy trade-offs. We also look at the communication cost from a server-centric perspective by fixing the total communication cost at the server, and comparing the relative error versus privacy. This allows for a different number of users for each of the two schemes.²

Figure 2, gives an instance of this comparison for a fixed communication cost. The key takeaway from this comparison is that there are two regimes, (i) a high privacy regime where for small values of the privacy parameter, ϵ , Q&A outperforms RG; (ii) a low privacy regime where for large enough privacy parameter, ϵ , RG outperforms Q&A. This is because, as ϵ goes to infinity, the error of the Q&A scheme converges to a constant strictly bounded away from zero as we cannot further tune the parameters of the scheme. On the other hand, the error of the RG scheme converges to zero. We defer a more detailed comparison to Section VI.

IV. THE QUERY AND AGGREGATE (Q&A) SCHEME

In this section, we describe the Q&A scheme. We begin by an example that illustrates the key ideas of this scheme

²This is motivated by the idea that, in practice, the server might be choosing a batch of users from a larger population.

by focusing on the special case of two groups and a binary alphabet. We then give the description of the general (Q&A) scheme in Section IV-B.

A. 1-Bit Example: Two Groups and a Binary Alphabet

We focus on the special case of two groups, $k = 2$, and a binary alphabet, $\mathcal{V} = \{-1, 1\}$. In this case, the Q&A scheme needs only a single bit of communication per user.

1) *Scheme Description:* The scheme is composed of the following three steps.

1. *Queries:* Each user i responds to a random query q_i which is a 2 by 2 matrix. More specifically, the query q_i is chosen uniformly at random from the set

$$\mathcal{Q} = \left\{ \begin{bmatrix} -1 & +1 \\ +1 & -1 \end{bmatrix}, \begin{bmatrix} -1 & +1 \\ -1 & +1 \end{bmatrix}, \begin{bmatrix} +1 & -1 \\ -1 & +1 \end{bmatrix}, \begin{bmatrix} +1 & -1 \\ +1 & -1 \end{bmatrix} \right\}.$$

The user's assigned query is independent of his group and value. Moreover, it is assumed that the server knows the queries assigned to each user.

2. *User's answer:* Each user sends the server a 1-bit answer, a_i , depending on the query he received. The user only looks at the *row* of the query matrix that corresponds to his group, *i.e.*, row 1 if he is in group 1 and row 2 if he is in group 2. He answers with the index of the *column* that contains his value, *i.e.*, $a_i = 1$ or $a_i = 2$.
3. *Server's estimation:* The server receives the 1-bit answer a_i from each user i . He maps the 1-bit answer into the vector $q_i(:, a_i)$, *i.e.*, the a_i^{th} column of the query matrix q_i . This is possible because he knows the user's assigned query. Then, the server forms the estimates of the aggregate for each group as follows:

$$\hat{\mathbf{S}}_{\text{QA}} = \sum_{i=1}^n q_i(:, a_i). \quad (5)$$

For example, consider a user i in group 2 who has value +1. If he receives the query $q_i = \begin{bmatrix} -1 & +1 \\ +1 & -1 \end{bmatrix}$, then his answer is $a_i = 1$, which the server maps into the vector $q_i(:, a_i) = \begin{bmatrix} -1 \\ +1 \end{bmatrix}$. Otherwise, if the user receives the query $q_i = \begin{bmatrix} -1 & +1 \\ -1 & +1 \end{bmatrix}$, then his answer is $a_i = 2$, which is mapped into $q_i(:, a_i) = \begin{bmatrix} +1 \\ +1 \end{bmatrix}$.

The key idea behind these queries is that they provide different, and equally likely, pairings of a value for a particular group with all possible values of the other group. For instance, if we look at the first column of the query matrices, notice that in the query $\begin{bmatrix} -1 & +1 \\ +1 & -1 \end{bmatrix}$, the value -1 for group 1 (first row) is paired with the value $+1$ of group 2 (second row), while in query $\begin{bmatrix} -1 & +1 \\ -1 & +1 \end{bmatrix}$ it is paired with the value -1 of group 2.

Next we give a brief analysis of this scheme, and see how it fairs on our three performance metrics: accuracy (MSE), privacy, and communication cost.

2) *Accuracy:* We show that the relative mean square error goes to zero as the number of users increases, allowing the server a better estimate of the true aggregate \mathbf{S} .

Without loss of generality, let us consider, $\mathbf{S}(1)$, the aggregate corresponding to group 1. Then, its estimate is

$$\begin{aligned} \hat{\mathbf{S}}_{\text{QA}}(1) &= \sum_{i \in [n]: g_i=1} q_i(1, a_i) + \sum_{i \in [n]: g_i=2} q_i(1, a_i) \\ &= \underbrace{\mathbf{S}(1)}_{\substack{\text{True Aggregate} \\ \text{for Group 1}}} + \underbrace{\sum_{i \in [n]: g_i=2} q_i(1, a_i)}_{\text{Noise}}. \end{aligned} \quad (6)$$

Therefore, the estimate $\hat{\mathbf{S}}_{\text{QA}}(1)$ can be interpreted as the true aggregate with an added noise term. The noise corresponds to the contribution of the users who do not belong to group 1. Since the queries were assigned uniformly at random, the distribution of the answers corresponding to the noise is uniform and independent of the true aggregate $\mathbf{S}(1)$. It follows from our choice of query matrices that the contribution to the estimate, of each user i in group 2, $q_i(1, a_i)$, is a realization of the random variable,

$$Q_i(1, A_i) = \begin{cases} -1 & \text{with probability } \frac{1}{2}, \\ +1 & \text{with probability } \frac{1}{2}. \end{cases} \quad (7)$$

The noise term can be interpreted as the position of a point on the integer number line, \mathbb{Z} , after n steps of a simple random walk starting at zero. Alternatively, the noise is the sum of i.i.d. random variables with bounded variance that converges to a zero mean additive Gaussian noise. Either way, this implies that the expectation of the norm of the noise grows as $\mathcal{O}(\sqrt{n})$. And indicates that the relative mean square error, \mathcal{E}_{QA} , goes to zero as $\mathcal{O}(n^{-1})$.

3) *Privacy:* We show that the Q&A scheme is ϵ_{QA} locally differentially private. From Definition 1,

$$e^{\epsilon_{\text{QA}}} = \max_{\substack{g, g' \in \{1, 2\}, \\ a \in \{1, 2\}, q \in \mathcal{Q}}} \frac{\Pr(A_i = a | G_i = g, Q_i = q)}{\Pr(A_i = a | G_i = g', Q_i = q)}. \quad (8)$$

The first thing we notice is that the ratio in (8) is equal to 1 for $g = g'$, and the maximum is always greater than or equal to 1 when $g \neq g'$. Therefore, we can limit the maximization in (8) to $g \neq g'$. Moreover, a user's value (+1 or -1) is a deterministic function of the answer, the query, and the group. Therefore, we can simplify (8) to

$$\begin{aligned} e^{\epsilon_{\text{QA}}} &= \max_{\substack{g, g' \in \{1, 2\}, g \neq g' \\ v, v' \in \{-1, 1\}, q \in \mathcal{Q}}} \frac{\Pr(V_i = v | G_i = g, Q_i = q)}{\Pr(V_i = v' | G_i = g', Q_i = q)} \\ &= \max_{\substack{g, g' \in \{1, 2\}, g \neq g' \\ v, v' \in \{-1, 1\}}} \frac{p_g(v)}{p_{g'}(v')}, \end{aligned} \quad (9)$$

which follows from the independence of the random variables representing the user's value, V_i , and his assigned query, Q_i , and the definition $p_g(v) = \Pr(V_i = v | G_i = g)$.³ Thus, we obtain an expression of the privacy which only depends on the users' value distributions.

³To simplify our discussion, in the rest of this paper, we assume that the probabilities $p_g(v)$ are in $(0, 1)$, for all $g \in [k]$ and $v \in \mathcal{V}$.

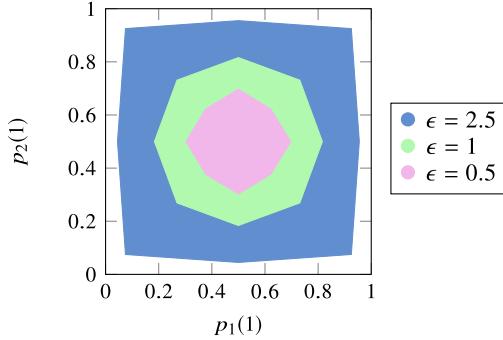


Fig. 4. Privacy for the Q&A Scheme. The values of $p_1(1)$ and $p_2(1)$ in the shaded regions of the figures above guarantee the fixed privacy parameters $\epsilon = 2.5$, $\epsilon = 1$, and $\epsilon = 0.5$, respectively. The higher the privacy requirement, *i.e.*, smaller ϵ , the smaller the region. We note that the indicated region is the full interior of the polygon.

We refer to the privacy parameter ϵ_{QA} , described in (9), as the *intrinsic privacy* of the scheme. Notice that it depends on the users' value distributions, $p_1(\cdot)$ and $p_2(\cdot)$; however, neither the server nor the users know these $p_1(\cdot)$ and $p_2(\cdot)$. Therefore, they cannot directly calculate the privacy parameter ϵ_{QA} . Nonetheless, the privacy parameter, ϵ_{QA} , can be bounded if the users have prior information about their value distributions. For example, suppose the users know that $p_1(1)$ and $p_2(1)$ are bounded such that $c_{\min} \leq p_1(1), p_2(1) \leq c_{\max}$, where the constants $c_{\min}, c_{\max} \in (0, 1)$. In this case, we can upper bound the intrinsic privacy level $\epsilon_{\text{QA}} \leq \ln\left(\frac{c_{\max}}{c_{\min}}\right)$.

Next we give more insights about the relationship between the users' value distributions, $p_g(\cdot)$, and the privacy parameter. Let us fix a privacy level ϵ , and define the region that describes the users' value distributions, $p_1(1), p_2(1) \in (0, 1)$, which guarantee that the scheme is ϵ -LDP. In Figure 4, we plot this region for different values of ϵ . Looking at Figure 4 and (9), we observe the following.

- The less privacy we require, *i.e.*, the larger the privacy level ϵ , the larger the highlighted region, *i.e.*, more values of $p_1(1)$ and $p_2(1)$ can guarantee this level of privacy.
- The closer $p_1(1)$ and $p_2(1)$ are to 0.5, the higher the privacy guarantee. And perfect privacy, *i.e.*, $\epsilon = 0$, is only guaranteed when $p_1(1) = p_2(1) = 0.5$. Intuitively, this occurs because when $p_1(1) = p_2(1) = 0.5$, a user's answer to the query is independent of his group.

