Distributed Learning with Sublinear Communication

Jayadev Acharya [1] **Christopher De Sa** [1] **Dylan J. Foster** [2] **Karthik Sridharan** [1]

Abstract

In distributed statistical learning, N samples are split across m machines and a learner wishes to use minimal communication to learn as well as if the examples were on a single machine. This model has received substantial interest in machine learning due to its scalability and potential for parallel speedup. However, in high-dimensional settings, where the number examples is smaller than the number of features ("dimension"), the speedup afforded by distributed learning may be overshadowed by the cost of communicating a single example. This paper investigates the following question: When is it possible to learn a d-dimensional model in the distributed setting with total communication sublinear in d? Starting with a negative result, we observe that for learning ℓ_1-bounded or sparse linear models, no algorithm can obtain optimal error until communication is linear in dimension. Our main result is that by slightly relaxing the standard boundedness assumptions for linear models, we can obtain distributed algorithms that enjoy optimal error with communication *logarithmic* in dimension. This result is based on a family of algorithms that combine mirror descent with randomized sparsification/quantization of iterates, and extends to the general stochastic convex optimization model.

1. Introduction

In statistical learning, a learner receives examples $z_1, \ldots, z_N$ i.i.d. from an unknown distribution $\mathcal{D}$. Their goal is to output a hypothesis $\hat{h} \in \mathcal{H}$ that minimizes the prediction error $L_{\mathcal{D}}(h) := \mathbb{E}_{z\sim\mathcal{D}}\, \ell(h, z)$, and in particular to guarantee that *excess risk* of the learner is small, i.e.

$$L_{\mathcal{D}}(\hat{h}) - \inf_{h\in\mathcal{H}} L_{\mathcal{D}}(h) \leq \varepsilon(\mathcal{H}, N), \qquad (1)$$

where $\varepsilon(\mathcal{H}, N)$ is a decreasing function of N. This paper focuses on *distributed* statistical learning. We consider a distributed setting, where the N examples are split evenly across m machines, with $n := N/m$ examples per machine, and the learner wishes to achieve an excess risk guarantee such as (1) with minimal overhead in computation or communication.

Distributed learning has been the subject of extensive investigation due to its scalability for processing massive data: We may wish to efficiently process datasets that are spread across multiple data-centers, or we may want to distribute data across multiple machines to allow for parallelization of learning procedures. The question of parallelizing computation via distributed learning is a well-explored problem (Bekkerman et al., 2011; Recht et al., 2011; Dekel et al., 2012; Chaturapruek et al., 2015). However, one drawback that limits the practical viability of these approaches is that the communication cost between machines may overshadow gains in parallel speedup (Bijral et al., 2016). Indeed, for high-dimensional statistical inference tasks where N could be much smaller than the dimension d, or in modern deep learning where the number of model parameters exceeds the number of examples (e.g. (He et al., 2016)), communicating a single gradient or sending the raw model parameters between machines constitutes a significant overhead.

Algorithms with reduced communication complexity in distributed learning have received significant recent development (Seide et al., 2014; Alistarh et al., 2017; Zhang et al., 2017; Suresh et al., 2017; Bernstein et al., 2018; Tang et al., 2018), but typical results here take as a given that when gradients or examples live in d dimensions, communication will scale as $\Omega(d)$. Our goal is to revisit this tacit assumption and understand when it can be relaxed. We explore the question of *sublinear communication*:

Suppose a hypothesis class $\mathcal{H}$ has d parameters. When is it possible to achieve optimal excess risk for $\mathcal{H}$ in the distributed setting using $o(d)$ communication?

1.1. Sublinear Communication for Linear Models?

Let us first focus on linear models, which are a special case of the general learning setup (1). We restrict to linear hypotheses of the form $h_w(x) = \langle w, x\rangle$ where $w, x \in \mathbb{R}^d$ and write $\ell(h_w, z) = \phi(\langle w, x\rangle, y)$, where $\phi(\cdot, y)$ is a fixed

[1] Cornell University [2] Massachusetts Institute of Technology. Correspondence to: Dylan Foster <dylanf@mit.edu>.

Proceedings of the 36th International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019. Copyright 2019 by the author(s).

link function and $z = (x, y)$. We overload notation slightly and write

$$L_{\mathcal{D}}(w) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\, \phi(\langle w, x\rangle, y). \tag{2}$$

This formulation captures standard learning tasks such as square loss regression, where $\phi(\langle w, x\rangle, y) = (\langle w, x\rangle - y)^2$, logistic regression, where $\phi(\langle w, x\rangle, y) = \log\left(1 + e^{-y\langle w,x\rangle}\right)$, and classification with surrogate losses such as the hinge loss, where $\phi(\langle w, x\rangle, y) = \max\{1 - \langle w, x\rangle \cdot y, 0\}$.

Our results concern the communication complexity of learning for linear models in the *ℓ_p/ℓ_q-bounded setup*: weights belong to $\mathcal{W}_p := \left\{w \in \mathbb{R}^d \mid \|w\|_p \le B_p\right\}$ and feature vectors belong to $\mathcal{X}_q := \left\{x \in \mathbb{R}^d \mid \|x\|_q \le R_q\right\}$.[1] This setting is a natural starting point to investigate sublinear-communication distributed learning because *learning is possible even when $N \ll d$.*

Consider the case where p and q are dual, i.e. $\frac{1}{p} + \frac{1}{q} = 1$, and where ϕ is 1-Lipschitz. Here it is well known (Zhang, 2002; Kakade et al., 2009) that whenever $q \ge 2$, the optimal sample complexity for learning, which is achieved by choosing the learner's weights $\widehat{w}$ using empirical risk minimization (ERM), is

$$L_{\mathcal{D}}(\widehat{w}) - \inf_{w\in\mathcal{W}_p} L_{\mathcal{D}}(w) = \Theta\left(\sqrt{\frac{B_p^2 R_q^2 C_q}{N}}\right), \tag{3}$$

where $C_q = q - 1$ for finite q and $C_\infty = \log d$, namely

$$L_{\mathcal{D}}(\widehat{w}) - \inf_{w\in\mathcal{W}_1} L_{\mathcal{D}}(w) = \Theta\left(\sqrt{\frac{B_1^2 R_\infty^2 \log d}{N}}\right). \tag{4}$$

We see that when $q < \infty$ the excess risk for the dual ℓ_p/ℓ_q setting is *independent of dimension* so long as the norm bounds B_p and R_q are held constant, and that even in the ℓ_1/ℓ_∞ case there is only a mild logarithmic dependence. Hence, we can get nontrivial excess risk even when the number of examples N is arbitrarily small compared to the dimension d. This raises the intriguing question: *Given that we can obtain nontrivial excess risk when $N \ll d$, can we obtain nontrivial excess risk when* communication *is sublinear in d?*

To be precise, we would like to develop algorithms that achieve (3)/(4) with total bits of communication $\mathrm{poly}(N, m, \log d)$, permitting also $\mathrm{poly}(B_p, R_q)$ dependence. The prospect of such a guarantee is exciting because—in light of the discussion above—as this would imply that we can obtain nontrivial excess risk with fewer bits of total communication than are required to naively send a *single feature vector*.

