Fast Private Norm Estimation and Heavy Hitters

Joe Kilian*, André Madeira†, Martin J. Strauss‡, and Xuan Zheng§

Abstract. We consider the problems of computing the Euclidean norm of the difference of two vectors and, as an application, computing the large components (Heavy Hitters) in the difference. We provide protocols that are approximate but private in the semi-honest model and efficient in terms of time and communication in the vector length N. We provide the following, which can serve as building blocks to other protocols:

- *Euclidean norm problem:* we give a protocol with quasi-linear local computation and polylogarithmic communication in N leaking only the true value of the norm. For processing massive datasets, the intended application, where N is typically huge, our improvement over a recent result with quadratic runtime is significant.
- *Heavy Hitters problem:* suppose, for a prescribed B, we want the B largest components in the difference vector. We give a protocol with quasi-linear local computation and polylogarithmic communication leaking only the set of true B largest components and the Euclidean norm of the difference vector. We justify the leakage as (1) desirable, since it gives a measure of goodness of approximation; or (2) inevitable, since we show that there are contexts where linear communication is required for approximating the Heavy Hitters.

1 Introduction

Secure Multiparty Computation (SMC) has been studied for decades since [6, 22]. Any protocol for computing a function can be converted, gate-by-gate, to a *private* protocol, in which no party learns anything from the protocol messages other than what can be inferred from the function's input/output relation. The computational overhead is at most polynomial in the size of the inputs.

In recent years, however, input sizes in many problems have grown to the point where "polynomial computational overhead" is too coarse a measure; both computation and communication should be minimized. For example, absent privacy concerns, applications may require that a protocol use at most polylogarithmic communication—this occurs in processing distributed internet traffic

* Department of Computer Science, Rutgers University, Piscataway, NJ 08854 USA, jkilian@cs.rutgers.edu. Supported in part by NSF grant CCF-0728937

† Department of Computer Science, Rutgers University, Piscataway, NJ 08854 USA, amadeira@cs.rutgers.edu.

‡ Departments of Math and EECS, University of Michigan, Ann Arbor, MI 48109 USA, martinjs@umich.edu. Supported in part by NSF grant DMS-0354600.

§ Department of EECS, University of Michigan, Ann Arbor, MI 48109 USA, xuanzh@eecs.umich.edu. Supported in part by NSF grant DMS-0354600.

at line speeds or in performing data mining algorithms in very large datasets. General-purpose SMC may blow up communication exponentially, so additional techniques are needed. In one theoretical approach, individual protocols are designed for functions of interest such as database lookup [9, 18, 7] and building decision trees [19]. Another important approach [21], converts any protocol into a private one with little communication blowup, but imposes a computational blowup that may be exponential in the communication complexity.

The approach we follow, which was introduced in [11], is to substitute an approximate function for the desired exact function. Many functions of interest have good approximations that can be computed efficiently both in terms of computation and communication. A caveat is that the traditional definition of privacy is no longer appropriate. Instead, a protocol π computing an approximation g to a function f is a private approximation protocol [11] for f if

- π is a private protocol for g in the traditional sense that the messages of π leak nothing beyond what is implied by inputs and g; *and*,
- the output g leaks nothing beyond what is implied by the inputs and f.

Several examples were given in [11]. Another important example was given in [15], where the authors provided an estimate $\|a - b\|_{\sim}$ (for integer-valued vectors a and b held by Alice and Bob respectively) as the first non-trivial example of polylogarithmic communication and polynomial computation. We will analyze and make use of this protocol for our results.

1.1 Our Results

Consider the general problem where Alice and Bob hold vectors, a and b, of dimension N, and they want an efficient summary for the vector sum $c = a + b$. We analyze two problems in this setting.

First, we consider the Euclidean norm estimation problem, in which we wish to output a tight approximation to $\|c\|_2 = \left(\sum_i c_i^2\right)^{1/2}$, the l_2 norm of vector c. The problem is a well-known building block for other protocols in the non-private setting, since it is used to estimate the skew of the data. A private protocol approximating $\|c\|_2$ using polylogarithmic communication in the vector length first appeared in [15]. Our results are based on their protocol; we strengthen its privacy guarantees and decrease its local computational costs. Specifically, we obtain a $O(N \log N)$ local computational cost versus the implied $\Omega(N^2)$, while keeping equivalent communication and round complexity costs.

Second, we consider the Euclidean approximate Heavy Hitters problem, in which there is a parameter, B, and the players ideally want c_{opt}, the B largest terms in c; *i.e.*, the B biggest values together with the corresponding indices. Unfortunately, finding c_{opt} exactly requires linear communication. Instead, the players use polylogarithmic communication (and polynomial work and $O(1)$ rounds) to output a vector $\widetilde{c}$ with $\|\widetilde{c} - c\|_2 \leq (1 + \epsilon)\|c_{\text{opt}} - c\|_2$. In our protocol, the players learn nothing more than what can be deduced from c_{opt} and $\|c\|_2$. (We discuss below the significance of leaking $\|c\|_2$.) We can immediately use this result as a black box for *taxicab* approximate heavy hitters, *i.e.*, finding $\widetilde{c}$ with

$\|\widetilde{c} - c\|_1 \le (1+\epsilon)\|c_{\text{opt}} - c\|_1$, leaking c_{opt} and $\|c\|_2$. We omit the development of this extension in the interest of space.

In the basic result, we give an at-most-B-term representation that is nearly as good (in the Euclidean sense) as the best B-term representation and leaks no more than the best B-term representation and *the exact Euclidean norm.* Although leaking the Euclidean norm represents a weaker result than not leaking it, actually (i) leaking $\|c\|_2$ is necessary and (ii) computing or approximating $\|c\|_2$ is desirable in some circumstances. First, we sketch a straightforward lower bound showing that, for some (reasonable) values of parameters $M, N, \ldots$, computing $\widetilde{c}$ leaking only c_{opt} requires $\Omega(N)$ communication. In fact, for some (artificial) classes of inputs, $\Omega(N)$ communication is needed unless $\|c\|_2$ itself is not only potentially leaked, but also actually computed exactly. On the other hand, one can regard the Euclidean norm as semantically interesting, so that we can regard the top B terms *together with the Euclidean norm* as a compound, extended summary. In particular, since $\widetilde{c}$ is computed, leaking $\|c\|_2$ is equivalent to leaking $\|c\|_2^2 - \|\widetilde{c}\|_2^2 = \|\widetilde{c} - c\|_2^2$, *i.e.*, the error in our representation, which is a useful and common desired result. Our protocol indeed can be modified to output an approximation $\|\widetilde{c} - c\|_\sim$ with $\|\widetilde{c} - c\|_2 \le \|\widetilde{c} - c\|_\sim \le (1+\epsilon)\|\widetilde{c} - c\|_2$, so we can regard the protocol as solving two cascaded approximation problems: find a near-best representation $\widetilde{c}$, then find an approximation $\|\widetilde{c} - c\|_\sim$ to $\|\widetilde{c} - c\|_2$. It is natural to expect a protocol for $\widetilde{c}$ to leak c_{opt} and a protocol for $\|\widetilde{c} - c\|_\sim$ to leak $\|\widetilde{c} - c\|_2$; while lower bounds prevent that, we can compute $\widetilde{c}$ and $\|\widetilde{c} - c\|_\sim$ *simultaneously* and guarantee that, *overall*, we leak only c_{opt} and $\|\widetilde{c} - c\|_2$.

1.2 Related Work

Other works in private communication-efficient protocols include the Private Information Retrieval problem [9, 18, 7], building decision trees [19], the set intersection and matching problem [12], and computing the k^{th}-ranked element [2]. The breakthrough work of [21] gives a general technique for converting any protocol into a private protocol with little communication overhead. However, this comes at the expense of local computational costs, which may increase exponentially. Thus, other general or application-specific techniques are needed.

