Electronic Journal of Statistics
Vol. 11 (2017) 4896–4944
ISSN: 1935-7524
https://doi.org/10.1214/17-EJS1334SI

Sketching meets random projection in the dual: A provable recovery algorithm for big and high-dimensional data

Jialei Wang

Department of Computer Science, University of Chicago, Chicago, IL 60637
e-mail: jialei@uchicago.edu

Jason D. Lee

Marshall School of Business, University of Southern California, Los Angeles, CA 90089
e-mail: jasonlee@marshall.usc.edu

Mehrdad Mahdavi

Computer Science and Engineering, Pennsylvania State University, State College, PA 16801
e-mail: mahdavi@cse.psu.edu

Mladen Kolar

Booth School of Business, University of Chicago, Chicago, IL 60637
e-mail: mkolar@chicagobooth.edu

and

Nathan Srebro

Toyota Technological Institute at Chicago, Chicago, IL 60637
e-mail: nati@ttic.edu

Abstract: Sketching techniques scale up machine learning algorithms by reducing the sample size or dimensionality of massive data sets, without sacrificing their statistical properties. In this paper, we study sketching from an optimization point of view. We first show that the iterative Hessian sketch is an optimization process with *preconditioning* and develop an *accelerated* version using this insight together with conjugate gradient descent. Next, we establish a primal-dual connection between the Hessian sketch and dual random projection, which allows us to develop an *accelerated* iterative dual random projection method by applying the preconditioned conjugate gradient descent on the dual problem. Finally, we tackle the problems of large sample size and high-dimensionality in massive data sets by developing the *primal-dual sketch*. The primal-dual sketch iteratively sketches the primal and dual formulations and requires only a logarithmic number of calls to solvers of small sub-problems to recover the optimum of the original problem up to *arbitrary* precision. Our iterative sketching techniques can also be applied for solving distributed optimization problems where data are partitioned by samples or features. Experiments on synthetic and real data sets complement our theoretical results.

MSC 2010 subject classifications: 62H12, 68T05, 90C06.

Keywords and phrases: Iterative Hessian sketch, dual random projection, preconditioning, primal-dual conversion, acceleration, conjugate gradient, primal-dual sketch.

Received June 2017.

Contents

1. Introduction

Machine learning is nowadays successfully applied to massive data sets collected from various domains. One of the major challenges in applying machine learning methods to massive data sets is how to effectively utilize available computational resources when building predictive and inferential models, while utilizing data in a statistically optimal way. One approach to tackling massive data sets is via building distributed computer systems and developing distributed learning algorithms. However, distributed systems may not always be available. Furthermore, the cost of running a distributed system can be much higher than one can afford, making distributed learning unsuitable for all scenarios. An alternative approach is to use the state-of-the-art randomized optimization algorithms to accelerate the training process. For example, many optimization algorithms are available for solving regularized empirical risk minimization problems, with provably fast convergence and low computational cost per iteration (see [16, 51, 7] for examples). It is worth pointing out at this point that the speed of these optimization methods still heavily depends on the condition number of the problem at hand, which is undesirable for many real world problems.

Sketching has emerged as a technique for big data analytics [46]. The idea behind sketching is to approximate the solution of the original problem by solving a sketched, smaller scale problem. For example, sketching has been used to *approximately* solve various large-scale problems, ranging from least square regression and robust regression to low-rank approximation and singular value decomposition (see [12, 23, 21, 2, 46, 34, 48, 29, 30, 9] and references therein), and has been implemented in high-quality software packages of least-square solvers [3, 25]. However, one major drawback of sketching is that it is typically not suitable in scenarios where a *highly accurate* solution is needed. To obtain a solution with exponentially smaller approximation error, we often also need to increase the sketching dimension *exponentially* as well.

Recent work on "iterative sketch", namely iterative Hessian sketch (IHS) [33] and iterative dual random projection (IDRP) [53], has improved the situation. These methods are able to refine the accuracy of their solution by iteratively solving small scale sketched problem. Hessian sketch [33] is designed to only reduce the sample size of the original problem, while dual random projection [53] is proposed to only reduce the dimensionality of data. As a consequence, when the sample size and feature dimension are both large, IHS and IDRP still need to solve relatively large-scale subproblems as they can only sketch the problem from one perspective.

In this paper, we address the problem of the *recovery* of optimal solution for big and high-dimensional data by solving small sketched problems of original problem. We make the following contributions. First, we propose an accelerated version of IHS that is computationally as effective as IHS at each iteration, but requires provably fewer number of sketching iterations to reach a certain accuracy. Next, we reveal a primal-dual connection between IHS [33] and IDRP [53], that were independently proposed by two different groups of researchers. We show that these two methods are equivalent in the sense that the dual random projection is essentially performing the Hessian sketch in the dual space. This connection allows us to provide a unified analysis of IHS and IDRP, and also develop an accelerated iterative sketching schema. Finally, we alleviate the computational issues raised by big and high-dimensional learning problems. We propose a *primal-dual sketching* method that can simultaneously reduce the sample size and dimension of the problem, and recover the optimal solution to the original large-scale high-dimensional problem with provable convergence guarantees. We also demonstrate applicability of the iterative sketching techniques for the distributed optimization problems where the data are partitioned across machines, either by samples or features.

Organization The rest of this paper is organized as follows: in Section 2 we review the iterative Hessian sketch as an optimization process and propose a new algorithm with faster convergence rate. In Section 3 we show that the dual random projection is equivalent to the Hessian sketch. This equivalence allows us to propose the corresponding accelerated dual random projection. In Section 4 we combine the sketching from both primal and dual perspectives, and propose an iterative algorithm that reduces both sample size and problem dimension. Theoretical properties of are investigated in Section 5, while technical details are deferred to Appendix. In Section 6 we discuss an application of the iterative sketching for distributed optimization. We present experiments in Section 7 to support our theoretical results, while Section 8 provides a final summary and a discussion of several future directions.

Notation We use bold-faced letters, such as $\mathbf{w}$, to denote vectors, and bold-faced capital letters, such as $\mathbf{X}$, to denote matrices. The set of real numbers is denoted by $\mathbb{R}$. Given a matrix $\mathbf{X} \in \mathbb{R}^{n\times p}$, we define the following matrix induced norm for any vector $\mathbf{w} \in \mathbb{R}^p$,

$$\|\mathbf{w}\|_{\mathbf{X}} = \sqrt{\frac{\mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w}}{n}}.$$

We use $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ to denote the multivariate normal distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. We use $\mathbf{I}_n$ to denote the identity matrix of size $n \times n$. The maximum and minimum eigenvalues of $\mathbf{H}$ are $\lambda_{\max}(\mathbf{H})$ and $\lambda_{\min}(\mathbf{H})$, respectively. The condition number of a matrix $\mathbf{H}$ is denoted by $\kappa(\mathbf{H})$, which is the ratio of the largest to smallest singular value in the singular value decomposition of $\mathbf{H}$. For two sequences $\{a_n\}_{n=1}^\infty$ and $\{a_n\}_{n=1}^\infty$, we denote $a_n \lesssim b_n$ if $a_n \leq Cb_n$ always holds for n large enough with some constant C, and denote $a_n \gtrsim b_n$ if $b_n \lesssim a_n$. We also use the notation $a_n = \mathcal{O}(b_n)$ if $a_n \lesssim b_n$, and use $\widetilde{\mathcal{O}}(\cdot)$ for $\mathcal{O}(\cdot)$ to hide logarithmic factors.

2. Iterative Hessian sketch as optimization with preconditioning

In this section, we first review the iterative Hessian sketch proposed in [33]. We present the iterative Hessian sketch as an iterative preconditioned optimization process. This allows us to propose a faster iterative algorithm by solving a different sketched problem.

Consider the following ℓ_2 regularized least-squares problem, also known as the ridge regression:

$$\min_{\mathbf{w}\in\mathbb{R}^p} P(\mathbf{X}, \mathbf{y}; \mathbf{w}) = \min_{\mathbf{w}\in\mathbb{R}^p} \frac{1}{2n} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|_2^2 + \frac{\lambda}{2} \|\mathbf{w}\|_2^2 . \tag{2.1}$$

where $\mathbf{X} \in \mathbb{R}^{n\times p}$ is the data matrix, $\mathbf{y} \in \mathbb{R}^n$ is the response vector, and λ is the tuning parameter. Let $\mathbf{w}^*$ denote the optimum of problem (2.1) which can be computed in a closed form as

$$\mathbf{w}^* = \left(\lambda \mathbf{I}_p + \frac{\mathbf{X}^\top \mathbf{X}}{n}\right)^{-1} \frac{\mathbf{X}^\top \mathbf{y}}{n},$$

however, to compute the closed-form solution requires one to construct and invert the covariance matrix, which can take $\mathcal{O}(np^2 + p^3)$ time to finish.

Sketching has become a widely used technique for efficiently finding an approximate solution to (2.1) when both n and p are large [10, 23, 46]. To avoid solving a problem of huge sample size, the traditional sketching techniques (for example, [38, 31]) reduce the sample size from n to m, with $m \ll n$, and solve the following sketched ℓ_2 regularized least-squares problem:

$$\min_{\mathbf{w}\in\mathbb{R}^p} P(\mathbf{\Pi}^\top \mathbf{X}, \mathbf{\Pi}^\top \mathbf{y}; \mathbf{w}) = \min_{\mathbf{w}\in\mathbb{R}^p} \frac{1}{2n} \left\|\mathbf{\Pi}^\top \mathbf{y} - \mathbf{\Pi}^\top \mathbf{X}\mathbf{w}\right\|_2^2 + \frac{\lambda}{2} \left\|\mathbf{w}\right\|_2^2, \tag{2.2}$$

where $\mathbf{\Pi} \in \mathbb{R}^{n\times m}$ is a sketching matrix. The problem (2.2) can be solved faster and with less storage as long as we can choose $m \ll n$. Typical choice of $\mathbf{\Pi}$ includes a random matrix with Gaussian or Rademacher entries, sub-sampled randomized Hadamard transform [4], and sub-sampled Randomized Fourier Transform [36]. See discussions in Section 2.1 of [33] for more details.

Though the classical sketching has been successful in various problems and has provable guarantees, as shown in [33], there is an approximation precision limit for classical sketching methods could achieve, given a fixed sketching dimension. To obtain an approximate solution with high precision, the sketching dimension m often needs to be of the same order as n. This is impractical as the goal of sketching is to speed up the algorithms via reducing the sample size.

The main idea behind the Hessian sketch [33] is based on the following equivalent formulation of (2.1):

$$\min_{\mathbf{w}\in\mathbb{R}^p} P(\mathbf{X}, \mathbf{y}; \mathbf{w}) = \min_{\mathbf{w}\in\mathbb{R}^p} \frac{1}{2n} \left\|\mathbf{y}\right\|_2^2 + \frac{1}{2n} \left\|\mathbf{X}\mathbf{w}\right\|_2^2 - \frac{1}{n}\langle \mathbf{y}, \mathbf{X}\mathbf{w}\rangle + \frac{\lambda}{2} \left\|\mathbf{w}\right\|_2^2. \tag{2.3}$$

In the Hessian sketch one only sketches the quadratic part $\left\|\mathbf{X}\mathbf{w}\right\|_2^2$ with respect to $\mathbf{X}$, but not the linear part $\langle \mathbf{y}, \mathbf{X}\mathbf{w}\rangle$, leading to the following problem:

$$\min_{\mathbf{w}\in\mathbb{R}^p} P_{\mathrm{HS}}(\mathbf{X}, \mathbf{y}; \mathbf{\Pi}, \mathbf{w}) = \min_{\mathbf{w}\in\mathbb{R}^p} \frac{1}{2n} \left\|\mathbf{y}\right\|_2^2 + \frac{1}{2n} \left\|\mathbf{\Pi}^\top \mathbf{X}\mathbf{w}\right\|_2^2 - \frac{1}{n}\langle \mathbf{y}, \mathbf{X}\mathbf{w}\rangle + \frac{\lambda}{2} \left\|\mathbf{w}\right\|_2^2. \tag{2.4}$$

The solution to the problem (2.4) has the following closed form solution:

$$\widehat{\mathbf{w}}_{\mathrm{HS}} = \left(\lambda \mathbf{I}_p + \frac{\mathbf{X}^\top \mathbf{\Pi}\mathbf{\Pi}^\top \mathbf{X}}{n}\right)^{-1} \frac{\mathbf{X}^\top \mathbf{y}}{n}. \tag{2.5}$$

Algorithm 1: Iterative Hessian Sketch (IHS).

```
Input: Data $\mathbf{X}, \mathbf{y}$, sketching matrix $\mathbf{\Pi}$.
Initialization: $\widehat{\mathbf{w}}_{\mathrm{HS}}^{(0)} = \mathbf{0}$.
for $t = 0, 1, 2, \ldots$ do
    Update the approximation by $\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)} = \widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)} + \widehat{\mathbf{u}}^{(t)}$, where $\widehat{\mathbf{u}}^{(t)}$ is obtained by
    solving the sketched problem (2.6).
end
```

Compared to the classical sketch where both the data matrix $\mathbf{X}$ and the response vector $\mathbf{y}$ are sketched, in the Hessian sketch one only sketches the Hessian matrix, through the following transform:

$$\mathbf{X}^\top \mathbf{X} \to \mathbf{X}^\top \mathbf{\Pi}\mathbf{\Pi}^\top \mathbf{X}.$$

The Hessian sketch suffers from the same approximation limit as the classical sketch. However, one notable feature of the Hessian sketch is that one can implement an iterative extension to refine the accuracy of the approximation. Define the initial Hessian sketch approximation as $\widehat{\mathbf{w}}_{\mathrm{HS}}^{(1)}$:

$$\widehat{\mathbf{w}}_{\mathrm{HS}}^{(1)} = \arg\min_{\mathbf{w}} \mathbf{w}^\top \left(\frac{\mathbf{X}^\top \mathbf{\Pi}\mathbf{\Pi}^\top \mathbf{X}}{2n} + \frac{\lambda}{2}\mathbf{I}_p \right) \mathbf{w} - \frac{1}{n}\langle \mathbf{y}, \mathbf{X}\mathbf{w} \rangle.$$

A refinement of $\widehat{\mathbf{w}}_{\mathrm{HS}}^{(1)}$ can be obtained by considering the following optimization problem

$$\begin{aligned}
&\arg\min_{\mathbf{u}} \frac{1}{2n} \left\| \mathbf{y} - \mathbf{X}(\mathbf{u} + \widehat{\mathbf{w}}_{\mathrm{HS}}^{(1)}) \right\|_2^2 + \frac{\lambda}{2} \left\| (\mathbf{u} + \widehat{\mathbf{w}}_{\mathrm{HS}}^{(1)}) \right\|_2^2 \\
&= \arg\min_{\mathbf{u}} \mathbf{u}^\top \left(\frac{\mathbf{X}^\top \mathbf{X}}{2n} + \frac{\lambda}{2}\mathbf{I}_p \right) \mathbf{u} - \left\langle \frac{\mathbf{X}^\top (\mathbf{y} - \mathbf{X}\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)})}{n} - \lambda \widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)}, \mathbf{u} \right\rangle,
\end{aligned}$$

whose optimum is $\mathbf{w}^* - \widehat{\mathbf{w}}_{\mathrm{HS}}^{(1)}$. The main idea of the iterative Hessian sketch is to approximate the residual solution $\mathbf{w}^* - \widehat{\mathbf{w}}_{\mathrm{HS}}^{(1)}$ by the Hessian sketch. At iteration t, $\mathbf{w}^* - \widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)}$ is approximated by $\widehat{\mathbf{u}}^{(t)}$ that minimizes the following problem

$$\widehat{\mathbf{u}}^{(t)} = \arg\min_{\mathbf{u}} \mathbf{u}^\top \left(\frac{\mathbf{X}^\top \mathbf{\Pi}\mathbf{\Pi}^\top \mathbf{X}}{2n} + \frac{\lambda}{2}\mathbf{I}_p \right) \mathbf{u} - \left\langle \frac{\mathbf{X}^\top (\mathbf{y} - \mathbf{X}\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)})}{n} - \lambda \widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)}, \mathbf{u} \right\rangle, \tag{2.6}$$

and the new approximation $\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}$ is updated as

$$\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)} = \widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)} + \widehat{\mathbf{u}}^{(t)}.$$

The algorithm for IHS is shown in Algorithm 1. Since (2.6) is a sketched problem with sample size m, it can be solved more efficiently than the original problem (2.1). Notice that we can reuse the previously sketched data matrix $\mathbf{\Pi}^\top \mathbf{X}$ without constructing any new random sketching matrices. [33] showed that the approximation error of IHS is exponentially decreasing with the number of sketching iterations. Thus IHS can find an approximate solution with an ϵ-approximation error within $\mathcal{O}(\log(1/\epsilon))$ iterations, as long as the sketching dimension m is large enough. IHS was originally developed for the least-squares problem in (2.1), the idea can be extended to solve more general problems, such as constrained least-squares [33], optimization with self-concordant loss [32], as well as non-parametric methods [49].