A takeaway from the above observations is that not all privacy levels can be guaranteed for fixed user value distributions $p_1(\cdot)$ and $p_2(\cdot)$. In other words, the intrinsic privacy of the scheme may not always be enough. The reason is that, in its basic form, the Q&A scheme described above, does not guarantee privacy over the user's value. Thus, when the user's value and group are sufficiently correlated, the user's value might leak more information about his group than permitted by the ϵ -LDP requirement. In such cases, the general Q&A scheme adds a second layer of privacy to the user's value to further hide his group. In addition, this provides flexible privacy guarantees which do not depend only on the user's value distributions. This second layer of privacy is obtained by adding a randomized response block, parameterized by

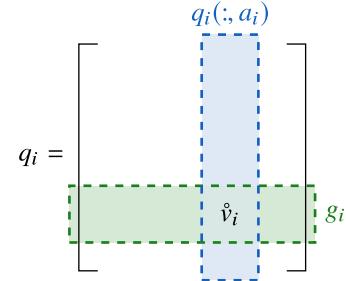


Fig. 5. User i sends the answer a_i based on his assigned query matrix q_i , his group g_i , and his randomized value \hat{v}_i . His answer is the index of the column that contains his randomized value. The server maps this answer to the a_i^{th} column of q_i .

the probability of lying λ , which hides a user's value (see Figure 3). We give a full description of the general Q&A scheme in Section IV-B.

4) *Communication*: Since the user's answer is either 1 or 2, *i.e.*, $a_i \in \{1, 2\}$, the scheme's communication cost is one bit per user. Moreover, we show in Theorem 1 that the general scheme's communication cost is always 1 bit per user when the alphabet, \mathcal{V} , is binary, irrespective of the number of groups. This is the fundamental limit on the zero-error communication cost if there were no groups and no privacy requirements.

Note that the query assignment must be known to both the server and the user. This can be accomplished without incurring communication cost. For instance, it can be implemented as the output of a public hash function that takes as input the user's index $i \in [n]$, or simply considered part of the scheme agreement that does not depend on a user's group and value.

B. The General Q&A Scheme

In this section, we describe the general Q&A scheme, for any number of groups $k \geq 2$, and alphabet parameter $m \in \mathbb{N}$. This scheme is obtained by generalizing the query matrices of the previous example and is presented in Figure 5. The Q&A scheme includes an additional randomized response block for improved privacy as described in Figure 3.

- 1) **Queries:** Each user is assigned a random query matrix of dimension $k \times 2m$ and elements in $\mathcal{V} = \{\pm 1, \dots, \pm m\}$. The query matrices assigned to each user are chosen independently and uniformly at random from the set \mathcal{Q} defined as

$$\mathcal{Q} := \{q \in \mathcal{V}^{k \times 2m} \mid q(g, :) \in \text{Sym}(\mathcal{V}) \text{ for all } g \in [k]\}, \quad (10)$$

where $q(g, :) = (q(g, 1), \dots, q(g, k))$ and $\text{Sym}(\mathcal{V})$ is the set of all row vectors which are an ordered permutation of the finite set \mathcal{V} .⁴ Each row of a matrix $q \in \mathcal{Q}$ is a permutation of all the possible $2m$ values. Notice that the values cannot be repeated within a row but rows can be repeated. We denote by q_i the query assigned to

⁴An ordered permutation of a set \mathcal{V} is a vector where each element is a distinct element of \mathcal{V} , *e.g.*, $\text{Sym}(\{\pm 1, \pm 2\})$ has $4! = 24$ elements including $(-2, -1, 1, 2)$ and $(1, 2, -2, -1)$.

user i . More details about the choice of set (10) are in Appendix A-B.

We assume that the server also knows the query assigned to the user. As previously mentioned, since the query does not depend on the user's group or value, it can be assigned offline as part of the scheme agreement, or implemented as the output of a public hash function.

2) **User's Answer:** Given his assigned query, user i hides his value using the randomized response block parameterized by λ , as in Figure 3. That is, given his true value v_i , the user first chooses a randomized value \dot{v}_i according to the distribution

$$\Pr(\dot{V}_i = \dot{v}_i | V_i = v_i) = \begin{cases} 1 - \lambda & \text{for } \dot{v}_i = v_i \\ \frac{\lambda}{2m-1} & \text{for } \dot{v}_i \in \mathcal{V} - \{v_i\}, \end{cases} \quad (11)$$

where $\lambda \in [0, 1 - \frac{1}{2m}]$. When $\lambda = 0$, *i.e.*, no privacy over the user's value, then $V_i = \dot{V}_i$.

Then, user i looks at the g_i^{th} row (g_i is the user's group) of the query matrix q_i , and sends to the server the answer a_i , which is the index of the column that has his randomized value \dot{v}_i . More precisely, a_i is such that $q_i(g_i, a_i) = \dot{v}_i$ as explained in Figure 5.

Remark 1 (On Sending Columns and Not Rows): If the user's answer corresponded to a row of the query matrix, instead of a column, then the server would obtain full information about the user's group and none about his value, *i.e.*, the scheme would fail in providing both privacy and utility.

3) **Server's Estimation:** Upon receiving user i 's answer, the server maps it into the a_i^{th} column of query q_i , *i.e.*,

$$q_i(:, a_i) = (q_i(1, a_i), q_i(2, a_i), \dots, q_i(k, a_i))^{\top}.$$

The server sums the mapped answers from all the users, and multiplies by an unbiasing term (see Appendix A-A for more details), to find the estimate of the true aggregate, \mathbf{S} , *i.e.*,

$$\hat{\mathbf{S}}_{\text{QA}} = \frac{2m-1}{2m-2m\lambda-1} \sum_{i \in [n]} q_i(:, a_i). \quad (12)$$

Below we give examples of possible queries and answers.

Example 1: Consider the setting where there are $k = 3$ groups and the alphabet of values is $\mathcal{V} = \{\pm 1, \pm 2\}$. Let $\lambda = 0$, *i.e.*, $V_i = \dot{V}_i$. Suppose that user 1 has value $v_1 = -1$ and belongs to group $g_1 = 2$. For instance, if he is assigned the query

$$q_1 = \begin{bmatrix} -2 & -1 & +1 & +2 \\ -2 & +1 & -1 & +2 \\ +2 & -1 & -2 & +1 \end{bmatrix},$$

then his answer is $a_1 = 3$, because his value, $v_1 = -1$, is the third element of the second row (corresponding to his group $g_1 = 2$) of q_1 . Upon receiving this answer, the server decodes it into the third column of q_1 , *i.e.*, $q_1(:, a_1) = (+1, -1, -2)^{\top}$.

If the user is assigned the query

$$q_1 = \begin{bmatrix} -2 & -1 & +1 & +2 \\ +1 & -2 & +2 & -1 \\ +1 & -2 & +2 & -1 \end{bmatrix},$$

his answer will be $a_1 = 4$, which the server decodes into $q_1(:, a_1) = (+2, -1, -1)^{\top}$. In both cases $q_1(g_1, a_1) = v_1$.

We note a few characteristics of this design of queries and answers. Since every row of any query matrix $q \in \mathcal{Q}$ contains all possible values, the user's value is always one of the elements of the row vector corresponding to his group. Moreover, from the server's perspective, looking at the mapped answer $q_i(:, a_i)$, *i.e.*, a column vector of the user's assigned query q_i , the user's value (or randomized value) is in row g_i of q_i . As for all the other elements of the vector, they are uniformly distributed over \mathcal{V} . This follows from the design of the query alphabet \mathcal{Q} and mirrors (7) from the previous section. It is also the key idea for the accuracy proof of Theorem 1.

An interesting property of the Q&A scheme is that, depending on the required privacy, one can choose $\lambda = 0$, *i.e.*, no randomized response block in Figure 3. The Q&A scheme still guarantees local differential privacy with

$$\epsilon_{\text{QA}} = \ln \left(\max_{\substack{v, v' \in \mathcal{V} \\ g, g' \in \mathcal{G}, g' \neq g}} \left\{ \frac{p_g(v)}{p_{g'}(v')} \right\} \right), \quad (13)$$

which follows from (3). As in the previous section, we refer to this as the *intrinsic privacy* of the scheme, which corresponds to the special case of $\lambda = 0$. If the intrinsic privacy is not enough because of a high correlation between the user's group and value, external noise can be added to the values through the randomized response block with λ chosen appropriately depending on the required privacy ϵ .

Remark 2 (The Choice of λ): Given a required privacy parameter ϵ , the parameter λ that can guarantee this given ϵ is determined using (3). However, this requires the knowledge of the value distributions, $p_g(\cdot)$ for all $g \in \mathcal{G}$. Nevertheless, one can still use (3) to find a bound on λ that is independent of the users' value distributions as follows,

$$\lambda \geq \frac{2m-1}{2m+e^{\epsilon}-1}. \quad (14)$$

This bound can be tightened if some side information is known about the users' value distributions. For instance, suppose that $c_{\min} < p_g(v) < c_{\max}$ for all $g \in \mathcal{G}$ and $v \in \mathcal{V}$, for some constants $c_{\max}, c_{\min} \in [0, 1]$, $c_{\max} > c_{\min}$. In this case, the following tighter bound can be shown

$$\lambda \geq \frac{(2m-1)c_{\max} - c_{\min}e^{\epsilon}}{2m(c_{\max} - c_{\min}e^{\epsilon}) + e^{\epsilon} - 1}. \quad (15)$$

Evidently, smaller values of λ are better for accuracy because the mean square error is increasing in λ .

Remark 3 (Error Calculation): Computing the mean square error relies on generalizing the approach in the example in Section IV-A. We have two types of errors in the estimate of the aggregate per group, *i.e.*, $\hat{\mathbf{S}}_{\text{QA}}(g)$. The first is the error introduced by the users that are not in group g . This can be approximated by a zero mean additive noise as shown in (6). The second is the error introduced by the randomized response block acting on the users' value. This error biases the sum $\sum q_i(:, a_i)$. Therefore, to unbias the estimator we multiply by $(2m-1)/(2m-2m\lambda-1)$ as seen in (12). The details of the error calculation can be found in the proof in Appendix A-A.

Theorem 1, first stated in Section III, provides the performance of the Q&A scheme with respect to communication cost, privacy, and accuracy. For its proof see Appendix A-A. Moreover, we elaborate on our choice of (10) and (11) for the Q&A scheme in Appendix A-B.