The seminal work of [11] introduced the notion of private approximations and gave several protocols. Some negative results followed in [14] for approximations to NP-hard functions and more on NP-hard search problems appears in [5]. Recently, [15] gave a private approximation to the Euclidean norm that is central to our paper. Statistical work such as [8] also addresses approximate summaries over large databases, but differs from our work in many parameters, such as the number of players and the allowable communication.

Several papers address the Heavy Hitters problem, in a variety of contexts. Many of the needed ideas can be seen in [16] as well as in [3, 4, 10, 13]. However, none are directly suitable when privacy is a concern.

Road map

This paper is organized as follows. In Section 2, we present some necessary definitions used throughout the paper. We review private approximations in Section 3. In Section 4 we present our results for the private Euclidean norm estimation. Finally, in Section 5, we present our private approximate Euclidean Heavy Hitters protocol and some suitable lower bounds that motivate our results.

2 Preliminaries

Fix parameters N, M, B, k, and a distortion parameter ϵ. In this paper, we consider only two players, Alice and Bob, holding input vectors a and b respectively, each of dimension N, and taking integer values in the range $[-M, +M]$. Let k be a security and failure probability parameter and $\mathrm{neg}(k, N)$ be an arbitrary negligible function of k and N, i.e. a function that shrinks faster than any inverse polynomial in k and N. We guarantee summaries whose error is at most the factor $(1+\epsilon)$ times the error of the best possible summary; and we will be interested in protocols that use communication $\mathrm{poly}(B, \log(N), k, \log(M), 1/\epsilon)$, local computation $\mathrm{poly}(B, N, k, \log(M), 1/\epsilon)$, and $O(1)$ of rounds.

The Euclidean norm of a vector c is $\|c\|_2 = \left(\sum_i c_i^2\right)^{1/2}$. For the Heavy Hitters protocol, we are interested in summaries of size B for the combined vector $c = a + b$. For example, we are interested ideally in the largest B terms of c. A vector c is written $c = (c_0, c_1, c_2, \ldots, c_{N-1}) = \sum_j c_j \delta_j$, where j is an *index*, c_j is a *value*, δ_j is the vector that is 1 at index j and 0 elsewhere, and $c_j\delta_j$, which can be implemented compactly and equivalently written as the pair (j, c_j), is a *term*, in which c_j is the *coefficient*. We compare terms by the *magnitudes* of their coefficients, breaking ties by the indices. That is, we will say that $(j, c_j) < (k, c_k)$ if $|c_j| < |c_k|$ or both $|c_j| = |c_k|$ and $j < k$. Thus all terms are strictly comparable. A heavy hitter summary is an expression of the form $\sum_{i\in\Lambda} \eta_i \delta_i$. If $|\Lambda|$ must be at most B, then the best heavy hitter summary c_{opt} for a vector c occurs where $\{(i, \eta_i) : i \in \Lambda\}$ consists of the B largest terms.

2.1 Approximate Data Summaries

A function g is said to be an $\langle\epsilon, \delta\rangle$-approximation of f if, for all inputs x, $\Pr[(1-\epsilon)f(x) \le g(x) \le (1+\epsilon)f(x)] \ge 1-\delta$ holds for an approximation error $\epsilon \in (0, 1)$ and confidence parameter $\delta \in (0, 1)$. The probabilistic guarantee is over the randomness of g.

In the exact Heavy Hitters problem, we are given parameters B and N and the goal is to find the B largest terms in a vector of dimension N. In the approximate Heavy Hitters problem, however, we want a summary $\widetilde{c} = \sum_{i\in\Lambda} \eta_i\delta_i$ such that $\|\widetilde{c} - c\| \le (1+\epsilon)\|c_{\mathrm{opt}} - c\|$, where the norms are all Euclidean norms.

In order to describe previous relevant algorithms, we first need some definitions. Fix a vector $c = (c_0, c_1, c_2, \ldots, c_{N-1}) = \sum_{0\le i<N} c_i\delta_i$, whose terms are $t_0 = (0, c_0), t_1 = (1, c_1), \ldots, t_{N-1} = (N-1, c_{N-1})$. Suppose the sequence $i'_0, i'_1, \ldots$ is a decreasing rearrangement of c, *i.e.*, $t_{i'_0} > t_{i'_1} > \cdots > t_{i'_{N-1}}$.

Definition 1 (Significant index) *Let $I \subseteq [0,N)$ be a set of indices. Then i is a (I,θ)-*significant index *for c if and only if $c_i^2 \geq \theta \sum_{j\in I} |c_j|^2$.*

That is, an index is significant if the corresponding value is large compared with all the other values. In some of the algorithms below, we will find the largest term (if it is sufficiently large), subtract it off, then recurse on the residual signal. This motivates the following definitions.

Definition 2 (Qualified index set) *Fix parameters ℓ and θ. The set $Q = \{i'_0, i'_1, \ldots, i'_{m-1}\}$ is a (ℓ,θ)-*qualified index set *for c* if and only if *(a) $m \leq \ell$; (b) $\forall j \in [0, m-1]$, i'_j is a $(\{i'_j, i'_{j+1}, \ldots, i'_{N-1}\}, \theta)$-significant index; and (c) i'_m is NOT a $(\{i'_m, i'_{m+1}, \ldots, i'_{N-1}\}, \theta)$-significant index.*

That is, a qualified index set consists of the largest possible dimension m for a prefix of $i'_0, i'_1, \ldots, i'_{m-1}$ such that, for each $j < m$, we have $c^2_{i'_j} \geq \theta(c^2_{i'_j} + c^2_{i'_{j+1}} + c^2_{i'_{j+2}} + \cdots + c^2_{i'_{N-1}})$. In particular, if the terms happen to be in decreasing order to begin with, *i.e.*, if $|c_0| > |c_1| > \cdots$, then a qualified index set is $\{0,1,2,\ldots,m-1\}$ for the largest m such that, for each $j < m$, we have $c_j^2 \geq \theta(c_j^2 + c_{j+1}^2 + c_{j+2}^2 + \cdots c_{N-1}^2)$. Note that for each ℓ, θ, and vector c, there is only one (ℓ,θ)-qualified index set for c. We use $Q_{c,\ell,\theta}$ to denote it and sometimes write $Q_{\ell,\theta}$ when c is understood. The following Proposition is then straightforward.

Proposition 3 *For any $\theta_1 < \theta_2$, Q_{ℓ,θ_2} set is a subset of Q_{ℓ,θ_1}.*

Proposition 4 *Fix parameters N, M, B, k, ϵ and the vector c as above. If $\widetilde{c} = \sum_{i\in Q_{c,B,\frac{\epsilon}{B(1+\epsilon)}}} c_i\delta_i$, then $\|\widetilde{c} - c\|_2^2 \leq (1+\epsilon)\|c_{\text{opt}} - c\|_2^2$.*

Proof. Assume without loss of generality that $|c_0| > |c_1| > \cdots$ and let $q = |Q_{c,B,\frac{\epsilon}{B(1+\epsilon)}}|$. If $q = B$, then $\widetilde{c} = c_{\text{opt}}$ and we are done. Otherwise we have $\|\widetilde{c} - c\|_2^2 = \sum_{q\leq i<B} |c_i|^2 + \|c_{\text{opt}} - c\|_2^2 \leq B|c_q|^2 + \|c_{\text{opt}} - c\|_2^2 \leq \frac{\epsilon}{1+\epsilon}\|\widetilde{c} - c\|_2^2 + \|c_{\text{opt}} - c\|_2^2$, whence $(1 - \epsilon/(1+\epsilon))\,\|\widetilde{c} - c\|_2^2 \leq \|c_{\text{opt}} - c\|_2^2$. The result follows. □

The algorithms below work from a linear *sketch* of a vector.