Though IHS improved the classical sketching by enabling us to find a high quality approximation more efficiently, it is imperfect due to the following two reasons. First, the guarantee that the approximation error decreases exponentially for IHS relies on the sketching dimension being large enough. The necessary sketching dimension depends on the intrinsic complexity of the problem, and, if the sketching dimension is too small, IHS can diverge, obtaining arbitrary worse approximation. Second, even when the sketching dimension is large enough, the speed at which the approximation error decreases in IHS can be significantly improved.

Now, we show that the iterative Hessian sketch is in fact an optimization process with *preconditioning*. This view allows us to develop better iterative algorithms by searching the conjugate directions. For notation simplicity, let

$$\mathbf{H} = \frac{\mathbf{X}^\top \mathbf{X}}{n} + \lambda \mathbf{I}_p \quad \text{and} \quad \widetilde{\mathbf{H}} = \frac{\mathbf{X}^\top \mathbf{\Pi}\mathbf{\Pi}^\top \mathbf{X}}{n} + \lambda \mathbf{I}_p.$$

Let

$$\nabla P(\mathbf{w}) = -\frac{\mathbf{X}^\top(\mathbf{y} - \mathbf{X}\mathbf{w})}{n} + \lambda \mathbf{w}$$

denote the gradient of $P(\mathbf{X}, \mathbf{y}; \mathbf{w})$ with respect to $\mathbf{w}$. Then the IHS algorithm can be seen as performing the following iterative update

$$\widehat{\mathbf{w}}_{\text{HS}}^{(t+1)} = \widehat{\mathbf{w}}_{\text{HS}}^{(t)} - \widetilde{\mathbf{H}}^{-1} \nabla P(\widehat{\mathbf{w}}_{\text{HS}}^{(t)}),$$

which is like a Newton update where we replace the true Hessian $\mathbf{H}$ with the sketched Hessian $\widetilde{\mathbf{H}}$. Another way to think about this update is via the change of variable $\mathbf{z} = \widetilde{\mathbf{H}}^{1/2}\mathbf{w}$ and then applying the gradient descent in the $\mathbf{z}$ space

$$\widehat{\mathbf{z}}^{(t+1)} = \widehat{\mathbf{z}}^{(t)} - \nabla_{\mathbf{z}} P(\widetilde{\mathbf{H}}^{-1/2}\widehat{\mathbf{z}}^{(t)}) = \widehat{\mathbf{z}}^{(t)} - \widetilde{\mathbf{H}}^{-1/2} \nabla_{\mathbf{x}} P(\widetilde{\mathbf{H}}^{-1/2}\widehat{\mathbf{z}}^{(t)}).$$

Multiplying by $\widetilde{\mathbf{H}}^{-1/2}$, changes the update back to the original space, leading back to the IHS update

$$\widehat{\mathbf{w}}_{\text{HS}}^{(t+1)} = \widehat{\mathbf{w}}_{\text{HS}}^{(t)} - \widetilde{\mathbf{H}}^{-1} \nabla P(\widehat{\mathbf{w}}_{\text{HS}}^{(t)}).$$

With above discussion, we see that the iterative Hessian sketch is in fact an optimization process with the sketched Hessian as preconditioning.

2.1. Accelerated IHS via preconditioned conjugate gradient

In this section, we present the accelerated iterative Hessian sketch (Acc-IHS) algorithm by utilizing the idea of preconditioned conjugate gradient. Conjugate gradient is known to have better convergence properties than gradient descent in solving linear systems [15, 28]. Since the iterative Hessian sketch is performing the gradient descent (with stepsize 1) in the transformed space $\mathbf{z} = \widetilde{\mathbf{H}}^{1/2}\mathbf{w}$, it can be accelerated by performing the conjugate gradient descent instead. Equivalently, we can implicitly transform the space by defining inner product as $\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^\top \widetilde{\mathbf{H}}\mathbf{y}$.

This leads to the algorithm Acc-IHS as detailed in Algorithm 2. At each iteration, the solver is called for the following sketched linear system:

$$\widehat{\mathbf{u}}^{(t)} = \arg\min_{\mathbf{u}} \mathbf{u}^\top \left(\frac{\mathbf{X}^\top \mathbf{\Pi}\mathbf{\Pi}^\top \mathbf{X}}{2n} + \frac{\lambda}{2}\mathbf{I}_p \right) \mathbf{u} - \left\langle \mathbf{r}^{(t)}, \mathbf{u} \right\rangle . \tag{2.7}$$

Unlike IHS, which uses $\widetilde{\mathbf{H}}^{-1}\nabla P(\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)})$ as the update direction at iteration t, Acc-IHS uses $\mathbf{p}^{(t)}$ as the update direction where $\mathbf{p}^{(t)}$ is chosen to satisfy the conjugate condition: $\forall t_1, t_2 \geq 0, t_1 \neq t_2$

$$\left(\mathbf{p}^{(t_1)}\right)^\top \widetilde{\mathbf{H}}^{-1/2}\mathbf{H}\widetilde{\mathbf{H}}^{-1/2}\mathbf{p}^{(t_2)} = 0.$$

Since the updating direction is conjugate to the previous directions, it is guaranteed that after p iterations we reach the exact minimizer, that is,

$$\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)} = \mathbf{w}^*, \quad \forall t \geq p.$$

Moreover, Acc-IHS has the same computational cost as the standard IHS in solving each sketched sub-problem. However, the convergence rate of Algorithm 2 is much faster than IHS, that is, it requires solving much smaller number of sketched sub-problems compared to IHS to reach the same approximation accuracy.

3. Equivalence between dual random projection and Hessian sketch

While Hessian sketch [33] tries to resolve the issue of huge sample size, Dual Random Projection [52, 53] is aimed at resolving the issue of high-dimensionality by using random projections as a tool for reducing the dimension of data. Again, we consider the standard ridge regression problem in (2.1). A random projection is now used to transform the original problem (2.1) to a low-dimensional problem:

$$\min_{\mathbf{z}\in\mathbb{R}^p} P_{\mathrm{RP}}(\mathbf{XR}, \mathbf{y}; \mathbf{z}) = \min_{\mathbf{z}\in\mathbb{R}^d} \frac{1}{2n} \|\mathbf{y} - \mathbf{XRz}\|_2^2 + \frac{\lambda}{2} \|\mathbf{z}\|_2^2 , \tag{3.1}$$

where $\mathbf{R} \in \mathbb{R}^{p\times d}$ is a random projection matrix, and $d \ll p$.

Algorithm 2: Accelerated Iterative Hessian Sketch (Acc-IHS).

```
Input: Data $\mathbf{X}, \mathbf{y}$, sketching matrix $\mathbf{\Pi}$.
Initialization: $\widehat{\mathbf{w}}_{\mathrm{HS}}^{(0)} = \mathbf{0}, \mathbf{r}^{(0)} = -\frac{\mathbf{X}^\top \mathbf{y}}{n}$.
Compute $\widehat{\mathbf{u}}^{(0)}$ by solving (2.7), and update $\mathbf{p}^{(0)} = -\widehat{\mathbf{u}}^{(0)}$, calculate
  $\mathbf{v}^{(0)} = \left(\frac{\mathbf{X}^\top \mathbf{X}}{n} + \lambda \mathbf{I}_p\right) \mathbf{p}^{(0)}$.
for $t = 0, 1, 2, \dots$ do
    Calculate $\alpha^{(t)} = \frac{\langle \mathbf{r}^{(t)}, \mathbf{u}^{(t)} \rangle}{\langle \mathbf{p}^{(t)}, \mathbf{v}^{(t)} \rangle}$
    Update the approximation by $\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)} = \widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)} + \alpha^{(t)} \mathbf{p}^{(t)}$.
    Update $\mathbf{r}^{(t+1)} = \mathbf{r}^{(t)} + \alpha^{(t)} \mathbf{v}^{(t)}$.
    Update $\mathbf{u}^{(t+1)}$ by solving (2.7).
    Update $\beta^{(t+1)} = \frac{\langle \mathbf{r}^{(t+1)}, \mathbf{u}^{(t)} \rangle}{\langle \mathbf{r}^{(t)}, \mathbf{r}^{(t)} \rangle}$.
    Update $\mathbf{p}^{(t+1)} = -\mathbf{u}^{(t+1)} + \beta^{(t+1)} \mathbf{p}^{(t)}$.
    Update $\mathbf{v}^{(t+1)} = \left(\frac{\mathbf{X}^\top \mathbf{X}}{n} + \lambda \mathbf{I}_p\right) \mathbf{p}^{(t+1)}$.
end
```

Let $\widehat{\mathbf{z}} = \arg\min_{\mathbf{z}} P_{\mathrm{RP}}(\mathbf{XR}, \mathbf{y}; \mathbf{z})$. If we want to recover the original high-dimensional solution, [53] observed that the naive solution $\widehat{\mathbf{w}}_{\mathrm{RP}} = \mathbf{R}\widehat{\mathbf{z}}$ results in a bad approximation. Instead, the optimal solution of the original problem, $\mathbf{w}^*$, is recovered from the dual solution, leading to the dual random projection (DRP) approach that we explain below. The dual problem of the optimization problem in (2.1) is

$$\max_{\boldsymbol{\alpha} \in \mathbb{R}^n} D(\mathbf{X}, \mathbf{y}; \boldsymbol{\alpha}) = \max_{\boldsymbol{\alpha} \in \mathbb{R}^n} -\frac{1}{2n} \boldsymbol{\alpha}^\top \boldsymbol{\alpha} + \frac{\mathbf{y}^\top \boldsymbol{\alpha}}{n} - \frac{1}{2\lambda n^2} \boldsymbol{\alpha}^\top \mathbf{X}\mathbf{X}^\top \boldsymbol{\alpha}. \tag{3.2}$$

Let $\boldsymbol{\alpha}^* = \arg\max_{\boldsymbol{\alpha} \in \mathbb{R}^n} D(\mathbf{X}, \mathbf{y}; \boldsymbol{\alpha})$ be the dual optimal solution. By the standard primal-dual theory [6], we have the following connection between the optimal primal and dual solutions:

$$\boldsymbol{\alpha}^* = \mathbf{y} - \mathbf{X}\mathbf{w}^* \quad \text{and} \quad \mathbf{w}^* = \frac{1}{\lambda n} \mathbf{X}^\top \boldsymbol{\alpha}^*. \tag{3.3}$$

The dual random projection procedure works as follows. First, we construct and solve the low-dimensional, randomly projected problem (3.1) and obtain the solution $\widehat{\mathbf{z}}$. Next, we calculate the approximated dual variables by

$$\widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}} = \mathbf{y} - \mathbf{XR}\widehat{\mathbf{z}}, \tag{3.4}$$

based on the approximated dual solution $\widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}$. Finally, we recover the primal solution as:

$$\widehat{\mathbf{w}}_{\mathrm{DRP}} = \frac{1}{\lambda n} \mathbf{X}^\top \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}. \tag{3.5}$$

Combining the steps above, it is easy to see that the dual random projection for ridge regression has the following closed form solution:

$$\widehat{\mathbf{w}}_{\mathrm{DRP}} = \frac{\mathbf{X}^\top}{n} \left(\lambda \mathbf{I}_n + \frac{\mathbf{XR}\mathbf{R}^\top \mathbf{X}^\top}{n} \right)^{-1} \mathbf{y}. \tag{3.6}$$

Algorithm 3: Iterative Dual Random Projection (IDRP).

```
Input: Data $\mathbf{X}, \mathbf{y}$, projection matrix $\mathbf{R}$.
Initialization: $\widehat{\mathbf{w}}_{\mathrm{DRP}}^{(0)} = \mathbf{0}$.
for $t = 0, 1, 2, \ldots$ do
    Solve the projected problem in (3.8) and obtain solution $\widehat{\mathbf{z}}^{(t)}$.
    Update dual approximation: $\widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}^{(t+1)} = \mathbf{y} - \mathbf{X}\mathbf{w}_{\mathrm{DRP}}^{(t)} - \mathbf{X}\mathbf{R}\widehat{\mathbf{z}}^{(t)}$.
    Update primal approximation: $\widehat{\mathbf{w}}_{\mathrm{DRP}}^{(t+1)} = \frac{1}{\lambda n}\mathbf{X}^\top \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}^{(t+1)}$.
end
```

The recovered solution from the dual, $\widehat{\mathbf{w}}_{\mathrm{DRP}}$, has much better approximation compared to the solution recovered directly from primal problem $\widehat{\mathbf{w}}_{\mathrm{RP}}$. Specifically, $\widehat{\mathbf{w}}_{\mathrm{RP}}$ is always a poor approximation of $\mathbf{w}^*$, because $\widehat{\mathbf{w}}_{\mathrm{RP}}$ lives in a random subspace spanned by the random projection matrix $\mathbf{R}$, while $\widehat{\mathbf{w}}_{\mathrm{DRP}}$ can be a good approximation of $\mathbf{w}^*$ as long as the projected dimension d is large enough [53]. Finally, an iterative extension of DRP can exponentially reduce the approximation error [53], that we explain next.

Suppose at iteration t we have the approximate solution $\widehat{\mathbf{w}}_{\mathrm{DRP}}^{(t)}$. Consider the following optimization problem:

$$\min_{\mathbf{u}\in\mathbb{R}^p} \frac{1}{2n}\left\|\mathbf{y} - \mathbf{X}(\mathbf{u} + \widehat{\mathbf{w}}_{\mathrm{DRP}}^{(t)})\right\|_2 + \frac{\lambda}{2}\left\|\mathbf{u} + \widehat{\mathbf{w}}_{\mathrm{DRP}}^{(t)}\right\|_2^2, \tag{3.7}$$

with optimum at $\mathbf{w}^* - \widehat{\mathbf{w}}_{\mathrm{DRP}}^{(t)}$. Similar to the iterative Hessian sketch, the idea behind the iterative dual random projection (IDRP) is to approximate the residual $\mathbf{w}^* - \widehat{\mathbf{w}}_{\mathrm{DRP}}^{(t)}$ by applying dual random projection again. Given $\widehat{\mathbf{w}}_{\mathrm{DRP}}^{(t)}$ we construct the following randomly projected problem:

$$\min_{\mathbf{z}\in\mathbb{R}^d} \frac{1}{2n}\left\|\mathbf{y} - \mathbf{X}\mathbf{w}_{\mathrm{DRP}}^{(t)} - \mathbf{X}\mathbf{R}\mathbf{z}\right\|_2^2 + \frac{\lambda}{2}\left\|\mathbf{z} + \mathbf{R}^\top\mathbf{w}_{\mathrm{DRP}}^{(t)}\right\|_2^2. \tag{3.8}$$

Let $\widehat{\mathbf{z}}^{(t)}$ to be the solution of (3.8), which is used to refine the dual and primal variables as:

$$\widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}^{(t+1)} = \mathbf{y} - \mathbf{X}\mathbf{w}_{\mathrm{DRP}}^{(t)} - \mathbf{X}\mathbf{R}\widehat{\mathbf{z}},$$

and

$$\widehat{\mathbf{w}}_{\mathrm{DRP}}^{(t+1)} = \frac{1}{\lambda n}\mathbf{X}^\top \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}^{(t+1)}.$$

The iterative dual random projection (IDRP) algorithm is shown in Algorithm 3. Here we presented the idea in the context of ℓ_2 regularized least-squares. However, the iterative dual random projection can be used to solve any ℓ_2 regularized empirical loss minimization problem, as long as the loss function is smooth [53], such as, logistic regression or support vector machines with smoothed hinge loss.

The iterative dual random projection algorithm is a powerful algorithm for dealing with high-dimensional problems, but it suffers from the same limitations as the iterative Hessian sketch. First, the projection dimension needs to be large enough to guarantee that the approximation error decreases exponentially. Second, the convergence speed is not optimal. We address both of these issues next. We will show that the dual random projection is equivalent to an application of the Hessian sketch procedure on the dual problem. This connection will allow us to develop an accelerated iterative dual random projection akin to accelerated the iterative Hessian sketch algorithm presented earlier.

