# An Iterative, Sketching-based Framework for Ridge Regression

**Agniva Chowdhury** [1] **Jiasen Yang** [1] **Petros Drineas** [2]

## Abstract

Ridge regression is a variant of regularized least squares regression that is particularly suitable in settings where the number of predictor variables greatly exceeds the number of observations. We present a simple, iterative, sketching-based algorithm for ridge regression that guarantees high-quality approximations to the optimal solution vector. Our analysis builds upon two simple structural results that boil down to randomized matrix multiplication, a fundamental and well-understood primitive of randomized linear algebra. An important contribution of our work is the analysis of the behavior of sub-sampled ridge regression problems when the ridge leverage scores are used: we prove that accurate approximations can be achieved by a sample whose size depends on the degrees of freedom of the ridge-regression problem rather than the dimensions of the design matrix. Our empirical evaluations verify our theoretical results on both real and synthetic data.

## 1. Introduction

In statistics and machine learning, ridge regression (Gunst & Mason, 1977; Hoerl & Kennard, 1970) (also known as Tikhonov regularization or weight decay) is a variant of regularized least squares problems where the choice of the penalty function is the squared $\ell_2$-norm. Formally, let $\mathbf{A} \in \mathbb{R}^{n\times d}$ be the design matrix and let $\mathbf{b} \in \mathbb{R}^n$ be the response vector. Then, the linear algebraic formulation of the ridge regression problem is as follows:

$$\mathcal{Z}^* = \min_{\mathbf{x}\in\mathbb{R}^d} \left\{\|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_2^2\right\}, \tag{1}$$

where $\lambda > 0$ is the regularization parameter. There are two fundamental motivations underlying the use of ridge regression. First, when $d \gg n$, *i.e.*, the number of predictor variables $d$ greatly exceeds the number of observations $n$, fitting the full model without regularization (*i.e.*, setting $\lambda$ to zero) will result in large prediction intervals and a non-unique regression estimator. Second, if the design matrix $\mathbf{A}$ is ill-conditioned, solving the standard least-squares problem without regularization would depend on $(\mathbf{A}^\top\mathbf{A})^{-1}$. This inversion would be problematic if $\mathbf{A}^\top\mathbf{A}$ were singular or nearly singular and thus adding even a little noise to the elements of $\mathbf{A}$ could result in large changes in $(\mathbf{A}^\top\mathbf{A})^{-1}$. Due to these two considerations, solving standard least-squares problems without regularization may provide a good fit to the training data but may not generalize well to test data.

Ridge regression abandons the requirement of an unbiased estimator in order to address the aforementioned problems. At the cost of introducing bias, ridge regression reduces the variance and thus might reduce the overall mean squared error (MSE). The minimizer of eqn. (1) is

$$\mathbf{x}^* = \left(\mathbf{A}^\top\mathbf{A} + \lambda\mathbf{I}_d\right)^{-1}\mathbf{A}^\top\mathbf{b}, \tag{2}$$

or, equivalently (see Saunders et al. (1998) and Lemma 9 in Appendix A),

$$\mathbf{x}^* = \mathbf{A}^\top\left(\mathbf{A}\mathbf{A}^\top + \lambda\mathbf{I}_n\right)^{-1}\mathbf{b}. \tag{3}$$

Both formulations work for any $\lambda > 0$ for either under-constrained or over-constrained ridge regression problems, regardless of the rank of the design matrix $\mathbf{A}$. It is easy to see that $\mathbf{x}^*$ can be computed in time

$$\mathcal{O}(nd\min\{n,d\} + \min\{n^3,d^3\}) = \mathcal{O}(nd\min\{n,d\}).$$

In our work, we will focus on design matrices $\mathbf{A} \in \mathbb{R}^{n\times d}$ with $d \gg n$, which is the most common setting for ridge regression. For simplicity of exposition, we will assume that the rank of $\mathbf{A}$ is equal to $n$.[1] In the context of ridge regression, a much more important quantity than the rank of the design matrix is the effective *degrees of freedom*:

$$d_\lambda = \sum_{i=1}^n \frac{\sigma_i^2}{\sigma_i^2+\lambda} \le n, \tag{4}$$

where $\sigma_i$ are the singular values of $\mathbf{A}$.

[1] Department of Statistics, Purdue University, West Lafayette, IN [2] Department of Computer Science, Purdue University, West Lafayette, IN. Correspondence to: Agniva Chowdhury <chowdhu5@purdue.edu>.

*Proceedings of the $35^{th}$ International Conference on Machine Learning*, Stockholm, Sweden, PMLR 80, 2018. Copyright 2018 by the author(s).

[1] Our results can be slightly improved to depend on the rank $\rho$ of the matrix $\mathbf{A}$ instead of $n$.

The recent flurry of activity on Randomized Linear Algebra (RLA) (Drineas & Mahoney, 2016) and the widespread use of *sketching* as a tool for matrix computations (Woodruff, 2014), resulted in many novel results for ridge regression. In Section 1.2 we discuss relevant prior work.

### 1.1. Our Contributions

We present a novel iterative algorithm (Algorithm 1) for *sketched* ridge regression and two simple sketching-based structural conditions under which Algorithm 1 guarantees highly accurate approximations to the optimal solution $\mathbf{x}^*$. More precisely, Algorithm 1 guarantees that, as long as a simple structural constraint is satisfied, the resulting approximate solution vector $\widehat{\mathbf{x}}^*$ satisfies (after $t$ iterations)

$$\|\mathbf{x}^* - \widehat{\mathbf{x}}^*\|_2 \leq \varepsilon^t \|\mathbf{x}^*\|_2. \tag{5}$$

Prior to discussing the aforementioned constraint, we note that error guarantees of the above form are highly desirable. Indeed, beyond being a relative error guarantee, the dependency on $\varepsilon$ drops exponentially fast as the number of iterations increases. It is easy to see that by setting $\varepsilon^t = \varepsilon'$, $\mathcal{O}(\ln(1/\varepsilon'))$ iterations would suffice to provide a relative error guarantee with accuracy parameter $\varepsilon'$. This means that converging to, say, ten decimal digits of accuracy would necessitate only a constant number of iterations. See Section 1.2 for a comparison of this bound with prior work.

Let $\mathbf{V} \in \mathbb{R}^{d \times n}$ be the matrix of right singular vectors of $\mathbf{A}$; recall that $\mathbf{A}$ has rank $n$. For eqn. (5) to hold, a sketching matrix $\mathbf{S} \in \mathbb{R}^{d \times s}$ is to be constructed such that (for an appropriate choice of the sketching dimension $s \ll d$)

$$\|\mathbf{V}^\mathsf{T}\mathbf{S}\mathbf{S}^\mathsf{T}\mathbf{V} - \mathbf{I}_n\|_2 \leq \frac{\varepsilon}{2}. \tag{6}$$

We note that the constraint of eqn. (6) has been the topic of intense research in the RLA literature; this is precisely the reason why we use eqn. (6) as the building block in our analysis. Indeed, assuming that $n \ll d$, one can use the (exact or approximate) column leverage scores (Mahoney & Drineas, 2009; Mahoney, 2011) of $\mathbf{A}$ to satisfy the aforementioned constraint, in which case $\mathbf{S}$ is a sampling-and-rescaling matrix. Perhaps more interestingly, a variety of oblivious sketching matrix constructions for $\mathbf{S}$ can be used to satisfy eqn. (6). We discuss various constructions for $\mathbf{S}$ in Section 2.1.

One deficiency of the structural constraint of eqn. (6) is that all known constructions for $\mathbf{S}$ that satisfy the constraint need a number of columns $s$ that is proportional to $n$. As a result, the running time of any algorithm that computes the sketch $\mathbf{AS}$ is also proportional to $n$. It would be much better to design algorithms whose running time depends on the *degrees of freedom* $d_\lambda$, which is upper bounded by $n$, but could be significantly smaller depending on the distribution of the singular values and the choice of $\lambda$.