Definition 5 (Sketch of a vector) *Given a vector c, a* linear sketch *of c is Rc, where R is a random matrix, called the* measurement matrix*, generated from a prescribed distribution*

In our case, as is typical, the matrix R is a pseudorandom matrix that can be generated from a short pseudorandom seed. We use sketching for the NORM_ESTIMATION protocol (Protocol 1 in Section 4), in which the generator needs to be secure against small space, and a different measurement matrix in the non-private Euclidean Heavy Hitters protocol, where, *e.g.*, pairwise independence suffices for the pseudorandom number generator.

An algorithm in connection with the approximate Euclidean Heavy Hitters problem satisfying the following is known [13]:

Theorem 6. *Fix N, M, B, k, ϵ as above, and $\theta \geq \mathrm{poly}(\log(N), \log(M), B, k, 1/\epsilon)^{-1}$. There is a distribution on sketch matrices R and a corresponding algorithm that, from R and sketch Rc of a vector c, outputs a superset of $Q_{c,B,\theta}$, in time* $\mathrm{poly}(\log(N), \log(M), B, k, 1/\epsilon)$.

In particular, the number or rows in R and the size of the output is bounded by the expression $\mathrm{poly}(\log(N), \log(M), B, k, 1/\epsilon)$ in accordance with the time bound on the algorithm. The algorithm admits efficient Secure Function Evaluation protocols, and can be modified to run privately in $\mathrm{poly}(\log(N), \log(M), B, k, 1/\epsilon)$ time. Note that the algorithm returns a superset of $Q_{c,B,\theta}$ but that even $Q_{c,B,\theta}$ itself suffices for a good approximation.

2.2 Private Two-Player Protocol

SMC allows two or more parties to evaluate a previously-agreed-upon function of their inputs, while hiding their inputs from each other. Here, we assume that all parties are computationally bounded and semi-honest, meaning they follow the protocol but may keep message histories in an attempt to learn more than is prescribed. The adversary is thus passive and can't modify the behavior of corrupted parties. In [21], the authors have shown how to transform a semi-honest protocol into a protocol secure in the malicious model, where parties deviate from the protocol arbitrarily using a different input or outputing the wrong answer, or even exiting from the protocol prematurely. Therefore, we assume parties are semi-honest for the remainder of the paper.

Formally, a two-party computation is specified by a (possibly randomized) mapping g from a pair of inputs $(a, b) \in \{0,1\}^* \times \{0,1\}^*$ to a pair of outputs $(c, d) \in \{0,1\}^* \times \{0,1\}^*$. Let $\pi = (\pi_A, \pi_B)$ be a two-party protocol computing g. Consider the probability space induced by the execution of π on input $\mathbf{x} = (a, b)$ (induced by the independent choices of random inputs r_A, r_B). Let $\mathsf{view}_A^\pi(\mathbf{x})$ (resp., $\mathsf{view}_B^\pi(\mathbf{x})$) denote the entire view of Alice (resp., Bob) in this execution, including her input, random input, and all messages she has received. Let $\mathsf{output}_A^\pi(\mathbf{x})$ (resp., $\mathsf{output}_B^\pi(\mathbf{x})$) denote Alice's (resp., Bob's) output. Note that the above four random variables are defined over the same probability space. Two distributions (or ensembles) $\mathcal{D}_1$ and $\mathcal{D}_2$ are said to be *computationally indistinguishable* with security parameter k, $\mathcal{D}_1 \stackrel{c}{\equiv} \mathcal{D}_2$, if, for any $X_1 \sim \mathcal{D}_1$ and $X_2 \sim \mathcal{D}_2$ and, for any family of polynomial-size circuits $\{C_k\}$, we have $|\Pr(C_k(X_1) = 1) - \Pr(C_k(X_2) = 1)| \leq \mathrm{neg}(k)$.

Definition 7 (Private two-party protocol) *Let X be the set of all valid inputs $\mathbf{x} = (a, b)$. A protocol π is a* private protocol *computing g if the following properties hold:*

Correctness. *The joint outputs are distributed according to $g(a, b)$. Formally,*

$$\{(\mathsf{output}_A^\pi(\mathbf{x}), \mathsf{output}_B^\pi(\mathbf{x}))\}_{\mathbf{x} \in X} \equiv \{(g_A(\mathbf{x}), g_B(\mathbf{x}))\}_{\mathbf{x} \in X},$$

where $(g_A(\mathbf{x}), g_B(\mathbf{x}))$ is the joint distribution of the outputs of $g(\mathbf{x})$.

Privacy. *There exist probabilistic polynomial-time algorithms $\mathcal{S}_A, \mathcal{S}_B$, also known as* simulators*, such that:*

$$\{(\mathcal{S}_A(a, g_A(\mathbf{x})), g_B(\mathbf{x}))\}_{\mathbf{x}=(a,b)\in X} \stackrel{c}{\equiv} \{(\mathsf{view}_A^\pi(\mathbf{x}), \mathsf{output}_B^\pi(\mathbf{x}))\}_{\mathbf{x}\in X}$$
$$\{(g_A(\mathbf{x}), \mathcal{S}_B(b, g_B(\mathbf{x}))\}_{\mathbf{x}=(a,b)\in X} \stackrel{c}{\equiv} \{(\mathsf{output}_A^\pi(\mathbf{x}), \mathsf{view}_B^\pi(\mathbf{x}))\}_{\mathbf{x}\in X}$$

Yao, in its seminal work [22], provided a general technique:

Proposition 8 (General-Purpose SMC [22]) *Two parties holding inputs x and y can privately compute any circuit C with communication and computation $O(k(|C| + |x| + |y|))$, where k is a security parameter, in $O(1)$ rounds.*

We also require the following notion of evaluating a circuit with ROM securely. In this context, the i^{th} party has a table $R_i \in (\{0,1\}^r)^s$, a function of its inputs. Then, the circuit has *lookup gates*, which on inputs (i, j) returns $R_i[j]$.

Proposition 9 (Secure Circuit with ROM [21]) *If C is a circuit with ROM, then it can be securely evaluated with $O(k|C|T(r,s))$ communication in $O(1)$ rounds, where $T(r,s)$ is the communication of 1-out-of-s Oblivious Transfer (OT) protocol on words of size r.*

We will need the following standard definitions for our results in Section 5.

Definition 10 (Additive Secret Sharing) *An intermediate value x of a joint computation is said to be* secret shared *between Alice and Bob if Alice holds r and Bob holds $x - r$, modulo some large prime, where r is a random number independent of all inputs and outputs.*

Definition 11 (Private Sample Sum) *At the start, Alice holds a vector a of dimension N and Bob holds a vector b. Alice and Bob also hold a secret sharing of an index i. At the end, Alice and Bob hold a secret sharing of $a_i + b_i$.*

That is, neither the index i nor the value $a_i + b_i$ becomes known to the parties. Efficient protocols for this problem can be found (or can be constructed immediately from related results) in [21, 11], under various assumptions about the existence of Private Information Retrieval, such as in [7].

Proposition 12 *There is a protocol* PRIVATE-SAMPLE-SUM *for the Private Sample Sum problem that requires* $\mathrm{poly}(N,k)$ *computation,* $\mathrm{poly}(\log(N),k)$ *communication, and $O(1)$ rounds.*

3 Private Approximations

In this Section, we review the notion of private approximations introduced in [11].