V. THE RANDOMIZED GROUP (RG) SCHEME

To better gauge the performance of the Q&A scheme we compare it to the Randomized Group (RG) scheme which directly hides a user's group by adding noise to it through a randomized response step. In RG, each user i sends the server an answer $a_i = (\dot{g}_i, \dot{v}_i)$ of his privatized group and value. That is, \dot{g}_i is chosen randomly according to the distribution

$$\Pr(\dot{G}_i = \dot{g}_i | G_i = g_i) = \begin{cases} 1 - \lambda_{gr} & \text{for } \dot{g}_i = g_i \\ \frac{\lambda_{gr}}{k-1} & \text{for } \dot{g}_i \in [k] - \{g_i\}, \end{cases} \quad (16)$$

where g_i is user i 's group and the parameter $\lambda_{gr} \in (0, 1)$. As for the value \dot{v}_i , there are two cases:

- 1) $\dot{g}_i \neq g_i$: In this case, the user chooses \dot{v}_i , uniformly at random, *i.e.*,

$$\Pr(\dot{V}_i = \dot{v}_i | \dot{g}_i \neq g_i) = \frac{1}{2m} \quad (17)$$

for all $\dot{v}_i \in \mathcal{V}$. This choice ensures that when users lie about their groups, the aggregate of their contribution has a zero mean.

- 2) $\dot{g}_i = g_i$: In this case, the user lies about his true value with probability $\lambda_{vl} \in [0, 1 - \frac{1}{2m}]$. That is, he randomly chooses a value, \dot{v}_i , according to the distribution

$$\Pr(\dot{V}_i = \dot{v}_i | V_i = v_i, \dot{g}_i = g_i) = \begin{cases} 1 - \lambda_{vl} & \text{for } \dot{v}_i = v_i, \\ \frac{\lambda_{vl}}{2m-1} & \text{for } \dot{v}_i \in \mathcal{V} - \{v_i\}. \end{cases} \quad (18)$$

The server aggregates the received answers and re-scales the aggregate to unbias the estimator, such that, for all $g \in [k]$ the estimate of the true aggregate of group g , $\mathbf{S}(g)$, is

$$\hat{\mathbf{S}}_{\text{RG}}(g) := \frac{2m-1}{(1-\lambda_{gr})(2m(1-\lambda_{vl})-1)} \sum_{i: a_i(1)=g} a_i(2).$$

Note that there are no queries assigned to users in this scheme.

Theorem 2 characterizes the scheme's performance with respect to communication cost, privacy, and accuracy.

Theorem 2: *Given a PMGA instance with n users, k groups, alphabet $\mathcal{V} = \{\pm 1, \dots, \pm m\}$, and the users' value distribution $p_g(v)$ for all $g \in \mathcal{G}, v \in \mathcal{V}$; the Randomized Group scheme (RG) is parameterized by the randomization parameters $\lambda_{gr} \in (0, 1)$, and $\lambda_{vl} \in [0, 1 - \frac{1}{2m}]$ and satisfies the following properties.*

- 1) *The RG scheme has a communication cost of $\log(2km)$ bits per user.*
- 2) *The RG scheme is ϵ_{RG} -LDP with*

$$e^{\epsilon_{\text{RG}}} = \max \left\{ \beta_1 (p_{\max} \beta_2 + \lambda_{vl}), \frac{1}{\beta_1 (p_{\min} \beta_2 + \lambda_{vl})} \right\}, \quad (19)$$

where $p_{\max} = \max_{g \in \mathcal{G}, v \in \mathcal{V}} p_g(v)$, $p_{\min} = \min_{g \in \mathcal{G}, v \in \mathcal{V}} p_g(v)$, $\beta_1 = \frac{2m(k-1)(1-\lambda_{gr})}{(2m-1)\lambda_{gr}}$, and $\beta_2 = (2m(1-\lambda_{vl})-1)$.

- 3) *The estimator of the RG scheme is unbiased and has relative mean square error*

$$\mathcal{E}_{\text{RG}} = \beta_3 n^{-1}, \quad (20)$$

where $\beta_3 = \mathbb{E}[V_1^2] \left(\frac{2m-1}{(1-\lambda_{gr})(2m(1-\lambda_{vl})-1)} - 1 \right) + \frac{(4m^2-1)(m+1)(2m\lambda_{vl}(1-\lambda_{gr})+\lambda_{gr}(2m-1))}{6(1-\lambda_{gr})^2(2m(1-\lambda_{vl})-1)^2}$. The relative mean square error is $\mathcal{O}\left(\frac{m^4 k^2}{n e^\epsilon}\right)$.

Proof: See Appendix B-A. \square

The following corollary characterizes the relationship between the randomization parameters, λ_{gr} , λ_{vl} , and the privacy parameter, ϵ_{RG} .

Corollary 1: *Let $\epsilon > 0$ be the required privacy parameter. Then, the optimal parameters λ_{gr}^* and λ_{vl}^* which guarantee the required privacy and minimize the relative error of the RG scheme are given below.*

- *If $e^{2\epsilon} < \frac{p_{\max}}{p_{\min}}$ and $p_{\max} \neq \frac{1}{2m}$, then*

$$\lambda_{vl}^* = \frac{(2m-1)(p_{\max} - e^{2\epsilon} p_{\min})}{(2p_{\max} - 1) + (1 - 2p_{\min})e^{2\epsilon}}$$

and

$$\lambda_{gr}^* = \frac{2m(k-1)(p_{\max} - p_{\min})e^\epsilon}{2m(k-1)(p_{\max} - p_{\min})e^\epsilon + (1 - 2p_{\min})e^{2\epsilon} + 2p_{\max} - 1}.$$

- *If $e^{2\epsilon} \geq \frac{p_{\max}}{p_{\min}}$ or $p_{\max} = \frac{1}{2m}$, then $\lambda_{vl}^* = 0$ and*

$$\lambda_{gr}^* = \frac{2m(k-1)p_{\max}}{2m(k-1)p_{\max} + e^\epsilon}.$$

Proof: See Appendix C. \square

The above Corollary describes the choice of parameters λ_{vl} and λ_{gr} that minimize the error for a given required privacy $\epsilon > 0$. It also shows that for a high privacy requirement, *i.e.*, privacy parameter $\epsilon < \frac{1}{2} \ln\left(\frac{p_{\max}}{p_{\min}}\right)$, the parameter λ_{vl} cannot be zero. Intuitively, since the user's value and group are correlated, applying a privacy preserving mechanism only over the group is not always enough. This is similar to what occurs with the Q&A scheme. In the RG scheme, the second layer of privacy, that hides the user's value, is characterized by the parameter λ_{vl} .

VI. COMPARISON OF THE RG AND Q&A SCHEMES

From the user-centric perspective, the Q&A scheme has a communication cost of $\log(2m)$ bits per user, *i.e.*, it does not depend on the number of groups k . However, the communication cost of the RG scheme is $\log(2km)$ bits per user. Thus, the Q&A scheme can achieve a communication cost per user that the RG scheme cannot.

From the server-centric perspective, to compare the accuracy vs. privacy trade-offs of the two schemes, we fix the total communication cost, *i.e.*, the number of bits communicated by all the users to the server. We choose the parameter λ of the Q&A scheme that guarantees the required (given) privacy parameter ϵ (see Remark 2) and minimizes the error.

We find this parameter λ by solving the following optimization problem numerically

$$\begin{aligned} & \text{minimize}_{\lambda} \mathcal{E}_{\text{QA}} \\ & \text{subject to } e^{\epsilon} \leq \max_{\substack{v, v' \in \mathcal{V}, \\ g, g' \in \mathcal{G}, g' \neq g}} \left\{ \frac{(2m(1 - \lambda) - 1)p_g(v) + \lambda}{(2m(1 - \lambda) - 1)p_{g'}(v') + \lambda} \right\}, \\ & \quad 0 \leq \lambda < 1 - \frac{1}{2m}, \end{aligned}$$

where \mathcal{E}_{QA} is from (4). Similarly, for the RG scheme, we choose the parameters, λ_{gr} and λ_{vl} , as in Corollary 1.

Figure 2 illustrates this comparison for a fixed total communication cost. Typically for a high enough privacy constraint, *i.e.*, small ϵ , Q&A outperforms RG, while for a low enough privacy constraint, RG outperforms Q&A. Thus, we have two privacy regimes, a high privacy regime where Q&A should be used, and a low privacy regime where RG should be used. These observations are made rigorous in Theorem 3.

We begin by expressing the relative mean square error as a function of the privacy parameter ϵ for a fixed total communication cost of $b \geq 2$ bits. Since the Q&A scheme's communication cost per user is $\log(2m)$ bits, its number of users is given by $n_{\text{QA}} := b / \log(2m)$. Analogously, the number of users for the RG scheme is given by $n_{\text{RG}} := b / \log(2km)$.⁵ Therefore, we normalize each scheme's mean square error by its respective number of users (squared), obtaining

$$\mathcal{E}_{\text{QA}}(\epsilon, b) = \text{MSE}(\hat{\mathbf{S}}_{\text{QA}})n_{\text{QA}}^{-2} = \text{MSE}(\hat{\mathbf{S}}_{\text{QA}}) \left(\frac{\log(2m)}{b} \right)^2, \quad (21)$$

and

$$\mathcal{E}_{\text{RG}}(\epsilon, b) = \text{MSE}(\hat{\mathbf{S}}_{\text{RG}})n_{\text{RG}}^{-2} = \text{MSE}(\hat{\mathbf{S}}_{\text{RG}}) \left(\frac{\log(2km)}{b} \right)^2. \quad (22)$$

With this notation we present the following theorem.

Theorem 3: Let $\mathcal{V} = \{\pm 1, \dots, \pm m\}$ be the alphabet of values, $\mathcal{G} = [k]$ the set of possible groups, and fix the total communication cost to $b \in \{x \in \mathbb{N} | x / \log(2m), x / \log(2km) \in \mathbb{N}\}$. Unless $k = 2$, $m = 1$ and $p_1(v) = p_2(v') \neq 0.5$ for all $v, v' \in \{-1, 1\}$; then, there exists,

- (i) an $\epsilon_h > 0$, such that for all $\epsilon < \epsilon_h$, the relative error $\mathcal{E}_{\text{QA}}(\epsilon, b) < \mathcal{E}_{\text{RG}}(\epsilon, b)$, and
- (ii) an $\epsilon_\ell > 0$, such that for all $\epsilon > \epsilon_\ell$, the relative error $\mathcal{E}_{\text{QA}}(\epsilon, b) > \mathcal{E}_{\text{RG}}(\epsilon, b)$.