1.2. Contributions

We provide new communication-efficient distributed learning algorithms and lower bounds for ℓ_p/ℓ_q-bounded linear models, and more broadly, stochastic convex optimization. We make the following observations:

- For ℓ_2/ℓ_2-bounded linear models, sublinear communication *is* achievable, and is obtained by using a derandomized Johnson-Lindenstrauss transform to compress examples and weights.
- For ℓ_1/ℓ_∞-bounded linear models, no distributed algorithm can obtain optimal excess risk until communication is *linear* in dimension.

These observations lead to our main result. We show that by relaxing the ℓ_1/ℓ_∞-boundedness assumption and instead learning ℓ_1/ℓ_q-bounded models for a constant $q < \infty$, one unlocks a plethora of new algorithmic tools for sublinear distributed learning:

1. We give an algorithm with optimal rates matching (3), with total communication $\mathrm{poly}(N, m^q, \log d)$.
2. We extend the sublinear-communication algorithm to give refined guarantees, including instance-dependent *small loss* bounds for smooth losses, *fast rates* for strongly convex losses, and optimal rates for matrix learning problems.

Our main algorithm is a distributed version of mirror descent that uses randomized sparsification of weight vectors to reduce communication. Beyond learning in linear models, the algorithm enjoys guarantees for the more general distributed stochastic convex optimization model.

To elaborate on the fast rates mentioned above, another important case where learning is possible when $N \ll d$ is the sparse high-dimensional linear model setup, central to compressed sensing and statistics. Here, the standard result is that when ϕ is strongly convex and the benchmark class consists of k-sparse linear predictors, i.e. $\mathcal{W}_0 := \left\{w \in \mathbb{R}^d \mid \|w\|_0 \le k\right\}$, one can guarantee

$$L_{\mathcal{D}}(\widehat{w}) - \inf_{w\in\mathcal{W}_0} L_{\mathcal{D}}(w) = \Theta\left(\frac{k \log(d/k)}{N}\right). \tag{5}$$

With ℓ_∞-bounded features, no algorithm can obtain optimal excess risk for this setting until communication is linear in dimension, even under compressed sensing-style assumptions. When features are ℓ_q-bounded however, our general machinery gives optimal fast rates matching (5) under Lasso-style assumptions, with communication $\mathrm{poly}(N^q, \log d)$.

The remainder of the paper is organized as follows. In Section 2 we develop basic upper and lower bounds for the ℓ_2/ℓ_2 and ℓ_1/ℓ_∞-bounded settings. In Section 3 we shift to the ℓ_1/ℓ_q-bounded setting, where we introduce the family of

[1] Recall the definition of the ℓ_p norm: $\|w\|_p = \left(\sum_{i=1}^d |w_i|^p\right)^{1/p}$.

sparsified mirror descent algorithms that leads to our main results and sketch the analysis.

1.3. Related Work

Much of the work in algorithm design for distributed learning and optimization does not explicitly consider the number of bits used in communication per messages, and instead tries to make communication efficient via other means, such as decreasing the communication frequency or making learning robust to network disruptions (Duchi et al., 2012; Zhang et al., 2012). Other work reduces the number of bits of communication, but still requires that this number be linear in the dimension d. One particularly successful line of work in this vein is low-precision training, which represents the numbers used for communication and elsewhere within the algorithm using few bits (Alistarh et al., 2017; Zhang et al., 2017; Seide et al., 2014; Bernstein et al., 2018; Tang et al., 2018; Stich et al., 2018; Alistarh et al., 2018). Although low-precision methods have seen great success and adoption in neural network training and inference, low-precision methods are fundamentally limited to use bits proportional to d; once they go down to one bit per number there is no additional benefit from decreasing the precision. Some work in this space tries to use sparsification to further decrease the communication cost of learning, either on its own or in combination with a low-precision representation for numbers (Alistarh et al., 2017; Wangni et al., 2018; Wang et al., 2018). While the majority of these works apply low-precision and sparsification to gradients, a number of recent works apply sparsification to model parameters (Tang et al., 2018; Stich et al., 2018; Alistarh et al., 2018); We also adopt this approach. The idea of sparsifying weights is not new (Shalev-Shwartz et al., 2010), but our work is the first to provably give communication logarithmic in dimension. To achieve this, our assumptions and analysis are quite a bit different from the results mentioned above, and we crucially use mirror descent, departing from the gradient descent approaches in (Tang et al., 2018; Stich et al., 2018; Alistarh et al., 2018).

Lower bounds on the accuracy of learning procedures with limited memory and communication have been explored in several settings, including mean estimation, sparse regression, learning parities, detecting correlations, and independence testing (Shamir, 2014; Duchi et al., 2014; Garg et al., 2014; Steinhardt & Duchi, 2015; Braverman et al., 2016; Steinhardt et al., 2016; Acharya et al., 2019a;b; Raz, 2018; Han et al., 2018; Sahasranand & Tyagi, 2018; Dagan & Shamir, 2018; Dagan et al., 2019). In particular, the results of (Steinhardt & Duchi, 2015) and (Braverman et al., 2016) imply that optimal algorithms for distributed sparse regression need communication much larger than the sparsity level under various assumptions on the number of machines and the communication protocol.

2. Linear Models: Basic Results

In this section we develop basic upper and lower bounds for communication in ℓ_2/ℓ_2- and ℓ_1/ℓ_∞-bounded linear models. Our goal is to highlight that the communication complexity of distributed learning and the statistical complexity of centralized learning do not in general coincide, and to motivate the ℓ_1/ℓ_q-boundedness assumption under which we derive communication-efficient algorithms in Section 3.

2.1. Preliminaries

We formulate our results in a distributed communication model following Shamir (2014). Recalling that $n = N/m$, the model is as follows.

- For machine $i = 1, \dots, m$:
  - Receive n i.i.d. examples $S_i := z_1^i, \dots, z_n^i$.
  - Compute message $W_i = f_i(S_i\,; W_1, \dots, W_{i-1})$, where W_i is at most b_i bits.
- Return $W = f(W_1, \dots, W_m)$.

We refer to $\sum_{i=1}^m b_i$ as the *total communication*, and we refer to any protocol with $b_i \leq b\ \forall i$ as a (b, n, m) *protocol*. As a special case, this model captures a serial distributed learning setting where machines proceed one after another: Each machine does some computation on their data $z_1^i, \dots, z_n^i$ and previous messages $W_1, \dots, W_{i-1}$, then broadcasts their own message W_i to all subsequent machines, and the final model in (1) is computed from W, either on machine m or on a central server. The model also captures protocols in which each machine independently computes a local estimator and sends it to a central server, which aggregates the local estimators to produce a final estimator (Zhang et al., 2012). All of our upper bounds have the serial structure above, and our lower bounds apply to any (b, n, m) protocol.