Towards that end, we analyze Algorithm 1 under a second structural constraint. We define a *diagonal* matrix $\mathbf{\Sigma}_\lambda \in \mathbb{R}^{n \times n}$ whose $i$-th diagonal entry is given by

$$(\mathbf{\Sigma}_\lambda)_{ii} = \sqrt{\frac{\sigma_i^2}{\sigma_i^2 + \lambda}}, \quad i = 1, \ldots, n. \tag{7}$$

Notice that $\|\mathbf{\Sigma}_\lambda\|_F^2 = d_\lambda$. Our second structural condition is given by

$$\|\mathbf{\Sigma}_\lambda\mathbf{V}^\mathsf{T}\mathbf{S}\mathbf{S}^\mathsf{T}\mathbf{V}\mathbf{\Sigma}_\lambda - \mathbf{\Sigma}_\lambda^2\|_2 \leq \frac{\varepsilon}{4\sqrt{2}}. \tag{8}$$

Similarly to the constraint of eqn. (6), the constraint of eqn. (8) can also be satisfied by, for example, sampling with respect to the *ridge leverage scores* of Alaoui & Mahoney (2015); Cohen et al. (2017) or by oblivious sketching matrix constructions for $\mathbf{S}$. The difference is that, instead of having the column size $s$ of the matrix $\mathbf{S}$ depend on $n$, it now depends on $d_\lambda$, which could be considerably smaller. Indeed, it follows that by sampling-and-rescaling $\mathcal{O}(d_\lambda \ln d_\lambda)$ predictor variables from the design matrix $\mathbf{A}$ (using either exact or approximate *ridge* leverage scores (Alaoui & Mahoney, 2015; Cohen et al., 2017) we can satisfy the constraint of eqn. (8). Similarly, oblivious sketching matrix constructions for $\mathbf{S}$ can be used to satisfy eqn. (8). We discuss constructions for $\mathbf{S}$ in Section 2.1.

However, this improved dependency on $d_\lambda$ instead of $n$ comes with a mild loss in accuracy. For simplicity, we only state a result when $\lambda$ satisfies $\sigma_{k+1}^2 \leq \lambda \leq \sigma_k^2$ for some integer $k$, $1 \leq k \leq n$.[2] In words, $\lambda$ can be thought of as "regularizing" the bottom $n - k$ singular values of the design matrix $\mathbf{A}$, since it dominates them. In this case, we prove that the approximation $\widehat{\mathbf{x}}^*$ returned by Algorithm 1 satisfies

$$\|\mathbf{x}^* - \widehat{\mathbf{x}}^*\|_2 \leq \frac{\varepsilon^t}{2}\left(\|\mathbf{x}^*\|_2 + \frac{1}{\sqrt{2\lambda}}\left\|\mathbf{U}_{k,\perp}^\mathsf{T}\mathbf{b}\right\|_2\right). \tag{9}$$

Here $\mathbf{U}_{k,\perp} \in \mathbb{R}^{n \times (n-k)}$ denotes the matrix of the bottom $n - k$ left singular vectors of the design matrix $\mathbf{A}$. In words, we achieve an additive-relative error approximation, where the additive error part depends on the norm of the "piece" of the response vector $\mathbf{b}$ that lies on the regularized component of the design matrix $\mathbf{A}$. As this piece grows, the quality of the approximation worsens. The error decreases exponentially fast with the number of iterations.

Another contribution of our work is Theorem 4, which proves that the mean-square-error (MSE) of the approximate solution $\widehat{\mathbf{x}}^*$ is a relative error approximation to the MSE of $\mathbf{x}^*$, under the structural assumptions of eqns. (6) or (8), even after a single iteration.

[2] The bound of eqn. (9) can be easily generalized to hold when $c_1\sigma_{k+1}^2 \leq \lambda \leq c_2\sigma_k^2$ for some constants $c_1, c_2 > 0$. For simplicity of exposition, we assume that both $c_1$ and $c_2$ equal one.

To the best of our knowledge, our bounds are a first attempt to provide general structural results that guarantee high-quality approximations to the optimal solution vector of ridge regression. Our first structural result can be satisfied by sampling with respect to the leverage scores or by the use of oblivious sketching matrices whose size depends on the rank of the design matrix and guarantees relative error approximations. Our second structural result presents the first accuracy analysis for ridge regression when the *ridge leverage scores are used to sample predictor variables*. Interestingly, the ridge leverage scores have been used in a number of applications involving matrix approximation, cost-preserving projections, clustering, etc. (Cohen et al., 2017), but their performance in the context of ridge regression has not been analyzed in prior work. Our work here argues that the second structural condition can be satisfied by sampling with respect to the ridge leverage scores. The number of predictor variables to be sampled depends on the degrees of freedom of the ridge-regression problem rather than the dimensions of the design matrix, and results in a relative-additive error guarantee.

### 1.2. Prior Work

In this section, we discuss our contributions in the context of the large and ever-growing body of prior work on sketching-based algorithms for regression and ridge regression. The work more closely related to ours is Chen et al. (2015), which (in our notation) returns an approximation $\widehat{\mathbf{x}}^*$ to $\mathbf{x}^*$ that satisfies (with high probability) a relative error guarantee of the form

$$\|\mathbf{x}^* - \widehat{\mathbf{x}}^*\|_2 \leq \varepsilon \|\mathbf{x}^*\|_2.$$

The running time of the proposed approach is $\mathcal{O}(\mathrm{nnz}(\mathbf{A}) + \varepsilon^{-2} n^3 \ln(n/\varepsilon))$. The proposed approach is also based on sketching $\mathbf{A}$ using RLA tools such as the count-min sketch of Clarkson & Woodruff (2013) and the sub-sampled Randomized Hadamard Transform of Ailon & Chazelle (2009); Sarlós (2006); Drineas et al. (2011). Compared to our work, notice that their dependency on $\varepsilon$ is exponentially higher: our approach has a running time that grows with $\ln(1/\varepsilon)$ whereas the above bound grows proportionally to $1/\varepsilon^2$. Additionally, our analysis can be made to depend on the degrees of freedom of the ridge-regression problem (see Theorem 2 and Section 2.1). Finally, we complement the bounds on the MSE for the response vector presented in Theorem 6 of Chen et al. (2015) with a relative-error guarantee on the MSE of the solution vector (see Theorem 4). We should also mention that prior to Chen et al. (2015); Lu et al. (2013) proposed a fast approximation algorithm for the computation of the kernel matrix using the sub-sampled randomized Hadamard transformation (SRHT).

Recently, Wang et al. (2017) presented many results on ridge-regression problems assuming $n \gg d$. In this setting, the main motivation for ridge regression is to deal with the potential ill-conditioning of the design matrix $\mathbf{A}$. Wang et al. (2017) presented sketching-based approaches that guarantee relative error approximations to the value of the objective $\mathcal{Z}^*$, as opposed to the actual solution vector. Our approach and analysis is quite different and is applicable where $d \gg n$; the results of Wang et al. (2017) do not generalize to this setting. However, recent work by Avron et al. (2017a;b) also focused on $d \gg n$: for example, Theorem 17 of Avron et al. (2017b) presents structural conditions under which the value of the objective $\mathcal{Z}^*$ can be estimated up to relative error accuracy, but no bounds are presented for the approximate solution vector. This last result seems to necessitate two structural conditions: the first one is identical to the condition of eqn. (6), but the second one is on the spectral norm of an approximate matrix product that is not needed in our analysis.

Our work was partially motivated by Pilanci & Wainwright (2016), where an iterative algorithm (the so-called Iterative Hessian Sketch) was presented for standard (*i.e.*, $\lambda = 0$), over-constrained ($n \gg d$) regression problems. Indeed, the authors provide strong motivation that clarifies the need for algorithms for regression problems whose running times depends on $\ln(1/\varepsilon)$ in order to achieve $\varepsilon$-relative-error approximations. We emphasize that the transition from standard to regularized regression problems as well as from the over- to the under-constrained case is far from trivial. Indeed, algorithms and structural results for over-constrained regression problems date back to 2006 (Drineas et al., 2006b), whereas the analogous results for ridge-regression problems appeared after 2015. Similarly, the only result that we know for under-constrained regression problems ($\lambda = 0$, $n \ll d$) appeared in Section 6.2 of Drineas et al. (2012).

Another line of research that motivated our approach was the recent introduction of ridge leverage scores (Alaoui & Mahoney, 2015; Cohen et al., 2017). Indeed, our Theorem 2 presents a structural result that can be satisfied (with high probability) by sampling columns of $\mathbf{A}$ with probabilities proportional to (exact or approximate) ridge leverage scores (see Section 2.1). The number of sampled predictor variables (columns of $\mathbf{A}$) is proportional to $\mathcal{O}(d_\lambda \ln d_\lambda)$. To the best of our knowledge, this is the first result showing a strong accuracy guarantee for ridge regression problems when the ridge leverage scores are used to sample predictor variables, in one or more iterations. We also note a recent application of ridge leverage scores (Calandriello et al., 2017a;b) where the authors presented a row sampling algorithm in order to construct a kernel sketch which is eventually used in a second-order gradient-based method for online kernel convex optimization.