Proof: See Appendix C. \square

For the special case of $k = 2$ groups, and a binary alphabet of values, *i.e.*, $\mathcal{V} = \{-1, 1\}$, and $p_1(v) = p_2(v') \neq 0.5$ for some $v, v' \in \mathcal{V}$, there exists an $\epsilon_0 > 0$ such that for all $\epsilon < \epsilon_0$, the difference in relative errors $\mathcal{E}_{\text{QA}}(\epsilon_0, n_{\text{QA}}) - \mathcal{E}_{\text{RG}}(\epsilon_0, n_{\text{RG}}) = \frac{1}{b}$, where b is the total communication cost.

Remark 4 (User-Centric Accuracy vs. Privacy Trade-Off): Since both the RG and Q&A schemes have a fixed communication cost per user, restricting it can deem a scheme infeasible. To elaborate, consider the case of 2 groups and 2 values. If we fix the communication cost per user to 1 bit, the RG scheme cannot satisfy this constraint because it requires 2 bits of communication per user. Whereas the Q&A scheme directly satisfies the constraint. On the other hand,

⁵We assume that the parameter b is chosen such that $n_{\text{QA}}, n_{\text{RG}} \in \mathbb{N}$.

if we fix the communication cost per user to 2 bits, then the Q&A scheme would not utilize the full communication allotment.

VII. CONCLUSION

In this paper, we formulated the problem of private multi-group aggregation where the goal was to privately aggregate the users' values per group. Moreover, we used local differential privacy as our measure of privacy for a user's group. We characterized two schemes: Q&A and RG. The Q&A scheme generally outperformed the RG scheme, in terms of privacy vs. accuracy, in the high privacy regime.

Future work for this problem includes finding theoretic bounds characterizing the best performance achievable for a given privacy and total communication cost. Another direction would involve mapping a larger alphabet of values \mathcal{V} to a smaller alphabet \mathcal{V}' to reduce communication costs.

APPENDIX A THE Q&A SCHEME

A. Proof of Theorem 1

We separate the proof into three parts starting with communication, then privacy, and finally with the accuracy.

- 1) **Communication:** The user sends the server the index of a column of the query matrix. Since the query matrix has dimension $k \times 2m$, the user sends $\log(2m)$ bits to the server.
- 2) **Privacy:** From Definition 2, the privacy of user i is

$$\begin{aligned} & e^{\epsilon_{\text{QA}}} \\ & = \max \left\{ \max_{\substack{g, g' \in \mathcal{G}, g \neq g', \\ a \in [2m], q \in \mathcal{Q}}} \frac{\Pr(A_i = a | G_i = g, Q_i = q)}{\Pr(A_i = a | G_i = g', Q_i = q)}, 1 \right\} \\ & = \max_{\substack{g, g' \in \mathcal{G}, g \neq g', \\ a \in [2m], q \in \mathcal{Q}}} \frac{\Pr(A_i = a | G_i = g, Q_i = q)}{\Pr(A_i = a | G_i = g', Q_i = q)}, \end{aligned} \quad (23)$$

where \mathcal{Q} is as defined in (10). Notice that if $g = g'$, the ratio of probabilities is equal to 1, and if $g \neq g'$, the maximum of the ratio of probabilities is greater than or equal to 1. Consider

$$\begin{aligned} & \Pr(A_i = a | G_i = g, Q_i = q) \\ & \stackrel{(a)}{=} \sum_{v \in \mathcal{V}} \Pr(V_i = v | G_i = g) \sum_{\tilde{v} \in \mathcal{V}} \Pr(\tilde{V}_i = \tilde{v} | V_i = v) \\ & \Pr(A_i = a | G_i = g, Q_i = q, \tilde{V}_i = \tilde{v}) \\ & \stackrel{(b)}{=} \sum_{v \in \mathcal{V}} p_g(v) \Pr(\tilde{V}_i = v^* | V_i = v) \\ & \stackrel{(c)}{=} (1 - \lambda)p_g(v^*) + \frac{\lambda}{2m - 1}(1 - p_g(v^*)), \end{aligned} \quad (24)$$

where (a) follows from the law of total probability and the random variable relationships. As for (b), it follows from definition $p_g(v) := \Pr(V_i = v | G_i = g)$, and noticing that given a user's randomized value \tilde{V}_i , his group G_i , and assigned query Q_i , the user's answer A_i is deterministic. So, the probability

$\Pr(A_i = a | G_i = g, Q_i = q, \dot{V}_i = \dot{v}) = 1$ only for one realization of \dot{V}_i which we denote by $v^* = q(g, a)$, otherwise $\Pr(A_i = a | G_i = g, Q_i = q, \dot{V}_i = \dot{v}) = 0$. Finally, (c) follows from (11). Substituting (24) in (23), we obtain

$$e^{\epsilon_{\text{QA}}} = \max_{\substack{g, g' \in \mathcal{G}, g' \neq g, \\ v, v' \in \mathcal{V}}} \left\{ \frac{(2m(1-\lambda)-1)p_g(v) + \lambda}{(2m(1-\lambda)-1)p_{g'}(v') + \lambda} \right\},$$

where we replaced $v^*, v'^* \in \mathcal{V}$ by $v, v' \in \mathcal{V}$.

3) **Accuracy:** We start by finding probabilities relating to the user's assigned queries. User i is assigned query $Q_i = q$, which is chosen uniformly at random from the set \mathcal{Q} defined in (10). Therefore, for fixed row j and column a , the probability $\Pr(Q_i(j, a) = v) = \frac{1}{2m}$ for all $v \in \mathcal{V}$. Note that if user i 's answer is A_i , and given his assigned query, the server maps the user's answer into the vector $Q_i(:, A_i)$. Given user i 's group $G_i = g$ and group $V_i = v$, we find the distribution of $Q_i(j, A_i)$ for all $j \in [k]$. That is, for all $j \neq g$, $j, g \in \mathcal{G}$, and $v, v' \in \mathcal{V}$, we have

$$\Pr(Q_i(j, A_i) = v' | G_i = g, V_i = v) = \frac{1}{2m},$$

Otherwise, for all $j = g$, $j, g \in \mathcal{G}$, and $v, v' \in \mathcal{V}$,

$$\Pr(Q_i(g, A_i) = v' | G_i = g, V_i = v) = \begin{cases} 1 - \lambda & \text{for } v' = v \\ \frac{\lambda}{2m-1} & \text{for } v' \in \mathcal{V} - \{v\}. \end{cases} \quad (25)$$

For all $i \in [n]$, we introduce the auxiliary random variables X_i and Y_i for ease of notation. For all $i \in [n]$, user i 's group G_i and his value V_i are random variables as described in Section II. We define an auxiliary random variable X_i that functions as an indicator for both the user's group and value. More precisely, X_i is a random k dimensional vector (where k is the number of groups), such that $X_i(j) = 0$ if $j \neq G_i$ and $X_i(j) = V_i$ if $j = G_i$. Then one readily obtains, for all $j \in [k]$,

$$\Pr(X_i(j) = v) = \begin{cases} \theta_j p_j(v) & \text{for } v \in \mathcal{V}, \\ 1 - \theta_j & \text{for } v = 0, \end{cases} \quad (26)$$

and,

$$\mathbb{E}[X_i(j)] = \sum_{v \in \mathcal{V}} v p_j(v) \theta_j = \mathbb{E}[V_i | G_i = j] \theta_j. \quad (27)$$

Since the X_i 's are i.i.d. for all $i \in [n]$, we have

$$\begin{aligned} \mathbb{E} \left[\left(\sum_{i \in [n]} X_i(j) \right)^2 \right] &= n \mathbb{E}[V_1^2 | G_1 = j] \theta_j + (n^2 - n) \mathbb{E}[V_1 | G_1 = j]^2 \theta_j^2. \end{aligned} \quad (28)$$

For every user $i \in [n]$, we define an auxiliary random variable $Y_i = Q_i(:, A_i)$, which is a k dimensional random vector. Given user i 's group $G_i = g_i$ and his value $V_i = v_i$, the g_i^{th} coordinate of the vector Y_i contains user i 's randomized value. All the other coordinates of

the vector Y_i are randomly chosen from the alphabet \mathcal{V} . More precisely,

$$\Pr(Y_i(j) = v | G_i = g_i, V_i = v_i) = \begin{cases} 1 - \lambda & \text{for } v = v_i \text{ and } j = g_i, \\ \frac{\lambda}{2m-1} & \text{for } v \in \mathcal{V} - \{v_i\} \text{ and } j = g_i, \\ \frac{1}{2m} & \text{for } v \in \mathcal{V} \text{ and } j \neq g_i, \end{cases} \quad (29)$$

where $\lambda \in [0, 1 - \frac{1}{2m}]$. Then, following from (29), we obtain, $\Pr(Y_i(j) = v) = v = \theta_j \left((1 - \lambda)p_j(v) + \frac{(1 - p_j(v))\lambda}{2m-1} \right) + \frac{(1 - \theta_j)}{2m}$. Then,

$$\mathbb{E}[Y_i(j)] = \frac{2m - 2m\lambda - 1}{2m - 1} \mathbb{E}[V_1 | G_1 = j] \theta_j, \quad (30)$$

Since Y_1, \dots, Y_n are i.i.d., we have

$$\begin{aligned} \mathbb{E} \left[\left(\sum_{i \in [n]} Y_i(j) \right)^2 \right] &= \frac{n(1 - \theta_j)}{2m} + \frac{n\lambda\theta_j}{2m-1} \sum_{v \in \mathcal{V}} v^2 \\ &\quad + (n^2 - n) \left(\frac{2m - 2m\lambda - 1}{2m - 1} \theta_j \mathbb{E}[V_1 | G_1 = j] \right)^2 \\ &\quad + n \mathbb{E}[V_1^2 | G_1 = j] \theta_j \left(\frac{2m - 2m\lambda - 1}{2m - 1} \right). \end{aligned} \quad (31)$$

Note that $\sum_{v \in \mathcal{V}} v^2 = \frac{1}{3}m(m+1)(2m+1)$.

One readily obtains $\mathbb{E}[\hat{\mathbf{S}}_{\text{QA}} - \mathbf{S}] = 0$ by substituting (27) and (30), and observing that X_1, \dots, X_n are i.i.d. and Y_1, \dots, Y_n are i.i.d.. Then, the estimator $\hat{\mathbf{S}}_{\text{QA}}$ is unbiased.