2.2. ℓ_2/ℓ_2-Bounded Models

In the ℓ_2/ℓ_2-bounded setting, we can achieve sample optimal learning with sublinear communication by using dimensionality reduction. The idea is to project examples into $k = \tilde{O}(N)$ dimensions using the Johnson-Lindenstrauss transform, then perform a naive distributed implementation of any standard learning algorithm in the projected space. Here we implement the approach using stochastic gradient descent.

To project the examples onto the same subspace, the machines need to agree on a JL transformation matrix. To do so with little communication, we consider the derandomized sparse JL transform of Kane & Nelson (2010), which constructs a collection $\mathcal{A}$ of matrices in $\mathbb{R}^{k \times d}$ with $|\mathcal{A}| = \exp\left(O(\log(k/\delta) \cdot \log d)\right)$ for confidence parameter δ. The first machine randomly selects an entry of $\mathcal{A}$ and sends the identity of this matrix using $O(\log(k/\delta) \cdot \log d)$ bits to

```
Algorithm 1 (Maurey Sparsification).
Input: Weight vector $w \in \mathbb{R}^d$. Sparsity level $s$.
  • Define $p \in \Delta_d$ via $p_i \propto |w_i|$.
  • For $\tau = 1, \ldots, s$:
      – Sample index $i_\tau \sim p$.
  • Return $Q^s(w) := \frac{\|w\|_1}{s} \sum_{\tau=1}^{s} \mathrm{sgn}(w_{i_\tau}) e_{i_\tau}$.
```

the other $m-1$ machines. The dimension k and parameter δ are chosen as a function of N.

Each machine uses the matrix A to project its features down to k dimensions. Letting $x'_t = Ax_t$ denote the projected features, the first machine starts with a k-dimensional weight vector $u_1 = 0$ and performs the online gradient descent update (Zinkevich, 2003; Cesa-Bianchi & Lugosi, 2006) over its n projected samples as:

$$u_t \leftarrow u_{t-1} - \eta \nabla \phi(\langle u_t, x'_t \rangle, y_t),$$

where $\eta > 0$ is the learning rate. Once the first machine has passed over all its samples, it broadcasts the last iterate u_{n+1} as well the average $\sum_{s=1}^{n} u_s$, which takes $\tilde{O}(k)$ communication. The next machine machine performs the same sequence of gradient updates on its own data using u_{n+1} as the initialization, then passes its final iterate and the updated average to the next machine. This repeats until we arrive at the mth machine. The mth machine computes the k-dimensional vector $\widehat{u} := \frac{1}{N} \sum_{t=1}^{N} u_t$, and returns $\widehat{w} = A^\top \hat{u}$ as the solution.

Theorem 1. When ϕ is L-Lipschitz and $k = \Omega(N \log(dN))$, the strategy above guarantees that

$$\mathbb{E}_S \mathbb{E}_A \left[L_\mathcal{D}(\widehat{w}) \right] - \inf_{w \in \mathcal{W}_2} L_\mathcal{D}(w) \leq O\left(\sqrt{\frac{L^2 B_2^2 R_2^2}{N}} \right),$$

where $\mathbb{E}_S$ denotes expectation over samples and $\mathbb{E}_A$ denotes expectation over the algorithm's randomness. The total communication is $O(mN \log(dN) \log(LB_2 R_2 N) + m \log(dN) \log d)$ bits.

2.3. ℓ_1/ℓ_∞-Bounded Models: Model Compression

While the results for the ℓ_2/ℓ_2-bounded setting are encouraging, they are not useful in the common situation where features are dense. When features are ℓ_∞-bounded, Equation (4) shows that one can obtain nearly dimension-independent excess risk so long as they restrict to ℓ_1-bounded weights. This ℓ_1/ℓ_∞-bounded setting is particularly important because it captures the fundamental problem of learning from a finite hypothesis class, or *aggregation* (Tsybakov, 2003): Given a class $\mathcal{H}$ of $\{\pm 1\}$-valued predictors with $|\mathcal{H}| < \infty$ we can set $x = (h(z))_{h \in \mathcal{H}} \in \mathbb{R}^{|\mathcal{H}|}$, in which case (4) turns into the familiar finite class bound $\sqrt{\log|\mathcal{H}|/N}$ (Shalev-Shwartz & Ben-David, 2014). Thus, algorithms with communication sublinear in dimension for the ℓ_1/ℓ_∞ setting would lead to positive results in the general setting (1).

As first positive result in this direction, we observe that by using the well-known technique of *randomized sparsification* or *Maurey sparsification*, we can compress models to require only logarithmic communication while preserving excess risk.[2] The method is simple: Suppose we have a weight vector w that lies in the simplex Δ_d. We sample s elements of $[d]$ i.i.d. according to w and return the empirical distribution, which we will denote $Q^s(w)$. The empirical distribution is always s-sparse and can be communicated using at most $O(s \log(ed/s))$ bits when $s \leq d$,[3] and it follows from standard concentration tools that by taking s large enough the empirical distribution will approximate the true vector w arbitrarily well.

The following lemma shows that Maurey sparsification indeed provides a dimension-independent approximation to the excess risk in the ℓ_1/ℓ_∞-bounded setting. It applies to a version of the Maurey technique for general vectors, which is given in Algorithm 1.

Lemma 1. Let $w \in \mathbb{R}^d$ be fixed and suppose features belong to $\mathcal{X}_\infty$. When ϕ is L-Lipschitz, Algorithm 1 guarantees that

$$\mathbb{E}\, L_\mathcal{D}(Q^s(w)) \leq L_\mathcal{D}(w) + \left(\frac{2L^2 R_\infty^2 \|w\|_1^2}{s} \right)^{1/2},$$

where the expectation is with respect to the algorithm's randomness. Furthermore, when ϕ is β-smooth[4] Algorithm 1 guarantees:

$$\mathbb{E}\, L_\mathcal{D}(Q^s(w)) \leq L_\mathcal{D}(w) + \frac{\beta R_\infty^2 \|w\|_1^2}{s}.$$

The number of bits required to communicate $Q^s(w)$, including sending the scalar $\|w\|_1$ up to numerical precision, is at most $O(s \log(ed/s) + \log(LB_1 R_\infty s))$. Thus, if any single machine is able to find an estimator $\widehat{w}$ with good excess risk, they can communicate it to any other machine while preserving the excess risk with sublinear communication. In particular, to preserve the optimal excess risk guarantee in (4) for a Lipschitz loss such as absolute or hinge, the total bits of communication required is only

[2] We refer to the method as Maurey sparsification in reference to Maurey's early use of the technique in Banach spaces (Pisier, 1980), which predates its long history in learning theory (Jones, 1992; Barron, 1993; Zhang, 2002).

[3] That $O(s \log(ed/s))$ bits rather than, e.g., $O(s \log d)$ bits suffice is a consequence of the usual "stars and bars" counting argument. We expect one can bring the expected communication down further using an adaptive scheme such as Elias coding, as in Alistarh et al. (2017).

[4] A scalar function is said to be β-smooth if it has β-Lipschitz first derivative.

$O(N + \log(LB_1 R_\infty N))$, which is indeed sublinear in dimension! For smooth losses (square, logistic), this improves further to only $O(\sqrt{N \log(ed/N)} + \log(LB_1 R_\infty N))$ bits.