3.1. Dual random projection is Hessian sketch in dual space

We present the equivalence connection between Hessian sketch and dual random projection. Note that the Hessian sketch is used for *sample reduction*, while the dual random projection is utilized for *dimension reduction.* Recall that the dual maximization objective (3.2) is quadratic with respect to $\boldsymbol{\alpha}$. We can write it in the equivalent form as:

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^n} \boldsymbol{\alpha}^\top \left(\frac{\mathbf{X}\mathbf{X}^\top}{2\lambda n} + \frac{1}{2}\mathbf{I}_n\right)\boldsymbol{\alpha} - \langle \mathbf{y}, \boldsymbol{\alpha}\rangle. \tag{3.9}$$

By applying the Hessian sketch with sketching matrix $\mathbf{R} \in \mathbb{R}^{p\times d}$, we find an approximate solution for $\boldsymbol{\alpha}^*$ as:

$$\widehat{\boldsymbol{\alpha}}_{\text{HS}} = \arg\min_{\boldsymbol{\alpha}\in\mathbb{R}^n} \boldsymbol{\alpha}^\top \left(\frac{\mathbf{X}\mathbf{R}\mathbf{R}^\top\mathbf{X}^\top}{2\lambda n} + \frac{1}{2}\mathbf{I}_n\right)\boldsymbol{\alpha} - \langle \mathbf{y}, \boldsymbol{\alpha}\rangle, \tag{3.10}$$

which has the closed form solution as

$$\widehat{\boldsymbol{\alpha}}_{\text{HS}} = \lambda\left(\lambda\mathbf{I}_n + \frac{\mathbf{X}\mathbf{R}\mathbf{R}^\top\mathbf{X}^\top}{n}\right)^{-1}\mathbf{y}.$$

Substituting $\widehat{\boldsymbol{\alpha}}_{\text{HS}}$ in the primal-dual connection (3.3), gives us the following approximation to the original problem

$$\widehat{\mathbf{w}} = \frac{\mathbf{X}^\top}{n}\left(\lambda\mathbf{I}_n + \frac{\mathbf{X}\mathbf{R}\mathbf{R}^\top\mathbf{X}^\top}{n}\right)^{-1}\mathbf{y},$$

which is the same as the DRP approximation in (3.6). From this discussion, we see that *the Dual Random Projection is the Hessian sketch applied in the dual space*. To summarize, for ridge regression problem (2.1) one has closed form solutions for various sketching techniques as:

$$\begin{aligned}
\textbf{Original}: \quad \mathbf{w}^* &= \left(\lambda\mathbf{I}_p + \frac{\mathbf{X}^\top\mathbf{X}}{n}\right)^{-1}\frac{\mathbf{X}^\top\mathbf{y}}{n}\\
&= \frac{\mathbf{X}^\top}{n}\left(\lambda\mathbf{I}_n + \frac{\mathbf{X}\mathbf{X}^\top}{n}\right)^{-1}\mathbf{y};\\
\textbf{Classical Sketch}: \quad \widehat{\mathbf{w}}_{\text{CS}} &= \left(\lambda\mathbf{I}_p + \frac{\mathbf{X}^\top\boldsymbol{\Pi}\boldsymbol{\Pi}^\top\mathbf{X}}{n}\right)^{-1}\frac{\mathbf{X}^\top\boldsymbol{\Pi}\boldsymbol{\Pi}^\top\mathbf{y}}{n};
\end{aligned}$$

Algorithm 4: Accelerated Iterative Dual Random Projection (Acc-IDRP)—Primal Version.

```
Input: Data $\mathbf{X}, \mathbf{y}$, projection matrix $\mathbf{R}$.
Initialization: $\widehat{\mathbf{w}}_{\mathrm{DRP}}^{(0)} = \mathbf{0}, \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}^{(0)} = \mathbf{0}, \mathbf{r}^{(0)} = -\mathbf{y}$.
Compute $\mathbf{z}^{(0)}$ by solving (3.11), and update $\mathbf{u}^{(0)} = \mathbf{r}^{(0)} - \mathbf{X}\mathbf{R}\mathbf{z}^{(0)}$, $\mathbf{p}^{(0)} = -\mathbf{u}^{(0)}$, $\mathbf{v}^{(0)} = \left(\frac{\mathbf{X}\mathbf{X}^\top}{n} + \lambda \mathbf{I}_n\right)\mathbf{p}^{(0)}$.
for $t = 0, 1, 2, \ldots$ do
    Calculate $a^{(t)} = \frac{\langle \mathbf{r}^{(t)}, \mathbf{u}^{(t)} \rangle}{\langle \mathbf{p}^{(t)}, \mathbf{v}^{(t)} \rangle}$
    Update the dual approximation by $\widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}^{(t+1)} = \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}^{(t)} + a^{(t)}\mathbf{p}^{(t)}$.
    Update primal approximation: $\widehat{\mathbf{w}}_{\mathrm{DRP}}^{(t+1)} = \frac{1}{\lambda n}\mathbf{X}^\top \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}^{(t+1)}$.
    Update $\mathbf{r}^{(t+1)} = \mathbf{r}^{(t)} + a^{(t)}\mathbf{v}^{(t)}$.
    Solve the projected problem in (3.11) and obtain solution $\widehat{\mathbf{z}}^{(t+1)}$.
    Update $\mathbf{u}^{(t+1)} = \mathbf{r}^{(t+1)} - \mathbf{X}\mathbf{R}\widehat{\mathbf{z}}^{(t+1)}$.
    Update $\beta^{(t+1)} = \frac{\langle \mathbf{r}^{(t+1)}, \mathbf{u}^{(t)} \rangle}{\langle \mathbf{r}^{(t)}, \mathbf{r}^{(t)} \rangle}$.
    Update $\mathbf{p}^{(t+1)} = -\mathbf{u}^{(t+1)} + \beta^{(t+1)}\mathbf{p}^{(t)}$.
    Update $\mathbf{v}^{(t+1)} = \left(\frac{\mathbf{X}\mathbf{X}^\top}{n} + \lambda \mathbf{I}_n\right)\mathbf{p}^{(t+1)}$.
end
```

$$\textbf{Random Projection}: \quad \widehat{\mathbf{w}}_{\mathrm{RP}} = \mathbf{R}\left(\lambda \mathbf{I}_d + \frac{\mathbf{R}^\top \mathbf{X}^\top \mathbf{X}\mathbf{R}}{n}\right)^{-1} \mathbf{R}^\top \frac{\mathbf{X}^\top \mathbf{y}}{n};$$

$$\textbf{Hessian Sketch}: \quad \widehat{\mathbf{w}}_{\mathrm{HS}} = \left(\lambda \mathbf{I}_p + \frac{\mathbf{X}^\top \boldsymbol{\Pi}\boldsymbol{\Pi}^\top \mathbf{X}}{n}\right)^{-1} \frac{\mathbf{X}^\top \mathbf{y}}{n};$$

$$\textbf{Dual Random Projection}: \quad \widehat{\mathbf{w}}_{\mathrm{DRP}} = \frac{\mathbf{X}^\top}{n}\left(\lambda \mathbf{I}_n + \frac{\mathbf{X}\mathbf{R}\mathbf{R}^\top \mathbf{X}^\top}{n}\right)^{-1} \mathbf{y}.$$

As we can see above, the Hessian sketch is sketching the *covariance matrix*:

$$\mathbf{X}^\top \mathbf{X} \to \mathbf{X}^\top \boldsymbol{\Pi}\boldsymbol{\Pi}^\top \mathbf{X},$$

while DRP is sketching the *Gram matrix*:

$$\mathbf{X}\mathbf{X}^\top \to \mathbf{X}\mathbf{R}\mathbf{R}^\top \mathbf{X}^\top.$$

3.2. Accelerated iterative dual random projection

Based on the equivalence between dual random projection and Hessian sketch established in Section 3.1, we propose an accelerated iterative dual random projection algorithm, which improves the convergence speed of standard iterative DRP procedure [53]. The algorithm is shown in Algorithm 4. Notice that in each iteration t, we call the solver for the following randomly projected problem based on the residual $\mathbf{r}^{(t)}$:

$$\widehat{\mathbf{z}}^{(t)} = \arg\min_{\mathbf{z} \in \mathbb{R}^d} \mathbf{z}^\top \left(\frac{\mathbf{R}^\top \mathbf{X}^\top \mathbf{X}\mathbf{R}}{2n} + \frac{\lambda}{2}\mathbf{I}_d\right)\mathbf{z} - \langle \mathbf{R}^\top \mathbf{X}^\top \mathbf{r}^{(t)}, \mathbf{z} \rangle. \tag{3.11}$$

Algorithm 5: Accelerated Iterative Dual Random Projection (Acc-IDRP)—Dual Version.

```
Input: Data $\mathbf{X}, \mathbf{y}$, projection matrix $\mathbf{R}$.
Initialization: $\widehat{\mathbf{w}}_{\mathrm{DRP}}^{(0)} = \mathbf{0}, \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}^{(0)} = \mathbf{0}, \mathbf{r}^{(0)} = -\mathbf{y}$.
Compute $\mathbf{u}^{(0)}$ by solving (3.12), and update $\mathbf{p}^{(0)} = -\mathbf{u}^{(0)}$, $\mathbf{v}^{(0)} = \left(\frac{\mathbf{X}\mathbf{X}^\top}{n} + \lambda \mathbf{I}_n\right) \mathbf{p}^{(0)}$.
for $t = 0, 1, 2, \ldots$ do
    Calculate $a^{(t)} = \frac{\langle \mathbf{r}^{(t)}, \mathbf{u}^{(t)} \rangle}{\langle \mathbf{p}^{(t)}, \mathbf{v}^{(t)} \rangle}$
    Update the dual approximation by $\widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}^{(t+1)} = \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}^{(t)} + a^{(t)} \mathbf{p}^{(t)}$.
    Update primal approximation: $\widehat{\mathbf{w}}_{\mathrm{DRP}}^{(t+1)} = \frac{1}{\lambda n} \mathbf{X}^\top \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}^{(t+1)}$.
    Update $\mathbf{r}^{(t+1)} = \mathbf{r}^{(t)} + a^{(t)} \mathbf{v}^{(t)}$.
    Update $\mathbf{u}^{(t+1)}$ by solving (3.12).
    Update $\beta^{(t+1)} = \frac{\langle \mathbf{r}^{(t+1)}, \mathbf{u}^{(t)} \rangle}{\langle \mathbf{r}^{(t)}, \mathbf{r}^{(t)} \rangle}$.
    Update $\mathbf{p}^{(t+1)} = -\mathbf{u}^{(t+1)} + \beta^{(t+1)} \mathbf{p}^{(t)}$.
    Update $\mathbf{v}^{(t+1)} = \left(\frac{\mathbf{X}\mathbf{X}^\top}{n} + \lambda \mathbf{I}_n\right) \mathbf{p}^{(t+1)}$.
end
```

The accelerated IDRP algorithm runs the Acc-IHS Algorithm 2 in the dual space. However, Acc-IDRP is still a primal algorithm, since it updates the corresponding dual variables after solving the randomly projected primal problem (3.11). The dual version of Acc-IDRP algorithm would at each iteration solve the following dual optimization problem

$$\min_{\mathbf{u} \in \mathbb{R}^n} \mathbf{u}^\top \left(\frac{\mathbf{X}\mathbf{R}\mathbf{R}^\top \mathbf{X}^\top}{2n} + \frac{\lambda}{2} \mathbf{I}_n \right) \mathbf{u} - \langle \mathbf{r}^{(t)}, \mathbf{u} \rangle, \tag{3.12}$$

where $\mathbf{r}^{(t)}$ is the dual residual. This, however, is not a practical algorithm as it requires solving relatively more expensive dual problem. We present it in Algorithm 5 as it is easier to understand since it directly borrows the ideas of Acc-IHS described in Section 2.1.

Though the computational cost per iteration of Acc-IDRP and standard IDRP are the same, Acc-IDRP has two main advantages over IDRP. First, as a preconditioned conjugate gradient procedure, Acc-IDRP is guaranteed to converge, and reach the optimum $\mathbf{w}^*$ within n iterations, even when the projection dimension d is very small. Second, even when the projection dimension d is large enough for the standard IDRP converge quickly to the optimum, Acc-IDRP converges even faster.

4. Primal-dual sketch for big and high-dimensional problems

In this section, we combine the idea of the iterative Hessian sketch and iterative dual random projection from the primal-dual point of view. We propose a more efficient sketching technique named *Iterative Primal-Dual Sketch* (IPDS), which

Approach	Suitable Situation	Reduced Dimension	Recovery	Iterative
`Classical Sketch`	large n, small p	sample reduction	×	×
`Random Projection`	small n, large p	dimension reduction	×	×
`Hessian Sketch`	large n, small p	sample reduction	✓	✓
`DRP`	small n, large p	dimension reduction	✓	✓

Table 1

Comparison of various algorithms for data sketching in solving large-scale problems.

simultaneously reduces the sample size and dimensionality of the problem, while recovering the original solution to a high precision.

The Hessian sketch is particularly suitable for the case where the sample size is much larger than the problem dimension and the computational bottleneck is “big n”. Here the Hessian sketch reduces the sample size significantly, and as a consequence, speeds up the computation. By utilizing the iterative extension approximation error can be further reduced to recover the original solution to a high precision. In contrast, the dual random projection is aimed at dimensionality reduction and is suitable for the case of high-dimensional data, with relatively small sample size. Here the computational bottleneck is “large p” and the random projection is used to reduce dimensionality and speedup computations.

Hessian sketch is particularly suitable for the case where sample size is much larger than problem dimension, where the computational bottleneck is “big n”. Therefore, it is possible to use Hessian sketch to reduce the sample size significantly and consequently speed up the computation. It also possible to utilize the iterative extension to reduce the approximation error further to recover the original solution to a high precision. In contrast, dual random projection is aimed at dimension reduction and is mostly suitable for the case of high-dimensional data but relatively small sample size, where the computational bottleneck is “large p”, and we would like to use random projection to perform dimension reduction and gain speedup. Table 1 summarizes these characteristics.

As shown in Table 1, the Hessian sketch and dual random projection are suitable for problems where the number of observations n and the number of variables p are not balanced, that is, when one is much larger than the other. Modern massive data-sets have a characteristic that both n and p are very large, for example, the click-through rate (CTR) prediction data sets provided by Criteo[1] has $n \geq 4 \times 10^9$ and $p \geq 8 \times 10^8$. To tackle problems of this size, it is desirable to have a sketching method capable of simultaneously reducing the sample size and dimensionality.

Inspired by the primal-dual view described in Section 3.1, we propose the iterative Primal-Dual Sketch. The iterative Primal-Dual Sketch only involves solving small scale problems where the sample size and dimension are both small. For the original problem (2.1) with data $\{\mathbf{X}, \mathbf{y}\}$, we first construct the

[1] Available at `http://labs.criteo.com/downloads/download-terabyte-click-logs/`.

Algorithm 6: Iterative Primal-Dual Sketch (IPDS).

```
Input: Data $\mathbf{X} \in \mathbb{R}^{n\times p}, \mathbf{y} \in \mathbb{R}^n$, sketching matrix $\mathbf{R} \in \mathbb{R}^{p\times d}, \mathbf{\Pi} \in \mathbb{R}^{n\times m}$.
Initialization: $\widehat{\mathbf{w}}_{\mathrm{DS}}^{(0)} = \mathbf{0}$.
for $t = 0, 1, 2, \ldots$ do
    Initialization: $\widetilde{\mathbf{z}}^{(0)} = \mathbf{0}, k = 0$
    if not converged then
        Solve the sketched problem in (4.2) and obtain solution $\Delta\mathbf{z}^{(k)}$.
        Update $\widetilde{\mathbf{z}}^{(k+1)} = \widetilde{\mathbf{z}}^{(k)} + \Delta\mathbf{z}^{(k)}$.
        Update $k = k + 1$.
    end
    Update dual approximation: $\widehat{\boldsymbol{\alpha}}_{\mathrm{DS}}^{(t+1)} = \mathbf{y} - \mathbf{X}\widehat{\mathbf{w}}_{\mathrm{DS}}^{(t)} - \mathbf{X}\mathbf{R}\widetilde{\mathbf{z}}^{(k+1)}$.
    Update primal approximation: $\widehat{\mathbf{w}}_{\mathrm{DS}}^{(t+1)} = \frac{1}{\lambda n}\mathbf{X}^\top\widehat{\boldsymbol{\alpha}}_{\mathrm{DS}}^{(t+1)}$.
end
```

randomly projected data, as well as the *doubly sketched* data, as follows:

$$\mathbf{X} \to \mathbf{XR}, \qquad \mathbf{XR} \to \mathbf{\Pi}^\top\mathbf{XR},$$

where $\mathbf{XR}$ is the randomly projected data, and $\mathbf{\Pi}^\top\mathbf{XR}$ is doubly sketched data. Let $\widehat{\mathbf{w}}_{\mathrm{DS}}^{(0)} = \mathbf{0}$. At every iteration of IPDS, we first apply random projection on the primal problem (which is equivalent to the Hessian sketch on the dual problem), and obtain the following problem:

$$\min_{\mathbf{z}\in\mathbb{R}^d} \frac{1}{2n}\left\|\mathbf{y} - \mathbf{X}\widehat{\mathbf{w}}_{\mathrm{DS}}^{(t)} - \mathbf{XRz}\right\|_2^2 + \frac{\lambda}{2}\left\|\mathbf{z} + \mathbf{R}^\top\widehat{\mathbf{w}}_{\mathrm{DS}}^{(t)}\right\|_2^2, \tag{4.1}$$

which is the same as the iterative dual random projection subproblem (3.8). However, different from IDRP, we do not directly solve (4.1). Instead, we apply the iterative Hessian sketch to find an approximate solution to

$$\min_{\mathbf{z}\in\mathbb{R}^d} \mathbf{z}^\top\left(\frac{\mathbf{R}^\top\mathbf{X}^\top\mathbf{XR}}{2n} + \frac{\lambda}{2}\mathbf{I}_d\right)\mathbf{z} - \left\langle \frac{(\mathbf{y} - \mathbf{X}\widehat{\mathbf{w}}_{\mathrm{DS}}^{(t)})^\top\mathbf{XR}}{n} - \lambda\mathbf{R}^\top\widehat{\mathbf{w}}_{\mathrm{DS}}^{(t)}, \mathbf{z}\right\rangle.$$

Let $\widetilde{\mathbf{z}}^{(0)} = \mathbf{0}$. At iteration k in the inner loop, we solve the following sketched problem:

$$\begin{aligned}\Delta\mathbf{z}^{(k)} = \arg\min_{\Delta\mathbf{z}} \Delta\mathbf{z}^\top&\left(\frac{\mathbf{R}^\top\mathbf{X}^\top\mathbf{\Pi\Pi}^\top\mathbf{XR}}{2n} + \frac{\lambda}{2}\mathbf{I}_d\right)\mathbf{z}\\ &- \left\langle \frac{\mathbf{R}^\top\mathbf{X}^\top(\mathbf{y} - \mathbf{X}\widehat{\mathbf{w}}_{\mathrm{DS}}^{(t)} - \mathbf{XR}\widetilde{\mathbf{z}}^{(k)})}{n} - \lambda\mathbf{R}^\top\widehat{\mathbf{w}}_{\mathrm{DS}}^{(t)} - \lambda\widetilde{\mathbf{z}}^{(k)}, \Delta\mathbf{z}\right\rangle\end{aligned} \tag{4.2}$$

and update $\widetilde{\mathbf{z}}^{(k+1)}$ as

$$\widetilde{\mathbf{z}}^{(k+1)} = \widetilde{\mathbf{z}}^{(k)} + \Delta\mathbf{z}^{(k)}.$$

The key point is that for the subproblem (4.2), the sketched data matrix is only of size $m \times d$, compared to the original problem size $n \times p$, where $n \gg m, p \gg d$.