Definition 13 (Private Approximation Protocol (*strict* sense) [11]) *A two-party private approximation protocol for a deterministic, common-output function g on inputs a and b is* strict *if it computes an approximation $\widetilde{g}$ to g such that: (a) $\widetilde{g}$ is a good approximation to g (in the appropriate sense); (b) π is a private protocol for $\widetilde{g}$ in the traditional sense (Definition 7); and (c) (Functional Privacy) there exists a probabilistic polynomial-time (PPT) simulator $\mathcal{S}$ s.t.:*

$$\{\mathcal{S}(g(\mathbf{x}))\}_{\mathbf{x}=(a,b)\in X} \stackrel{c}{\equiv} \widetilde{g}(\mathbf{x}).$$

In the case the output to both parties is a deterministic function, a (weakly) equivalent definition is as follows, known as the "liberal" definition in [11]:

Definition 14 (Private Approximation Protocol (*liberal* sense) [11]) *A two-party private approximation protocol for a deterministic, common-output function g on inputs a and b is* liberal *if it computes an approximation $\widehat{g}$ to g such that: (a) $\widehat{g}$ is a good approximation to g (in the appropriate sense); (b) π is a private protocol for $\widehat{g}$, with correctness as in Definition 7 and privacy as in existing PPT simulators $\mathcal{S}_A$ and $\mathcal{S}_B$ such that:*

$$\{\mathcal{S}_A(a, g(\mathbf{x}))\}_{\mathbf{x}=(a,b)\in X} \stackrel{c}{\equiv} \{\mathsf{view}^{\pi}_A(\mathbf{x})\}_{\mathbf{x}\in X}$$
$$\{\mathcal{S}_B(b, g(\mathbf{x}))\}_{\mathbf{x}=(a,b)\in X} \stackrel{c}{\equiv} \{\mathsf{view}^{\pi}_B(\mathbf{x})\}_{\mathbf{x}\in X};$$

and (c) (Functional Privacy) there exists a PPT simulator $\mathcal{S}$ such that:

$$\{\mathcal{S}(g(\mathbf{x}))\}_{\mathbf{x}=(a,b)\in X} \stackrel{c}{\equiv} \widehat{g}(\mathbf{x}).$$

We elaborate on a general technique, originally sketched in [11], to construct a private protocol in the *strict* sense *given* a private protocol in the *liberal* sense. We also show the intuitive fact that the converse always holds.

Proposition 15 (equivalency between *liberal* and *strict* definitions) *Any private approximation protocol in the* liberal *sense requiring only polylogarithmic communication complexity can be transformed into a private approximation protocol in the* strict *sense with the same asymptotic communication complexity, local computational costs, and rounds. The converse holds true as well.*

Proof. Let $\widehat{g}$ and $\widetilde{g}$ be $\langle\epsilon, \delta\rangle$-approximations of g; and $\widehat{\pi}$ and $\widetilde{\pi}$ be private protocols computing $\widehat{g}$ and $\widetilde{g}$ in the *liberal* and *strict* sense respectively. Now, suppose there are simulators in the *strict* sense. Then, putting $\widehat{g} = \widetilde{g}$, a simulator for the *liberal* definition can be constructed by simulating $\widehat{g}(a, b) = \widetilde{g}(a, b)$ from $g(a, b)$ using the hypothesized simulator for functional privacy, then simulating Alice's (or Bob's) view from $\widehat{g}(a, b)$ and a (or b) using the hypothesized *strict* simulator.

In the other direction, suppose there is a simulator in the *liberal* definition. Let τ be a transcript of Alice's view except for input a. Define $\widetilde{g} = \widehat{g}.\tau$ to be $\widehat{g}$ with τ encoded into its low-order bits. We assume that this kind of encoding into approximations can be accomplished without significantly affecting the goodness of approximation; in fact, we will assume that the value represented does not change at all, even if the "approximate" value is zero—that is, τ is auxiliary data rather than an actual part of the value of $\widetilde{g}$. Furthermore, since τ is polylogarithmically bounded in the input size, the communication overhead of $\widetilde{g}$ over $\widehat{g}$ is at most the size of τ, since a protocol for $\widehat{g}$ also serves as a protocol for $\widetilde{g}$. It is trivial to simulate the protocol messages given a and $\widetilde{g}$. Use the hypothesized simulator in the *liberal* definition to show functional privacy of $\widetilde{g}$. □

In Section 4, we apply the technique above of encoding the transcript into the low-order bits to the NORM_ESTIMATION protocol from [15], originally presented in the *liberal* definition, to achieve a more secure version abiding by the *strict* definition. Furthermore, our Heavy Hitters result in Section 5 is formally proven in the *strict* sense using the same idea.

4 Private Euclidean Norm estimation

We consider the setting in which Alice and Bob hold integer-valued vectors a and b respectively, each of dimension N. In [15], the authors provided a protocol for privately approximating the Euclidean norm of the vector difference $\|c\| = \|a - b\|$ as well as the similar vector sum. Before we present our enhancements, it is instructive to review the inner workings of their protocol and its guarantees, given in Protocol 1 and Proposition 16 respectively.

NORM_ESTIMATION

Inputs: N-dimensional vectors a and b with integer values in the range $[-M, M]$.
Output: An $\langle\epsilon, \delta\rangle$-approximation of $\|c\|^2$, where $c = a - b$.

1. Alice and Bob exchange a seed of a pseudorandom generator G and generate a pseudorandom orthonormal matrix A.
2. Set $T = T_{\max} = NM^2$
3. Repeat ({Assertion: $\|c\|^2 \leq T$})
   (a) $\forall j \in [l]$, a secure circuit with ROM (with lookup tables on Aa and Ab) independently generates random coordinates i_j, computes $(Ax)^2_{i_j}$, and independently generates z_j from a Bernoulli$(N(Ax)^2_{i_j}/(TB))$ distribution.
   (b) $T = T/2$
4. Until $\sum_i z_i \geq l/(4B)$ or $T < 1$
5. Output $E = (2TB)/l \cdot \sum_i z_i$ as an estimate of $\|c\|^2$.

Protocol 1: Private approximation protocol of the square l_2 difference [15].

Proposition 16 *(Private l_2 approximation [15]) Suppose Alice and Bob have integer-valued vectors a and b in $[-M, M]^N$ and let $c = a - b$. Fix distortion ϵ and security parameter k. There is a protocol* NORM_ESTIMATION *that computes an approximation $\|c\|_\sim$ to the Euclidean norm of the vector difference, $\|c\|_2$, such that it (a) outputs $\frac{1}{1+\epsilon}\|c\|_2 \leq \|c\|_\sim \leq \|c\|_2$; (b) requires* poly$(k \log(M)N/\epsilon)$

local computation, $\text{poly}(k \log(M) \log(N)/\epsilon)$ *communication, and* $O(1)$ *rounds; and (c) is a private approximation protocol for* $\|c\|_2$ *in the* liberal *sense.*

Furthermore, the protocol's only access to a *and* b *is through the matrix-vector products* Aa *and* Ab*, where* A *is a pseudorandom matrix known to both players. The access is possible through evaluating a circuit with ROM securely; i.e. a circuit with* lookup gates *on inputs* Aa *and* Ab *(see Proposition 9).*

Observe that although the *communication* complexity of this protocol is low, the *computational* complexity of their protocol is quadratic in the vector dimension N. The protocol multiplies the matrix A, which has $\Theta(N^2)$ degrees of freedom by the input vectors a and b, thus requiring $\Omega(N^2)$ computations. Before we present our enhancements to Protocol 1, we first sketch the intuition behind its construction, correctness and privacy guarantees.