2.4. ℓ_1/ℓ_∞-Bounded Models: Impossibility

Alas, we have only shown that *if* we happen to find a good solution, we can send it using sublinear communication. If we have to start from scratch, is it possible to use Maurey sparsification to coordinate between all machines to find a good solution?

Unfortunately, the answer is no: For the ℓ_1/ℓ_∞ bounded setting, in the extreme case where each machine has a single example, *no algorithm* can obtain a risk bound matching (4) until the number of bits b allowed per machine is (nearly) linear in d.

Theorem 2. Consider the problem of learning with linear loss in the $(b, 1, N)$ model, where the risk is $L_{\mathcal{D}}(w) = \mathbb{E}_{(x,y)\sim\mathcal{D}}[-y\langle w, x\rangle]$. Let the benchmark class be the ℓ_1 ball $\mathcal{W}_1$, where $B_1 = 1$. For any algorithm $\widehat{w}$ there exists a distribution $\mathcal{D}$ with $\|x\|_\infty \le 1$ and $|y| \le 1$ such that

$$\Pr\left(L_{\mathcal{D}}(\widehat{w}) - \inf_{w\in\mathcal{W}_1} L_{\mathcal{D}}(w) \ge \frac{1}{16}\sqrt{\frac{d}{b}\cdot\frac{1}{N}} \wedge \frac{1}{2}\right) \ge \frac{1}{2}.$$

The lower bound also extends to the case of multiple examples per machine (i.e., $m > 1$), albeit with a less sharp tradeoff.

Proposition 1. Let m, n, and $\varepsilon > 0$ be fixed. In the setting of Theorem 2, any algorithm in the (b, n, m) protocol with $b \le O(d^{1-\varepsilon/2}/\sqrt{N})$ has excess risk at least $\Omega(\sqrt{d^\varepsilon/N})$ with constant probability.

This lower bound follows almost immediately from reduction to the "hide-and-seek" problem of Shamir (2014). The weaker guarantee from Proposition 1 is a consequence of the fact that the lower bound for the hide-and-seek problem from Shamir (2014) is weaker in the multi-machine case.

The value of Theorem 2 and Proposition 1 is to rule out the possibility of obtaining optimal excess risk with communication polylogarithmic in d in the ℓ_1/ℓ_∞ setting, even when there are many examples per machine. This motivates the results of the next section, which show that for ℓ_1/ℓ_q-bounded models it is indeed possible to get polylogarithmic communication for any value of m.

One might hope that it is possible to circumvent Theorem 2 by making compressed sensing-type assumptions, e.g. assuming that the vector $w^\star$ is sparse and that restricted eigenvalue or a similar property is satisfied. Unfortunately, this is not the case.[5]

Proposition 2. Consider square loss regression in the $(b, 1, N)$ model. For any algorithm $\widehat{w}$ there exists a distribution $\mathcal{D}$ with the following properties:

- $\|x\|_\infty \le 1$ and $|y| \le 1$ with probability 1.
- $\Sigma := \mathbb{E}[xx^\top] = I$, so that the population risk is 1-strongly convex, and in particular has restricted strong convexity constant 1.
- $w^\star := \arg\min_{w:\|w\|_1 \le 1} L_{\mathcal{D}}(w)$ is 1-sparse.
- $\Pr\left(L_{\mathcal{D}}(\widehat{w}) - L_{\mathcal{D}}(w^\star) \ge \frac{1}{256}\left(\frac{d}{b}\cdot\frac{1}{N}\right) \wedge \frac{1}{4}\right) \ge \frac{1}{2}$.

Moreover, any algorithm in the (b, n, m) protocol with $b \le O(d^{1-\varepsilon/2}/\sqrt{N})$ has excess risk at least $\Omega(d^\varepsilon/N)$ with constant probability.

That $\Omega(d)$ communication is required to obtain optimal excess risk for $m = N$ was proven in (Steinhardt & Duchi, 2015). The lower bound for general m is important here because it serves as a converse to the algorithmic results we develop for sparse regression in Section 3.[6]

3. Sparsified Mirror Descent

We now deliver on the promise outlined in the introduction and give new algorithms with logarithmic communication under an assumption we call ℓ_1/ℓ_q*-boundness*. The model for which we derive algorithms in this section is more general than the linear model setup (2) to which our lower bounds apply. We consider problems of the form

$$\underset{w\in\mathcal{W}}{\text{minimize}} \quad L_{\mathcal{D}}(w) := \mathbb{E}_{z\sim\mathcal{D}}\,\ell(w, z), \tag{6}$$

where $\ell(\cdot, z)$ is convex, $\mathcal{W} \subseteq \mathcal{W}_1 = \left\{w \in \mathbb{R}^d \mid \|w\|_1 \le B_1\right\}$ is a convex constraint set, and subgradients $\partial\ell(w, z)$ are assumed to belong to $\mathcal{X}_q = \left\{x \in \mathbb{R}^d \mid \|x\|_q \le R_q\right\}$. This setting captures linear models with ℓ_1-bounded weights and ℓ_q-bounded features as a special case, but is more general, since the loss can be any Lipschitz function of w.

We have already shown that one cannot expect sublinear-communication algorithms for ℓ_1/ℓ_∞-bounded models, and so the ℓ_q-boundedness of subgradients in (6) may be thought of as strengthening our assumption on the data generating process. That this is stronger follows from the elementary fact that $\|x\|_q \ge \|x\|_\infty$ for all q.

Statistical complexity and nontriviality. For the dual ℓ_1/ℓ_∞ setup in (2) the optimal rate is $\Theta(\sqrt{\log d/N})$. While our goal is to find minimal assumptions that allow for distributed learning with sublinear communication, the reader

[5] See Appendix B.2 for additional discussion of compressed sensing based assumptions under which sublinear communication may be possible.

[6] (Braverman et al., 2016) also prove a communication lower bound for sparse regression. Their lower bound applies for all values of m and for more sophisticated interactive protocols, but does not rule out the possibility of $\text{poly}(N, m, \log d)$ communication.

may wonder at this point whether we have made the problem easier *statistically* by moving to the ℓ_1/ℓ_q assumption. The answer is "yes, but only slightly." When q is constant the optimal rate for ℓ_1/ℓ_q-bounded models is $\Theta(\sqrt{1/N})$,[7] and so the effect of this assumption is to shave off the $\log d$ factor that was present in (4).

3.1. Lipschitz Losses

Our main algorithm is called *sparsified mirror descent* (Algorithm 2). The idea is to run the online mirror descent algorithm (Ben-Tal & Nemirovski, 2001; Hazan, 2016) in serial across the machines and sparsify the iterates whenever we move from one machine to the next.

In a bit more detail, Algorithm 2 proceeds from machine to machine sequentially. On each machine, the algorithm generates a sequence of iterates $w_1^i, \dots, w_n^i$ by doing a single pass over the machine's n examples $z_1^i, \dots, z_n^i$ using the mirror descent update with regularizer $\mathcal{R}(w) = \frac{1}{2}\|w\|_p^2$, where $\frac{1}{p} + \frac{1}{q} = 1$, and using stochastic gradients $\nabla_t^i \in \partial\ell(w_t^i, z_t^i)$. After the last example is processed on machine i, we compress the last iterate using Maurey sparsification (Algorithm 1) and send it to the next machine, where the process is repeated.