Next we calculate the expectation $\mathbb{E}[\sum_{i=1}^n Y_i(j) \sum_{l=1}^n X_l(j)]$ which will be helpful later in the proof. Notice that given user i 's group, $G_i = g_i$, and value, $V_i = v_i$, the product $X_i(g_i)Y_i(g_i)$ is equal to v_i^2 with probability λ , and equal to $v_i v$ with probability $\frac{1-\lambda}{2m-1}$ for all $v \in \mathcal{V} - \{v_i\}$. And since $X_i(j) = 0$ for all $j \neq g_i$, then $X_i(j)Y_i(j) = 0$ for all $j \neq g_i$. Following these observations,

$$\mathbb{E}[X_i(j)Y_i(j)] = \frac{2m - 2m\lambda - 1}{2m - 1} \theta_j \mathbb{E}[V_i^2 | G_i = g], \quad (32)$$

which follows from $\sum_{v' \in \mathcal{V} - \{v\}} v' = -v$. Then,

$$\begin{aligned} \mathbb{E} \left[\sum_{i=1}^n Y_i(j) \sum_{l=1}^n X_l(j) \right] &\stackrel{(a)}{=} \sum_{i, l \in [n], i \neq l} \mathbb{E}[Y_i(j)] \mathbb{E}[X_l(j)] + \sum_{i=1}^n \mathbb{E}[Y_i(j)X_i(j)] \\ &\stackrel{(b)}{=} \frac{2m - 2m\lambda - 1}{2m - 1} \left[(n^2 - n)\theta_j^2 \mathbb{E}[V_1 | G_1 = j]^2 \right. \\ &\quad \left. + n\theta_j \mathbb{E}[V_1^2 | G_1 = j] \right]. \end{aligned} \quad (33)$$

We have that (a) follows from that fact that if $i \neq l$, then $Y_i(j)$ is independent of $X_l(j)$. And (b) follows from

substituting (27), (30), and (32). Then,

$$\begin{aligned} \mathcal{E}_{QA} &= \frac{2m\lambda}{n(2m-2m\lambda-1)} \mathbb{E}[V_1^2] \\ &+ \frac{(4m^2-1)(m+1)}{6n(2m-2m\lambda-1)} \left(\frac{(2m-1)k}{2m-2m\lambda-1} - 1 \right), \end{aligned} \quad (34)$$

follows from substituting (28), (31), and (33) in (1).

Now we have the exact expression of \mathcal{E}_{QA} in terms of λ . To characterize the accuracy vs. privacy trade-off, one might be interested in the expression of \mathcal{E}_{QA} in terms of the privacy parameter ϵ . We can get a loose upper bound on \mathcal{E}_{QA} , as a function of ϵ , by substituting $\lambda = (2m-1)/(2m-1+e^\epsilon)$, from Remark 2, in (34), and we get $\mathcal{E}_{QA} = \mathcal{O}\left(\frac{km^4}{n}\right)$.

B. On the Choice of Parameters for the Q&A Scheme

We consider the class of schemes in which the user is assigned a random query matrix and in turn sends the server an answer, corresponding to a column of this matrix, based on his group and randomized value.

1) *On the Choice of Queries:* In the Q&A scheme we considered queries chosen uniformly at random from the set (10). Below we discuss variations on this set (10), and some of their disadvantages.

- Suppose the rows do not have all possible values from the set \mathcal{V} . Then, if the user's value does not exist in the corresponding row (group), he is not able to answer the query. For example, consider an assigned query matrix $\begin{bmatrix} -1 & -1 \\ +1 & -1 \end{bmatrix}$. This is asking a user of group 1 (row 1): "Which of these is your value: -1 or -1 ?" If the user has value $+1$, he cannot answer the question.
- If the rows have extra values that are not in \mathcal{V} , i.e., the set of possible user values, this can lead to breaking the privacy requirement because the server knows that no user will have that value. To clarify, consider the example of 2 groups and 2 values, i.e., $\mathcal{V} = \{-1, +1\}$. Let's assume the user is in group 1, has value $+1$, and is assigned the query matrix $\begin{bmatrix} +1 & -1 & +3 \\ +3 & -1 & +1 \end{bmatrix}$. If he sends the server the answer corresponding to column 1, the server knows with certainty that the user is in group 1, since he cannot assume the value $+3$. On the other hand, if both the additional numbers are in the same column, e.g., $\begin{bmatrix} +1 & -1 & +3 \\ +1 & -1 & +3 \end{bmatrix}$, then this column can effectively be removed because no user will answer it.
- One could also restrict to a subset of (10). Depending on the choice of subset this could perform equivalently or result in a biased estimator. For example, consider the case of 2 groups and 2 values. The query set defined in (10) for this case is

$$\mathcal{Q} = \left\{ \begin{bmatrix} -1 & +1 \\ +1 & -1 \end{bmatrix}, \begin{bmatrix} -1 & +1 \\ -1 & +1 \end{bmatrix}, \begin{bmatrix} +1 & -1 \\ -1 & +1 \end{bmatrix}, \begin{bmatrix} +1 & -1 \\ +1 & -1 \end{bmatrix} \right\}.$$

- We note that if a query is a result of permuting the columns of another query, then they convey the same information. For example, the following two queries, $\begin{bmatrix} -1 & +1 \\ +1 & -1 \end{bmatrix}$ and $\begin{bmatrix} +1 & -1 \\ -1 & +1 \end{bmatrix}$, are equivalent. This

is because the server's interpretation of the user's answer will be the same irrespective of which of these two queries is assigned. Therefore, restricting to a subset $\mathcal{Q}' \subseteq \mathcal{Q}$, such that,

$$\mathcal{Q}' = \left\{ \begin{bmatrix} -1 & +1 \\ +1 & -1 \end{bmatrix}, \begin{bmatrix} -1 & +1 \\ -1 & +1 \end{bmatrix} \right\},$$

results in a scheme equivalent to Q&A.

- On the other hand, consider restricting to a subset $\mathcal{Q}'' \subseteq \mathcal{Q}$, such that,

$$\mathcal{Q}'' = \left\{ \begin{bmatrix} -1 & +1 \\ +1 & -1 \end{bmatrix}, \begin{bmatrix} -1 & +1 \\ -1 & +1 \end{bmatrix}, \begin{bmatrix} +1 & -1 \\ -1 & +1 \end{bmatrix} \right\},$$

where the queries are chosen uniformly at random from \mathcal{Q}'' . Then, the noise term in (6) will not have zero mean, and thus the estimate will be biased.

Therefore, a reduction to a subset of \mathcal{Q} needs to be done carefully, and can complicate the definition in (10).

2) *Randomizing the User's Value:* One could consider a more general scheme in which the probability assigned to each possible random value ($v \in \mathcal{V}$ and $v \neq v_i$) is arbitrarily different. However, this can lead to a biased estimator, that cannot be unbiased without the knowledge of the parameters $p_g(v)$ and θ_g (which the server and user do not know).

Let us elaborate by considering the following modification to (11). Given his true value v , the user first chooses a randomized value \hat{v} according to the distribution

$$\Pr(\hat{V}_i = \hat{v} | V_i = v) = \begin{cases} \ell & \text{for } \hat{v} = v \\ \ell_{v,\hat{v}} & \text{for } \hat{v} \in \mathcal{V} - \{v\}, \end{cases} \quad (35)$$

such that $\sum_{\hat{v} \in \mathcal{V}, \hat{v} \neq v} \ell_{v,\hat{v}} = 1 - \ell$ for all $v \in \mathcal{V}$. Note that the ℓ and $\ell_{v,\hat{v}}$ have to be constants, and not a function of the problem parameters $p_g(v)$ or θ_g . Otherwise, because the user does not know these problem parameters, he would not be able to perform the randomization.

Following the same procedure described in the Accuracy discussion of Appendix A-A, we define the k -dimensional random vector \tilde{Y}_i whose coordinates are randomly chosen from the alphabet \mathcal{V} , such that,

$$\begin{aligned} \Pr(\tilde{Y}_i(j) = v | G_i = g_i, V_i = v_i) \\ = \begin{cases} \ell & \text{for } v = v_i \text{ and } j = g_i, \\ \ell_{v,\hat{v}} & \text{for } v \in \mathcal{V} - \{v_i\} \text{ and } j = g_i, \\ \frac{1}{2m} & \text{for } v \in \mathcal{V} \text{ and } j \neq g_i. \end{cases} \end{aligned}$$

Then, the expected value of $\tilde{Y}_i(j)$, for all $i \in [n]$ and $j \in [k]$,

$$\mathbb{E} \left[\tilde{Y}_i(j) \right] = \ell \mathbb{E}[V_i | G_i = j] \theta_j + \theta_j \sum_{v, \hat{v} \in \mathcal{V}, v \neq \hat{v}} \hat{v} p_j(v) \ell_{v,\hat{v}}. \quad (36)$$

Therefore, following from (27) and (36), we have

$$\begin{aligned} \mathbb{E} \left[\sum_{i \in [n]} \tilde{Y}_i - \sum_{i \in [n]} X_i \right] \\ = \sum_{i \in [n]} \mathbb{E}[\tilde{Y}_i - X_i] \\ = n(\ell - 1) \mathbb{E} \left[\sum_{i \in [n]} X_i \right] + n\theta_g \sum_{v, \hat{v} \in \mathcal{V}, v \neq \hat{v}} \hat{v} p_g(v) \ell_{v,\hat{v}}. \end{aligned}$$

Notice that the server does not know the parameters $p_g(v)$ and θ_g for $g \in [k]$ and $v \in \mathcal{V}$. Therefore, the server, in general, cannot unbias the estimator. However, the Q&A scheme considers a special case of (35), where we have $\ell_{v,\hat{v}} = \frac{\lambda}{2m-1}$ for all $v, \hat{v} \in \mathcal{V}$. Therefore, we unbias the estimator by multiplying the aggregate by the constant $\frac{2m-1}{2m-2m\lambda-1}$, as we did in (12).

Moreover, there exists other special cases where the server can unbias the estimator without the knowledge of $p_g(v)$ and θ_g , for example if $\sum_{\hat{v} \in \mathcal{V}, \hat{v} \neq v} \hat{v} \ell_{v,\hat{v}} = 0$ for all $v \in \mathcal{V}$. We chose the first method for the Q&A scheme, as it simplifies its presentation, and is an adaptation of randomized response from the literature. The analysis of other methods, including their privacy analysis, remains open.

On an additional note, in general, the randomization of the user's value could be done such that the incorrect values are chosen based on the mean square error of the scheme. However, this error is a function of the parameters θ_g and $p_g(v)$ for $g \in [k]$ and $v \in \mathcal{V}$ (see (4)). Therefore, implementing the scheme would require the user (and the server) to know these parameters, which is inconsistent with our model. Thus, the Q&A scheme does not depend on this error.