In [20], the authors have shown that picking a random $N \times N$ orthonormal matrix A from a distribution defined by the Haar measure ensures that each component of Ax, for any vector x, is tightly concentrated around its root mean square, $\|x\|/\sqrt{N}$. Formally, there exists a $c > 0$ such that

$$\Pr\left[|(Ax)_i| \geq t\|x\|/\sqrt{N}\right] \leq e^{-ct^2} \tag{1}$$

holds for any $i = 1, \ldots, N$, any $t > 1$ and any $x \in \mathbb{R}^N$. This transformation ensures that the "mass" of vector x is uniformly spread among the N coordinates while preserving the vector norm, i.e. $\|x\| = \|Ax\|$. Protocol 1 uses this fact and constructs A using pseudorandom generators instead, guaranteeing nonetheless that $(1 - 2^{-\Theta(k)})\|x\|^2 \leq \|Ax\|^2 \leq \|x\|^2$ holds except with $\text{neg}(k, N)$ probability. Note that with each component tightly concentrated around the root mean square, one can construct an unbiased sample estimator which is an $\langle \epsilon, \delta \rangle$-approximation by straightforward application of Chernoff bounds. However, to achieve privacy, the protocol must sample the coordinates $(Ax)_i$ *obliviously* as to prevent either party from learning the sampled values (it does so by using a secure circuit with ROM; see Section 2.2). Furthermore, the protocol ensures that the final estimate E depends only on $\|x\|$ by using Bernoulli trials to squash the higher moments of E, thus preventing non-simulatable information from leaking. In particular, this also achieves *Functional Privacy* as needed in Definition 14. For its correctness argument, the protocol guarantees that each z_j has enough information to approximate $l_2(x)$ tightly by scaling the Bernoulli trials by a loop variable T and exiting the loop when the sum of the trials is large enough for tight estimation. We refer the reader to [15] for complete analysis of Protocol 1.

4.1 Faster Approximation

As argued in the last Section, the computation bottleneck of Protocol 1 is the multiplication of the pseudorandom matrix A by the input vectors a and b. Computing Aa (and Ab) requires $\Omega(N^2)$ due to the $\Theta(N^2)$ degrees of freedom of matrix A. We recall that this multiplication step is crucial for both the *correctness* and *privacy* guarantees. The matrix transformation ensures that the "mass" of the vector is uniformly spread among all coordinates while preserving the norm. Such process allows a circuit to sample logarithmic many coordinates for a tight estimation of the same norm. Preserving the norm ensures that the Bernouilli trials can be simulated for the privacy proof.

We perform a similar but faster matrix transformation on Alice and Bob's input vectors. The transformation also spreads the "mass" of the vector uniformly and preserves the vector norms as required by the correctness and privacy arguments of Protocol 1. However, our matrix multiplication takes only $O(N \log N)$ time as opposed to $\Omega(N^2)$.

Our approach, based on the technique of Ailon and Chazelle [1], is to randomly choose from a "sufficiently random" family of easily computable orthonormal transformations, as follows. Given a vector $x = (x_1, x_2, \ldots, x_N)$, we flip the sign of each x_i independently with probability 1/2 and then apply a Hadamard transform to it, yielding a new vector x^*. Thus we choose uniformly from a family of 2^N linear transformations, each corresponding to a choice of which variables to sign-flip. Since the Hadamard transform is orthonormal and can be computed in $O(N \log N)$ time, it follows that each transformation in our family is orthonormal and computable in $O(N \log N)$ time, as flipping the sign of a variable is an orthonormal transformation with trivial computational overhead.

Next, we observe that each x_j^*, viewed in isolation, is a random linear combination of signed and unsigned x_i's, scaled by $1/\sqrt{N}$. We prove that each x_i^* is not larger than the root mean square of x, or $\|x\|/\sqrt{N}$, with high probability. Thus, we achieve a similar bound for each coordinate $(Ax)_i$ as in equation Eq. (1), which suffices for the *correctness* and *privacy* proofs of the original protocol.

The following lemma summarizes the above discussion and claims.

Lemma 1. *Let x and x' be vectors of dimension N, with each x'_i being the result of flipping the signal of the corresponding x_i with probability 1/2. Then, for any $\lambda > 0$, applying a Hadamard transform to vector x', yielding $x^* = \frac{1}{\sqrt{N}} H_N x'$, where H_N is the $N \times N$ Hadamard matrix, we have that*

$$\Pr\left[|x_i^*| \geq \lambda \frac{|x|}{\sqrt{N}}\right] \leq 2e^{-\lambda^2/2}. \tag{2}$$

Proof. We analyze the case for a particular x_j^*, for $j \in [1, N]$. Let $Z_1, \ldots, Z_N$ be independent variables such that $Z_i = (\zeta_i x_i)/\sqrt{N}$, where $\zeta_i \in_R \{+1, -1\}$. Here, $\in_R$ denotes drawing each ζ_i independently and uniformly at random. Note that $E[Z_i] = 0$. Now, let $S = \sum_i^N Z_i$. We then define a martingale sequence $X_0, X_1, \ldots, X_N$ by setting $X_0 = E[S]$ and, for $i \in [1, N]$, $X_i = E[S|Z_1, \ldots, Z_i]$. We now apply Azuma's inequality as follows. Recall that for a martingale sequence $X_0, X_1, \ldots, X_N$ s.t. $|X_k - X_{k-1}| \leq c_k$, $\Pr[|X_t - X_0| \geq \lambda] \leq 2\exp\left(-\frac{\lambda^2}{2\sum_{k=1}^t c_k^2}\right)$ for any $t \geq 0$ and any $\lambda > 0$. For our martingale difference sequence let $c_k = |X_k - X_{k-1}|$ and thus we get

$$\Pr\left[|X_N - X_0| \geq \lambda \frac{|x|}{\sqrt{N}}\right] \leq 2\exp\left(-\frac{\lambda^2}{2}\frac{|x|^2}{N}\frac{1}{\sum_{k=1}^N c_k^2}\right). \tag{3}$$

Thus, to prove Eq. (2) it suffices to show that $\sum_{k=1}^{N} c_k^2 \leq |x|^2/N$. Note that $X_k - X_{k-1} = Z_k = (\zeta_k x_k)/\sqrt{N}$, and thus $\sum_{k=1}^{N} c_k^2 = \sum_{k=1}^{N}(X_k - X_{k-1})^2 = \sum_{k=1}^{N} \left(\frac{\zeta_k x_k}{\sqrt{N}}\right)^2 = \frac{|x|^2}{N}$. Therefore, applying it to Eq. (3) guarantees Eq. (2). □

4.2 More Secure Approximation

In [15], the authors have shown that Protocol 1 is secure in the *liberal* sense. They provided the NORM_ESTIMATION simulator that guarantees both functional privacy and private computation of protocol $\widehat{\pi}$ (Protocol 1) computing an approximation $\widehat{g}$ of $g = \|x\|^2$. Their simulator receives the *exact* output $\|x\|^2$ for generating the protocol transcripts. To be secure in the *strict* sense, besides showing functional privacy, one must provide a simulator that is able to produce computationally indistingishable views from Alice's and Bob's *without access* to the exact output, but *only* to the approximation output $\widehat{g}$ (see Definition 13). The original NORM_ESTIMATION simulator from [15] is shown next.

NORM_ESTIMATION simulator

Input: $\|x\|^2$
Output: a computationally indistinguishable distribution from Protocol 1

1. Generate a random seed of G
2. Set $T = T_{\max} = nM^2$
3. Repeat:
   (a) $\forall j \in [l]$, independently generate z_j from a Bernoulli($\|x\|^2/(TB)$) distribution
   (b) $T = T/2$
4. Until $\sum_i z_i \geq l/(4B)$ or $T < 1$
5. Output $E = (2TB)/l \cdot \sum_i z_i$

Simulator 1: The NORM_ESTIMATION simulator from [15].

Simulator 1 above guarantees that the probabilities of the Bernoulli trials from the real and simulated views differ only by neg(k, N). Thus, given access to the exact and approximate outputs, $g(x)$ and $\widehat{g}(x)$ respectively, all messages exchanged —the seed, the oblivious transfer (OT) invocations by the secure circuit, and the output—are simulatable. Specifically, the final value of T is also simulatable. Note that simulating the final value of T is crucial to the privacy argument. If the number of invocations made by the secure circuit differ between the real and simulated views, the distribution on the resulting transcripts will no longer be indistinguishable. Furthermore, observe that the *exact* output $g(x)$ is necessary for simulating such number of steps since its magnitude dictates the loop exit condition. Clearly, using $\widehat{g}(x) = (1 \pm \epsilon)\|x\|^2$ to replace $g(x) = \|x\|^2$ in the Bernoulli trials would make the probabilities differ by a factor in the order of $O(\epsilon)$, a non-negligible factor in our security setting; i.e. we expect $O(2^{-\Theta(k)})$.