In yet another relevant line of work, much research recently focused on the computation and inversion of the kernel matrix $\mathbf{A}\mathbf{A}^\top$ (or $\mathbf{A}^\top\mathbf{A}$). A number of recent papers have considered the problem of fast kernel approximation for large datasets (Zhang et al., 2015; Avron et al., 2017b; Musco & Musco, 2017; Calandriello et al., 2017c; Wang et al., 2017). However, direct comparison of the bounds presented in the aforementioned papers and our work is not straightforward, since our objective (accuracy of the approximate solution vector) is different than the objective of the above papers. In this context, there are also several recent works (Cutajar et al., 2016; Rudi et al., 2017; Ma & Belkin, 2017) that considered preconditioned gradient-based methods to develop fast and scalable approaches for approximating kernels.

Finally, Gonen et al. (2016) presented a sketching-based preconditioned SVRG approach for ridge regression problems that converges to the optimal solution in a number of iterations that depends on $\ln(1/\varepsilon)$, returning an $\varepsilon$-relative-error approximation to the objective value $\mathcal{Z}^*$. However, no such bounds were presented for the actual solution vector.

### 1.3. Notation

We use $\mathbf{a}, \mathbf{b}, \dots$ to denote vectors and $\mathbf{A}, \mathbf{B}, \dots$ to denote matrices. For a matrix $\mathbf{A}$, $\mathbf{A}_{*i}$ ($\mathbf{A}_{i*}$) denotes the $i$-th column (row) of $\mathbf{A}$ as a column (row) vector. For vector $\mathbf{a}$, $\|\mathbf{a}\|_2$ denotes its Euclidean norm; for a matrix $\mathbf{A}$, $\|\mathbf{A}\|_2$ denotes its spectral norm and $\|\mathbf{A}\|_F$ denotes its Frobenius norm. We refer the reader to Golub & Van Loan (1996) for properties of norms that will be quite useful in our work. For a matrix $\mathbf{A} \in \mathbb{R}^{n\times d}$ with $d > n$ of rank $n$, its (thin) Singular Value Decomposition (SVD) is equal to the product $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$, with $\mathbf{U} \in \mathbb{R}^{n\times n}$ (the matrix of the left singular vectors), $\mathbf{V} \in \mathbb{R}^{d\times n}$ (the matrix of the right singular vectors), and $\mathbf{\Sigma} \in \mathbb{R}^{n\times n}$ a diagonal matrix whose diagonal entries are the singular values of $\mathbf{A}$. Computation of the SVD takes, in this setting, $\mathcal{O}(n^2 d)$ time. We will use the notation $\mathbf{U}_k \in \mathbb{R}^{n\times k}$ to denote the matrix of the top $k$ left singular vectors and $\mathbf{U}_{k,\perp} \in \mathbb{R}^{n\times(n-k)}$ to denote the matrix of the bottom $n-k$ left singular vectors. We will often use $\sigma_i$ to denote the singular values of a matrix implied by context. Additional notation will be introduced as needed.

## 2. Iterative, Sketching-based Ridge Regression

Algorithm 1 iteratively computes a sequence of vectors $\widetilde{\mathbf{x}}^{(j)} \in \mathbb{R}^d$ for $j = 1, \dots, t$ and returns the estimator $\widehat{\mathbf{x}}^* = \sum_{j=1}^t \widetilde{\mathbf{x}}^{(j)}$ to the true solution vector $\mathbf{x}^*$ of eqn. (3).

In words, Algorithm 1 is quite simple: roughly, it solves ridge regression problems with the residual vector $\mathbf{b}^{(j)}$ (*i.e.*, the part of the vector $\mathbf{b}^{(j-1)}$ that was *not* captured in the previous iteration) as the new response vector for $i = 1, \dots, t$. Our main quality-of-approximation results (Theorems 1 and 2) argue that returning the *sum* of those intermediate solutions results in a highly accurate approximation to the optimal solution vector $\mathbf{x}^*$. Theorem 1 presents a quality-of-approximation result under the assumption that the sketching matrix $\mathbf{S}$ satisfies the constraint of eqn. (6).

**Algorithm 1** Iterative, sketching-based ridge regression

```
**Input:** $\mathbf{A} \in \mathbb{R}^{n\times d}$, $\mathbf{b} \in \mathbb{R}^n$, $\lambda > 0$; number of iterations $t > 0$; sketching matrix $\mathbf{S} \in \mathbb{R}^{d\times s}$;
**Initialize:** $\mathbf{b}^{(0)} \leftarrow \mathbf{b}$, $\widetilde{\mathbf{x}}^{(0)} \leftarrow \mathbf{0}_d$, $\mathbf{y}^{(0)} \leftarrow \mathbf{0}_n$;
**for** $j = 1$ **to** $t$ **do**
    $\mathbf{b}^{(j)} \leftarrow \mathbf{b}^{(j-1)} - \lambda\mathbf{y}^{(j-1)} - \mathbf{A}\widetilde{\mathbf{x}}^{(j-1)}$;
    $\mathbf{y}^{(j)} \leftarrow (\mathbf{A}\mathbf{S}\mathbf{S}^\top\mathbf{A}^\top + \lambda\mathbf{I}_n)^{-1}\mathbf{b}^{(j)}$;
    $\widetilde{\mathbf{x}}^{(j)} \leftarrow \mathbf{A}^\top\mathbf{y}^{(j)}$;
**end for**
**Output:** Approximate solution vector $\widehat{\mathbf{x}}^* = \sum_{j=1}^t \widetilde{\mathbf{x}}^{(j)}$;
```

**Theorem 1.** *Let $\mathbf{A} \in \mathbb{R}^{n\times d}$, $\mathbf{b} \in \mathbb{R}^n$, and $\lambda > 0$ be the inputs of the ridge regression problem. Assume that for some constant $0 < \varepsilon < 1$, the sketching matrix $\mathbf{S} \in \mathbb{R}^{d\times s}$ satisfies the constraint of eqn. (6). Then, the estimator $\widehat{\mathbf{x}}^*$ returned by Algorithm 1 satisfies*

$$\left\|\widehat{\mathbf{x}}^* - \mathbf{x}^*\right\|_2 \le \varepsilon^t \left\|\mathbf{x}^*\right\|_2 .$$

*Here $\mathbf{x}^*$ is the true solution of the ridge regression problem.*

Similarly, Theorem 2 presents a quality-of-approximation result under the assumption that the sketching matrix $\mathbf{S}$ satisfies the constraint of eqn. (8).

**Theorem 2.** *Let $\mathbf{A} \in \mathbb{R}^{n\times d}$, $\mathbf{b} \in \mathbb{R}^n$, and $\lambda > 0$ be the inputs of the ridge regression problem. Assume that for some constant $0 < \varepsilon < 1$, the sketching matrix $\mathbf{S} \in \mathbb{R}^{d\times s}$ satisfies the constraint of eqn. (8). Then, the estimator $\widehat{\mathbf{x}}^*$ returned by Algorithm 1 satisfies*

$$\left\|\widehat{\mathbf{x}}^* - \mathbf{x}^*\right\|_2 \le \frac{\varepsilon^t}{2}\left(\left\|\mathbf{x}^*\right\|_2 + \frac{1}{\sqrt{2\lambda}}\left\|\mathbf{U}_{k,\perp}^\top\mathbf{b}\right\|_2\right).$$

*Here, $k \in \{1, \dots, n\}$ is an integer with $\sigma_{k+1}^2 \le \lambda \le \sigma_k^2$ and $\mathbf{x}^*$ is the true solution of the ridge regression problem.*

As we have already discussed, the bound of Theorem 2 is weaker. However, the structural condition of eqn. (8) on which the above theorem depends, can be satisfied with a sketching matrix $\mathbf{S}$ whose dimensionality depends only on the degrees of freedom $d_\lambda$ of the underlying ridge regression problem, as opposed to the dimensions of the design matrix. This could result in significant savings (see Section 2.1).