In contrast, the IHS still needs to solve sub-problems of size $m \times p$, while IDRP needs to solve sub-problems of size $n \times d$. We only need to call solvers of $m \times d$ problem (4.2) logarithmic times to obtain a solution of high approximation quality.

The pseudo code of Iterative Primal-Dual Sketch (IPDS) is summarized in Algorithm 6. It is also possible to perform iterative Primal-Dual Sketch via another direction, that is, first perform primal Hessian sketch, and then apply dual Hessian sketch to solve the sketched primal problem:

$$\mathbf{X} \to \mathbf{\Pi}^\top \mathbf{X}, \qquad \mathbf{\Pi}^\top \mathbf{X} \to \mathbf{\Pi}^\top \mathbf{X}\mathbf{R}.$$

The idea presented in Section 2.1 can also be adopted to further reduce the number of calls to $m \times d$ scale sub-problems, which leads to the accelerated iterative primal-dual sketch (Acc-IPDS) algorithm, of which the pseudo code is summarized in Algorithm 7. In Acc-IPDS, we maintain both the vectors in the primal space $\mathbf{u}_{\mathrm{P}}, \mathbf{v}_{\mathrm{P}}, \mathbf{r}_{\mathrm{P}}$ and the vectors in the dual space $\mathbf{u}_{\mathrm{D}}, \mathbf{v}_{\mathrm{D}}, \mathbf{r}_{\mathrm{D}}$, to make sure that the updating directions for both primal variables and dual variables are conjugate with previous updating directions. Moreover, based on the residual vector $\mathbf{r}_{\mathrm{P}}$, Acc-IPDS iteratively calls the solver to find a solution of the following sketched linear system of scale $m \times d$:

$$\widehat{\mathbf{u}}_P^{(k)} = \arg\min_{\mathbf{u}} \mathbf{u}^\top \left(\frac{\mathbf{R}^\top \mathbf{X}^\top \mathbf{\Pi}\mathbf{\Pi}^\top \mathbf{X}\mathbf{R}}{2n} + \frac{\lambda}{2}\mathbf{I}_d \right) \mathbf{u} - \left\langle \mathbf{r}_{\mathrm{P}}^{(k)}, \mathbf{u} \right\rangle . \tag{4.3}$$

As we will show in the subsequent section, the number of calls for solving problem (4.3) only grows logarithmically with the inverse of approximation error.

5. Theoretical analysis

We present theoretical properties of various iterative sketching procedures, while the proofs are deferred to Appendix A. First, we provide a unified analysis of the Hessian sketch and dual random projection. The unified analysis basically follows the analysis of [53] and [33], but provides recovery guarantees for both the *primal* and *dual* variables of interest, simultaneously. Next, we present convergence analysis for the proposed accelerated IHS and IDRP algorithms. We show improved convergence speed compared to the standard IHS and IDRP algorithms. Finally, we prove that the iterative primal-dual sketch converges to the optimum with the number of iterations growing only logarithmically with the target approximation accuracy. This makes the proposed primal-dual sketching schema suitable for large-scale learning problems with huge number of features.

5.1. A unified analysis of Hessian sketch and dual random projection

In this section we provide a simple and unified analysis for the recovery performance of the Hessian sketch and dual random projection. First, we define the

Algorithm 7: Accelerated Iterative Primal-Dual Sketch (Acc-IPDS).

```
Input: Data $\mathbf{X} \in \mathbb{R}^{n\times p}, \mathbf{y} \in \mathbb{R}^n$, sketching matrix $\mathbf{R} \in \mathbb{R}^{p\times d}, \mathbf{\Pi} \in \mathbb{R}^{n\times m}$.
Initialization: $\widehat{\mathbf{w}}_{\mathrm{DS}}^{(0)} = \mathbf{0}, \widehat{\boldsymbol{\alpha}}_{\mathrm{DS}}^{(0)} = \mathbf{0}, \mathbf{r}_{\mathrm{Dual}}^{(0)} = -\mathbf{y}$.
Initialization: $k = 0, \widetilde{\mathbf{z}}^{(k)} = \mathbf{0}, \mathbf{r}_{\mathrm{P}}^{(0)} = \mathbf{R}^\top\mathbf{X}^\top\mathbf{r}_{\mathrm{D}}^{(0)}$.
Compute $\widehat{\mathbf{u}}_{\mathrm{P}}^{(0)}$ by solving (4.3), and update $\mathbf{p}_{\mathrm{P}}^{(0)} = -\mathbf{u}_{\mathrm{P}}^{(0)}$, calculate
  $\mathbf{v}_{\mathrm{P}}^{(0)} = \left(\frac{\mathbf{R}^\top\mathbf{X}\mathbf{X}\mathbf{R}}{n} + \lambda\mathbf{I}_d\right)\mathbf{p}_{\mathrm{P}}^{(0)}$.
if not converged then
    Calculate $a_{\mathrm{P}}^{(k)} = \frac{\langle \mathbf{r}_{\mathrm{P}}^{(k)}, \mathbf{u}_{\mathrm{P}}^{(k)}\rangle}{\langle \mathbf{p}_{\mathrm{P}}^{(k)}, \mathbf{v}_{\mathrm{P}}^{(k)}\rangle}$, and update the approximation by
      $\widetilde{\mathbf{z}}^{(k+1)} = \widetilde{\mathbf{z}}^{(k)} + a_{\mathrm{P}}^{(k)}\mathbf{p}_{\mathrm{P}}^{(k)}$.
    Update $\mathbf{r}_{\mathrm{P}}^{(k+1)} = \mathbf{r}_{\mathrm{P}}^{(k)} + a_{\mathrm{P}}^{(k)}\mathbf{v}^{(k)}$, and update $\mathbf{u}_{\mathrm{P}}^{(k+1)}$ by solving (4.3).
    Update $\beta_{\mathrm{P}}^{(k+1)} = \frac{\langle \mathbf{r}_{\mathrm{P}}^{(k+1)}, \mathbf{u}_{\mathrm{P}}^{(k)}\rangle}{\langle \mathbf{r}_{\mathrm{P}}^{(k)}, \mathbf{r}_{\mathrm{P}}^{(k)}\rangle}$, and update $\mathbf{p}_{\mathrm{P}}^{(k+1)} = -\mathbf{u}_{\mathrm{P}}^{(k+1)} + \beta_{\mathrm{P}}^{(k+1)}\mathbf{p}_{\mathrm{P}}^{(k)}$.
    Update $\mathbf{v}_{\mathrm{P}}^{(k+1)} = \left(\frac{\mathbf{R}^\top\mathbf{X}^\top\mathbf{X}\mathbf{R}}{n} + \lambda\mathbf{I}_p\right)\mathbf{p}_{\mathrm{P}}^{(t+1)}$, and update $k = k + 1$.
end
Compute $\mathbf{u}_{\mathrm{D}}^{(0)} = \mathbf{r}_{\mathrm{D}}^{(0)} - \mathbf{X}\mathbf{R}\widetilde{\mathbf{z}}^{(k+1)}$, $\mathbf{p}_{\mathrm{D}}^{(0)} = -\mathbf{u}_{\mathrm{D}}^{(0)}$, $\mathbf{v}_{\mathrm{D}}^{(0)} = \left(\frac{\mathbf{X}\mathbf{X}^\top}{n} + \lambda\mathbf{I}_n\right)\mathbf{p}_{\mathrm{D}}^{(0)}$.
for $t = 0, 1, 2, \ldots$ do
    Calculate $a_{\mathrm{D}}^{(t)} = \frac{\langle \mathbf{r}_{\mathrm{D}}^{(t)}, \mathbf{u}_{\mathrm{D}}^{(t)}\rangle}{\langle \mathbf{p}_{\mathrm{D}}^{(t)}, \mathbf{v}_{\mathrm{D}}^{(t)}\rangle}$, and update the dual approximation by
      $\widehat{\boldsymbol{\alpha}}_{\mathrm{DS}}^{(t+1)} = \widehat{\boldsymbol{\alpha}}_{\mathrm{DS}}^{(t)} + a_{\mathrm{D}}^{(t)}\mathbf{p}_{\mathrm{D}}^{(t)}$.
    Update primal approximation: $\widehat{\mathbf{w}}_{\mathrm{DS}}^{(t+1)} = \frac{1}{\lambda n}\mathbf{X}^\top\widehat{\boldsymbol{\alpha}}_{\mathrm{DS}}^{(t+1)}$, and update
      $\mathbf{r}_{\mathrm{D}}^{(t+1)} = \mathbf{r}_{\mathrm{D}}^{(t)} + a_{\mathrm{D}}^{(t)}\mathbf{v}_{\mathrm{D}}^{(t)}$.
    Initialization: $k = 0, \widetilde{\mathbf{z}}^{(k)} = \mathbf{0}, \mathbf{r}_{\mathrm{P}}^{(0)} = \mathbf{R}^\top\mathbf{X}^\top\mathbf{r}_{\mathrm{D}}^{(t+1)}$.
    Compute $\widehat{\mathbf{u}}_{\mathrm{P}}^{(0)}$ by solving (4.3), and update $\mathbf{p}_{\mathrm{P}}^{(0)} = -\mathbf{u}_{\mathrm{P}}^{(0)}$, calculate
      $\mathbf{v}_{\mathrm{P}}^{(0)} = \left(\frac{\mathbf{R}^\top\mathbf{X}\mathbf{X}\mathbf{R}}{n} + \lambda\mathbf{I}_d\right)\mathbf{p}_{\mathrm{P}}^{(0)}$.
    if not converged then
        Calculate $a_{\mathrm{P}}^{(k)} = \frac{\langle \mathbf{r}_{\mathrm{P}}^{(k)}, \mathbf{u}_{\mathrm{P}}^{(k)}\rangle}{\langle \mathbf{p}_{\mathrm{P}}^{(k)}, \mathbf{v}_{\mathrm{P}}^{(k)}\rangle}$, and update the approximation by
          $\widetilde{\mathbf{z}}^{(k+1)} = \widetilde{\mathbf{z}}^{(k)} + a_{\mathrm{P}}^{(k)}\mathbf{p}_{\mathrm{P}}^{(k)}$.
        Update $\mathbf{r}_{\mathrm{P}}^{(k+1)} = \mathbf{r}_{\mathrm{P}}^{(k)} + a_{\mathrm{P}}^{(k)}\mathbf{v}^{(k)}$, and update $\mathbf{u}_{\mathrm{P}}^{(k+1)}$ by solving (4.3).
        Update $\beta_{\mathrm{P}}^{(k+1)} = \frac{\langle \mathbf{r}_{\mathrm{P}}^{(k+1)}, \mathbf{u}_{\mathrm{P}}^{(k)}\rangle}{\langle \mathbf{r}_{\mathrm{P}}^{(k)}, \mathbf{r}_{\mathrm{P}}^{(k)}\rangle}$, and update $\mathbf{p}_{\mathrm{P}}^{(k+1)} = -\mathbf{u}_{\mathrm{P}}^{(k+1)} + \beta_{\mathrm{P}}^{(k+1)}\mathbf{p}_{\mathrm{P}}^{(k)}$.
        Update $\mathbf{v}_{\mathrm{P}}^{(k+1)} = \left(\frac{\mathbf{R}^\top\mathbf{X}^\top\mathbf{X}\mathbf{R}}{n} + \lambda\mathbf{I}_p\right)\mathbf{p}_{\mathrm{P}}^{(t+1)}$, and update $k = k + 1$.
    end
    Update $\mathbf{u}_{\mathrm{D}}^{(t+1)} = \mathbf{r}_{\mathrm{D}}^{(t+1)} - \mathbf{X}\mathbf{R}\widetilde{\mathbf{z}}_{\mathrm{D}}^{(k+1)}$, and update $\beta_{\mathrm{D}}^{(t+1)} = \frac{\langle \mathbf{r}_{\mathrm{D}}^{(t+1)}, \mathbf{u}_{\mathrm{D}}^{(t)}\rangle}{\langle \mathbf{r}_{\mathrm{D}}^{(t)}, \mathbf{r}_{\mathrm{D}}^{(t)}\rangle}$.
    Update $\mathbf{p}_{\mathrm{D}}^{(t+1)} = -\mathbf{u}_{\mathrm{D}}^{(t+1)} + \beta_{\mathrm{D}}^{(t+1)}\mathbf{p}_{\mathrm{D}}^{(t)}$, and update $\mathbf{v}_{\mathrm{D}}^{(t+1)} = \left(\frac{\mathbf{X}\mathbf{X}^\top}{n} + \lambda\mathbf{I}_n\right)\mathbf{p}_{\mathrm{D}}^{(t+1)}$.
end
```

following notion of the *Gaussian width*, which will be useful in the statement of our results. For any set $\mathcal{K} \subseteq \mathbb{R}^p$, the Gaussian width is defined as:

$$\mathbb{W}(\mathcal{K}) = \mathbb{E}\left[\sup_{\mathbf{w}\in\mathcal{K}} \langle \mathbf{w}, \mathbf{g}\rangle\right], \tag{5.1}$$

where $\mathbf{g}$ is a random vector drawn from a Normal distribution $\mathcal{N}(\mathbf{0}, \mathbf{I}_p)$. Intuitively, if the set $\mathcal{K}$ is restricted to certain directions and magnitude, then $\mathbb{W}(\mathcal{K})$ will be small [43]. Given a set $\mathcal{K}$ and a random matrix $\mathbf{R} \in \mathbb{R}^{p\times d}$, and a unit-length vector $\mathbf{v} \in \mathbb{R}^p$, the following quantities will be important:

$$\rho_1(\mathcal{K}, \mathbf{R}) = \inf_{\mathbf{u}\in\mathcal{K}\cap\mathcal{S}^{p-1}} \mathbf{u}^\top \mathbf{R}\mathbf{R}^\top \mathbf{u}$$
$$\rho_2(\mathcal{K}, \mathbf{R}, \mathbf{v}) = \sup_{\mathbf{u}\in\mathcal{K}\cap\mathcal{S}^{p-1}} \left|\mathbf{u}^\top \left(\mathbf{R}\mathbf{R}^\top - \mathbf{I}_p\right)\mathbf{v}\right|,$$

where $\mathcal{S}^{p-1} = \{\mathbf{x} \in \mathbb{R}^p \mid \|\mathbf{x}\| = 1\}$ is the p-dimensional Euclidean unit-sphere. The sketching matrix $\mathbf{R}$ will be constructed to satisfy

$$\mathbb{E}\left[\mathbf{R}\mathbf{R}^\top\right] = \mathbf{I}_p,$$

and, as the sketching dimension d increases, the matrix $\mathbf{R}\mathbf{R}^\top$ will get closer to $\mathbf{I}_p$. Thus, we would like to have $\rho_1(\mathcal{K}, \mathbf{R})$ be close to 1, and $\rho_2(\mathcal{K}, \mathbf{R}, \mathbf{v})$ be close to 0. To simplify presentation, assume that the entries of a random matrix $\mathbf{R}$ are sampled i.i.d. from a sub-Gaussian distribution and $\mathbb{E}\left[\mathbf{R}\mathbf{R}^\top\right] = \mathbf{I}_p$. The following lemma states how large the sketching dimension d should be in order to control $\rho_1(\mathcal{K}, \mathbf{R})$ and $\rho_2(\mathcal{K}, \mathbf{R}, \mathbf{v})$.