To formally describe the algorithm, we recall the definition of the *Bregman divergence*. Given a convex regularization function $\mathcal{R} : \mathbb{R}^d \to \mathbb{R}$, the Bregman divergence for $\mathcal{R}$ is defined as $D_{\mathcal{R}}(w\|w') = \mathcal{R}(w) - \mathcal{R}(w') - \langle \nabla\mathcal{R}(w'), w - w' \rangle$. For the ℓ_1/ℓ_q setting we exclusively use the regularizer $\mathcal{R}(w) = \frac{1}{2}\|w\|_p^2$, where $\frac{1}{p} + \frac{1}{q} = 1$.

The main guarantee for Algorithm 2 is as follows.

Theorem 3. Let $q \geq 2$ be fixed. Suppose that subgradients belong to $\mathcal{X}_q$ and that $\mathcal{W} \subseteq \mathcal{W}_1$. If we run Algorithm 2 with $\eta = \frac{B_1}{R_q}\sqrt{\frac{1}{C_q N}}$ and with initial point $\bar{w} = 0$, then whenever $s = \Omega(m^{2(q-1)})$ and $s_0 = \Omega(N^{\frac{q}{2}})$ the algorithm guarantees

$$\mathbb{E}[L_{\mathcal{D}}(\widehat{w})] - L_{\mathcal{D}}(w^\star) \leq O\left(\sqrt{\frac{B_1^2 R_q^2 C_q}{N}}\right),$$

where $C_q = q - 1$ is a constant depending only on q.

The total number of bits sent by each machine—besides communicating the final iterate $\widehat{w}$—is at most $O(m^{2(q-1)}\log(d/m) + \log(B_1 R_q N))$, and so the total number of bits communicated globally is at most

$$O\left(N^{\frac{q}{2}}\log(d/N) + m^{2q-1}\log(d/m) + m\log(B_1 R_q N)\right).$$

In the linear model setting (2) with 1-Lipschitz loss ϕ it suffices to set $s_0 = \Omega(N)$, so the total bits of communication is $O(N\log(d/N) + m^{2q-1}\log(d/m) + m\log(B_1 R_q N))$.

```
Algorithm 2 (Sparsified Mirror Descent).
Input:
  Constraint set $\mathcal{W}$ with $\|w\|_1 \leq B_1$.
  Gradient norm parameter $q \in [2, \infty)$.
  Gradient $\ell_q$ norm bound $R_q$.
  Learning rate $\eta$, Initial point $\bar{w}$, Sparsity $s, s_0 \in \mathbb{N}$.
Define $p = \frac{q}{q-1}$ and $\mathcal{R}(w) = \frac{1}{2}\|w - \bar{w}\|_p^2$.
For machine $i = 1, \dots, m$:
  • Receive $\widehat{w}^{i-1}$ from machine $i-1$ and set $w_1^i = \widehat{w}^{i-1}$ (if machine 1 set $w_1^1 = \bar{w}$).
  • For $t = 1, \dots, n$: // Mirror descent step.
      – Get gradient $\nabla_t^i \in \partial\ell(w_t^i; z_t^i)$.
      – $\nabla\mathcal{R}(\theta_{t+1}^i) \leftarrow \nabla\mathcal{R}(w_t^i) - \eta\nabla_t^i$.
      – $w_{t+1}^i \leftarrow \arg\min_{w\in\mathcal{W}} D_{\mathcal{R}}(w\|\theta_{t+1}^i)$.
  • Let $\widehat{w}^i \leftarrow Q^s(w_{n+1}^i)$. // Sparsification.
  • Send $\widehat{w}^i$ to machine $i+1$.
Sample $i \in [m]$, $t \in [n]$ uniformly at random and return $\widehat{w} := Q^{s_0}(w_t^i)$.
```

We see that the communication required by sparsified mirror descent is exponential in the norm parameter q. This means that whenever q is constant, the overall communication is polylogarithmic in dimension. When $q = \log d$, $\|x\|_q \approx \|x\|_\infty$ up to a multiplicative constant. In this case the communication from Theorem 3 becomes polynomial in dimension, which we know from Section 2.4 is necessary.

The guarantee of Algorithm 2 extends beyond the statistical learning model to the first-order stochastic convex optimization model, as well as the online convex optimization model.

Proof sketch. They basic premise behind the algorithm and its analysis is that by using the same learning rate across all machines, we can pretend as though we are running a single instance of mirror descent on a centralized machine. The key difference from the usual analysis is that we need to bound the error incurred by sparsification between successive machines. Here, the choice of the regularizer is crucial. A fundamental property used in the analysis of mirror descent is *strong convexity* of the regularizer. To give convergence rates that do not depend on dimension (such as (3)) it is essential that the regularizer be $\Omega(1)$-strongly convex. Our regularizer $\mathcal{R}$ indeed has this property.

Proposition 3 (Ball et al. (1994))**.** For $p \in (1, 2]$, $\mathcal{R}$ is $(p-1)$-strongly convex with respect to $\|\cdot\|_p$. Equivalently, $D_{\mathcal{R}}(w\|w') \geq \frac{p-1}{2} \cdot \|w - w'\|_p^2 \quad \forall w, w' \in \mathbb{R}^d$.

On the other hand, to argue that sparsification has negligible impact on convergence, our analysis leverages *smoothness*

[7] The upper bound follows from (3) and the lower bound follows by reduction to the one-dimensional case.

of the regularizer. Strong convexity and smoothness are at odds with each other: It is well known that in infinite dimension, any norm that is both strongly convex and smooth is isomorphic to a Hilbert space (Pisier, 2011). What makes our analysis work is that while the regularizer $\mathcal{R}$ is not smooth, it is *Hölder-smooth* for any finite q. This is sufficient to bound the approximation error from sparsification. To argue that the excess risk achieved by mirror descent with the ℓ_p regularizer $\mathcal{R}$ is optimal, however, it is essential that the gradients are ℓ_q-bounded rather than ℓ_∞-bounded.

In more detail, the proof has three components:

- *Telescoping.* Mirror descent gives a regret bound that telescopes across all m machines up to the error introduced by sparsification. To match the optimal centralized regret, we only need to to bound m error terms of the form $D_{\mathcal{R}}(w^\star \| Q^s(w^i_{n+1})) - D_{\mathcal{R}}(w^\star \| w^i_{n+1})$.
- *Hölder-smoothness.* We prove (Theorem 7) that the difference above is of order
$$B_1\|Q^s(w^i_{n+1}) - w^i_{n+1}\|_p + B_1^{3-p}\|Q^s(w^i_{n+1}) - w^i_{n+1}\|_\infty^{p-1}.$$
- *Maurey for ℓ_p norms.* We prove (Theorem 6) that $\|Q^s(w^i_{n+1}) - w^i_{n+1}\|_p \lesssim \left(\frac{1}{s}\right)^{1-1/p}$ and likewise that $\|Q^s(w^i_{n+1}) - w^i_{n+1}\|_\infty \lesssim \left(\frac{1}{s}\right)^{1/2}$.