Our algorithm can also be viewed as a *preconditioned Richardson iteration* (see *e.g.*, Chapter 2 of Quarteroni & Valli (1994)) for solving the linear system $(\mathbf{A}\mathbf{A}^\top + \lambda\mathbf{I}_n)\mathbf{y} = \mathbf{b}$ with pre-conditioner $\mathbf{P}^{-1} = (\mathbf{A}\mathbf{S}\mathbf{S}^\top\mathbf{A}^\top + \lambda\mathbf{I}_n)^{-1}$ and step-size equal to one. More precisely, Algorithm 1 can be formulated as

$$\bar{\mathbf{y}}^{(j)} = \bar{\mathbf{y}}^{(j-1)} + \mathbf{P}^{-1}\left(\mathbf{b} - (\mathbf{A}\mathbf{A}^\top + \lambda\mathbf{I}_n)\bar{\mathbf{y}}^{(j-1)}\right),$$

where $\bar{\mathbf{y}}^{(j)} = \sum_{k=1}^{j} \mathbf{y}^{(k)}$ (see Appendix D for the derivation). Further, subject to the structural conditions of eqns. (6) and (8), it can be shown that $\bar{\mathbf{y}}^{(t)}$ converges to the true solution $\mathbf{y}^* = (\mathbf{A}\mathbf{A}^\mathsf{T} + \lambda\mathbf{I}_n)^{-1}\mathbf{b}$ in $\mathcal{O}(\ln(1/\varepsilon))$ steps (see Appendix D) and, consequently, the output of Algorithm 1 (which can be expressed as $\widehat{\mathbf{x}}^* = \mathbf{A}^\mathsf{T}\bar{\mathbf{y}}^{(t)}$) also converges to $\mathbf{x}^* = \mathbf{A}^\mathsf{T}(\mathbf{A}\mathbf{A}^\mathsf{T} + \lambda\mathbf{I}_n)^{-1}\mathbf{b}$, the true solution of the ridge regression problem. Our analysis offers several advantages over preconditioned Richardson iteration. In our case, $\mathbf{P}^{-1}(\mathbf{A}\mathbf{A}^\mathsf{T} + \lambda\mathbf{I}_n)$ is not symmetric positive definite which, according to existing literature, implies that the convergence of Richardson's method is monotone in terms of the energy-norm induced by $\mathbf{A}\mathbf{A}^\mathsf{T} + \lambda\mathbf{I}_n$, but not the Euclidean norm (see eqn. (2.4.17) in Quarteroni & Valli (1994)). Additionally, standard convergence analysis of the Richardson iteration is with respect to $\bar{\mathbf{y}}^{(t)}$, whereas our vector of interest is $\widehat{\mathbf{x}}^*$ (which is $\bar{\mathbf{y}}^{(t)}$ premultiplied by $\mathbf{A}^\mathsf{T}$). The equality $\|\bar{\mathbf{y}}^{(t)} - \mathbf{y}^*\|_2 = \|\widehat{\mathbf{x}}^* - \mathbf{x}^*\|_2$ holds if $\mathbf{A}$ has orthonormal rows, which is not true in general.

We now discuss the running time of Algorithm 1. First, we need to compute $\mathbf{A}\widetilde{\mathbf{x}}^{(j-1)}$ which takes time $\mathcal{O}(\mathrm{nnz}(\mathbf{A}))$. Next, computing the sketch $\mathbf{AS} \in \mathbb{R}^{n\times s}$ takes $T(\mathbf{A}, \mathbf{S})$ time and depends on the particular construction of $\mathbf{S}$ (see Section 2.1). Then, in order to invert the matrix $\mathbf{\Theta} = \mathbf{A}\mathbf{S}\mathbf{S}^\mathsf{T}\mathbf{A}^\mathsf{T} + \lambda\mathbf{I}_n$ it suffices to compute the SVD of the matrix $\mathbf{AS}$. Notice that given the singular values of $\mathbf{AS}$ we can compute the singular values of $\mathbf{\Theta}$; also note that the left and right singular vectors of $\mathbf{\Theta}$ are the same as the left singular vectors of $\mathbf{AS}$. Interestingly, we do not need to compute $\mathbf{\Theta}^{-1}$: we can store it implicitly by storing its left (and right) singular vectors $\mathbf{U}_{\mathbf{\Theta}}$ and its singular values $\mathbf{\Sigma}_{\mathbf{\Theta}}$. Then, we can compute all necessary matrix-vector products using this implicit representation of $\mathbf{\Theta}^{-1}$. Thus, inverting $\mathbf{\Theta}$ takes $\mathcal{O}(sn^2)$ time. Updating the vectors $\mathbf{b}^{(j)}$, $\mathbf{y}^{(j)}$, and $\widetilde{\mathbf{x}}^{(j)}$ is dominated by the aforementioned running times, as all updates amount to just matrix-vector products. Thus, summing over all $t$ iterations, the running time of Algorithm 1 is given by

$$\mathcal{O}(t \cdot \mathrm{nnz}(\mathbf{A})) + \mathcal{O}(sn^2) + T(\mathbf{A}, \mathbf{S}). \tag{10}$$

We conclude this section by noting that our results remain valid when *different* sampling matrices $\mathbf{S}_j$ are used in each iteration $j = 1, \dots, t$, as long as they satisfy the constraints of eqns. (6) or (8). As a matter of fact, the sketching matrices $\mathbf{S}_j$ do not even need to have the same number of columns. See Section 5 for an interesting open problem in this setting.

### 2.1. Satisfying the Conditions of Eqns. (6) or (8)

The conditions of eqns. (6) and (8) essentially boil down to randomized, approximate matrix multiplication (Drineas & Kannan, 2001; Drineas et al., 2006a), a task that has received much attention in the RLA community. We start by discussing *sketching-based* approaches: a particularly useful result for our purposes appeared in Cohen et al. (2016). Using our notation, Cohen et al. (2016) proved that for $\mathbf{X} \in \mathbb{R}^{d\times n}$ and for a (suitably constructed) sketching matrix $\mathbf{S} \in \mathbb{R}^{d\times s}$, with probability at least $1-\delta$,

$$\left\|\mathbf{X}^\mathsf{T}\mathbf{S}\mathbf{S}^\mathsf{T}\mathbf{X} - \mathbf{X}^\mathsf{T}\mathbf{X}\right\|_2 \le \varepsilon\left(\|\mathbf{X}\|_2^2 + \frac{\|\mathbf{X}\|_F^2}{r}\right), \tag{11}$$

for any arbitrary $r \ge 1$. The above bound holds for a very broad family of constructions for the sketching matrix $\mathbf{S}$ (see Cohen et al. (2016) for details). In particular, Cohen et al. (2016) demonstrated a construction for $\mathbf{S}$ with $s = \mathcal{O}(r/\varepsilon^2)$ columns such that, for any $n \times d$ matrix $\mathbf{A}$, the product $\mathbf{AS}$ can be computed in time $\mathcal{O}(\mathrm{nnz}(\mathbf{A})) + \tilde{\mathcal{O}}((r^3 + r^2 n)/\varepsilon^\gamma)$ for some constant $\gamma$. Thus, starting with eqn. (6) and using this particular construction for $\mathbf{S}$, let $\mathbf{X} = \mathbf{V}$ and note that $\|\mathbf{V}\|_F^2 = n$ and $\|\mathbf{V}\|_2 = 1$. Setting $r = n$, eqn. (11) implies that

$$\left\|\mathbf{V}^\mathsf{T}\mathbf{S}\mathbf{S}^\mathsf{T}\mathbf{V} - \mathbf{I}_n\right\|_2 \le 2\,\varepsilon.$$

In this case, the running time of the sketch computation is equal to $T(\mathbf{A}, \mathbf{S}) = \mathcal{O}(\mathrm{nnz}(\mathbf{A})) + \widetilde{\mathcal{O}}(n^3/\varepsilon^\gamma)$. The running time of the overall algorithm follows from eqn. (10) and our choices for $s$ and $r$:

$$\mathcal{O}(t \cdot \mathrm{nnz}(\mathbf{A})) + \widetilde{\mathcal{O}}(n^3/\varepsilon^{\max\{2,\gamma\}}).$$

The failure probability (hidden in the polylogarithmic terms) can be easily controlled using a union bound. Finally, a simple change of variables (using $\varepsilon/4$ instead of $\varepsilon$) suffices to satisfy the structural condition of eqn. (6) without changing the above running time.