Lemma 1 ([31])**.** *When $\mathbf{R}$ is sampled i.i.d. from a sub-Gaussian distribution and $\mathbb{E}\left[\mathbf{R}\mathbf{R}^\top\right] = \mathbf{I}_p$, then there exists universal constants C_0 such that*

$$\rho_1(\mathcal{K}, \mathbf{R}) \geq 1 - C_0\sqrt{\frac{\mathbb{W}^2(\mathcal{K}\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right) \tag{5.2}$$

and

$$\rho_2(\mathcal{K}, \mathbf{R}, \mathbf{v}) \leq C_0\sqrt{\frac{\mathbb{W}^2(\mathcal{K}\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right), \tag{5.3}$$

with probability at least $1-\delta$.

For a set $\mathcal{K} \subseteq \mathbb{R}^p$, define the transformed set $\mathbf{X}\mathcal{K}$ with $\mathbf{X} \in \mathbb{R}^{n\times p}$ as

$$\mathbf{X}\mathcal{K} = \{\mathbf{u} = \mathbf{X}\mathbf{v} \in \mathbb{R}^n \mid \mathbf{v} \in \mathcal{K}\}.$$

Before presenting the main unifying result, let us recall the reductions in the Hessian sketch and dual random projection. For the Hessian sketch, we perform *sample reduction* with the transformation $\mathbf{X} \to \mathbf{\Pi}^\top\mathbf{X}$, while for the dual random projection, we perform *dimension reduction* with the transformation $\mathbf{X} \to \mathbf{X}\mathbf{R}$, where $\mathbf{\Pi} \in \mathbb{R}^{n\times m}$ and $\mathbf{R} \in \mathbb{R}^{p\times d}$. Let $\widehat{\mathbf{w}}_{\text{HS}}$ be an approximate solution obtained via the Hessian sketch by solving (2.4). The corresponding dual variables are obtained using the following transformation

$$\widehat{\boldsymbol{\alpha}}_{\text{HS}} = \mathbf{y} - \mathbf{X}\widehat{\mathbf{w}}_{\text{HS}}.$$

Likewise, let $\widehat{\boldsymbol{\alpha}}_{\text{DRP}}$ and $\widehat{\mathbf{w}}_{\text{DRP}}$ be the approximate dual and primal variables obtained by the dual random projection. The following theorem established the recovery bound for $\widehat{\boldsymbol{\alpha}}_{\text{HS}}, \widehat{\boldsymbol{\alpha}}_{\text{DRP}}$ and $\widehat{\mathbf{w}}_{\text{HS}}, \widehat{\mathbf{w}}_{\text{DRP}}$ simultaneously.

Theorem 2. *Suppose we perform the Hessian sketch or the dual random projection for the problem (2.1) with a sub-Gaussian sketching matrix $\mathbf{\Pi} \in \mathbb{R}^{n\times m}$ (for HS) or $\mathbf{R} \in \mathbb{R}^{p\times d}$ (for DRP), satisfying $\mathbb{E}\left[\mathbf{R}\mathbf{R}^\top\right] = \mathbf{I}_p$ and $\mathbb{E}\left[\mathbf{\Pi}\mathbf{\Pi}^\top\right] = \mathbf{I}_n$. Then there exists a universal constant C_0 such that with probability at least $1-\delta$, the following approximation error bounds hold for HS or DRP:* ***For Hessian sketch:***

$$\|\widehat{\mathbf{w}}_{\mathrm{HS}} - \mathbf{w}^*\|_{\mathbf{X}} \le \frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}}\log\left(\frac{1}{\delta}\right)}{1 - C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}}\log\left(\frac{1}{\delta}\right)}\|\mathbf{w}^*\|_{\mathbf{X}}, \quad \textit{and} \tag{5.4}$$

$$\|\widehat{\boldsymbol{\alpha}}_{\mathrm{HS}} - \boldsymbol{\alpha}^*\|_2 \le \frac{\sqrt{n}C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}}\log\left(\frac{1}{\delta}\right)}{1 - C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}}\log\left(\frac{1}{\delta}\right)}\|\mathbf{w}^*\|_{\mathbf{X}}. \tag{5.5}$$

For dual random projection:

$$\|\widehat{\mathbf{w}}_{\mathrm{DRP}} - \mathbf{w}^*\|_2 \le \frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}{1 - C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}\|\mathbf{w}^*\|_2, \quad \textit{and} \tag{5.6}$$

$$\|\widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}} - \boldsymbol{\alpha}^*\|_{\mathbf{X}^\top} \le \frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}{1 - C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}\|\boldsymbol{\alpha}^*\|_{\mathbf{X}^\top}. \tag{5.7}$$

Theorem 2 is proven in Appendix A.1. We have the following remarks on Theorem 2.

Remark 1. *For a general low-dimensional problem where $n \gg p$, $\mathbb{W}^2(\mathbf{X}\mathbb{R}^p \cap \mathcal{S}^{n-1}) = p$. Thus we have $\|\widehat{\mathbf{w}}_{\mathrm{HS}} - \mathbf{w}^*\|_{\mathbf{X}} \lesssim \sqrt{\frac{p}{m}}\log\left(\frac{1}{\delta}\right)\|\mathbf{w}^*\|_{\mathbf{X}}$. This is the recovery bound proved in [Proposition 1, 33].*

Remark 2. *For high-dimensional problems when p is large, $\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1}) = n$. Thus we have $\|\widehat{\mathbf{w}}_{\mathrm{DRP}} - \mathbf{w}^*\|_2 \lesssim \sqrt{\frac{n}{d}}\log\left(\frac{1}{\delta}\right)\|\mathbf{w}^*\|_2$. Moreover, when $\mathbf{X}$ is low-rank, that is $\operatorname{rank}(\mathbf{X}) = r$ and $r \ll \min(n,p)$, we have $\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n \cap \mathcal{S}^{p-1}) = r$, leading to $\|\widehat{\mathbf{w}}_{\mathrm{DRP}} - \mathbf{w}^*\|_2 \lesssim \sqrt{\frac{r}{d}}\log\left(\frac{1}{\delta}\right)\|\mathbf{w}^*\|_2$. This recovery bound removes a $\sqrt{\log r}$ factor from a bound obtained in Theorem 1 of [53].*

5.1.1. Analysis of IHS and DRP when $\mathbf{X}$ is approximately low-rank

In this section we provide recovery guarantees for the case when the data matrix $\mathbf{X}$ is *approximately* low rank. To make $\mathbf{X}$ be well approximated by a rank r matrix where $r \ll \min(n,p)$, we assume σ_{r+1}, the $r+1$-th singular value of $\mathbf{X}$, is small enough. Let us decompose $\mathbf{X}$ into two parts as

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top = \mathbf{U}_r\mathbf{\Sigma}_r\mathbf{V}_r^\top + \mathbf{U}_{\bar{r}}\mathbf{\Sigma}_{\bar{r}}\mathbf{V}_{\bar{r}}^\top = \mathbf{X}_r + \mathbf{X}_{\bar{r}},$$

where $\mathbf{X}_r$ corresponds to the largest r singular values and $\mathbf{X}_{\bar{r}}$ corresponds to the remaining singular values with $\bar{r} = \{r+1, ..., \max(n,p)\}$. Furthermore, suppose that the solution to (2.1), $\mathbf{w}^*$, is well approximated by the a linear combination of the top r left singular vectors of $\mathbf{X}$ and that the remaining singular vectors are almost orthogonal to $\mathbf{w}^*$. We can represent this notion more formally depending on the particular method. For the Hessian sketch we assume that for some ρ we have

$$\left\|\mathbf{X}_{\bar{r}}\mathbf{w}^*\right\|_2 \leq \rho \left\|\mathbf{X}\mathbf{w}^*\right\|_2 ,$$

while for the dual random projection we assume that

$$\left\|\mathbf{V}_{\bar{r}}^\top \mathbf{w}^*\right\|_2 \leq \varrho \left\|\mathbf{w}^*\right\|_2 ,$$

for some ϱ. For simplicity, assume that the entries of the sketching matrix $\mathbf{\Pi} \in \mathbb{R}^{m\times n}$ and $\mathbf{R} \in \mathbb{R}^{p\times d}$ are sampled i.i.d. from a zero-mean sub-Gaussian distributions, and satisfying $\mathbb{E}\left[\mathbf{R}\mathbf{R}^\top\right] = \mathbf{I}_p$ and $\mathbb{E}\left[\mathbf{\Pi}\mathbf{\Pi}^\top\right] = \mathbf{I}_n$ respectively. We have the following recovery bounds for the Hessian sketch and the dual random projection.

Theorem 3. *With probability at least $1-\delta$, we have:*
For the Hessian sketch:

$$m \geq \max\left(32(r+1), 4\log\left(\frac{2m}{\delta}\right), \frac{784\sigma_{r+1}^2}{9\lambda}\right)\log\left(\frac{n}{\delta}\right)$$

then

$$\left\|\widehat{\mathbf{w}}_{\mathrm{HS}} - \mathbf{w}^*\right\|_{\mathbf{X}} \leq 4\sqrt{\frac{1}{1-\epsilon_1} + \frac{\sigma_{r+1}^2}{\lambda n}} \cdot \sqrt{\frac{\epsilon_1^2 + \tau_1^2\rho^2}{1-\epsilon_1} + \frac{\tau_1^2\sigma_{r+1}^2 + \rho^2\upsilon_1^2\sigma_{r+1}^2}{\lambda n}} \left\|\mathbf{w}^*\right\|_{\mathbf{X}} ;$$

For the dual random projection: *if*

$$d \geq \max\left(32(r+1), 4\log\left(\frac{2d}{\delta}\right), \frac{784p\sigma_{r+1}^2}{9\lambda n}\right)\log\left(\frac{p}{\delta}\right)$$

then

$$\left\|\widehat{\mathbf{w}}_{\mathrm{DRP}} - \mathbf{w}^*\right\|_2 \leq 4\sqrt{\frac{1}{1-\epsilon_2} + \frac{\sigma_{r+1}^2}{\lambda n}} \cdot \sqrt{\frac{\epsilon_2^2 + \tau_2^2\varrho^2}{1-\epsilon_2} + \frac{\tau_2^2\sigma_{r+1}^2 + \varrho^2\upsilon_2^2\sigma_{r+1}^2}{\lambda n}} \left\|\mathbf{w}^*\right\|_2 , \tag{5.8}$$

where $\epsilon_1, \epsilon_2, \tau_1, \tau_2, \upsilon_1, \upsilon_2$ are defined as

$$\begin{aligned}
\epsilon_1 =& 2\sqrt{\frac{2(r+1)}{m}\log\frac{2r}{\delta}}, & \epsilon_2 = 2\sqrt{\frac{2(r+1)}{d}\log\frac{2r}{\delta}}, \\
\tau_1 =& \frac{7}{3}\sqrt{\frac{2(n-r)}{m}\log\frac{n}{\delta}}, & \tau_2 = \frac{7}{3}\sqrt{\frac{2(p-r)}{d}\log\frac{p}{\delta}}, \\
\upsilon_1 =& 2\sqrt{\frac{2(n-r+1)}{m}\log\frac{2(n-r)}{\delta}}, & \upsilon_2 = 2\sqrt{\frac{2(p-r+1)}{d}\log\frac{2(p-r)}{\delta}}.
\end{aligned}$$

The proof is provided in Appendix A.2. We make the following comments on Theorem 3.

Remark 3. *When $\sigma_{r+1} = 0$ and $\mathbf{X}$ is of exact rank r, the above result simplifies to*

$$\|\widehat{\mathbf{w}}_{\mathrm{HS}} - \mathbf{w}^*\|_{\mathbf{X}} \lesssim \sqrt{\frac{r}{m}}\,\|\mathbf{w}^*\|_{\mathbf{X}}\,, \qquad \|\widehat{\mathbf{w}}_{\mathrm{DRP}} - \mathbf{w}^*\|_2 \lesssim \sqrt{\frac{r}{d}}\,\|\mathbf{w}^*\|_2 \tag{5.9}$$

These are the results of Theorem 2.

Remark 4. *We see that if we have σ_{r+1}, ρ, and ϱ sufficiently small, then the guarantees (5.9) still hold. In particular, for the Hessian sketch we need that*

$$\sigma_{r+1} \lesssim \sqrt{\lambda}, \qquad \rho \lesssim \sqrt{\frac{r}{n}},$$

while for the dual random projection we need

$$\sigma_{r+1} \lesssim \sqrt{\frac{\lambda n}{p}}, \qquad \varrho \lesssim \sqrt{\frac{r}{p}}.$$

5.2. Analysis of the accelerated IHS and IDRP methods

In this section we provide convergence analysis for the Acc-IHS and Acc-IDRP algorithms. Recall that Acc-IHS and Acc-IDRP algorithms are preconditioned conjugate gradient methods on primal and dual problems, with a sketched Hessian as a pre-conditioner. Therefore, we will use a classical analysis of preconditioned conjugate gradient [22] to obtain the following convergence guarantees.

Proposition 4. *Iterates obtain by Acc-IHS satisfy*

$$\left\|\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)} - \mathbf{w}^*\right\|_{\mathbf{X}} \leq 2\left(\frac{\sqrt{\kappa_{\mathrm{HS}}(\mathbf{X}, \mathbf{\Pi}, \lambda)} - 1}{\sqrt{\kappa_{\mathrm{HS}}(\mathbf{X}, \mathbf{\Pi}, \lambda)} + 1}\right)^t \|\mathbf{w}^*\|_{\mathbf{X}}\,, \tag{5.10}$$

where

$$\kappa_{\mathrm{HS}}(\mathbf{X}, \mathbf{\Pi}, \lambda) = \kappa\left(\left(\frac{\mathbf{X}^\top\mathbf{\Pi}\mathbf{\Pi}^\top\mathbf{X}}{n} + \lambda\mathbf{I}_p\right)^{-1}\left(\frac{\mathbf{X}^\top\mathbf{X}}{n} + \lambda\mathbf{I}_p\right)\right).$$

Iterates obtain by Acc-IDRP satisfy

$$\left\|\widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}^{(t)} - \boldsymbol{\alpha}^*\right\|_{\mathbf{X}^\top} \leq 2\left(\frac{\sqrt{\kappa_{\mathrm{DRP}}(\mathbf{X}, \mathbf{R}, \lambda)} - 1}{\sqrt{\kappa_{\mathrm{DRP}}(\mathbf{X}, \mathbf{R}, \lambda)} + 1}\right)^t \|\boldsymbol{\alpha}^*\|_{\mathbf{X}^\top}\,, \tag{5.11}$$

where

$$\kappa_{\mathrm{DRP}}(\mathbf{X}, \mathbf{R}, \lambda) = \kappa\left(\left(\frac{\mathbf{X}\mathbf{R}\mathbf{R}^\top\mathbf{X}^\top}{n} + \lambda\mathbf{I}_n\right)^{-1}\left(\frac{\mathbf{X}\mathbf{X}^\top}{n} + \lambda\mathbf{I}_n\right)\right).$$

From Proposition 4, we see that the convergence of Acc-IHS and Acc-IDRP heavily depend on the respective condition number, $\kappa_{\mathrm{HS}}(\mathbf{X}, \mathbf{\Pi}, \lambda)$ or

$\kappa_{\mathrm{DRP}}(\mathbf{X}, \mathbf{R}, \lambda)$. Therefore, it is crucial to obtain a refined upper bound on these condition numbers. We make use of the following result in [24].

Lemma 5. *Suppose the elements of $\mathbf{\Pi} \in \mathbb{R}^{n\times m}$ are sampled i.i.d. from a zero-mean sub-Gaussian distribution satisfying $\mathbb{E}\left[\mathbf{\Pi}\mathbf{\Pi}^\top\right] = \mathbf{I}_n$, then there exists a universal constant C_0 such that, for any subset $\mathcal{K} \subseteq \mathbb{R}^n$, we have*

$$\sup_{\mathbf{u}\in\mathcal{K}\cap\mathcal{S}^{n-1}} \left|\mathbf{u}^\top\left(\mathbf{\Pi}\mathbf{\Pi}^\top - \mathbf{I}_n\right)\mathbf{u}\right| \le C_0\sqrt{\frac{\mathbb{W}^2(\mathcal{K})}{m}\log\left(\frac{1}{\delta}\right)}$$

with probability at least $1-\delta$.

An application of this lemma gives us the following bounds on the condition numbers $\kappa_{\mathrm{HS}}(\mathbf{X}, \mathbf{\Pi}, \lambda)$ and $\kappa_{\mathrm{DRP}}(\mathbf{X}, \mathbf{R}, \lambda)$.

Theorem 6. *If the sketching matrices $\mathbf{\Pi} \in \mathbb{R}^{n\times m}$ and $\mathbf{R} \in \mathbb{R}^{p\times d}$ are sampled from sub-Gaussian distributions, and satisfying $\mathbb{E}\left[\mathbf{R}\mathbf{R}^\top\right] = \mathbf{I}_p$ and $\mathbb{E}\left[\mathbf{\Pi}\mathbf{\Pi}^\top\right] = \mathbf{I}_n$ respectively, then*

$$\kappa_{\mathrm{HS}}(\mathbf{X}, \mathbf{\Pi}, \lambda) \le \left(1 - 2C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}\right)^{-1}$$

and

$$\kappa_{\mathrm{DRP}}(\mathbf{X}, \mathbf{R}, \lambda) \le \left(1 - 2C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}\log\left(\frac{1}{\delta}\right)}\right)^{-1}$$

with probability at least $1-\delta$.