Nonetheless, we show how can we transform this *liberal* protocol into a *strict* one, by using the general technique outlined in Section 3. We define a new approximation function $\widetilde{g}$ based on $\widehat{g}$. Let $\widehat{\tau}$ denote the transcript of protocol $\widehat{\pi}$ and let $\widetilde{g} = \widehat{g}.\widehat{\tau}$, meaning that the output of the new approximation function is the output of the original approximation function $\widehat{g}$ concatenated to the entire transcript of protocol $\widehat{\pi}$ (one can view $\widehat{\tau}$ as encoded into the low-order bits of the approximation—in which case we assume the goodness of approximation is not substantially changed—or, alternatively, as auxiliary data and not part of the output itself). The transcript $\widehat{\tau}$ in this case is just a concatenation of the seed used for the pseudorandom generator, all OT invocations by the secure circuit, and the final approximate output. Thus, the communication costs at most doubled; and thus still remain asymptotically $\mathrm{poly}(k \log(M) \log(N)/\epsilon)$.

Let the new protocol $\widetilde{\pi}$ computing $\widetilde{g}$ be identical to $\widehat{\pi}$ with the additional output of $\widehat{\tau}$ along with $\widehat{g}$. It remains to show that $\widetilde{\pi}$ can be privately computed. We thus create Simulator 2, which clearly generates indistinguishable views for Alice and Bob, since all messages exchanged in protocol $\widetilde{\pi}$ are simulated properly: the random seed messages, a matching number of OT calls as well as the final output. Finally, it is clear that $\widetilde{g}$ is *functionally private* to g since one can use the NORM_ESTIMATION simulator to output $\widehat{\tau}$ along with $\widehat{g}(x)$ *given only* $g(x) = \|x\|$.

$\widetilde{\pi}$ simulator

Input: $\widetilde{g}(x) = \widehat{g}(x).\widehat{\tau}$
Output: a computationally indistinguishable transcript from $\widetilde{\pi}(x)$

1. Extract $\widehat{g}(x)$ and $\widehat{\tau}$ from the input $\widetilde{g}(x) = \widehat{g}(x).\widehat{\tau}$
2. Extract the random seed for the pseudorandom generator from $\widehat{\tau}$ and send it to the other party.
3. Simulate the OT calls from Step 3 in Protocol 1 by playing back the messages exchanged in $\widehat{\tau}$.
4. Output $\widetilde{g}(x)$

Simulator 2: Simulator for $\widetilde{\pi}$.

5 Private Euclidean Heavy Hitters

Consider the same input setting from the previous Section. Here, both parties want to learn a representation $\widetilde{c} = \sum_{t \in T_{\mathrm{out}}} t$ such that $\|c - \widetilde{c}\|_2^2 \leq (1 + \epsilon)\|c - c_{\mathrm{opt}}\|_2^2$ and such that at most c_{opt} and $\|c\|_2$ is revealed. Unless otherwise stated, we consider the private Euclidean Heavy Hitters problem as simply the private Heavy Hitters problem. A protocol is given in Figure 2.

5.1 Analysis

First, to gain intuition, we consider some easy special cases of the protocol's operation. For our analysis, assume that the terms in c are already positive and in decreasing order, $c_0 > c_1 > \cdots > c_{N-1} > 0$. We will be able to find the coefficient value of any desired term, so we focus on the set of indices. Let $I_{\text{opt}} = \{0, 1, 2, \ldots, B-1\}$ denote the set of indices for the optimal B terms. The set I of indices is defined in Figure 2. Thus $Q_{c,B,\theta} \subseteq Q_{c,B,\frac{\theta}{1+\epsilon}} \subseteq I_{\text{opt}}$ and $Q_{c,B,\frac{\theta}{1+\epsilon}} \subseteq I$.

The ideal output is I_{opt}, though any superset of $Q_{c,B,\theta}$ suffices to get an approximation with error at most $(1+\epsilon)$ times optimal. This includes the set $I \supseteq Q_{c,B,\theta}$ that the non-private algorithm has recovered. The set I_B of the largest B terms indexed by I contains $Q_{c,B,\theta}$, so I_B is a set of at most B terms with error at most $(1+\epsilon)$ times optimal. If $|Q_{c,B,\theta}| = B$, then $I_B = Q_{c,B,\theta} = I_{\text{opt}}$, and I_B is a private and correct output.

PRIVATE_HEAVY_HITTERS

- *Known parameters*: N, M, B, ϵ, k, which determine $\theta = \frac{\epsilon}{B(1+\epsilon)}$ and B'.
- *Inputs*: N-dimensional vectors a and b with integer values in the range $[-M, M]$.
- *Output*: With probability at least $1 - 2^{-k}$, a set T_{out} of at most B terms, such that $\left\| c - \sum_{t \in T_{\text{out}}} t \right\|_2^2 \leq (1+\epsilon) \left\| c - \sum_{t \in T_{\text{opt}}} t \right\|_2^2$.

---

1. Exchange pseudorandom seeds (in the clear). Generate measurement matrices R_1 and R_2. Alice locally constructs sketches $R_1 a$ and $R_2 a = (R_2^0 a, R_2^1 a, \ldots R_2^{B-1} a)$, where the matrix R_1 is used for a non-private Euclidean Heavy Hitters and the matrix $R_2 = (R_2^0, R_2^1, \ldots, R_2^{B-1})$ is used for B independent repetitions of NORM_ESTIMATION. Bob similarly constructs $R_1 b$ and $R_2 b$.
2. Using general-purpose SMC, do
   - Use an existing (non-private) Euclidean Heavy Hitters protocol to get, from $R_1 a$ and $R_1 b$, a secret-sharing of a superset I of $Q_{c,B,\frac{\theta}{1+\epsilon}}$, in which I has exactly $B' \leq \text{poly}(\log(N), \log(M), B, k, 1/\epsilon)$ indices. (Pad, if necessary.)
3. Use PRIVATE-SAMPLE-SUM to compute, from I, a, and b, secret-shared values for each index in I. Let T denote the corresponding set of secret-shared terms. (Both the index and value of each term in T is secret shared.) Enumerate I as $I = \{i_0, i_1, \ldots\}$ with $t_{i_0} > t_{i_1} > \cdots$.
4. Using SMC, do
   - for $j = 0$ to $B-1$
     (a) From $R_2^j, R_2^j a, R_2^j b, t_0, t_1, \ldots, t_{i_{j-1}}$, sketch $r_j = c - (t_{i_0} + t_{i_1} + \cdots + t_{i_{j-1}})$ as $R_2^j r_j = (R_2^j a + R_2^j b - R_2^j (t_{i_0} + t_{i_1} + \cdots + t_{i_{j-1}}))$.
     (b) use NORM_ESTIMATION to estimate $\|r_j\|_2^2$ as $\|r_j\|_\sim^2$, satisfying $\frac{1}{1+\epsilon}\|r_j\|_2^2 \leq \|r_j\|_\sim^2 \leq \|r_j\|_2^2$.
     (c) If $|c_{i_j}|^2 < \theta \|r_j\|_\sim^2$, break (out of for-loop)
     (d) Output t_j
5. Encode the pseudorandom seeds for R_1 and R_2 into the low-order bits of the output or (as we assume here) provide R_1 and R_2 as auxiliary output.

Protocol 2: Protocol for the Euclidean Heavy Hitters problem.