With a bit more work these inequalities yield Theorem 3. We close this section with a few more notes about Algorithm 2 and its performance.

Remark 1. *For the special case of ℓ_1/ℓ_q-bounded linear models with Lipschitz link function, it is straightforward to show that the following strategy also leads to sublinear communication: Truncate each feature vector to the top $\Theta(N^{q/2})$ coordinates, then send all the truncated examples to a central server, which returns the empirical risk minimizer. This strategy matches the risk of Theorem 3 with total communication $\tilde{O}(N^{q/2+1})$, but has two deficiencies. First, the total communication is larger than the $\tilde{O}(N + m^{2q-1})$ bound achieved by Theorem 3, unless m is very large. Second, it does not extend to the general optimization setting.*

3.2. Smooth Losses

We can improve the statistical guarantee and total communication further for the case where $L_{\mathcal{D}}$ is *smooth* with respect to ℓ_q rather than just Lipschitz. We assume that ℓ has β_q-Lipschitz gradients, in the sense that for all $w, w' \in \mathcal{W}_1$ for all z, $\|\nabla\ell(w,z) - \nabla\ell(w',z)\|_q \le \beta_q\|w - w'\|_p$, where p is such that $\frac{1}{p} + \frac{1}{q}$.

Theorem 4. Suppose in addition to the assumptions of Theorem 3 that $\ell(\cdot, z)$ is non-negative and has β_q-Lipschitz gradients with respect to ℓ_q. Let $L^\star = \inf_{w\in\mathcal{W}} L_{\mathcal{D}}(w)$. If we run Algorithm 2 with learning rate $\eta = \sqrt{\frac{B_1^2}{C_q\beta_q L^\star N}} \wedge \frac{1}{4C_q\beta_q}$ and $\bar{w} = 0$ then, if $s = \Omega(m^{2(q-1)})$ and $s_0 = \sqrt{\frac{\beta_q B_1^2 N}{C_q L^\star}} \wedge \frac{N}{C_q}$, the algorithm guarantees

$$\mathbb{E}[L_{\mathcal{D}}(\widehat{w})] - L^\star \le O\left(\sqrt{\frac{C_q\beta_q B_1^2 L^\star}{N}} + \frac{C_q\beta_q B_1^2}{N}\right).$$

The total number of bits sent by each machine—besides communicating the final iterate $\widehat{w}$—is at most $O(m^{2(q-1)}\log(d/m))$, and so the total number of bits communicated globally is at most $O(m\log(\beta_q B_1 N))+$

$$O\left(\left(\sqrt{\frac{\beta_q B_1^2 N}{C_q L^\star}} \wedge \frac{N}{C_q}\right)\log(d/N) + m^{2q-1}\log(d/m)\right).$$

Compared to the previous theorem, this result provides a so-called "small-loss bound" (Srebro et al., 2010), with the main term scaling with the optimal loss $L^\star$. The dependence on N in the communication cost can be as low as $O(\sqrt{N})$ depending on the value of $L^\star$.

3.3. Fast Rates under Restricted Strong Convexity

So far all of the algorithmic results we have present scale as $O(N^{-1/2})$. While this is optimal for generic Lipschitz losses, we mentioned in Section 2 that for strongly convex losses the rate can be improved in a nearly-dimension independent fashion to $O(N^{-1})$ for sparse high-dimensional linear models. As in the generic Lipschitz loss setting, we show that making the assumption of ℓ_1/ℓ_q-boundedness is sufficient to get statistically optimal distributed algorithms with sublinear communication, thus providing a way around the lower bounds for fast rates in Section 2.4. The key assumption for the results in this section is that the population risk satisfies a form of *restricted strong convexity*.

Assumption 1. There is a constant γ_q such that $\forall w \in \mathcal{W}$, $L_{\mathcal{D}}(w) - L_{\mathcal{D}}(w^\star) - \langle\nabla L_{\mathcal{D}}(w^\star), w - w^\star\rangle \ge \frac{\gamma_q}{2}\|w - w^\star\|_p^2$.

In a moment we will show how to relate this property to the standard restricted eigenvalue property in high-dimensional statistics (Negahban et al., 2012) and apply it to sparse regression.

Our main algorithm for strongly convex losses is Algorithm 3, which is stated in Appendix C due to space constraints. The algorithm does not introduce any new tricks for distributed learning over Algorithm 2; rather, it invokes Algorithm 2 repeatedly in an inner loop, relying on these invocations to take care of communication. This reduction is based on techniques developed in (Juditsky & Nesterov, 2014), whereby restricted strong convexity is used to establish that error decreases geometrically as a function of the number of invocations to the sub-algorithm. We refer the reader to Appendix C for additional details.

Theorem 5. Suppose Assumption 1 holds, that subgradients belong to $\mathcal{X}_q$ for $q \ge 2$, and that $\mathcal{W} \subset \mathcal{W}_1$. When the parameter $c > 0$ is a sufficiently large absolute constant, Algorithm 3 guarantees that

$$\mathbb{E}[L_{\mathcal{D}}(\widehat{w}_T)] - L_{\mathcal{D}}(w^\star) \le O\left(\frac{C_q R_q^2}{\gamma_q N}\right).$$

The total numbers of bits communicated is

$$O\left(\left(N^{2(q-1)} m^{2q-1}\left(\frac{\gamma_q^2 B_q^2}{C_q R_q^2}\right)^{2(q-1)} + N^q\left(\frac{\gamma_q B_1}{C_q R_q}\right)^q\right)\log d\right),$$

plus $O(m\log(B_1 R_q N))$. Treating scale parameters as constants, the total communication simplifies to $O\big(N^{2q-2} m^{2q-1}\log d\big)$.

Application: Sparse Regression. As an application of Algorithm 3, we consider the sparse regression setting (5), where $L_{\mathcal{D}}(w) = \mathbb{E}_{x,y}(\langle w, x\rangle - y)^2$. We assume $\|x\|_q \le R_q$ and $|y| \le 1$. We let $w^\star = \arg\min_{w\in\mathcal{W}_1} L_{\mathcal{D}}(w)$, so $\|w^\star\|_1 \le B_1$. We assume $w^\star$ is k-sparse with support set $S \subset [d]$.

We invoke Algorithm 3 constraint set $\mathcal{W} := \{w \in \mathbb{R}^d \mid \|w\|_1 \le \|w^\star\|_1\}$ and let $\Sigma = \mathbb{E}[xx^\top]$. Our bound depends on the restricted eigenvalue parameter: $\gamma := \inf_{\nu\in\mathcal{W}-w^\star\setminus\{0\}} \|\Sigma^{1/2}\nu\|_2^2/\|\nu\|_2^2$.