Proof is provided in Appendix A.3. With Theorem 6, we immediately obtain the following corollary, which states the overall convergence of Acc-IHS and Acc-IDRP.

Corollary 7. *Suppose conditions of Theorem 6 hold. If the number of iterations of Acc-IHS satisfy*

$$t \ge \left\lceil \left(1 - 2C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}\right)^{-1/2} \log\left(\frac{2\left\|\mathbf{w}^*\right\|_{\mathbf{X}}}{\epsilon}\right)\right\rceil,$$

then with probability at least $1-\delta$, we have

$$\left\|\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)} - \mathbf{w}^*\right\|_{\mathbf{X}} \le \epsilon.$$

If the number of iterations of Acc-IDRP satisfies

$$t \ge \left\lceil \left(1 - 2C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}\log\left(\frac{1}{\delta}\right)}\right)^{-1/2} \log\left(\frac{2\left\|\mathbf{w}^*\right\|_2}{\epsilon}\right)\right\rceil$$

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

$$\left\|\widehat{\mathbf{w}}_{\mathrm{DRP}}^{(t)} - \mathbf{w}^*\right\|_2 \leq \epsilon.$$

We can compare the convergence rate of Acc-IHS and Acc-IDRP with that of the standard IHS [33] and the IDRP [53].

Remark 5. *The number of iterations to reach* ϵ*-accuracy for IHS is* $\mathcal{O}\left(\left(\frac{1+\rho}{1-\rho}\right)\log\left(\frac{2\|\mathbf{w}^*\|_{\mathbf{X}}}{\epsilon}\right)\right)$, *where* $\rho = C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}$ *[Corollary 1 33]. Acc-IHS reduces the number of iterations to* $\mathcal{O}\left(\left(\sqrt{\frac{1}{1-2\rho}}\right)\log\left(\frac{2\|\mathbf{w}^*\|_{\mathbf{X}}}{\epsilon}\right)\right)$, *which is significantly smaller when* ρ *is relatively large. Furthermore, IHS requires* $m \gtrsim \mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})$ *for the convergence to happen, while Acc-IHS is always guaranteed to converge. This will be illustrated in simulations.*

Remark 6. *In a setting with low-rank data, [Theorem 7 53] showed that IDRP requires* $\mathcal{O}\left(\frac{1+\rho}{1-\rho}\right)\log\left(\frac{2\|\mathbf{w}^*\|_2}{\epsilon}\right)$ *to* ϵ*-accuracy where* $\rho = C_0\sqrt{\frac{r}{d}\log\left(\frac{r}{\delta}\right)}$. *Acc-IDRP reduces the number of iterations to* $\mathcal{O}\left(\sqrt{\frac{1}{1-2\rho}}\right)\log\left(\frac{2\|\mathbf{w}^*\|_2}{\epsilon}\right)$ *and, furthermore, relaxes the stringent condition* $d \gtrsim r\log r$ *needed for IDRP to converge, since Acc-IDRP always converges.*

5.3. Analysis of the primal-dual sketch method

In this section, we provide analysis for the primal-dual sketch method. Recall that here the sketched dual problem is not solved exactly, but is approximately solved by sketching the primal problem again.

Consider an outer loop iteration t. The analysis of the iterative Hessian sketch gives us the following lemma on the decrease of the error.

Lemma 8. *Let* $\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}$ *be the iterate defined in Algorithm 1. Then we have*

$$\left\|\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)} - \mathbf{w}^*\right\|_{\mathbf{X}} \leq \frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}}{1 - C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}}\left\|\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)} - \mathbf{w}^*\right\|_{\mathbf{X}}.$$

Note, however, that in the iterative primal-dual sketch, we do not have access to the exact minimizer $\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}$. Instead, we have an *approximate* minimizer $\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}$, which is close to $\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}$. So it remains to analyze the iteration complexity of the inner loop and see how close the approximate minimizer $\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}$ is to the optimal solution $\mathbf{w}^*$. We have the following theorem.

Theorem 9. *With probability at least* $1-\delta$, *we have the following approximation error bound for* $\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}$ *in the iterative primal-dual sketch:*

$$\begin{aligned}
&\left\|\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)} - \mathbf{w}^*\right\|_{\mathbf{X}} \\
&\leq \left(\frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}}\log\left(\frac{1}{\delta}\right)}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}}\log\left(\frac{1}{\delta}\right)}\right)^t \|\mathbf{w}^*\|_{\mathbf{X}} \\
&\quad + \frac{10\lambda_{\max}^2\left(\frac{\mathbf{X}^\top\mathbf{X}}{n}\right)}{\lambda^2}\left(\frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}\right)^k \|\mathbf{w}^*\|_2 .
\end{aligned}$$

The proof is given in Appendix A.5.

With Theorem 9, we have the following iterative complexity for the proposed IPDS approach.

Corollary 10. *If the number of outer iterations t and number of inner iterations k in the IPDS satisfy*

$$\begin{aligned}
t &\geq \left\lceil \frac{1+C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}}\log\left(\frac{1}{\delta}\right)}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}}\log\left(\frac{1}{\delta}\right)}\right\rceil \log\left(\frac{4\|\mathbf{w}^*\|_{\mathbf{X}}}{\epsilon}\right), \\
k &\geq \left\lceil \frac{1+C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}\right\rceil \log\left(\frac{40\lambda_{\max}^2\left(\frac{\mathbf{X}^\top\mathbf{X}}{n}\right)\|\mathbf{w}^*\|_2}{\lambda\epsilon}\right),
\end{aligned}$$

then with probability at least $1-\delta$:

$$\left\|\widetilde{\mathbf{w}}_{\mathrm{IPDS}}^{(t+1)} - \mathbf{w}^*\right\|_{\mathbf{X}} \leq \epsilon.$$

Proof. The result directly follows by an application of Theorem 9. □

Remark 7. *The total number of sketched subproblem to solve in iterative primal-dual sketch is $t \cdot k$. To obtain ϵ approximation error, the total number of subproblems is*

$$tk \lesssim \left\lceil \frac{1+\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}}}{1-\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}}} \cdot \frac{1+\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}}{1-\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}}\right\rceil \log^2\left(\frac{1}{\epsilon}\right).$$

Thus the iterative primal-dual sketch will be efficient when the Gaussian width of set $\mathbf{X}\mathbb{R}^p$ and $\mathbf{X}^\top\mathbb{R}^n$ is relatively small. For example, when $\mathrm{rank}(\mathbf{X}) = r \ll \min(n,p)$, we can choose the sketching dimension in IPDS to be $m, d \gtrsim r$. In this case the IPDS can return a solution with ϵ-approximation error by solving $\log^2(1/\epsilon)$ small scale subproblems of scale $r \times r$.

We next provide iteration complexity for the proposed Acc-IPDS algorithms in Algorithm 7.

Corollary 11. *If the number of outer loops t and number of inner loops k in IPDS satisfy*

$$t \geq \left\lceil \left(1 - 2C_0 \sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p \cap \mathcal{S}^{n-1})}{m}} \log\left(\frac{1}{\delta}\right)\right)^{-1/2} \right\rceil \log\left(\frac{4\left\|\mathbf{w}^*\right\|_{\mathbf{X}}}{\epsilon}\right),$$

$$k \geq \left\lceil \left(1 - 2C_0 \sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n \cap \mathcal{S}^{p-1})}{d}} \log\left(\frac{1}{\delta}\right)\right)^{-1/2} \right\rceil$$
$$\times \log\left(\frac{40\lambda_{\max}^2\left(\frac{\mathbf{X}^\top\mathbf{X}}{n}\right)\left\|\mathbf{w}^*\right\|_2}{\lambda\epsilon}\right),$$

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

$$\left\|\widetilde{\mathbf{w}}_{\mathrm{IPDS}}^{(t+1)} - \mathbf{w}^*\right\|_{\mathbf{X}} \leq \epsilon.$$

Proof. The proof is similar to the proof of Theorem 9. We simply need to substitute the lower bounds for t and k to obtain the desired result. □

5.4. Runtime comparison for large n, large p, and low-rank data

To solve problem (2.1), the runtime usually depends on several quantities including the sample size n, the dimension p, as well as the condition number. To make the comparison between different algorithms, we simply assume $\mathbf{X}$ is of rank r, noting that r might be much smaller than n and p. In (2.1), the regularization parameter λ is generally chosen at the order of $\mathcal{O}(1/\sqrt{n})$ to $\mathcal{O}(1/n)$ [42, 8]. Here, we simply consider the large value for λ, that is, of order $\mathcal{O}(1/\sqrt{n})$, which gives a better condition number for the problem. For iterative optimization algorithms, the convergence depends on the smoothness parameter of the problem. In (2.1), the smoothness parameter is $\lambda_{\max}\left(\frac{\mathbf{X}^\top\mathbf{X}}{n} + \lambda\mathbf{I}_p\right)$, which is often of the order $\mathcal{O}(p)$, for example, under a random sub-Gaussian design. To attain the runtime of solving (2.1) in different scenarios, we consider the following methods which are summarized in Table 2 with their time complexities in terms of stated parameters:

Solving Linear System: which solves the problem exactly using matrix inversion, and requires $\mathcal{O}(np^2 + p^3)$.

Linear System with Low-rank SVD: if we have the factorization $\mathbf{X} = \mathbf{U}\mathbf{V}^\top$ available, where $\mathbf{U} \in \mathbb{R}^{n\times r}$, $\mathbf{V} \in \mathbb{R}^{p\times r}$, then we can solve the matrix inversion efficiently using the Sherman-Morrison-Woodbury formula: $\left(\lambda\mathbf{I}_p + \frac{\mathbf{X}^\top\mathbf{X}}{n}\right)^{-1} = \frac{1}{\lambda}\mathbf{I}_p - \frac{1}{\lambda^2}\mathbf{V}\mathbf{U}^\top\mathbf{U}(\mathbf{I}_r + \mathbf{V}^\top\mathbf{V}\mathbf{U}^\top\mathbf{U})^{-1}\mathbf{V}^\top$. This can be done in $\mathcal{O}(npr + r^3)$ in total.

Gradient Descent: standard analysis [27] shows that the gradient descent requires $\mathcal{O}\left(\left(\frac{L}{\lambda}\right)\log\left(\frac{1}{\varepsilon}\right)\right)$ iterations, with each iteration has a time complexity of

Approach / Runtime	$\mathcal{O}(\cdot)$	Comment
Linear System	$np^2 + p^3$	
LS with Low-rank SVD	$npr + r^3$	
Gradient Descent	$\left(n^{1.5}p^2\right) \log\left(\frac{1}{\varepsilon}\right)$	
Acc.Gradient Descent	$\left(n^{1.25}p^{1.5}\right) \log\left(\frac{1}{\varepsilon}\right)$	
Coordinate Descent	$\left(n^{1.5}p\right) \log\left(\frac{1}{\varepsilon}\right)$	
SVRG,SDCA,SAG	$\left(np + n^{0.5}p^2\right) \log\left(\frac{1}{\varepsilon}\right)$	
Catalyst,APPA	$\left(np + n^{0.75}p^{1.5}\right) \log\left(\frac{1}{\varepsilon}\right)$	
DSPDC	$npr + \left(nr + n^{0.75}p^{1.5}r\right) \log\left(\frac{1}{\varepsilon}\right)$	
IHS + Catalyst	$np \log p + n^{0.25}p^{1.5}r \log^2\left(\frac{1}{\varepsilon}\right)$	Fast when $p \ll n$
DRP + Exact	$np \log n + (nr^2 + r^3) \log\left(\frac{1}{\varepsilon}\right)$	Fast when $n \ll p$
Iter.primal-dual sketch	$np \log p + (n + r^3) \log^2\left(\frac{1}{\varepsilon}\right)$	Fast when $r \ll \max(p, n)$

TABLE 2

Comparison of runtime of different approaches for solving the large scale optimization problem in (2.1) stated in terms of number of samples n, the dimensionality of data points p, the rank of data matrix r, and the target accuracy of recovered solution ε.

$\mathcal{O}(np)$ to compute the full gradient for all training samples. Since $L = \mathcal{O}(p), \lambda = \mathcal{O}\left(1/\sqrt{n}\right)$, the overall runtime becomes $\mathcal{O}\left(\left(n^{1.5}p^2\right) \log\left(\frac{1}{\varepsilon}\right)\right)$.

Accelerated Gradient Descent [27]: which requires $\mathcal{O}\left(\sqrt{\left(\frac{L}{\lambda}\right)} \log\left(\frac{1}{\varepsilon}\right)\right)$ iterations, where the cost of each iteration is $\mathcal{O}(np)$. For the stated values of parameters $L = \mathcal{O}(p)$ and $\lambda = \mathcal{O}\left(1/\sqrt{n}\right)$, the overall runtime would be $\mathcal{O}\left(\left(n^{1.25}p^{1.5}\right) \log\left(\frac{1}{\varepsilon}\right)\right)$.

Randomized Coordinate Descent [26]: which requires $\mathcal{O}\left(p\left(\frac{1}{\lambda}\right) \log\left(\frac{1}{\varepsilon}\right)\right)$ iterations, with each iteration $\mathcal{O}(n)$, since $\lambda = \mathcal{O}\left(1/\sqrt{n}\right)$. We have the overall runtime is $\mathcal{O}\left(\left(n^{1.5}p\right) \log\left(\frac{1}{\varepsilon}\right)\right)$.

SVRG, SDCA, SAG [16, 51, 39, 37]: which requires $\mathcal{O}\left(\left(n + \frac{L}{\lambda}\right) \log\left(\frac{1}{\varepsilon}\right)\right)$ iterations, with the time complexity of $\mathcal{O}(p)$ for each iteration to computed the gradient of simple sample. Since $L = \mathcal{O}(p), \lambda = \mathcal{O}\left(1/\sqrt{n}\right)$, the overall runtime for this family of algorithms would be $\mathcal{O}\left(\left(np + n^{0.5}p^2\right) \log\left(\frac{1}{\varepsilon}\right)\right)$.

Accelerated SVRG: Catalyst, APPA, SPDC, RPDG [20, 11, 54, 18]: thanks to acceleration, this algorithm requires $\mathcal{O}\left(\left(n + \sqrt{n\frac{L}{\lambda}}\right) \log\left(\frac{1}{\varepsilon}\right)\right)$ iterations, with each iteration shares the same $\mathcal{O}(p)$ complexity per iteration as SVRG. Since $L = \mathcal{O}(p), \lambda = \mathcal{O}\left(1/\sqrt{n}\right)$, the overall runtime becomes $\mathcal{O}\left(\left(np + n^{0.75}p^{1.5}\right) \log\left(\frac{1}{\varepsilon}\right)\right)$.

DSPDC [50]: requires $\mathcal{O}\left(\left(n + \sqrt{n\frac{L}{\lambda}p}\right) \log\left(\frac{1}{\varepsilon}\right)\right)$ iterations, and each iteration is in order of $\mathcal{O}(r)$. Here $L = \mathcal{O}(p), \lambda = \mathcal{O}\left(1/\sqrt{n}\right)$. Also, to apply DSPDC, one should compute the low-rank factorization as a preprocessing step which takes $\mathcal{O}(npr)$. Thus we have the overall runtime for this algorithm as $\mathcal{O}\left(npr + \left(nr + n^{0.75}p^{0.5}r\right) \log\left(\frac{1}{\varepsilon}\right)\right)$.

Iterative Hessian Sketch + Accelerated SVRG [33]: computing the sketched problem takes $\mathcal{O}(np \log p)$ (e.g., via fast Johnson-Lindenstrauss transforms [1]). The algorithms solves $\mathcal{O}\left(\log\left(\frac{1}{\varepsilon}\right)\right)$ sketched problems using accelerated SVRG type algorithm that takes $\mathcal{O}\left(n^{0.25}p^{1.5}r \log\left(\frac{1}{\varepsilon}\right)\right)$. This leads to the overall runtime of $\mathcal{O}\left(np \log p + n^{0.25}p^{1.5}r \log^2\left(\frac{1}{\varepsilon}\right)\right)$.

DRP + Matrix inversion [53]: computing the sketched problem takes $\mathcal{O}(np\log n)$. The algorithms needs to solve $\mathcal{O}\left(\log\left(\frac{1}{\varepsilon}\right)\right)$ reduced problems where each of them requires a matrix inversion with time complexity of $\mathcal{O}\left(nr^2+r^3\right)$. This leads to the overall runtime of $\mathcal{O}\left(np\log n+(nr^2+r^3)\log\left(\frac{1}{\varepsilon}\right)\right)$ for this algorithm.

Iterative Primal-Dual Sketch: computing the sketched problem takes $\mathcal{O}(np\log p)$. The algorithms iterates for $\mathcal{O}\left(\log^2\left(\frac{1}{\varepsilon}\right)\right)$ rounds, and at each iteration it needs to solve a reduced problem that exactly takes $\mathcal{O}\left(n+r^3\right)$. As a result the overall runtime becomes as $\mathcal{O}\left(np\log p+(n+r^3)\log^2\left(\frac{1}{\varepsilon}\right)\right)$.