APPENDIX B THE RANDOMIZED GROUP (RG) SCHEME

In Section B-A of this appendix we prove Theorem 2, and in Section B-B we prove Corollary 1.

A. Proof of Theorem 2

We separate the proof into three parts starting with communication, then privacy, and finally the accuracy.

- 1) **Communication:** Each user i sends the server an answer a_i , which is a 2 dimensional vector. The first coordinate has information about the user's group, *i.e.*, $\hat{g} \in \mathcal{G}$, and the second coordinate has information about the user's value, *i.e.*, $\hat{v} \in \mathcal{V}$. Therefore, to represent the user's answer, a_i , we need $\log(|\mathcal{V}|) + \log(|\mathcal{G}|) = 1 + \log(m) + \log(k)$ bits.
- 2) **Privacy:** To prove (19), we consider user i and look at the distribution $\Pr(A_i = a|G_i = g) = \Pr(\hat{G}_i = g', \hat{V}_i = v|G_i = g)$, for all $v \in \mathcal{V}$ and $g, g' \in \mathcal{G}$. We separately consider the two cases of $g' = g$ and $g' \neq g$ mirroring the two cases described in Section V. For all $v \in \mathcal{V}$ and $g, g' \in \mathcal{G}$ such that $g' = g$,

$$\begin{aligned} \Pr(\hat{G}_i = g, \hat{V}_i = v|G_i = g) &= \left((1 - \lambda_{vl})p_g(v) + \frac{(1 - p_g(v))\lambda_{vl}}{2m-1} \right) \\ &\quad \times (1 - \lambda_{gr}), \end{aligned} \quad (37)$$

which follows from (16), (18), and

$$\begin{aligned} \Pr(\hat{V}_i = v|\hat{G}_i = g, G_i = g) &= (1 - \lambda_{vl})p_g(v) + \frac{(1 - p_g(v))\lambda_{vl}}{2m-1} \end{aligned}$$

However, for all $v \in \mathcal{V}$ and $g, g' \in \mathcal{G}$ such that $g' \neq g$,

$$\Pr(\hat{G}_i = g', \hat{V}_i = v|G_i = g) = \frac{\lambda_{gr}}{2m(k-1)}, \quad (38)$$

which follows from (16) and (17).

From Definition 1, we drop Q from the conditioning because there are no queries assigned to the users in this scheme, also the conditioning on P and Θ is implicit. Therefore, by substituting (37) and (38) in (2), we get

$$\begin{aligned} e^{\text{erg}} &= \max \left\{ \rho \max_{g \in \mathcal{G}, v \in \mathcal{V}} p_g(v)(2m(1 - \lambda_{vl}) - 1) + \lambda_{vl}, \right. \\ &\quad \left. \left(\rho \min_{g \in \mathcal{G}, v \in \mathcal{V}} p_g(v)(2m(1 - \lambda_{vl}) - 1) + \lambda_{vl} \right)^{-1} \right\}, \end{aligned}$$

$$\text{where } \rho = \frac{2m(k-1)(1 - \lambda_{gr})}{\lambda_{gr}}.$$

- 3) **Accuracy:** For all $i \in [n]$, user i sends the server the answer $A_i = (\hat{G}_i, \hat{V}_i)$, where the user's randomized group, \hat{G}_i , is described in (16), and his randomized value, \hat{V}_i , is described in equations (18) and (17).

We define an auxiliary random variable Z_i that functions as an indicator for both user i 's randomized group and randomized value. More precisely, Z_i is a random k dimensional vector (where k is the number of groups), such that $Z_i(j) = V_i$ if $j = \hat{G}_i$ and $Z_i(j) = 0$ otherwise, *i.e.*, $j \neq \hat{G}_i$. For all $j \in [k]$ and $v \in \mathcal{V}$, one readily obtains

$$\begin{aligned} \Pr(Z_i(j) = v) &= \frac{(1 - \theta_g)\lambda_{gr}}{2m(k-1)} \\ &\quad + \theta_g(1 - \lambda_{gr}) \left((1 - \lambda_{vl})p_j(v) + \frac{(1 - p_j(v))\lambda_{vl}}{2m-1} \right). \end{aligned} \quad (39)$$

Since Z_1, Z_2, \dots, Z_n are i.i.d., then following from (39) for all $j \in [k]$ and $i \in [n]$ the expectation

$$\begin{aligned} \mathbb{E} \left[\sum_{i \in [n]} Z_i(j) \right] &= \frac{n\theta_g(1 - \lambda_{gr})(2m(1 - \lambda_{vl}) - 1)}{2m-1} \mathbb{E}[V_1|G_1 = j]. \end{aligned} \quad (40)$$

Then, $\mathbb{E}[\hat{\mathbf{S}}_{\text{RG}} - \mathbf{S}] = 0$, which follows from (27) and (40). Thus, the estimator of the RG scheme is unbiased. Moreover, (41), as shown at the bottom of the next page. Consider the random variables X_i , for all $i \in [n]$, described in (26). For all $i \in [n]$ and $j \in [k]$, notice that the product $X_i(j)Z_i(j)$ can take on one of these values:

$$X_i(j)Z_i(j)$$

$$= \begin{cases} v^2 & \forall v \in \mathcal{V}, \text{ if } X_i(j) = Z_i(j) = v, \\ vv' & \forall v, v' \in \mathcal{V}, v' \neq v, \text{ if } X_i(j) = v, \\ & \quad \text{and } Z_i(j) = v', \\ 0 & \forall v, v' \in \mathcal{V}, \text{ if } X_i(j) = 0 \text{ or } Z_i(j) = 0, \end{cases}$$

Then, we can use this to find the expectation

$$\begin{aligned} \mathbb{E}[X_i(j)Z_i(j)] &= \frac{2m(1-\lambda_{vr})-1}{2m-1} \mathbb{E}[V_i^2|G_i=j] \theta_j(1-\lambda_{gr}). \end{aligned} \quad (42)$$

Moreover, since Z_1, \dots, Z_n are i.i.d., X_1, \dots, X_n are i.i.d., and Z_i is independent of X_ℓ if $i \neq \ell$,

$$\begin{aligned} &\mathbb{E}\left[\sum_{i \in [n]} Z_i(j) \sum_{\ell \in [n]} X_\ell(j)\right] \\ &= \sum_{i \in [n]} \mathbb{E}[Z_i(j)X_i(j)] + (n^2 - n) \\ &\quad \times \sum_{i, \ell \in [n], i \neq \ell} \mathbb{E}[Z_i(j)][X_\ell(j)] \\ &= \frac{(n^2 - n)(1 - \lambda_{gr})(2m(1 - \lambda_{vl}) - 1)}{2m - 1} \\ &\quad \mathbb{E}[V_1|G_1=j]^2 \theta_g^2 \\ &\quad + n(1 - \lambda_{gr})(1 - \lambda_{vl}) \mathbb{E}[V_1^2|G_1=j] \theta_g, \end{aligned} \quad (43)$$

which follows from the substitution of (27), (40), and (42).

Notice that $\hat{\mathbf{S}}_{\text{RG}}(j) = \frac{2m-1}{(1-\lambda_{gr})(2m(1-\lambda_{vl})-1)} \sum_{i=1}^n Z_i(j)$, and $\mathbf{S}(j) = \sum_{i=1}^n X_i(j)$. This implies in

$$\begin{aligned} \mathcal{E}_{\text{RG}} &= \frac{2m\lambda_{vl}(1 - \lambda_{gr}) + \lambda_{gr}(2m - 1)}{n(1 - \lambda_{gr})(2m(1 - \lambda_{vl}) - 1)} \left(\mathbb{E}[V_1^2] \right. \\ &\quad \left. + \frac{(4m^2 - 1)(m + 1)}{6(1 - \lambda_{gr})(2m(1 - \lambda_{vl}) - 1)} \right), \end{aligned} \quad (44)$$

which follows from substituting (28), (41), (43) in (1), and noting that $\sum_{v \in \mathcal{V}} v^2 = \frac{1}{3}m(m+1)(2m+1)$.

This proves how we obtained the expression of \mathcal{E}_{RG} as a function of λ_{gr} and λ_{vl} . Moreover, one could be interested in the error as a function of a given required privacy $\epsilon > 0$. We give an upper bound of \mathcal{E}_{RG} as a function of ϵ . We substitute λ_{gr} and λ_{vl} that minimize the error from Corollary 1, in (44). Then, the error \mathcal{E}_{RG} is upper bounded by $\mathcal{O}\left(\frac{m^4 k^2}{n \epsilon^2}\right)$.

B. Proof of Corollary 1

We first assume that $p_{\max} > \frac{1}{2m}$ and $p_{\min} < \frac{1}{2m}$, and consider the special case of $p_{\max} = p_{\min} = \frac{1}{2m}$ separately in

the end. For ease of notation define

$$\begin{aligned} f(\lambda_{gr}, \lambda_{vl}) &= \text{MSE}(\hat{\mathbf{S}}_{\text{RG}}) \\ &= \frac{2m\lambda_{vl}(1 - \lambda_{gr}) + \lambda_{gr}(2m - 1)}{(1 - \lambda_{gr})(2m(1 - \lambda_{vl}) - 1)} \\ &\quad \times \left(\mathbb{E}[V_1^2] + \frac{(4m^2 - 1)(m + 1)}{6(1 - \lambda_{gr})(2m(1 - \lambda_{vl}) - 1)} \right) n \end{aligned}$$

which follows directly from (20). To minimize the error of the RG scheme we solve the following optimization problem,

$$\begin{aligned} &\underset{\lambda_{vl}, \lambda_{gr}}{\text{minimize}} f(\lambda_{gr}, \lambda_{vl}) \\ &\text{subject to } e^\epsilon = \max \left\{ \frac{\frac{2m(k-1)(1-\lambda_{gr})(p_{\max}(2m(1-\lambda_{vl})-1)+\lambda_{vl})}{\lambda_{gr}(2m-1)}}{2m(k-1)(1-\lambda_{gr})(p_{\min}(2m(1-\lambda_{vl})-1)+\lambda_{vl})}, \right. \\ &\quad \left. \frac{\lambda_{gr}(2m-1)}{2m(k-1)(1-\lambda_{gr})(p_{\min}(2m(1-\lambda_{vl})-1)+\lambda_{vl})} \right\}, \\ &0 \leq \lambda_{vl} < \frac{2m-1}{2m}, 0 < \lambda_{gr} < 1. \end{aligned} \quad (45)$$