Similarly, starting with eqn. (8), let $\mathbf{X} = \mathbf{V}\mathbf{\Sigma}_\lambda$ and note that $\|\mathbf{V}\mathbf{\Sigma}_\lambda\|_F^2 = d_\lambda$ and $\|\mathbf{V}\mathbf{\Sigma}_\lambda\|_2 \le 1$. Setting $r = d_\lambda$, eqn. (11) implies that $\left\|\mathbf{\Sigma}_\lambda\mathbf{V}^\mathsf{T}\mathbf{S}\mathbf{S}^\mathsf{T}\mathbf{V}\mathbf{\Sigma}_\lambda - \mathbf{\Sigma}_\lambda^2\right\|_2 \le 2\varepsilon$. In this case, the running time of the sketch computation is equal to $T(\mathbf{A}, \mathbf{S}) = \mathcal{O}(\mathrm{nnz}(\mathbf{A})) + \tilde{\mathcal{O}}(d_\lambda^2 n/\varepsilon^\gamma)$. The running time of the overall algorithm follows from eqn. (10) and our choices for $s$ and $r$:

$$\mathcal{O}(t \cdot \mathrm{nnz}(\mathbf{A})) + \widetilde{\mathcal{O}}(d_\lambda n^2/\varepsilon^{\max\{2,\gamma\}}).$$

Again, a change of variables suffices to satisfy the structural condition of eqn. (8) without changing the running time.

We now discuss how to satisfy the conditions of eqns. (6) or (8) by *sampling*, *i.e.*, by selecting a small number of predictor variables. Towards that end, consider Algorithm 2 for the construction of the sampling-and-rescaling matrix $\mathbf{S}$.

The following theorem (see Appendix G for its proof) is of independent interest and is a strengthening of Theorem 4.2 of Holodnak & Ipsen (2015), since the sampling complexity $s$ is improved to depend only on $\|\mathbf{X}\|_F^2$ instead of the stable rank of $\mathbf{X}$ for the special case where $\|\mathbf{X}\|_2 \le 1$.[3]

[3] We do note that Theorem 3 is implicit in Cohen et al. (2017).

**Algorithm 2** Construct sampling-and-rescaling matrix

```
Input: Probabilities $p_i$, $i = 1, \dots, d$; integer $s \ll d$;
$\mathbf{S} \leftarrow \mathbf{0}_{d\times s}$;
for $j = 1$ to $s$ do
    Pick $i_j \in \{1, \dots, d\}$ with $\mathbb{P}(i_j = i) = p_i$;
    $\mathbf{S}_{i_j j} \leftarrow (s\, p_{i_j})^{-\frac{1}{2}}$;
end for
Output: Sampling-and-rescaling matrix $\mathbf{S}$;
```

**Theorem 3.** *Let $\mathbf{X} \in \mathbb{R}^{d\times n}$ with $\|\mathbf{X}\|_2 \le 1$ and let $\mathbf{S}$ be constructed by Algorithm 2 with $p_i = \|\mathbf{X}_{i*}\|_2^2 \,/\, \|\mathbf{X}\|_F^2$ for $i = 1, \dots, d$. Let $\delta$ be a failure probability and let $\varepsilon \in (0, 1]$ be an accuracy parameter. If the number of sampled columns $s$ satisfies*

$$s \ge \frac{8\,\|\mathbf{X}\|_F^2}{3\,\varepsilon^2} \ln\left(\frac{4\,(1 + \|\mathbf{X}\|_F^2)}{\delta}\right),$$

*then, with probability at least $1 - \delta$,*

$$\left\|\mathbf{X}^\mathsf{T}\mathbf{S}\mathbf{S}^\mathsf{T}\mathbf{X} - \mathbf{X}^\mathsf{T}\mathbf{X}\right\|_2 \le \varepsilon.$$

Using Theorem 3 with $\mathbf{X} = \mathbf{V}$ we can satisfy the condition of eqn. (6) by simply using the sampling probabilities $p_i = \|\mathbf{V}_{i*}\|_2^2 \,/ n$ (recall that $\|\mathbf{V}\|_F^2 = n$ and $\|\mathbf{V}\|_2 = 1$), which are the column *leverage scores* of the design matrix $\mathbf{A}$. Setting $s = \mathcal{O}(\varepsilon^{-2} n \ln n)$ suffices to satisfy the condition of eqn. (6). We note that approximate leverage scores also suffice and that their computation can be done efficiently without computing $\mathbf{V}$ (Drineas et al., 2012).

Finally, using Theorem 3 with $\mathbf{X} = \mathbf{V}\mathbf{\Sigma}_\lambda$ we can satisfy the condition of eqn. (8) using the sampling probabilities $p_i = \|(\mathbf{V}\mathbf{\Sigma}_\lambda)_{i*}\|_2^2 \,/ d_\lambda$ (recall that $\|\mathbf{V}\mathbf{\Sigma}_\lambda\|_F^2 = d_\lambda$ and $\|\mathbf{V}\mathbf{\Sigma}_\lambda\|_2 \le 1$). It is easy to see that these probabilities are proportional to the column *ridge leverage scores* of the design matrix $\mathbf{A}$ (see Lemma 21 in Appendix F). Setting $s = \mathcal{O}(\varepsilon^{-2} d_\lambda \ln d_\lambda)$ suffices to satisfy the condition of eqn. (8). We note that approximate ridge leverage scores also suffice and that their computation can be done efficiently without computing $\mathbf{V}$ (Cohen et al., 2017).

### 2.2. Bounding the MSE of $\widehat{\mathbf{x}}^*$

Consider the data-generation model

$$\mathbf{b} = \mathbf{A}\mathbf{x}_0 + \boldsymbol{\varepsilon}, \tag{12}$$

where $\mathbf{b} \in \mathbb{R}^n$ is the response vector, $\mathbf{A} \in \mathbb{R}^{n\times d}$ is the design matrix, $\mathbf{x}_0 \in \mathbb{R}^n$ is the "true" parameter vector, and $\boldsymbol{\varepsilon} \in \mathbb{R}^n$ is the noise satisfying $\mathbb{E}(\boldsymbol{\varepsilon}) = \mathbf{0}$ and $\mathbb{E}(\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^\mathsf{T}) = \sigma^2\mathbf{I}_n$, $\sigma > 0$. Then, the ridge regression estimator $\mathbf{x}^*$ of the parameter vector $\mathbf{x}_0$ can be expressed as in eqn. (3), with mean squared error (MSE) given by (see Lemma 16 in Appendix E for the derivation)

$$\begin{aligned}\mathrm{MSE}(\mathbf{x}^*) &= \sigma^2 \left\|(\mathbf{A}\mathbf{A}^\mathsf{T} + \lambda\mathbf{I}_n)^{-1}\mathbf{A}\right\|_F^2 \\ &\quad + \left\|\left(\mathbf{A}^\mathsf{T}(\mathbf{A}\mathbf{A}^\mathsf{T} + \lambda\mathbf{I}_n)^{-1}\mathbf{A} - \mathbf{I}_d\right)\mathbf{x}_0\right\|_2^2.\end{aligned} \tag{13}$$

Similarly, we can prove that the MSE of $\widehat{\mathbf{x}}^*$ for the special case where $t = 1$ in Algorithm 1 is equal to

$$\begin{aligned}\mathrm{MSE}(\widehat{\mathbf{x}}^*) &= \sigma^2 \left\|(\mathbf{A}\mathbf{S}\mathbf{S}^\mathsf{T}\mathbf{A}^\mathsf{T} + \lambda\mathbf{I}_n)^{-1}\mathbf{A}\right\|_F^2 \\ &\quad + \left\|\left(\mathbf{A}^\mathsf{T}(\mathbf{A}\mathbf{S}\mathbf{S}^\mathsf{T}\mathbf{A}^\mathsf{T} + \lambda\mathbf{I}_n)^{-1}\mathbf{A} - \mathbf{I}_d\right)\mathbf{x}_0\right\|_2^2.\end{aligned} \tag{14}$$

We present bounds on the MSE of $\widehat{\mathbf{x}}^*$ for the special case where Algorithm 1 is run for a single iteration ($t = 1$) under the assumptions of eqns. (6) or (8). Bounds for $t > 1$ (more than one iteration) are delegated to future work.

**Theorem 4.** *Let $\mathbf{A} \in \mathbb{R}^{n\times d}$ be the design matrix and let $\widehat{\mathbf{x}}^*$ be the output of Algorithm 1 for $t = 1$. If the condition of eqn. (6) is satisfied for some constant $0 < \varepsilon < 1$, then,*

$$\mathrm{MSE}(\widehat{\mathbf{x}}^*) \le (1 + 3\varepsilon\gamma_1^2)\,\mathrm{MSE}(\mathbf{x}^*),$$

*where $\gamma_1 = 1 + \frac{\sigma_1^2}{\lambda}$. If the condition of eqn. (8) is satisfied for some constant $0 < \varepsilon < 1$, then,*

$$\mathrm{MSE}(\widehat{\mathbf{x}}^*) \le (1 + 3\varepsilon\gamma_2^2)\,\mathrm{MSE}(\mathbf{x}^*),$$

*where $\gamma_2 = \max\left\{1 + \sigma_1^2/\lambda,\, \sqrt{1 + \lambda/\sigma_n^2}\right\}$.*

## 3. Sketching the Proof of Theorem 2

Due to space considerations, essentially all our proofs have been deferred to the Appendix. However, to give a flavor of the mathematical derivations underlying our contributions, we present an outline of the proof of Theorem 2, starting with the special case where Algorithm 1 is run for a single iteration ($t = 1$).