6. Application to distributed optimization

In this section we apply the improved iterative sketching in the distributed optimization problems. Typically, distributed optimization approaches can be divided into two categories, depending how the data set is partitioned across different machines: data could be partitioned across features [14, 13, 45, 47] or it could be partitioned by samples [40, 55, 19, 44, 17, 41]. For the setting where features are partitioned across machines, we propose the (accelerated) iterative distributed dual random projection (DIDRP). In the setting where samples are partitioned across machines, we propose the (accelerated) iterative distributed Hessian sketch (DIHS). We discuss in detail how these proposals compare to and improve over existing work.

6.1. Distributed iterative dual random projection

We first consider a setting where features are distributed across different machines. In this setting, LOCO [14] and Dual-LOCO [13] considered sketching based approaches, where randomly projected data are transmitted across machines to approximate the original data. However, as predicted by theory, these one-shot approaches require communicating a very large number of vectors in order to obtain a high accuracy solution for the original optimization problem. On the other hand, iterative sketching methods are very powerful in reducing the approximation error by solving a different problem using the same sketched data. At the same time, once we have transmitted the sketched data matrix, at every iterative sketching round each machine only needs to communicate two vectors in $\mathbb{R}^n$ to solve the next sketched problem.

Suppose $\mathbf{X}\in\mathbb{R}^{n\times d}$ is partitioned across features over m machines, $\mathbf{X}=[\mathbf{X}_{[1]},\mathbf{X}_{[2]},\ldots,\mathbf{X}_{[m]}]$, such that machine k holds $\mathbf{X}_{[k]}$ consisting of p/m features (for simplicity we assume that m divides p). Without loss of generality, assume the first machine serves as the master machine, and it contains the local data $\mathbf{X}_{[1]}$, as well as the transmitted, randomly projected data $[\mathbf{X}_{[2]}\mathbf{R}_{[2]},\mathbf{X}_{[3]}\mathbf{R}_{[3]},\ldots,\mathbf{X}_{[m]}\mathbf{R}_{[m]}]$. Let $\widetilde{\mathbf{X}}=[\mathbf{X}_{[1]},\mathbf{X}_{[2]}\mathbf{R}_{[2]},\mathbf{X}_{[3]}\mathbf{R}_{[3]},\ldots,\mathbf{X}_{[m]}\mathbf{R}_{[m]}]$ be the concatenated data matrix which contains full local data and the sketched global data. Here each random matrix $\mathbf{R}_{[k]}$, $k=2,\ldots,m$ is of dimension $(p/m)\times(d/(m-1))$, so that the dimension of $\widetilde{\mathbf{X}}$ is $n \times (p/m) + d$. At each iteration, the master machine solves the following problem:

$$\min_{\mathbf{z}} \frac{1}{2n} \left\| \mathbf{y} - \mathbf{X}\widehat{\mathbf{w}}^{(t)} - \widetilde{\mathbf{X}}\mathbf{z} \right\|_2^2 + \frac{\lambda}{2} \left\| \mathbf{z} + \widehat{\mathbf{w}}^{(t)} \right\|_2^2, \tag{6.1}$$

where $\widehat{\mathbf{w}}^{(t)} = [\mathbf{w}_{[1]}^{(t)}; \mathbf{R}_{[2]}^\top \mathbf{w}_{[2]}^{(t)}; \ldots; \mathbf{R}_{[m]}^\top \mathbf{w}_{[m]}^{(t)}]$. In order to do so, each machine communicates $\mathbf{X}_{[k]}\widehat{\mathbf{w}}_{[k]}^{(t)}$, and the master machine aggregates and computes $\mathbf{X}\widehat{\mathbf{w}}^{(t)} = \sum_{k=1}^m \mathbf{X}_{[k]}\widehat{\mathbf{w}}_{[k]}^{(t)}$. With $\widehat{\mathbf{z}}$ obtained by solving (6.1), the master machine can update the dual solution $\widehat{\boldsymbol{\alpha}}^{(t+1)}$ and communicate it back to each machine. Each machine, in turn, uses the obtained $\widehat{\boldsymbol{\alpha}}^{(t+1)}$ to updated their local primal solution as $\widehat{\mathbf{w}}_{[k]}^{(t+1)} = \frac{1}{\lambda n} X_{[k]}^\top \widehat{\boldsymbol{\alpha}}^{(t+1)}$. The details of the algorithm are presented in Algorithm 8. It is noteworthy to point out that after the initial transmission stage, in each iteration, each worker only communicates two vectors in $\mathbb{R}^n$ to the master machine.

The following corollary states the communication complexity of Algorithm 8 which is a direct consequence of Theorem 2.

Corollary 12. *Suppose that sub-Gaussian sketching matrices were used in Algorithm 8. For Algorithm 8 to reach ϵ accuracy, $\left\| \widehat{\mathbf{w}}^{(t)} - \mathbf{w}^* \right\|_2 \leq \epsilon$, the total number of vectors (in $\mathbb{R}^n$) each machine needs to communicate is upper bounded by*

$$\mathcal{O}\left(\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p)}{m-1} + \log\left(\frac{\|\mathbf{w}^*\|_2}{\epsilon} \right) \right).$$

Remark 8. *We can compare our result with that established for Dual-LOCO [13]. Dual-LOCO requires the number of communication rounds to linearly with $1/\epsilon^2$. On the other hand, the number of communication rounds of DIDRP only grow logarithmically with $1/\epsilon$. Therefore, DIDRP presents a significant improvement over Dual-LOCO. This is also verified by the empirical results.*

6.2. Distributed iterative Hessian sketch

Next, we consider a setting where the data are partitioned by samples. The data matrix $\mathbf{X}$ is partitioned as $\mathbf{X} = [\mathbf{X}_{(1)}; \mathbf{X}_{(2)}; \ldots; \mathbf{X}_{(m)}]$ where each machine k holds the local data $\mathbf{X}_{(k)} \in \mathbb{R}^{n/m \times p}$, which contains n/m samples. In this setting, our main idea is to approximate the Hessian matrix with a mix of local data and sketched global data. Again, assume the first machine serves as the master. At the beginning of the algorithm, workers compute and communicate their sketched local data $\mathbf{\Pi}_{(k)}^\top \mathbf{X}_{(k)}$. The master constructs the sketched data matrix as $\widetilde{\mathbf{X}} = [\mathbf{X}_{(1)}; \mathbf{\Pi}_{(2)}^\top \mathbf{X}_{(2)}; \ldots; \mathbf{\Pi}_{(m)}^\top \mathbf{X}_{(m)}]$, which will be used for constructing an approximation to the Hessian matrix as $\widetilde{\mathbf{H}} = \frac{\widetilde{\mathbf{X}}^\top \widetilde{\mathbf{X}}}{n/m+d}$. At each iteration of the algorithm, the master solves a sub-problem of form

$$\widehat{\mathbf{u}}^{(t)} = \arg\min_{\mathbf{u}} \mathbf{u}^\top \left(\widetilde{\mathbf{H}} + \frac{\lambda}{2}\mathbf{I}_p \right) \mathbf{u} - \left\langle \frac{\mathbf{X}^\top(\mathbf{y} - \mathbf{X}\widehat{\mathbf{w}}^{(t)})}{n} - \lambda\widehat{\mathbf{w}}^{(t)}, \mathbf{u} \right\rangle, \tag{6.2}$$

Algorithm 8: Distributed Iterative Dual Random Projection (DIDRP).

```
Input: Data $\mathbf{X}, \mathbf{y}$.
Initialization: $\widehat{\mathbf{w}}^{(0)} = \mathbf{0}$.
for Each worker $k = 2, ..., m$ do
    Compute and communicate randomly projected data $\mathbf{X}_{[k]}\mathbf{R}_{[k]}$.
end
for $t = 0, 1, 2, \ldots$ do
    The master machine solves the projected problem in (6.1), and obtains $\widehat{\mathbf{z}}^{(t)}$.
    The master machine computes and communicates the dual approximation:
    $\widehat{\boldsymbol{\alpha}}^{(t+1)} = \mathbf{y} - \mathbf{X}\widehat{\mathbf{w}}^{(t)} - \widetilde{\mathbf{X}}\widehat{\mathbf{z}}^{(t)}$.
    for Each worker $k = 2, ..., m$ do
        Update local primal approximation: $\widehat{\mathbf{w}}_{[k]}^{(t+1)} = \frac{1}{\lambda n}\mathbf{X}_{[k]}^\top \widehat{\boldsymbol{\alpha}}^{(t+1)}$.
        Compute and communicate $\mathbf{X}_{[k]}\widehat{\mathbf{w}}_{[k]}^{(t+1)}$.
    end
end
```

which is inspired by the iterative Hessian sketch. The quantity $\mathbf{X}^\top\mathbf{X}\widehat{\mathbf{w}}^{(t)}$ is computed by communicating and aggregating the local information $\mathbf{X}^\top\mathbf{X}\widehat{\mathbf{w}}^{(t)} = \sum_{k=1}^{m}\mathbf{X}_{(k)}^\top\mathbf{X}_{(k)}\widehat{\mathbf{w}}^{(t)}$. The details of the algorithm DIHS are presented in Algorithm 9. The following corollary on its communication efficiency is a direct consequence of Theorem 2.

Corollary 13. *Suppose we use sub-Gaussian sketching in Algorithm 8 and for Algorithm 9 to reach ϵ approximation: $\left\|\widehat{\mathbf{w}}^{(t)} - \mathbf{w}^*\right\|_{\mathbf{X}} \leq \epsilon$, the total number of vectors (in $\mathbb{R}^p$) each machine need to communicate is upper bounded by*

$$\mathcal{O}\left(\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p)}{m-1} + \log\left(\frac{\|\mathbf{w}^*\|_{\mathbf{X}}}{\epsilon}\right)\right).$$

Acceleration The acceleration techniques presented in Section 2 and 3 can also be applied in the distributed optimization setting to further improve the communication efficiency of DIDRP and DIHS. In the experiments, we found that the accelerated algorithms can often help in saving communication because of their faster convergence.

7. Experiments

In this section we present extensive comparisons for the proposed iterative sketching approaches on both simulated and real world data sets. We first demonstrate the improved convergence of the proposed Acc-IHS and Acc-IDRP algorithms on simulated data sets. Then we show that the proposed iterative primal-dual sketch procedure and its accelerated version could simultaneously reduce the sample size and dimension of the problem, while still maintaining high approximation precision. Finally, we evaluate these algorithms on some real world data sets.

Algorithm 9: Distributed Iterative Hessian Sketch (DIHS).

```
Input: Data $\mathbf{X}, \mathbf{y}$.
Initialization: $\widehat{\mathbf{w}}^{(0)} = \mathbf{0}$.
for Each work $k = 2, ..., m$ do
    Compute and communicate randomly projected data $\mathbf{\Pi}_{(k)}^\top \mathbf{X}_{(k)}$.
end
for $t = 0, 1, 2, \ldots$ do
    for Each worker $k = 2, ..., m$ do
        Compute and communicate $\mathbf{X}_{(k)}^\top \mathbf{X}_{(k)} \widehat{\mathbf{w}}^{(t)}$.
    end
    The master machine computes $\widehat{\mathbf{w}}^{(t+1)} = \widehat{\mathbf{w}}^{(t)} + \widehat{\mathbf{u}}^{(t)}$, where $\widehat{\mathbf{u}}^{(t)}$ is obtained by
    solving the sketched problem (6.2), and communicates $\widehat{\mathbf{w}}^{(t+1)}$.
end
```

7.1. Simulations for Acc-IHS and Acc-IDRP

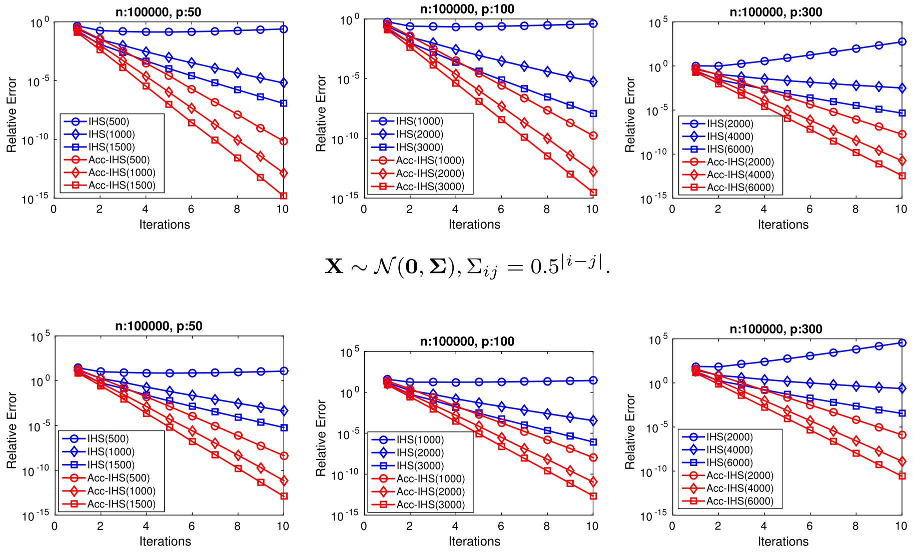
We first examine the effectiveness of the proposed Acc-IHS and Acc-DRP algorithms on simulated data. We generate the response $\{y_i\}_{i\in[n]}$ from the following linear model

$$y_i = \langle \mathbf{x}_i, \boldsymbol{\beta}^* \rangle + \epsilon_i,$$

where the noise ϵ_i is sampled from a standard Normal distribution. The true model $\boldsymbol{\beta}^*$ is a p-dimensional vector where the entries are sampled i.i.d. from a uniform distribution in $[0, 1]$.

We first compare the proposed Acc-IHS with the standard IHS on some "big n", but relatively low-dimensional problems. We generate $\{\mathbf{x}_i\}_{i\in[n]}$ from a multivariate Normal distribution with mean zero vector, and covariance matrix $\mathbf{\Sigma}$, which controls the condition number of the problem. We will varying $\mathbf{\Sigma}$ to see how it affects the performance of various methods. We set $\Sigma_{ij} = 0.5^{|i-j|}$ for the well-conditioned setting, and $\Sigma_{ij} = 0.5^{|i-j|/10}$ for the ill-conditioned setting. We fix the sample size $n = 10^5$ and vary the dimension $p \in \{50, 100, 300\}$. The results are shown in Figure 1. For each problem setting, we test 3 different sketching dimensions (number inside parentheses in legend). We have the following observations:

- For both IHS and Acc-IHS, the larger the sketching dimension m, the faster the iterative algorithms converges to the optimum, which is consistent with the theory that characterize the benefit of using larger sketching dimension. And this has also been observed in [33] and [53] for IHS and IDRP algorithms.
- When compared with IHS, we observe that Acc-IHS converges significantly faster. Moreover, when the sketching dimension is small, IHS can diverge and go far away from the optimum, while Acc-IHS still converges.
- For all the simulation setting we tried, Acc-IHS converges faster than IHS, even when its sketching dimension is only 1/3 of the sketching dimension of IHS.



$$\mathbf{X} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}), \Sigma_{ij} = 0.5^{|i-j|}.$$

$$\mathbf{X} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}), \Sigma_{ij} = 0.5^{|i-j|/10}.$$

Fig 1. *Comparison of IHS and Acc-IHS on various simulated datasets. The sketching dimension for each algorithm is shown inside parentheses.*

Next, we compare the proposed Acc-IDRP with the standard IDRP on high-dimensional, but relatively low-rank data. We generate $\{\mathbf{x}_i\}_{i\in[n]}$ from a low-rank factorization $\mathbf{X} = \mathbf{U}\mathbf{V}^\top$, where the entries in $\mathbf{U} \in \mathbb{R}^{n\times r}$ and $\mathbf{V} \in \mathbb{R}^{p\times r}$ are sampled i.i.d. from a standard Normal distribution. We fix the sample size $n = 10^4$ and vary the dimensions $p \in \{2000, 5000, 20000\}$. We also vary the rank $r \in \{20, 50\}$. The results are shown in Figure 2. For each problem setting, we test 3 different sketching dimensions (number inside parentheses in legend). We have similar observations as in the IHS case. Acc-IDRP always converges significantly faster than IDRP. When the low sketching dimension causes IDRP to diverge, Acc-IDRP still converges to the optimum.

Above simulations validate the theoretical analysis, which showed that the accelerated procedures for IHS and IDRP could significantly boost the convergence speed of their standard counterparts. Since the computational cost per iteration of the standard iterative sketching techniques and their accelerated versions is almost the same, Acc-IHS and Acc-IDRP will be useful practical techniques.