Proposition 4. Algorithm 3, with constraint set $\mathcal{W}$ and appropriate choice of parameters, guarantees:

$$\mathbb{E}[L_{\mathcal{D}}(\widehat{w}_T)] - L_{\mathcal{D}}(w^\star) \le O\left(C_q B_1^2 R_q^2 \cdot \frac{k}{\gamma N}\right).$$

Suppressing problem-dependent constants, total communication is of order $O((N^{2q-2}m^{2q-1}\log d)/k^{4q-4})$.

3.4. Extension: Matrix Learning and Beyond

The basic idea behind sparsified mirror descent—that by assuming ℓ_q-boundedness one can get away with using a Hölder-smooth regularizer that behaves well under sparsification—is not limited to the ℓ_1/ℓ_q setting. To extend the algorithm to more general geometry, all that is required is the following:

- The constraint set $\mathcal{W}$ can be written as the convex hull of a set of atoms $\mathcal{A}$ that has sublinear bit complexity.
- The data should be bounded in some norm $\|\cdot\|$ such that the dual $\|\cdot\|_\star$ admits a regularizer $\mathcal{R}$ that is strongly convex and Hölder-smooth with respect to $\|\cdot\|_\star$
- $\|\cdot\|_\star$ is preserved under sparsification. We remark in passing that this property and the previous one are closely related to the notions of type and cotype in Banach spaces (Pisier, 2011).

Here we deliver on this potential and sketch how to extend the results so far to *matrix learning* problems where $\mathcal{W} \subseteq \mathbb{R}^{d\times d}$ is a convex set of matrices. As in Section 3.1 we work with a generic Lipschitz loss $L_{\mathcal{D}}(W) = \mathbb{E}_z\,\ell(W, z)$. Letting $\|W\|_{S_p} = \mathsf{tr}((WW^\top)^{\frac{p}{2}})$ denote the Schatten p-norm, we make the following spectral analogue of the ℓ_1/ℓ_q-boundedness assumption: $\mathcal{W} \subseteq \mathcal{W}_{S_1} := \{W \in \mathbb{R}^{d\times d} \mid \|W\|_{S_1} \le B_1\}$ and subgradients $\partial\ell(\cdot, z)$ belong to $\mathcal{X}_{S_q} := \left\{X \in \mathbb{R}^{d\times d} \mid \|X\|_{S_q} \le R_q\right\}$, where $q \ge 2$. Recall that S_1 and S_∞ are the nuclear norm and spectral norm, respectively. The S_1/S_∞ setup has many applications in learning (Hazan et al., 2012).

We make the following key changes to Algorithm 2:

- Use the Schatten regularizer $\mathcal{R}(W) = \frac{1}{2}\|W\|_{S_p}^2$.
- Use the following spectral version of the Maurey operator $Q^s(W)$: Let W have singular value decomposition $W = \sum_{i=1}^d \sigma_i u_i v_i^\top$ with $\sigma_i \ge 0$ and define $P \in \Delta_d$ via $P_i \propto \sigma_i$.[8] Sample $i_1, \ldots, i_s$ i.i.d. from P and return $Q^s(W) = \frac{\|W\|_{S_1}}{s}\sum_{\tau=1}^s u_{i_\tau} v_{i_\tau}^\top$.
- Encode and transmit $Q^s(W)$ as the sequence $(u_{i_1}, v_{i_1}), \ldots, (u_{i_s}, v_{i_s})$, plus the scalar $\|W\|_{S_1}$. This takes $\tilde{O}(sd)$ bits.

Proposition 5. Let $q \ge 2$ be fixed, and suppose that subgradients belong to $\mathcal{X}_{S_q}$ and that $\mathcal{W} \subseteq \mathcal{W}_{S_1}$. If we run the variant of Algorithm 2 described above with learning rate $\eta = \frac{B_1}{R_q}\sqrt{\frac{1}{C_q N}}$ and initial point $\bar{W} = 0$, then whenever $s = \Omega(m^{2(q-1)})$ and $s_0 = \Omega(N^{\frac{q}{2}})$, the algorithm guarantees

$$\mathbb{E}\big[L_{\mathcal{D}}(\widehat{W})\big] - \inf_{W\in\mathcal{W}} L_{\mathcal{D}}(W) \le O\left(\sqrt{\frac{B_1^2 R_q^2 C_q}{N}}\right),$$

where $C_q = q - 1$. The total number of bits communicated globally is at most $\tilde{O}(m^{2q-1}d + N^{\frac{q}{2}}d)$.

In the matrix setting, the number of bits required to naively send weights $W \in \mathbb{R}^{d\times d}$ or subgradients $\partial\ell(W, z) \in \mathbb{R}^{d\times d}$ is $O(d^2)$. The communication required by our algorithm scales only as $\tilde{O}(d)$, so it is indeed sublinear. The proof of Proposition 5 is sketched in Appendix C.

4. Discussion

We hope our work will lead to further development of algorithms with sublinear communication. A few immediate questions:

- Can we get matching upper and lower bounds for communication in terms of m, N, $\log d$, and q?
- Currently all of our algorithms work serially. Can we extend the techniques to give parallel speedup?
- Returning to the general setting (1), what abstract properties of the hypothesis class $\mathcal{H}$ are required to guarantee that learning with sublinear-communication is possible?

[8] We may assume $\sigma_i \ge 0$ without loss of generality.

Acknowledgements

Part of this work was completed while DF was a student at Cornell University and supported by the Facebook PhD fellowship.

References

Acharya, J., Canonne, C. L., and Tyagi, H. Communication constrained inference and the role of shared randomness. In *International Conference on Machine Learning*, 2019a.

Acharya, J., Canonne, C. L., and Tyagi, H. Inference under information constraints I: lower bounds from chi-square contraction. In *Conference on Learning Theory*, 2019b.

Alistarh, D., Grubic, D., Li, J., Tomioka, R., and Vojnovic, M. Qsgd: Communication-efficient sgd via gradient quantization and encoding. In *Advances in Neural Information Processing Systems*, pp. 1709–1720, 2017.

Alistarh, D., Hoefler, T., Johansson, M., Konstantinov, N., Khirirat, S., and Renggli, C. The convergence of sparsified gradient methods. In *Advances in Neural Information Processing Systems*, pp. 5977–5987, 2018.

Ball, K., Carlen, E. A., and Lieb, E. H. Sharp uniform convexity and smoothness inequalities for trace norms. *Inventiones mathematicae*, 115(1):463–482, 1994.

Barron, A. R. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information Theory*, 39(3):930–945, May 1993.

Bekkerman, R., Bilenko, M., and Langford, J. *Scaling up machine learning: Parallel and distributed approaches*. Cambridge University Press, 2011.

Ben-Tal, A. and Nemirovski, A. *Lectures on modern convex optimization: analysis, algorithms, and engineering applications*, volume 2. Siam, 2001.

Bernstein, J., Wang, Y.-X., Azizzadenesheli, K., and Anandkumar, A. Signsgd: Compressed optimisation for non-convex problems. In *International Conference on Machine Learning*, pp. 559–568, 2018.

Bijral, A. S., Sarwate, A. D., and Srebro, N. On data dependence in distributed stochastic optimization. *arXiv preprint arXiv:1603.04379*, 2016.