Using the quantities defined in Algorithm 1, let

$$\mathbf{x}^{*(j)} = \mathbf{A}^\mathsf{T}(\mathbf{A}\mathbf{A}^\mathsf{T} + \lambda\mathbf{I}_n)^{-1}\mathbf{b}^{(j)} \tag{15}$$

for $j = 1, \dots, t$. Notice that $\mathbf{x}^* = \mathbf{x}^{*(1)}$. Our next result expresses the intermediate vectors $\widetilde{\mathbf{x}}^{(j)}$ of Algorithm 1 in terms of the vectors $\mathbf{x}^{*(j)}$. We remind the reader that $\mathbf{U} \in \mathbb{R}^{n\times n}$ and $\mathbf{\Sigma} \in \mathbb{R}^{n\times n}$ are, respectively, the matrices of the left singular vectors and singular values of $\mathbf{A}$. We will make extensive use of the matrix $\mathbf{\Sigma}_\lambda$ defined in eqn. (7).

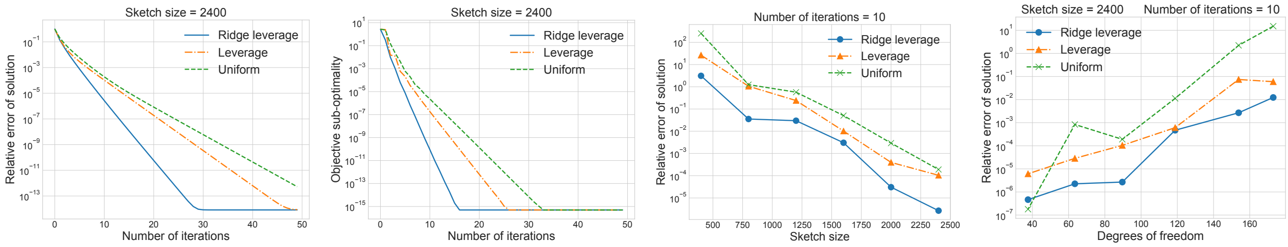
**Lemma 5.** *Let $\mathbf{A} \in \mathbb{R}^{n\times d}$, $\mathbf{b} \in \mathbb{R}^n$, and $\lambda > 0$ be the inputs of the ridge regression problem. Let $\mathbf{S} \in \mathbb{R}^{d\times s}$ be the sketching matrix and define*

$$\mathbf{E} = \mathbf{\Sigma}_\lambda\mathbf{V}^\mathsf{T}\mathbf{S}\mathbf{S}^\mathsf{T}\mathbf{V}\mathbf{\Sigma}_\lambda - \mathbf{\Sigma}_\lambda^2.$$

*If $\|\mathbf{E}\|_2 < 1$, then for all $j = 1, \dots, t$,*

$$\widetilde{\mathbf{x}}^{(j)} = \mathbf{x}^{*(j)} + \mathbf{V}\mathbf{\Sigma}_\lambda\mathbf{R}\mathbf{\Sigma}_\lambda\mathbf{\Sigma}^{-1}\mathbf{U}^\mathsf{T}\mathbf{b}^{(j)}, \tag{16}$$

*where $\mathbf{R} = \sum_{\ell=1}^{\infty}(-1)^\ell\mathbf{E}^\ell$.*



(a) Solution error vs. iterations (b) Objective error vs. iterations (c) Solution error vs. sketch size (d) Solution error vs. $d_\lambda$

*Figure 1.* Experiment results on real data (errors are on log-scale).

Now, consider the case when $t = 1$. Algorithm 1 returns $\widehat{\mathbf{x}}^* = \widetilde{\mathbf{x}}^{(1)}$; also recall that $\mathbf{x}^* = \mathbf{x}^{*(1)}$ and $\mathbf{b} = \mathbf{b}^{(1)}$. Therefore, applying Lemma 5 yields

$$\widehat{\mathbf{x}}^* = \mathbf{x}^* + \mathbf{V}\boldsymbol{\Sigma}_\lambda \mathbf{R} \boldsymbol{\Sigma}_\lambda \boldsymbol{\Sigma}^{-1}\mathbf{U}^\mathsf{T}\mathbf{b}. \tag{17}$$

Further, for any $j = 1, \ldots, t$,

$$\begin{aligned}\|\mathbf{R}\|_2 &= \Big\|\sum_{\ell=1}^{\infty}(-1)^\ell \mathbf{E}^\ell\Big\|_2 \le \sum_{\ell=1}^{\infty}\|\mathbf{E}^\ell\|_2 \le \sum_{\ell=1}^{\infty}\|\mathbf{E}\|_2^\ell \\ &\le \sum_{\ell=1}^{\infty}\left(\frac{\varepsilon}{4\sqrt{2}}\right)^\ell = \frac{\frac{\varepsilon}{4\sqrt{2}}}{1-\frac{\varepsilon}{4\sqrt{2}}} \le \frac{\varepsilon}{2\sqrt{2}}. \end{aligned}\tag{18}$$

where we used the triangle inequality, sub-multiplicativity of the spectral norm, and the fact that $\frac{\varepsilon}{4\sqrt{2}} \le \frac{1}{2}$. Now, using eqn. (17), we have

$$\begin{aligned}\|\widehat{\mathbf{x}}^* - \mathbf{x}^*\|_2 &= \|\mathbf{V}\boldsymbol{\Sigma}_\lambda \mathbf{R}\boldsymbol{\Sigma}_\lambda\boldsymbol{\Sigma}^{-1}\mathbf{U}^\mathsf{T}\mathbf{b}\|_2 \\ &\le \|\boldsymbol{\Sigma}_\lambda\|_2\|\mathbf{R}\|_2\|\boldsymbol{\Sigma}_\lambda\boldsymbol{\Sigma}^{-1}\mathbf{U}^\mathsf{T}\mathbf{b}\|_2 \\ &\le \frac{\varepsilon}{2\sqrt{2}}\,\|\boldsymbol{\Sigma}_\lambda\boldsymbol{\Sigma}^{-1}\mathbf{U}^\mathsf{T}\mathbf{b}\|_2 \\ &= \frac{\varepsilon}{2\sqrt{2}}\,\|\boldsymbol{\Sigma}_\lambda^{-1}\boldsymbol{\Sigma}_\lambda^2\boldsymbol{\Sigma}^{-1}\mathbf{U}^\mathsf{T}\mathbf{b}\|_2. \end{aligned}\tag{19}$$

where the first inequality follows from the unitary invariance and sub-multiplicativity of the spectral norm, and the second inequality is due to eqn. (18) and the fact that $\|\boldsymbol{\Sigma}_\lambda\|_2 \le 1$.

Now, let $(\boldsymbol{\Sigma}_\lambda^{-1})_k$ denote the diagonal matrix whose first $k$ diagonal entries are equal to the first $k$ diagonal entries of $\boldsymbol{\Sigma}_\lambda^{-1}$ and the bottom $n-k$ diagonal entries are set to zero. Let $(\boldsymbol{\Sigma}_\lambda^{-1})_{k,\perp} = \boldsymbol{\Sigma}_\lambda^{-1} - (\boldsymbol{\Sigma}_\lambda^{-1})_k$. Then, we have

$$\begin{aligned}\|\boldsymbol{\Sigma}_\lambda^{-1}\boldsymbol{\Sigma}_\lambda^2\boldsymbol{\Sigma}^{-1}\mathbf{U}^\mathsf{T}\mathbf{b}\|_2 &\le \underbrace{\|(\boldsymbol{\Sigma}_\lambda^{-1})_k\;\boldsymbol{\Sigma}_\lambda^2\boldsymbol{\Sigma}^{-1}\mathbf{U}^\mathsf{T}\mathbf{b}\|_2}_{\Delta_1} \\ &+ \underbrace{\|(\boldsymbol{\Sigma}_\lambda^{-1})_{k,\perp}\;\boldsymbol{\Sigma}_\lambda^2\boldsymbol{\Sigma}^{-1}\mathbf{U}^\mathsf{T}\mathbf{b}\|_2}_{\Delta_2}. \end{aligned}\tag{20}$$

where eqn. (20) follows from the triangle inequality and the fact that $\boldsymbol{\Sigma}_\lambda^{-1} = (\boldsymbol{\Sigma}_\lambda^{-1})_k + (\boldsymbol{\Sigma}_\lambda^{-1})_{k,\perp}$.