The difficulty arises when $|Q_{c,B,\theta}| < B$, in which case some of I_B may be arbitrary and should not be allowed to leak. So the algorithm needs to find a private subset I_{out} with $Q_{c,B,\theta} \subseteq I_{\text{out}} \subseteq I_B$. The challenge is subtle. Let s denote $|Q_{c,B,\theta}|$. If the algorithm knew s, the algorithm could easily output $Q_{c,B,\theta}$, which is the indices of the top s terms, a correct and private output. Unfortunately, determining $Q_{c,B,\theta}$ or $s = |Q_{c,B,\theta}|$ requires $\Omega(N)$ communication (see Section 5.2), so we cannot hope to find $Q_{c,B,\theta}$ exactly. Non-private norm estimation can be used to find a subset I_{out} with $Q_{c,B,\theta} \subseteq I_{\text{out}} \subseteq Q_{c,B,\frac{\theta}{1+\epsilon}} \subseteq I_{\text{opt}}$, which is correct, but not quite private. Given $|I_{\text{out}}|$, the contents of $I_{\text{out}} \subseteq I_{\text{opt}}$ are indeed private, but the *size* of I_{out} is, generally, non-private. Fortunately, if we use a *private* protocol for norm estimation, $|I_{\text{out}}|$ remains private. We now proceed to a formal analysis.

Theorem 17. *Protocol* PRIVATE_HEAVY_HITTERS *requires* $\text{poly}(N, \log(M), B, k, 1/\epsilon)$ *local computation,* $\text{poly}(\log(N), \log(M), B, k, 1/\epsilon)$ *communication, and* $O(1)$ *rounds.*

Proof. By existing work, all costs of Steps 1 to 3 are as claimed. Now consider Step 4. Observe that the function being computed there has inputs and outputs of size bounded by $\text{poly}(\log(N), \log(M), B, k, 1/\epsilon)$ and takes time polynomial in the size of its inputs. In particular, the instances of NORM_ESTIMATION do *not* start from scratch with respect to a or b; rather, they pick up from the precomputed short sketches $R_2 a$ and $R_2 b$. It follows that this function can be wrapped with SMC, preserving the computation and communication up to polynomial blowup in the size of the input and keeping the round complexity to $O(1)$. □

We now turn to correctness and privacy. Let I_{out} denote the set of indices corresponding to the set T_{out} of output terms.

Theorem 18. *Protocol* PRIVATE_HEAVY_HITTERS *is correct.*

Proof. The correctness of Steps 2 and 3 follows from previous work. In Step 4, we first show that $Q_{B,\frac{\epsilon}{B(1+\epsilon)}} \subseteq I_{\text{out}}$. We assume that $\frac{1}{1+\epsilon}\|r_j\|_2^2 \leq \|r_j\|_\sim^2 \leq \|r_j\|_2^2$ always holds; by Proposition 16, this happens with high probability. Thus, if $|c_{i_j}|^2 \geq \frac{\epsilon}{B(1+\epsilon)}\|r_j\|_2^2$, then $|c_{i_j}|^2 \geq \frac{\epsilon}{B(1+\epsilon)}\|r_j\|_2^2 \geq \frac{\epsilon}{B(1+\epsilon)}\|r_i\|_\sim^2$. By construction, $Q_{B,\frac{\epsilon}{B(1+\epsilon)}} \subseteq I$. A straightforward induction shows that, if $j \in Q_{B,\frac{\epsilon}{B(1+\epsilon)}}$, then iteration j outputs t_{i_j} and the previous iterations output exactly the set of the j larger terms in I. By Proposition 4, since I_{out} is a superset of $Q_{B,\frac{\epsilon}{B(1+\epsilon)}}$, if $\widetilde{c} = \sum_{j \in I_{\text{out}}} c_{i_j}\delta_{i_j}$, then $\|\widetilde{c} - c\|_2^2 \leq (1+\epsilon)\|c_{\text{opt}} - c\|_2^2$, as desired. □

Before giving the complete privacy argument, we give a lemma, similar to the above. Suppose a set P of indices is a subset of another set Q of indices. We will say that P is a *prefix* of Q if $i \in P, t_j > t_i$, and $j \in Q$ imply $j \in P$.

Lemma 2. *Output set* I_{out} *is a prefix of* $Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$ *except with probability* 2^{-k}.

Proof. Note that $Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$ is a subset of I and $Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$ is a prefix of the universe, so $Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$ is a prefix of I. The set I_{out} is also a prefix of I. Thus, of the sets I_{out} and $Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$, one is a prefix of the other (or they are equal).

So suppose, toward a contradiction, that $Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$ is a proper prefix of I_{out}. Let $q = \left|Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}\right|$, so q is the least number such that i_q is *not* in $Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$. If the protocol halts before considering q, then $I_{\text{out}} \subseteq Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$, a contradiction. So we may assume that $q < B$ (so the for-loop doesn't terminate). Then, by definition of $Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$, we have $|c_{i_q}|^2 < \frac{\epsilon}{B(1+\epsilon)^2}\sum_{j\geq q}|c_{i_j}|^2$. It follows that $|c_{i_q}|^2 < \frac{\epsilon}{B(1+\epsilon)^2}\sum_{i\geq q}|c_i|^2 = \frac{\epsilon}{B(1+\epsilon)^2}\|r_q\|_2^2 \leq \frac{\epsilon}{B(1+\epsilon)}\|r_q\|_{\sim}^2$. Thus the protocol halts without outputting t_q, after outputting exactly $Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$. □

Finally, Theorem 19 ensures privacy and Theorem 20 summarizes our results.

Theorem 19. *Protocol* PRIVATE_HEAVY_HITTERS *leaks* only $\|c\|_2^2$ *and* c_{opt}.

Proof. With the random inputs R_1 and R_2 encoded into the output, it is straightforward to show that Protocol PRIVATE_HEAVY_HITTERS is a private protocol in the traditional sense that the protocol messages leak no more than the inputs and outputs. This is done by composing simulators for PRIVATE-SAMPLE-SUM and SMC. It remains only to show only that we can simulate the joint distribution on $(\widetilde{c}, R_1, R_2)$ given as simulator-input c_{opt} and $\|c\|$. We will show that R_1 is indistinguishable from independent of the joint distribution of $(\widetilde{c}, R_2)$, which we will simulate directly.

First, we show that R_1 is independent. Except with probability $2^{-\Omega(k)}$, the intermediate set I is a superset of $Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$ and the norm estimation is correct. In that case, the protocol outputs a prefix of $Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$ and we get identical output if I is replaced by $Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$. Also, $Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$ can be constructed from c_{opt} and $\|c\|_2$. Since the protocol proceeds without further reference to R_1, we have shown that the pair $(\widetilde{c}, R_2)$ is indistinguishable from being independent of R_1. It remains only to simulate $(\widetilde{c}, R_2)$.

Note that the output $\widetilde{c}$ does depend non-negligibly on R_2. If $|c_{i_j}|^2$ is very close to $\theta\|r_j\|_2^2$, then the test $|c_{i_j}|^2 < \theta\|r_j\|_{\sim}^2$ in the protocol may succeed with probability non-negligibly far from 0 and from 1, depending on R_2, since the distortion guarantee on $\|r_j\|_{\sim}^2$ is only the factor $(1 \pm \epsilon)$.

The simulator is as follows. Assume that the terms in c_{opt} are $t_0, t_1, \ldots, t_{B-1}$ with decreasing order, $t_0 > t_1 > \cdots > t_{B-1}$. For each $j \leq B$, compute $E_j = \|c - (t_0 + t_1 + \cdots + t_{j-1})\|_2^2 = \|c\|_2^2 - \|t_0 + t_1 + \cdots + t_{j-1}\|_2^2$ and then run the NORM_ESTIMATION simulator on input E_j and ϵ to get a sample from the joint distribution $(\widetilde{E}_j, \overline{R}_2)$, where $\widetilde{E}_j$ is a good estimate to E_j. Our simulator then outputs t_{i_j} if $|c_{i_j}|^2 \geq \frac{\epsilon}{B(1+\epsilon)}\widetilde{E}_j$, and halts, otherwise, following the final for-loop of the protocol. Call the output of the simulator $\widetilde{s} = \sum_j t_{i_j}\delta_{i_j}$.