To solve it, we consider two optimization problems. Consider this first optimization problem, assume its optimal value is attained, and let $\lambda_{gr}^{(1)}$ and $\lambda_{vl}^{(1)}$ be its optimal points,

$$\begin{aligned} &\underset{\lambda_{vl}, \lambda_{gr}}{\text{minimize}} f(\lambda_{gr}, \lambda_{vl}) \\ &\text{subject to } e^\epsilon = \frac{2m(k-1)(1-\lambda_{gr})(p_{\max}(2m(1-\lambda_{vl})-1)+\lambda_{vl})}{\lambda_{gr}(2m-1)}, \\ &\quad \frac{2m(k-1)(1-\lambda_{gr})(p_{\max}(2m(1-\lambda_{vl})-1)+\lambda_{vl})}{\lambda_{gr}(2m-1)} \\ &\geq \frac{\lambda_{gr}(2m-1)}{2m(k-1)(1-\lambda_{gr})(p_{\min}(2m(1-\lambda_{vl})-1)+\lambda_{vl})}, \\ &0 \leq \lambda_{vl} < \frac{2m-1}{2m}, 0 < \lambda_{gr} < 1. \end{aligned} \quad (46)$$

Consider this second optimization problem, assume its optimal value is attained, and let $\lambda_{gr}^{(2)}$ and $\lambda_{vl}^{(2)}$ be its optimal points,

$$\begin{aligned} &\underset{\lambda_{vl}, \lambda_{gr}}{\text{minimize}} f(\lambda_{gr}, \lambda_{vl}) \\ &\text{subject to } e^\epsilon = \frac{\lambda_{gr}(2m-1)}{2m(k-1)(1-\lambda_{gr})(p_{\min}(2m(1-\lambda_{vl})-1)+\lambda_{vl})}, \\ &\quad \frac{\lambda_{gr}(2m-1)}{2m(k-1)(1-\lambda_{gr})(p_{\min}(2m(1-\lambda_{vl})-1)+\lambda_{vl})} \\ &\geq \frac{2m(k-1)(1-\lambda_{gr})(p_{\max}(2m(1-\lambda_{vl})-1)+\lambda_{vl})}{\lambda_{gr}(2m-1)}, \\ &0 \leq \lambda_{vl} < 1 - \frac{1}{2m}, 0 < \lambda_{gr} < 1. \end{aligned} \quad (47)$$

Then, the solution of (45) is $\min \left\{ f(\lambda_{gr}^{(1)}, \lambda_{vl}^{(1)}), f(\lambda_{gr}^{(2)}, \lambda_{vl}^{(2)}) \right\}$.

Therefore, to solve (45), we first solve (46) and (47).

- *Solution of (46):* Since $0 \leq \lambda_{vl} < \frac{2m-1}{2m}$, we have the following two cases.

$$\begin{aligned} &\mathbb{E}\left[\left(\frac{2m-1}{(1-\lambda_{gr})(2m(1-\lambda_{vl})-1)} \sum_{i=1}^n Z_i(j)\right)^2\right] \\ &= \frac{n(4m^2-1)[2m\theta_g(k-1)(1-\lambda_{gr})\lambda_{vl} + \lambda_{gr}(2m-1)(1-\theta_g)]}{6(k-1)(1-\lambda_{gr})^2(2m(1-\lambda_{vl})-1)^2(m+1)^{-1}} \\ &\quad + \frac{n\theta_g(2m-1)}{(1-\lambda_{gr})(2m(1-\lambda_{vl})-1)} \mathbb{E}[V_1^2|G_1=j] + (n^2 - n)\theta_g^2 \mathbb{E}[V_1|G_1=j]^2. \end{aligned} \quad (41)$$

- o If $e^{2\epsilon} < \frac{p_{\max}}{p_{\min}}$, from the first condition directly follows
$$\lambda_{gr}^{(1)} = \frac{2m(k-1)(p_{\max}(2m(1-\lambda_{vl})-1)+\lambda_{vl})}{2m(k-1)(p_{\max}(2m(1-\lambda_{vl})-1)+\lambda_{vl})+e^{\epsilon}}. \quad (48)$$

Since $e^{2\epsilon} < \frac{p_{\max}}{p_{\min}}$, then $(2m-1)(p_{\max}-p_{\min}e^{2\epsilon}) > 0$, and by substituting (48) in the conditions of (46), we get

$$0 < \frac{(2m-1)p_{\max}-(2m-1)p_{\min}e^{2\epsilon}}{(1-2mp_{\min})e^{2\epsilon}+2mp_{\max}-1} \leq \lambda_{vl} < \frac{1}{2}.$$

Since $f(\lambda_{gr}^{(1)}, \lambda_{vl})$ is increasing in λ_{vl} , then its minimum is at achieved at the boundary of the domain, *i.e.*,

$$\lambda_{vl}^{(1)} = \frac{(2m-1)p_{\max}-(2m-1)p_{\min}e^{2\epsilon}}{(1-2mp_{\min})e^{2\epsilon}+2mp_{\max}-1}.$$

Then substituting this back in (48), we get

$$\lambda_{gr}^{(1)} = \frac{2m(k-1)(p_{\max}-p_{\min})e^{\epsilon}}{2m(k-1)(p_{\max}-p_{\min})e^{\epsilon}+(1-2p_{\min})e^{2\epsilon}+2mp_{\max}-1}.$$

- o If $e^{2\epsilon} \geq \frac{p_{\max}}{p_{\min}}$, then $(2m-1)(p_{\max}-p_{\min}e^{2\epsilon}) \leq 0$, and

$$\frac{(2m-1)p_{\max}-(2m-1)p_{\min}e^{2\epsilon}}{(1-2mp_{\min})e^{2\epsilon}+2mp_{\max}-1} \leq 0 \leq \lambda_{vl} < \frac{1}{2}.$$

Similarly, since $f(\lambda_{gr}^{(1)}, \lambda_{vl})$ is increasing in λ_{vl} , then $\lambda_{vl}^{(1)} = 0$. And we have that, $\lambda_{gr}^{(1)} = \frac{2m(k-1)p_{\max}}{2m(k-1)p_{\max}+e^{\epsilon}}$,

- *Solution of (47):* The solution of (47) follows similarly as that of (46), and we have the following two cases.

- o If $e^{2\epsilon} < \frac{p_{\max}}{p_{\min}}$, $\lambda_{vl}^{(2)} = \frac{(2m-1)p_{\max}-(2m-1)p_{\min}e^{2\epsilon}}{(1-2mp_{\min})e^{2\epsilon}+2mp_{\max}-1}$, and

$$\lambda_{gr}^{(2)} = \frac{2m(k-1)(p_{\max}-p_{\min})e^{\epsilon}}{2m(k-1)(p_{\max}-p_{\min})e^{\epsilon}+(1-2p_{\min})e^{2\epsilon}+2mp_{\max}-1}.$$

- o If $e^{2\epsilon} \geq \frac{p_{\max}}{p_{\min}}$, then $\lambda_{vl}^{(2)} = 0$, and $\lambda_{gr}^{(2)} = \frac{2m(k-1)p_{\min}e^{\epsilon}}{2m(k-1)p_{\min}e^{\epsilon}+1}$.

- *Combining the two solutions:* We also have to look at the two cases separately as follows. *leftmargin=+.3in*

- o If $e^{2\epsilon} < \frac{p_{\max}}{p_{\min}}$, the solution is straightforward. The optimal points for (45) are $\lambda_{vl}^* = \frac{(2m-1)p_{\max}-(2m-1)p_{\min}e^{2\epsilon}}{(1-2mp_{\min})e^{2\epsilon}+2mp_{\max}-1}$, and

$$\lambda_{gr}^* = \frac{2m(k-1)(p_{\max}-p_{\min})e^{\epsilon}}{2m(k-1)(p_{\max}-p_{\min})e^{\epsilon}+(1-2p_{\min})e^{2\epsilon}+2mp_{\max}-1}.$$

- o If $e^{2\epsilon} \geq \frac{p_{\max}}{p_{\min}}$, one readily obtains $f(\lambda_{vl}^{(1)}, \lambda_{gr}^{(1)}) \leq f(\lambda_{vl}^{(2)}, \lambda_{gr}^{(2)})$. Therefore, for this case, the optimal points for (45) are $\lambda_{vl}^* = 0$ and $\lambda_{gr}^* = \frac{2m(k-1)p_{\max}}{2m(k-1)p_{\max}+e^{\epsilon}}$.

This completes the proof for $p_{\max} > \frac{1}{2m}$. If $p_{\max} = p_{\min} = \frac{1}{2m}$, then the second condition of (45) reduces to

$$e^{\epsilon} = \max \left\{ \frac{(k-1)(1-\lambda_{gr})}{\lambda_{gr}}, \frac{\lambda_{gr}}{(k-1)(1-\lambda_{gr})} \right\}.$$

i.e., e^{ϵ} is not a function of λ_{vl} . Therefore, for this case, the optimal points for (45) can be readily obtained such that $\lambda_{vl}^* = 0$ and $\lambda_{gr}^* = \frac{k-1}{k-1+e^{\epsilon}}$. Combining all the described cases completes the proof.

Remark 5: In Corollary 1, we minimize the relative error subject to a fixed privacy parameter ϵ . Because of the

monotonicity of the relative error as a function of ϵ , an increase in privacy, *i.e.*, smaller ϵ , cannot decrease the error. Thus, minimizing the error subject to

$$e^{\epsilon} \leq \max \left\{ \frac{2m(k-1)(1-\lambda_{gr})(p_{\max}(2m(1-\lambda_{vl})-1)+\lambda_{vl})}{\lambda_{gr}(2m-1)}, \frac{\lambda_{gr}(2m-1)}{2m(k-1)(1-\lambda_{gr})(p_{\min}(2m(1-\lambda_{vl})-1)+\lambda_{vl})} \right\},$$

is equivalent to solving the optimization (45).

APPENDIX C COMPARISON: PROOF OF THEOREM 3

We start by sketching the proof of (i) in Theorem 3.

- o For $k = 2$, $m = 1$, and $p_1(v) \neq p_2(v')$ or $p_1(v) = p_2(v) = 0.5$ for all $v, v' \in \mathcal{V} = \{-1, 1\}$, we can easily find the exact value of λ that satisfies (3); therefore, we can find the expression for the error of the Q&A scheme $\mathcal{E}_{QA}(\epsilon, b)$. Moreover, the minimum error of the RG scheme $\mathcal{E}_{RG}(\epsilon, b)$ follows from Corollary 1. We find that the limit of the difference of the errors, $\mathcal{E}_{RG}(\epsilon, b) - \mathcal{E}_{QA}(\epsilon, b)$, as ϵ goes to zero, is positive.
- o For $k > 2$ and $m = 1$ or $k \geq 2$ and $m > 1$, the minimum error of the RG scheme $\mathcal{E}_{RG}(\epsilon, b)$ follows from Corollary 1. From Remark 2, to guarantee a required privacy ϵ , we can choose any $\lambda \geq \frac{(2m-1)p_{\max}-p_{\min}e^{\epsilon}}{2m(p_{\max}-p_{\min}e^{\epsilon})+e^{\epsilon}-1}$. We use this λ to bound the error of the Q&A scheme. Finally, we find that the bound on the limit of the difference of the errors, $\mathcal{E}_{RG}(\epsilon, b) - \mathcal{E}_{QA}(\epsilon, b)$, as ϵ goes to zero, is positive.