Next, we bound $\Delta_1$ and $\Delta_2$ separately using eqns. (60) and (62) in Appendix C:

$$\Delta_1 \le \sqrt{2}\,\|\mathbf{x}^*\|_2\,, \quad \Delta_2 \le \frac{1}{\sqrt{\lambda}}\left\|\mathbf{U}_{k,\perp}^\mathsf{T}\mathbf{b}\right\|_2. \tag{21}$$

Finally, combining eqns. (19), (20) and (21), we obtain

$$\begin{aligned}\|\widehat{\mathbf{x}}^* - \mathbf{x}^*\|_2 &\le \frac{\varepsilon}{2\sqrt{2}}\left(\sqrt{2}\,\|\mathbf{x}^*\|_2 + \frac{1}{\sqrt{\lambda}}\,\|\mathbf{U}_{k,\perp}^\mathsf{T}\mathbf{b}\|_2\right) \\ &= \frac{\varepsilon}{2}\left(\|\mathbf{x}^*\|_2 + \frac{1}{\sqrt{2\lambda}}\,\|\mathbf{U}_{k,\perp}^\mathsf{T}\mathbf{b}\|_2\right), \end{aligned}\tag{22}$$

which concludes the proof for the $t = 1$ case.

Interestingly, the eqn. (22) holds more generally and can be used to bound the distance between the intermediate approximate solution vectors $\widetilde{\mathbf{x}}^{(j)}$ and the intermediate true solution vectors $\mathbf{x}^{*(j)}$of eqn. (15). Indeed, for $j = 1, \ldots, t$, we have

$$\|\widetilde{\mathbf{x}}^{(j)} - \mathbf{x}^{*(j)}\|_2 \le \frac{\varepsilon}{2}\left(\|\mathbf{x}^{*(j)}\|_2 + \frac{1}{\sqrt{2\lambda}}\|\mathbf{U}_{k,\perp}^\mathsf{T}\mathbf{b}^{(j)}\|_2\right). \tag{23}$$

The next lemma (see Appendix C for its proof) presents a structural result for the optimal solution $\mathbf{x}^*$.

**Lemma 6.** *Let $\widetilde{\mathbf{x}}^{(j)}$, $j = 1, \ldots, t$ be the sequence of vectors introduced in Algorithm 1 and let $\mathbf{x}^{*(t)} \in \mathbb{R}^d$ be defined as in eqn. (15). Then,*

$$\mathbf{x}^* = \mathbf{x}^{*(t)} + \sum_{j=1}^{t-1}\widetilde{\mathbf{x}}^{(j)}, \tag{24}$$

*where $\mathbf{x}^*$ is the true solution of the ridge regression problem.*

Repeated application of eqns. (23) and (24) yields

$$\begin{aligned}\|\widehat{\mathbf{x}}^* - \mathbf{x}^*\|_2 &= \Big\|\sum_{j=1}^{t}\widetilde{\mathbf{x}}^{(j)} - \mathbf{x}^*\Big\|_2 \\ &= \Big\|\widetilde{\mathbf{x}}^{(t)} - \Big(\mathbf{x}^* - \sum_{j=1}^{t-1}\widetilde{\mathbf{x}}^{(j)}\Big)\Big\|_2 = \Big\|\widetilde{\mathbf{x}}^{(t)} - \mathbf{x}^{*(t)}\Big\|_2 \\ &\le \frac{\varepsilon}{2}\left(\|\mathbf{x}^{*(t)}\|_2 + \frac{1}{\sqrt{2\lambda}}\|\mathbf{U}_{k,\perp}^\mathsf{T}\mathbf{b}^{(t)}\|_2\right). \end{aligned}\tag{25}$$

The next bound (see Appendix C for its proof) provides a critical inequality that can be used recursively in order to establish Theorem 2.

**Lemma 7.** *Let $\mathbf{b}^{(j)}$, $j = 1, \ldots, t$, be the intermediate response vectors of Algorithm 1 and let $\mathbf{x}^{*(j)}$ be the vector defined in eqn. (15) for $j = 1, \ldots, t-1$. If the structural condition of eqn. (8) is satisfied, then*

$$\|\mathbf{x}^{*(j+1)}\|_2 + \frac{1}{\sqrt{2\lambda}}\|\mathbf{U}_{k,\perp}^{\mathsf{T}}\mathbf{b}^{(j+1)}\|_2 \leq \varepsilon\left(\|\mathbf{x}^{*(j)}\|_2 + \frac{1}{\sqrt{2\lambda}}\|\mathbf{U}_{k,\perp}^{\mathsf{T}}\mathbf{b}^{(j)}\|_2\right). \quad (26)$$

Applying eqn. (26) iteratively, we obtain

$$\begin{aligned}\|\mathbf{x}^{*(t)}\|_2 &+ \frac{1}{\sqrt{2\lambda}}\|\mathbf{U}_{k,\perp}^{\mathsf{T}}\mathbf{b}^{(t)}\|_2 \\ &\leq \varepsilon\left(\|\mathbf{x}^{*(t-1)}\|_2 + \frac{1}{\sqrt{2\lambda}}\|\mathbf{U}_{k,\perp}^{\mathsf{T}}\mathbf{b}^{(t-1)}\|_2\right) \\ &\leq \cdots \leq \varepsilon^{t-1}\left(\|\mathbf{x}^{*}\|_2 + \frac{1}{\sqrt{2\lambda}}\|\mathbf{U}_{k,\perp}^{\mathsf{T}}\mathbf{b}\|_2\right). \quad (27)\end{aligned}$$

Finally, combining eqns. (25) and (27), we conclude that

$$\|\widehat{\mathbf{x}}^* - \mathbf{x}^*\|_2 \leq \frac{\varepsilon^t}{2}\left(\|\mathbf{x}^*\|_2 + \frac{1}{\sqrt{2\lambda}}\|\mathbf{U}_{k,\perp}^{\mathsf{T}}\mathbf{b}\|_2\right). \quad (28)$$

## 4. Empirical Evaluation

We perform experiments on the ARCENE dataset (Guyon et al., 2005) from the UCI repository (Lichman, 2013). The design matrix contains 200 samples with $10,000$ real-valued features; we normalize the entries to be within the interval $[0, 1]$. The response vector consists of $\pm 1$ labels. We also perform experiments on synthetic data generated as in Chen et al. (2015); see Appendix H for details.

In our experiments, we compare three different choices of sampling probabilities: selecting columns (*i*) uniformly at random, (*ii*) proportional to their leverage scores, or (*iii*) proportional to their ridge leverage scores. For each sampling method, we run Algorithm 1 for 50 iterations with a variety of sketch sizes, and measure (*i*) the relative error of the solution vector $\frac{\|\widehat{\mathbf{x}}^* - \mathbf{x}^*\|_2}{\|\mathbf{x}^*\|_2}$, where $\mathbf{x}^*$ is the true optimal solution and (*ii*) the objective sub-optimality $\frac{f(\widehat{\mathbf{x}}^*)}{f(\mathbf{x}^*)} - 1$, where $f(\mathbf{x}) = \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_2^2$ is the objective function for the ridge-regression problem.

The results are shown in Figure 1. Figures 1a and 1b plot the relative error of the solution vector and the objective sub-optimality (for a fixed sketch size) as the iterative algorithm progresses. Figure 1c plots the relative error of the solution with respect to varying sketch sizes (the plots for objective sub-optimality are analogous and thus omitted). We observe that both the solution error and the objective sub-optimality decay *exponentially* as our iterative algorithm progresses.[4]

Next, we show that the approximation quality depends directly on the *degrees of freedom* $d_\lambda$ of the ridge-regression problem (eqn. (4)), rather than the dimensions of the design matrix. To this end, we keep the design matrix unchanged ($n$ remains fixed), and vary the regularization parameter $\lambda \in \{1, 2, 5, 10, 20, 50\}$. Figure 1d plots the relative solution error against the degrees of freedom $d_\lambda$ (for a fixed sketch size and number of iterations); we observe that the relative error decreases roughly exponentially as $d_\lambda$ decreases (as $\lambda$ increases). Thus, the sketch size or number of iterations necessary to achieve a certain precision in the solution also decreases with $d_\lambda$, even though $n$ remains fixed.