Again using the fact that a prefix of $Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$ is output, if $j \in Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$, then $i_j = j$; *i.e.*, the j^{th} largest output term is the j^{th} largest overall, so that, if j is output, $E_j = \|r_j\|_2^2$. Thus $(\widetilde{E}_j, \overline{R}_2)$ is distributed indistinguishably from $(\|r_j\|_{\sim}^2, R_2)$. The protocol finishes deterministically using I and $\|r_j\|_{\sim}^2$ and the simulator finishes deterministically using $Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$ and $\widetilde{E}_j$, but, since the protocol output is identical if I is replaced by $Q_{B,\frac{\epsilon}{B(1+\epsilon)^2}}$, the distributions on output $(\widetilde{c}, R_2)$ of the protocol and $(\widetilde{s}, \overline{R}_2)$ of the simulator are indistinguishable. □

Theorem 20. *Suppose Alice and Bob hold integer-valued vectors a and b in $[-M, M]^N$, respectively. Let B, k and ϵ be user-defined parameters. Let $c = a + b$. Let T_{opt} be the set of the largest B terms in c. There is a protocol, taking a, b, B k and ϵ as input, that computes a representation $\widetilde{c}$ of at most B terms such that it: (a) outputs $\widetilde{c}$ with $\|\widetilde{c} - c\|_2 \leq (1 + \epsilon)\|c_{\mathrm{opt}} - c\|_2$; (b) uses* $\mathrm{poly}(N, \log(M), B, k, 1/\epsilon)$ *time,* $\mathrm{poly}(\log(N), \log(M), B, k, 1/\epsilon)$ *communication, and $O(1)$ rounds; and (c) succeeds with probability $1 - 2^{-k}$ and leaks only c_{opt} and $\|c\|_2$ on security parameter k.*

Corollary 21 *With the same hyptotheses and resource bounds, there is a protocol that computes $\widetilde{c}$ and an approximation $\|\widetilde{c} - c\|_\sim$ to $\|\widetilde{c} - c\|_2$ such that $\frac{1}{1+\epsilon}\|\widetilde{c} - c\|_2 \leq \|\widetilde{c} - c\|_\sim \leq \|\widetilde{c} - c\|_2$ and the protocol leaks only c_{opt} and $\|\widetilde{c} - c\|_2$.*

Proof. Run the main protocol and output also $\|\widetilde{c} - c\|_\sim$, computed in the course of the main protocol. Note that $\|\widetilde{c} - c\|_2^2 = \|c\|_2^2 - \|\widetilde{c}\|_2^2$ and both $\|c\|_2$ and $\widetilde{c}$ are available to the main simulator (as input and output, resp.), so we can modify the main simulator to compute $\|\widetilde{c} - c\|_2^2$ as well. □

5.2 Lower Bounds

In this Section, we state some lower bounds for problems related to our main problem in this Section, such as computing an approximation to c_{opt} without leaking $\|c\|_2$. The results are straightforward, but we include Theorem 22 to motivate the approximation and Theorem 23 to motivate leakage of the Euclidean norm in protocols we present. The proofs, based on the set disjointness problem, will appear in the journal version of this article.

Theorem 22. *There is an infinite family of settings of parameters M, N, B, k such that any protocol that computes the Euclidean norm exactly on the sum c of individually-held inputs a and b, uses communication $\Omega(N)$. Similarly, any protocol that computes the exact Heavy Hitters or computes the qualified set $Q_{c,1,1}$ exactly uses communication $\Omega(N)$.*

Theorem 23. *There is an infinite family of settings of parameters M, N, B, k, ϵ such that any protocol that solves the Euclidean Heavy Hitters problem on the sum c of individually-held inputs a and b, leaking only c_{opt}, uses communication $\Omega(N)$. Furthermore, for an infinite class of inputs in which $\|c\|_2$ is not constant, any such protocol either computes $\|c\|_2$ or uses communication $\Omega(N)$.*

References

1. Nir Ailon and Bernard Chazelle. Approximate nearest neighbors and the fast Johnson-Lindenstrauss transform. In *Proc. 38th Annual ACM STOC*, pages 557–563, 2006.

2. Gagan Aggarwal, Nina Mishra, and Benny Pinkas. Secure computation of the k th-ranked element. In *EUROCRYPT04*, pages 40–55, 2004.
3. N. Alon, P. B. Gibbons, Y. Matias, and M. Szegedy. Tracking join and self-join sizes in limited storage. *J. Comput. Syst. Sci.*, 64(3):719–747, 2002.
4. N. Alon, Y. Matias, and M. Szegedy. The space complexity of approximating the frequency moments. *J. Comput. Syst. Sci.*, 58(1):137–147, 1999.
5. A. Beimel, P. Carmi, K. Nissim, and E. Weinreb. Private approximation of search problems. In *Proc. 38th Annual ACM STOC*, pages 119–128, 2006.
6. M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *Proc. 20th Annual ACM STOC*, pages 1–10. ACM Press, 1988.
7. C. Cachin, S. Micali, and M. Stadler. Computationally private information retrieval with polylogarithmic communication. In *EUROCRYPT'99*, LNCS 1592, pages 404–414. Springer-Verlag, 1999.
8. S. Chawla, C. Dwork, F. McSherry, A. Smith, and H. Wee. Toward privacy in public databases. In *Proc. Second TCC*, pages 363–385, 2005.
9. B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan. Private information retrieval. *Journal of the ACM*, 45:965–981, 1998.
10. G. Cormode and S. Muthukrishnan. What's hot and what's not: Tracking most frequent items dynamically. In *Proc. ACM PODS*, pages 296–306, 2003.
11. J. Feigenbaum, Y. Ishai, T. Malkin, K. Nissim, M. Strauss, and R. N. Wright. Secure multiparty computation of approximations. *Transactions on Algorithms*, 2006. An extended abstract appeared in ICALP 2001.
12. M. Freedman, K. Nissim, and B. Pinkas. Efficient private matching and set intersection. In *EUROCRYPT'04*, LNCS 3027, pages 1–19. Springer-Verlag, 2004.
13. A. Gilbert, S. Guha, P. Indyk, Y. Kotidis, S. Muthukrishnan, and M. Strauss. Fast, small-space algorithms for approximate histogram maintenance. In *Proc. 34th Annual ACM STOC*, pages 389–398, 2002.
14. S. Halevi, E. Kushilevitz, R. Krauthgamer, and K. Nissim. Private approximations of NP-hard functions. In *Proc. 33th Annual ACM STOC*, pages 550–559, 2001.
15. P. Indyk and D. P. Woodruff. Polylogarithmic private approximations and efficient matching. In *Proc. Third Theory of Cryptography Conference*, pages 245–264, 2006.
16. E. Kushilevitz and Y. Mansour. Learning decision trees using the fourier sprectrum. In *Proc. 23th Annual ACM STOC*, pages 455–464, 1991.
17. E. Kushilevitz and N. Nisan. *Communication complexity*. Cambridge University Press, 1997.
18. E. Kushilevitz and R. Ostrovsky. Replication is NOT needed: SINGLE database, computationally-private information retrieval. In *Proc. 38th IEEE FOCS*, pages 364–373, 1997.
19. Y. Lindell and B. Pinkas. Privacy preserving data mining. *J. Cryptology*, 15(3):177–206, 2002.
20. V. D. Milman and G. Schechtman, Asymptotic Theory of Finite Dimensional Normed Spaces. *Lecture Notes in Mathematics*, Volume 1200, 1986.
21. M. Naor and K. Nissim. Communication preserving protocols for secure function evaluation. In *Proc. 33th Annual ACM STOC*, pages 590–599, 2001.
22. A. Yao. Protocols for secure computation. In *Proc. 23rd IEEE FOCS*, pages 160–164, 1982.