Braverman, M., Garg, A., Ma, T., Nguyen, H. L., and Woodruff, D. P. Communication lower bounds for statistical estimation problems via a distributed data processing inequality. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pp. 1011–1020. ACM, 2016.

Cesa-Bianchi, N. and Lugosi, G. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

Chaturapruek, S., Duchi, J. C., and Ré, C. Asynchronous stochastic convex optimization: the noise is in the noise and sgd don't care. In *Advances in Neural Information Processing Systems*, pp. 1531–1539, 2015.

Dagan, Y. and Shamir, O. Detecting correlations with little memory and communication. In *Conference on Learning Theory*, pp. 1145–1198, 2018.

Dagan, Y., Kur, G., and Shamir, O. Space lower bounds for linear prediction. In *Conference on Learning Theory*, 2019.

Dekel, O., Gilad-Bachrach, R., Shamir, O., and Xiao, L. Optimal distributed online prediction using mini-batches. *Journal of Machine Learning Research*, 13(Jan):165–202, 2012.

Duchi, J. C., Agarwal, A., and Wainwright, M. J. Dual averaging for distributed optimization: Convergence analysis and network scaling. *IEEE Transactions on Automatic control*, 57(3):592–606, 2012.

Duchi, J. C., Jordan, M. I., Wainwright, M. J., and Zhang, Y. Optimality guarantees for distributed statistical estimation. *arXiv preprint arXiv:1405.0782*, 2014.

Garg, A., Ma, T., and Nguyen, H. On communication cost of distributed statistical estimation and dimensionality. In *Advances in Neural Information Processing Systems*, pp. 2726–2734, 2014.

Han, Y., Özgür, A., and Weissman, T. Geometric lower bounds for distributed parameter estimation under communication constraints. In *Conference On Learning Theory*, pp. 3163–3188, 2018.

Hazan, E. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.

Hazan, E., Kale, S., and Shalev-Shwartz, S. Near-optimal algorithms for online matrix prediction. *Conference on Learning Theory*, 2012.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Jones, L. K. A simple lemma on greedy approximation in hilbert space and convergence rates for projection pursuit regression and neural network training. *The annals of Statistics*, 20(1):608–613, 1992.

Juditsky, A. and Nesterov, Y. Deterministic and stochastic primal-dual subgradient algorithms for uniformly convex minimization. *Stochastic Systems*, 4(1):44–80, 2014.

Kakade, S. M., Sridharan, K., and Tewari, A. On the complexity of linear prediction: Risk bounds, margin bounds, and regularization. In *Advances in neural information processing systems*, pp. 793–800, 2009.

Kakade, S. M., Shalev-Shwartz, S., and Tewari, A. Regularization techniques for learning with matrices. *Journal of Machine Learning Research*, 13(Jun):1865–1890, 2012.

Kane, D. M. and Nelson, J. A derandomized sparse johnson-lindenstrauss transform. *arXiv preprint arXiv:1006.3585*, 2010.

Loh, P.-L., Wainwright, M. J., et al. Support recovery without incoherence: A case for nonconvex regularization. *The Annals of Statistics*, 45(6):2455–2482, 2017.

Negahban, S. N., Ravikumar, P., Wainwright, M. J., Yu, B., et al. A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. *Statistical Science*, 27(4):538–557, 2012.

Pisier, G. Remarques sur un résultat non publié de b. maurey. *Séminaire Analyse fonctionnelle (dit)*, pp. 1–12, 1980.

Pisier, G. Martingales in banach spaces (in connection with type and cotype). course ihp, feb. 2–8, 2011. 2011.

Raz, R. Fast learning requires good memory: A time-space lower bound for parity learning. *Journal of the ACM (JACM)*, 66(1):3, 2018.

Recht, B., Re, C., Wright, S., and Niu, F. Hogwild: A lock-free approach to parallelizing stochastic gradient descent. In *Advances in neural information processing systems*, pp. 693–701, 2011.

Sahasranand, K. and Tyagi, H. Extra samples can reduce the communication for independence testing. In *2018 IEEE International Symposium on Information Theory (ISIT)*, pp. 2316–2320. IEEE, 2018.

Seide, F., Fu, H., Droppo, J., Li, G., and Yu, D. 1-bit stochastic gradient descent and its application to data-parallel distributed training of speech dnns. In *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

Shalev-Shwartz, S. and Ben-David, S. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Shalev-Shwartz, S., Srebro, N., and Zhang, T. Trading accuracy for sparsity in optimization problems with sparsity constraints. *SIAM Journal on Optimization*, 20(6): 2807–2832, 2010.

Shamir, O. Fundamental limits of online and distributed algorithms for statistical learning and estimation. In *Advances in Neural Information Processing Systems*, pp. 163–171, 2014.

Srebro, N., Sridharan, K., and Tewari, A. Smoothness, low noise and fast rates. In *Advances in Neural Information Processing Systems*, pp. 2199–2207, 2010.

Steinhardt, J. and Duchi, J. Minimax rates for memory-bounded sparse linear regression. In *Conference on Learning Theory*, pp. 1564–1587, 2015.

Steinhardt, J., Valiant, G., and Wager, S. Memory, communication, and statistical queries. In *Conference on Learning Theory*, pp. 1490–1516, 2016.

Stich, S. U., Cordonnier, J.-B., and Jaggi, M. Sparsified sgd with memory. In *Advances in Neural Information Processing Systems*, pp. 4452–4463, 2018.

Suresh, A. T., Yu, F. X., Kumar, S., and McMahan, H. B. Distributed mean estimation with limited communication. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 3329–3337. JMLR. org, 2017.

Tang, H., Gan, S., Zhang, C., Zhang, T., and Liu, J. Communication compression for decentralized training. In *Advances in Neural Information Processing Systems*, pp. 7663–7673, 2018.

Tsybakov, A. B. Optimal rates of aggregation. In *Learning Theory and Kernel Machines*, pp. 303–313. Springer, 2003.

Wang, H., Sievert, S., Liu, S., Charles, Z., Papailiopoulos, D., and Wright, S. Atomo: Communication-efficient learning via atomic sparsification. In *Advances in Neural Information Processing Systems*, pp. 9872–9883, 2018.

Wangni, J., Wang, J., Liu, J., and Zhang, T. Gradient sparsification for communication-efficient distributed optimization. In *Advances in Neural Information Processing Systems*, pp. 1306–1316, 2018.

Zhang, H., Li, J., Kara, K., Alistarh, D., Liu, J., and Zhang, C. Zipml: Training linear models with end-to-end low precision, and a little bit of deep learning. In *International Conference on Machine Learning*, pp. 4035–4043, 2017.

Zhang, T. Covering number bounds of certain regularized linear function classes. *Journal of Machine Learning Research*, 2(Mar):527–550, 2002.

Zhang, Y., Wainwright, M. J., and Duchi, J. C. Communication-efficient algorithms for statistical optimization. In *Advances in Neural Information Processing Systems*, pp. 1502–1510, 2012.

Zinkevich, M. Online convex programming and generalized infinitesimal gradient ascent. In *International Conference on Machine Learning*, pp. 928–936, 2003.