## 5. Conclusion and Open Problems

We have presented simple structural results that guarantee high-quality approximations to the optimal solution vector of ridge regression. In particular, our second structural result presents the first accuracy analysis for ridge regression when the ridge leverage scores are used to sample predictor variables. The sample size depends on the degrees of freedom of the ridge regression problem and not the dimensions of the design matrix. An obvious open problem is to either improve the sample size or present lower bounds showing that our bounds are tight. Additionally, the results of Theorem 4 should be generalized to cover the $t > 1$ case.

Finally, an interesting open problem would be to investigate whether the use of different sampling matrices in each iteration of Algorithm 1 (*i.e.*, introducing new "randomness" in each iteration) could lead to *provably* improved bounds for our main theorems. We conjecture that this is indeed the case, and we present further experiment results in Appendix H which support our conjecture. In particular, the results show that using a newly sampled sketching matrix at every iteration enables faster convergence as the iterations progress, and also reduces the minimum sketch size necessary for Algorithm 1 to converge.

**Acknowledgements.** We thank an anonymous reviewer for pointing out the connection between our method and the preconditioned Richardson iteration. AC and PD were partially supported by NSF IIS-1661760 and IIS-1661756. JY was supported by NSF IIS-1149789 and IIS-1618690.

## References

Ailon, N. and Chazelle, B. The fast Johnson–Lindenstrauss transform and approximate nearest neighbors. *SIAM Journal on Computing*, 39(1):302–322, 2009.

Alaoui, A. E. and Mahoney, M. W. Fast randomized kernel ridge regression with statistical guarantees. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, pp. 775–783, 2015.

[4] For these experiments, we have set the regularization parameter $\lambda = 10$ in the ridge regression objective as well as when computing the ridge leverage score sampling probabilities.

Avron, H., Clarkson, K. L., and Woodruff, D. P. Sharper bounds for regularized data fitting. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pp. 27:1–27:22, 2017a.

Avron, H., Clarkson, K. L., and Woodruff, D. P. Faster kernel ridge regression using sketching and preconditioning. *SIAM Journal on Matrix Analysis and Applications*, 38 (4):1116–1138, 2017b.

Calandriello, D., Lazaric, A., and Valko, M. Second-order kernel online convex optimization with adaptive sketching. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 645–653, 2017a.

Calandriello, D., Lazaric, A., and Valko, M. Efficient second-order online kernel learning with adaptive embedding. In *Advances in Neural Information Processing Systems 30*, pp. 6142–6151. 2017b.

Calandriello, D., Lazaric, A., and Valko, M. Distributed adaptive sampling for kernel matrix approximation. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pp. 1421–1429, 2017c.

Chen, S., Liu, Y., Lyu, M. R., King, I., and Zhang, S. Fast relative-error approximation algorithm for ridge regression. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, pp. 201–210, 2015.

Clarkson, K. L. and Woodruff, D. P. Low rank approximation and regression in input sparsity time. In *Proceedings of the 45th annual ACM symposium on Symposium on Theory of Computing*, pp. 81, 2013.

Cohen, M. B., Nelson, J., and Woodruff, D. P. Optimal approximate matrix product in terms of stable rank. In *43rd International Colloquium on Automata, Languages, and Programming*, pp. 11:1–11:14, 2016.

Cohen, M. B., Musco, C., and Musco, C. Input sparsity time low-rank approximation via ridge leverage score sampling. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1758–1777, 2017.

Cutajar, K., Osborne, M. A., Cunningham, J. P., and Filippone, M. Preconditioning kernel matrices. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning*, pp. 2529–2538, 2016.

Drineas, P. and Kannan, R. Fast monte-carlo algorithms for approximate matrix multiplication. In *Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science*, pp. 452–459, 2001.

Drineas, P. and Mahoney, M. W. RandNLA: Randomized numerical linear algebra. *Communications of the ACM*, 59(6):80–90, 2016.

Drineas, P., Kannan, R., and Mahoney, M. W. Fast monte carlo algorithms for matrices I: Approximating matrix multiplication. *SIAM Journal on Computing*, 36(1):132–157, 2006a.

Drineas, P., Mahoney, M. W., and Muthukrishnan, S. Sampling algorithms for $\ell_2$ regression and applications. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1127–1136, 2006b.

Drineas, P., Mahoney, M. W., Muthukrishnan, S., and Sarlós, T. Faster least squares approximation. *Numerische Mathematik*, 117:219–249, 2011.

Drineas, P., Magdon-Ismail, M., Mahoney, M. W., and Woodruff, D. P. Fast approximation of matrix coherence and statistical leverage. *Journal of Machine Learning Research*, 13(1):3475–3506, 2012.

Golub, G. H. and Van Loan, C. F. *Matrix Computations*. Johns Hopkins University Press, 1996.

Gonen, A., Orabona, F., and Shalev-Shwartz, S. Solving ridge regression using sketched preconditioned SVRG. In *Proceedings of the 33nd International Conference on Machine Learning*, pp. 1397–1405, 2016.

Gunst, R. F. and Mason, R. L. Biased estimation in regression: An evaluation using mean squared error. *Journal of the American Statistical Association*, 72(359):616–628, 1977.

Guyon, I., Gunn, S., Ben-Hur, A., and Dror, G. Result analysis of the NIPS 2003 feature selection challenge. In *Advances in Neural Information Processing Systems 17*, pp. 545–552. 2005.

Hoerl, A. E. and Kennard, R. W. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12:55–67, 1970.

Holodnak, J. T. and Ipsen, I. C. F. Randomized approximation of the gram matrix: Exact computation and probabilistic bounds. *SIAM Journal on Matrix Analysis and Applications*, 36(1):110–137, 2015.

Kyng, R. *Approximate Gaussian Elimination*. Ph.D Thesis, Yale University, 2017.

Lichman, M. UCI machine learning repository, 2013. URL http://archive.ics.uci.edu/ml.

Lu, Y., Dhillon, P., Foster, D. P., and Ungar, L. Faster ridge regression via the subsampled randomized hadamard transform. In *Advances in Neural Information Processing Systems 26*, pp. 369–377. 2013.

Ma, S. and Belkin, M. Diving into the shallows: a computational perspective on large-scale shallow learning. In *Advances in Neural Information Processing Systems 30*, pp. 3778–3787. 2017.

Mahoney, M. W. *Randomized algorithms for matrices and data*. Foundations and Trends in Machine Learning. 2011.

Mahoney, M. W. and Drineas, P. CUR matrix decompositions for improved data analysis. *Proceedings of the National Academy of Sciences of the United States of America*, 106(3), 2009.

Musco, C. and Musco, C. Recursive sampling for the nystrom method. In *Advances in Neural Information Processing Systems 30*, pp. 3836–3848. 2017.

Pilanci, M. and Wainwright, M. J. Iterative Hessian sketch: Fast and accurate solution approximation for constrained least-squares. *Journal of Machine Learning Research*, 17 (53):1–38, 2016.

Quarteroni, A. M. and Valli, A. *Numerical Approximation of Partial Differential Equations*. Springer, 1994.

Rudi, A., Carratino, L., and Rosasco, L. Falkon: An optimal large scale kernel method. In *Advances in Neural Information Processing Systems 30*, pp. 3888–3898. 2017.

Sarlós, T. Improved approximation algorithms for large matrices via random projections. In *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science*, pp. 143–152, 2006.

Saunders, C., Gammerman, A., and Vovk, V. Ridge regression learning algorithm in dual variables. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pp. 515–521, 1998.

Tropp, J. A. An introduction to matrix concentration inequalities. *Foundations and Trends in Machine Learning*, 8(1-2):1–230, 2015.

Wang, S., Gittens, A., and Mahoney, M. W. Sketched ridge regression: Optimization perspective, statistical perspective, and model averaging. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 3608–3616, 2017.

Woodruff, D. P. Sketching as a tool for numerical linear algebra. *Foundations and Trends in Theoretical Computer Science*, 10(1-2):1–157, 2014.

Zhan, X. Singular values of differences of positive semidefinite matrices. *SIAM Journal on Matrix Analysis and Applications*, 22(3):819–823, 2001.

Zhang, Y., Duchi, J., and Wainwright, M. Divide and conquer kernel ridge regression: A distributed algorithm with minimax optimal rates. *Journal of Machine Learning Research*, 16(1):3299–3340, 2015.