Now we prove (ii) of Theorem 3 by showing that there exists an $\epsilon_{\ell} > 0$, such that for all $\epsilon > \epsilon_{\ell}$, we have $\mathcal{E}_{QA}(\epsilon, b) > \mathcal{E}_{RG}(\epsilon, b)$. We first consider the Q&A scheme. From Remark 2, there exists an $\epsilon_0 > 0$, such that for all $\epsilon > \epsilon_0$, the parameter $\lambda = 0$ guarantees privacy level ϵ_0 . And the error of the Q&A scheme, as defined in (21), for $\lambda = 0$, *i.e.*, all $\epsilon > \epsilon_0$, is

$$\mathcal{E}_{QA}(\epsilon, b) = \frac{1}{6b} \log(2m)(2m+1)(m+1)(k-1) > 0.$$

Let $\epsilon_{\ell} > \epsilon_0 > \sqrt{\frac{p_{\max}}{p_{\min}}}$, then from Corollary 1, the parameters

$\lambda_{vl}^* = 0$ and $\lambda_{gr}^* = \frac{2m(k-1)p_{\max}}{2m(k-1)p_{\max}+e^{\epsilon_{\ell}}}$ minimize the error of the RG scheme. Thus, there exists ϵ_{ℓ} , such that for all $\epsilon > \epsilon_{\ell}$,

$$\mathcal{E}_{QA}(\epsilon_{\ell}, b) - \mathcal{E}_{RG}(\epsilon_{\ell}, b) = \frac{\log(2m)(2m+1)(m+1)(k-1)}{6b} > 0,$$

which completes the proof.

ACKNOWLEDGMENT

The authors would like to thank Peter Kairouz for his helpful discussions.

REFERENCES

- [1] S. Kim, M. K. Sung, and Y. D. Chung, “A framework to preserve the privacy of electronic health data streams,” *J. Biomed. Inf.*, vol. 50, pp. 95–106, Aug. 2014.
- [2] K. Bonawitz *et al.*, “Practical secure aggregation for privacy-preserving machine learning,” in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2017, pp. 1175–1191.

- [3] M. Abadi *et al.*, “Deep learning with differential privacy,” in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2016, pp. 308–318.
- [4] S. Truex, L. Liu, K.-H. Chow, M. E. Gursoy, and W. Wei, “LDP-fed: Federated learning with local differential privacy,” in *Proc. 3rd ACM Int. Workshop Edge Syst., Analytics Netw.*, Apr. 2020, pp. 61–66.
- [5] M. Kim, O. Günlü, and R. F. Schaefer, “Federated learning with local differential privacy: Trade-offs between privacy, utility, and communication,” 2021, *arXiv:2102.04737*.
- [6] T. H. Chan, E. Shi, and D. Song, “Optimal lower bound for differentially private multi-party aggregation,” in *Proc. Eur. Symp. Algorithms*, 2012, pp. 277–288.
- [7] C. Dwork, “Differential privacy,” in *Proc. 33rd Int. Colloq. Automata, Lang. Program. (ICALP)*, 2006, pp. 1–9.
- [8] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, “Local privacy and statistical minimax rates,” in *Proc. IEEE 54th Annu. Symp. Found. Comput. Sci.*, Oct. 2013, pp. 429–438.
- [9] B. Ghazi, N. Golowich, R. Kumar, P. Manurangsi, R. Pagh, and A. Velingker, “Pure differentially private summation from anonymous messages,” in *Proc. 1st Conf. Inf.-Theoretic Cryptogr. (ITC)*, 2020, pp. 1–40.
- [10] S. Goryczka and L. Xiong, “A comprehensive comparison of multiparty secure additions with differential privacy,” *IEEE Trans. Dependable Sec. Comput.*, vol. 14, no. 5, pp. 463–477, Oct. 2017.
- [11] S. Truex *et al.*, “A hybrid approach to privacy-preserving federated learning,” in *Proc. 12th ACM Workshop Artif. Intell. Secur. (AISec)*, 2019, pp. 1–11.
- [12] E. Shi, T. H. Chan, E. Rieffel, R. Chow, and D. Song, “Privacy-preserving aggregation of time-series data,” in *Proc. 18th Annu. Netw. Distrib. Syst. Secur. Symp. Proc. (NDSS)*, 2011, pp. 1–17.
- [13] H. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, “Communication-efficient learning of deep networks from decentralized data,” in *Proc. 20th Int. Conf. Artif. Intell. Statist. (AISTATS)*, 2017, pp. 1273–1282.
- [14] J. So, B. Guler, and A. S. Avestimehr, “Turbo-aggregate: Breaking the quadratic aggregation barrier in secure federated learning,” 2020, *arXiv:2002.04156*.
- [15] S. Kadhe, N. Rajaraman, O. O. Koyluoglu, and K. Ramchandran, “Fast-SecAgg: Scalable secure aggregation for privacy-preserving federated learning,” 2020, *arXiv:2009.11248*.
- [16] K. Pillutla, S. M. Kakade, and Z. Harchaoui, “Robust aggregation for federated learning,” 2019, *arXiv:1912.13445*.
- [17] J. Konečný and P. Richtárik, “Randomized distributed mean estimation: Accuracy vs. communication,” *Frontiers Appl. Math. Statist.*, vol. 4, p. 62, Dec. 2018.
- [18] A. T. Suresh, F. X. Yu, S. Kumar, and H. B. McMahan, “Distributed mean estimation with limited communication,” in *Proc. 34th Int. Conf. Mach. Learn. (ICML)*, 2017, pp. 3329–3337.
- [19] L. P. Barnes, Y. Han, and A. Ozgur, “Lower bounds for learning distributions under communication constraints via Fisher information,” 2019, *arXiv:1902.02890*.
- [20] I. Diakonikolas, E. Grigorescu, J. Li, A. Natarajan, K. Onak, and L. Schmidt, “Communication-efficient distributed learning of discrete distributions,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1–11.
- [21] P. Kairouz, K. Bonawitz, and D. Ramage, “Discrete distribution estimation under local privacy,” in *Proc. 33rd Int. Conf. Int. Conf. Mach. Learn. (ICML)*, 2016, pp. 2436–2444.
- [22] M. Ye and A. Barg, “Optimal schemes for discrete distribution estimation under local differential privacy,” in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2017, pp. 759–763.
- [23] I. Diakonikolas, M. Hardt, and L. Schmidt, “Differentially private learning of structured discrete distributions,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2566–2574.
- [24] J. Acharya, Z. Sun, and H. Zhang, “Hadamard response: Estimating distributions privately, efficiently, and with little communication,” in *Proc. 22nd Int. Conf. Artif. Intell. Statist.*, 2019, pp. 1120–1129.
- [25] S. Wang *et al.*, “Mutual information optimally local private discrete distribution estimation,” 2016, *arXiv:1607.08025*.
- [26] Ú. Erlingsson, V. Pihur, and A. Korolova, “RAPPOR: Randomized aggregatable privacy-preserving ordinal response,” in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Nov. 2014, pp. 1054–1067.
- [27] J. Acharya and Z. Sun, “Communication complexity in locally private distribution estimation and heavy hitters,” in *Proc. 36th Int. Conf. Mach. Learn. (ICML)*, 2019, pp. 51–60.
- [28] W. Zhu, P. Kairouz, H. Sun, B. McMahan, and W. Li, “Federated heavy hitters with differential privacy,” in *Proc. 23rd Int. Conf. Artif. Intell. Statist. (AISTATS)*, 2020, pp. 3837–3847.
- [29] W.-N. Chen, P. Kairouz, and A. Özgür, “Breaking the communication-privacy-accuracy trilemma,” 2020, *arXiv:2007.11707*.
- [30] C. Niu *et al.*, “Secure federated submodel learning,” 2019, *arXiv:1911.02254*.
- [31] M. Kim and J. Lee, “Information-theoretic privacy in federated submodel learning,” 2020, *arXiv:2008.07656*.
- [32] Z. Jia and S. A. Jafar, “X-secure T-private federated submodel learning,” *arXiv:2010.01059*, 2020.
- [33] S. L. Warner, “Randomized response: A survey technique for eliminating evasive answer bias,” *J. Amer. Statist. Assoc.*, vol. 60, no. 309, pp. 63–69, 1965.

Carolina Naim (Graduate Student Member, IEEE) received the B.S. degree in communication and computer engineering from the American University of Science and Technology, Beirut, in 2017, and the M.S. degree in electrical and computer engineering from the Rutgers, The State University of New Jersey, NJ, USA, in 2020, where she is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering. Her research interests include privacy and security with applications to information retrieval, distributed computing, and machine learning.

Rafael G. L. D’Oliveira (Member, IEEE) received the B.S. and M.S. degrees in mathematics and the Ph.D. degree in applied mathematics from the University of Campinas, Brazil, in 2009, 2012, and 2017, respectively. He was a Post-Doctoral Research Associate with the Rutgers, The State University of New Jersey, from 2018 to 2019 and the Illinois Institute of Technology in 2017, and he did a research internship at Télécom ParisTech from 2015 to 2016. He is a Post-Doctoral Research Associate with the Research Laboratory of Electronics, Massachusetts Institute of Technology. His research interests include privacy and security, distributed computing, coding theory, and information theory.

Salim El Rouayheb (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from Texas A&M University, College Station, in 2009. He was an Assistant Professor with the ECE Department, Illinois Institute of Technology, Chicago, from 2013 to 2017. He was a Research Scholar at the Electrical Engineering Department, Princeton University, from 2012 to 2013, and a Post-Doctoral Researcher at the EECS Department, University of California at Berkeley, Berkeley, from 2010 to 2011. He is an Associate Professor at the ECE Department, Rutgers, The State University of New Jersey, New Brunswick, NJ, USA. In 2019, he was the Rutgers University Walter Tyson Junior Faculty Chair. His research interests are in the broad area of information theory and coding theory with applications to reliability, security, and privacy in distributed systems. He received the Google Faculty Award in 2018 and the NSF CAREER Award in 2016.