7.2. Simulations for IPDS and Acc-IPDS

In this section we demonstrate how iterative primal-dual sketch and its accelerated version work on simulated data. We generated the data using the same procedure described in the previous section for Acc-DRP. We generate

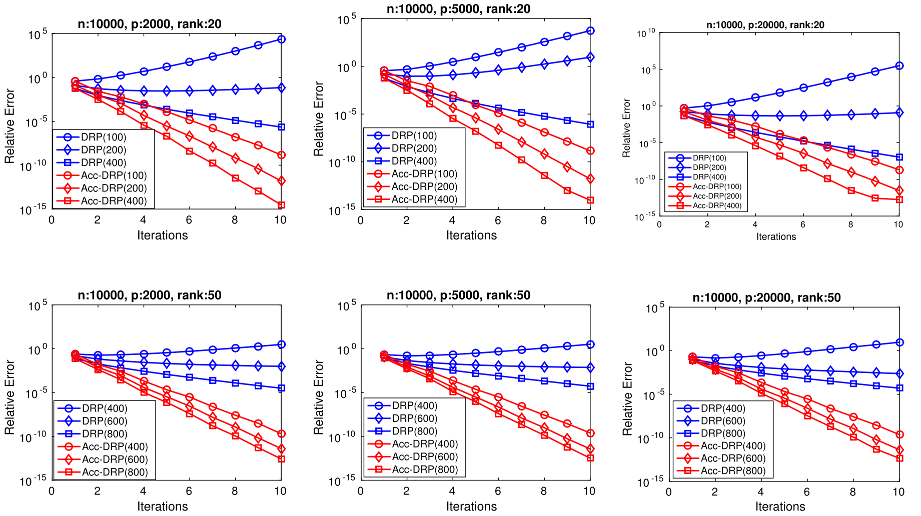
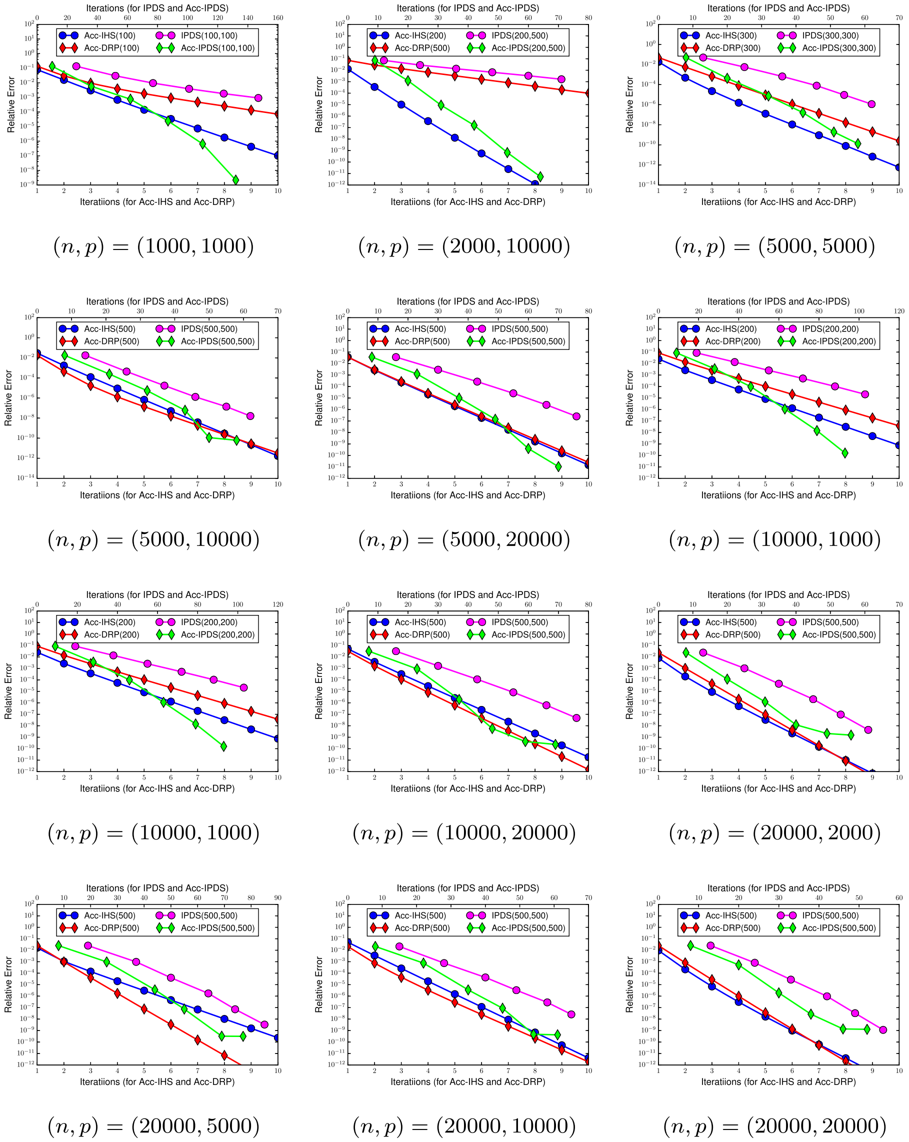


FIG 2. *Comparison of IDRP and Acc-IDRP on various simulated datasets.*

the low-rank data matrix $\mathbf{X}$ with rank 10 and vary the sample size n and dimension p. For primal-dual sketching, we reduce the sample size to m and the dimension to d, with $m \ll n, d \ll p$. We compare with the standard IHS and IDRP. For IHS, we perform the sample reduction from n to m, while for IDRP we perform data dimension reduction from p to d. Thus the sizes of the subproblems for IPDS (and Acc-IPDS), IHS, and IDRP are $m \times d$, $m \times p$, and $n \times d$, respectively. For IPDS and Acc-IPDS, we terminate the inner loop when the ℓ_∞ distance between two inner iterations is less than 10^{-10}. The results are shown in Figure 3, where the sketched dimension (m, d) is shown in legend.

We have the following observations:

- IPDS and Acc-IPDS are able to recover the optimum to a very high precision, even though they simultaneously reduce the sample size and data dimension. However, they generally require more iterations to reach certain approximation level compared with IHS and IDRP, which, on the other hand, need to solve a substantially larger subproblem at each iteration. Therefore, primal-dual sketching approach still enjoys computational advantages. For example, on a problem of size $(n, p) = (10000, 20000)$, IHS and IDRP need to solve 5 sub-problems of scale $(m, p) = (500, 20000)$ and $(n, d) = (10000, 500)$, respectively, while Acc-IPDS is only required to solve 35 sub-problems of scale $(m, d) = (500, 500)$ to obtain the same approximation accuracy.
- Acc-IPDS converges significantly faster than IPDS, which again verifies the effectiveness of the proposed acceleration procedure for the sketching techniques.



$(n, p) = (1000, 1000)$ $(n, p) = (2000, 10000)$ $(n, p) = (5000, 5000)$

$(n, p) = (5000, 10000)$ $(n, p) = (5000, 20000)$ $(n, p) = (10000, 1000)$

$(n, p) = (10000, 1000)$ $(n, p) = (10000, 20000)$ $(n, p) = (20000, 2000)$

$(n, p) = (20000, 5000)$ $(n, p) = (20000, 10000)$ $(n, p) = (20000, 20000)$

FIG 3. *Comparion of IPDS and Acc-IPDS versus with IHS and DRP various simulated datasets.*

7.3. Experiments on real data sets

In this section, we present experiments conducted on real-world data sets. Table 3 summarizes their statistics. Among these data sets, the first 3 are cases where sample size is significantly larger than the data dimension. We use them

Table 3
List of real-world data sets used in the experiments.

Name	#Instances	#Features
connect4	67,557	126
slice	53,500	385
year	51,630	90
colon-cancer	62	2,000
duke breast-cancer	44	7,129
leukemia	72	7,129
cifar	4,047	3,072
gisette	6,000	5,000
sector	6,412	15,000
mnist	60,000	780
tomes	28,179	96
twitter	582,350	77

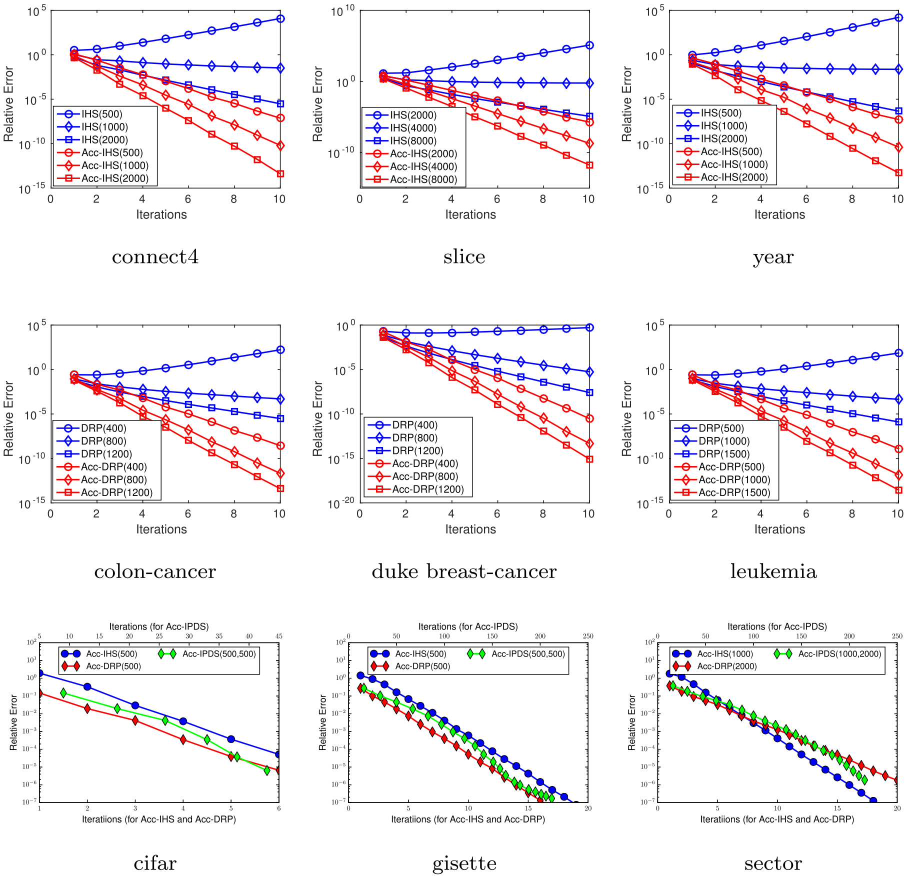
to compare the IHS and Acc-IHS algorithms. The middle 3 data sets are high-dimensional data sets with small sample sizes. We use them to compare the DRP and Acc-DRP algorithms. Finally, the last 3 data sets are cases where the sample size and data dimensions are both relatively large, which is suitable for iterative primal-dual sketching methods. For the last 3 data sets, we found that standard IHS and DRP often fail, unless a very large sketching dimension is used. As a result, we compared with Acc-IHS and Acc-DRP algorithms. We follow the same experimental setup used in the simulation study. The convergence plots are summarized in Figure 4.

We have the following observations:

- Acc-IHS and Acc-DRP converge significantly faster than IHS and DRP, respectively. This is consistent with the observation drawn from simulation studies.
- For the last 3 data sets, where n and p are both large, and the data are not exactly low-rank, IHS, DRP, and IPDS often diverge. This is because the requirement on the sketching dimension to ensure convergence is high. The accelerated versions still converge to the optimum. It is notable that the Acc-IPDS only requires solving several least squares problems with both sample size and dimension being relatively small.

7.4. Experiments for distributed optimization

We consider distributed optimization in both partition by features and partition by samples settings. We follow the data generating process as in the simulation study for Acc-DRP, where we fix a low-rank data matrix $\mathbf{X}$ with rank 10, and vary the sample size n and dimension p, as well as number of machines m. For partition by features scenario, we compare with LOCO and Dual-LOCO [14, 13]. We plot the relative approximation error versus the number of vectors (in $\mathbb{R}^n$) communicated. The results are shown in Figure 6. We LOCO and Dual-LOCO fail to quickly decrease the approximation error even with relatively large com-

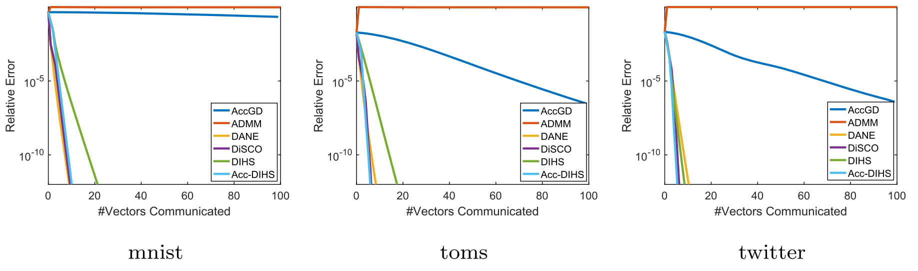


connect4 slice year

colon-cancer duke breast-cancer leukemia

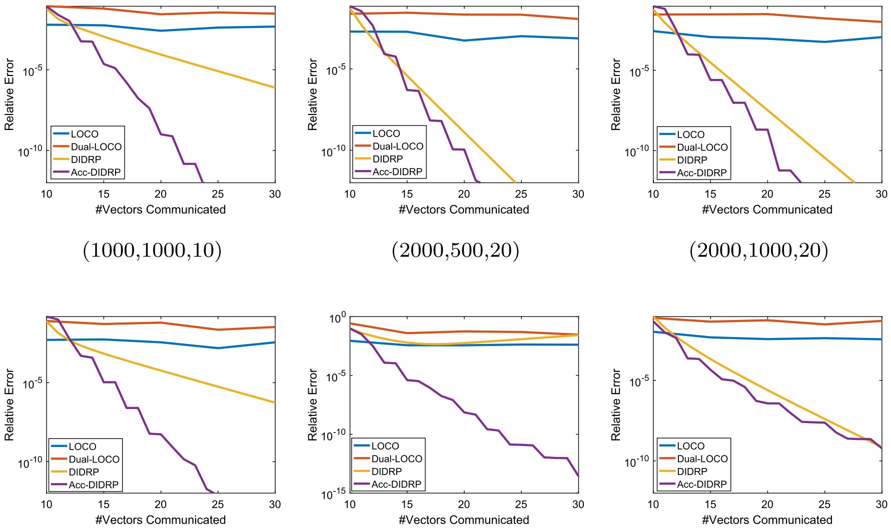
cifar gisette sector

Fig 4. *Comparion of various iterative sketching approaches on real-world data sets. Top row: Acc-IHS versus IHS, middle row: Acc-DRP versus DRP, bottom row: Acc-IPDS versus Acc-IHS and Acc-DRP.*



mnist toms twitter

Fig 5. *Comparion of various approaches for distributed optimization on several real world data sets under the partition by sample setting.*



(1000,1000,10) (2000,500,20) (2000,1000,20)

(2000,2000,120) (3000,3000,10) (5000,1000,20)

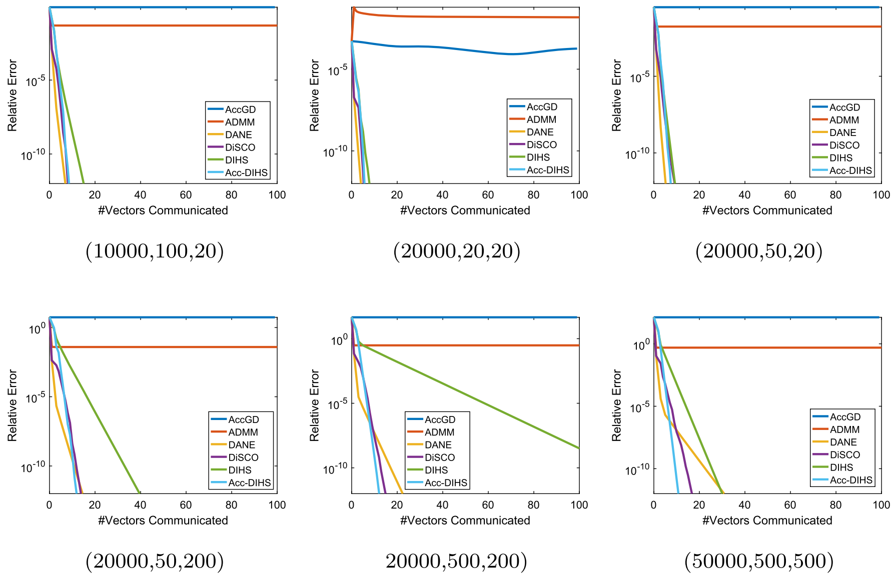
Fig 6. *Comparion of various approaches for distributed optimization under the partition by feature scenario, with different settings of* (n, p, m).

munication, this is consistent with theory that characterize the limit of one-shot sketching methods. The proposed DIDRP algorithm clearly outperforms LOCO methods as the number of communicated vectors grows. We further observe that Acc-DIDRP is more efficient than DIDRP, which again illustrates that the acceleration techniques can be helpful in further reducing the communication.

For the partition by sample scenario, we compare with several state-of-the-art algorithms including accelerated gradient descent (AccGD) [27], ADMM [5], DANE [40] and DiSCO [55]. The results are summarized in Figure 7. We observe that the methods leveraging higher-order information (DANE,DiSCO,DIHS,Acc-DIHS) are significantly more communication-efficient compared to AccGD and ADMM. Generally speaking Acc-DIHS has a slight advantage over existing approaches. We also tested on several real world data sets and the results are shown in Figure 5, where we observed a similar behavior.

8. Conclusion and discussion

In this paper, we focused on sketching techniques for solving large-scale ℓ_2 regularized least square problems. We established the equivalence between two recently proposed techniques, Hessian sketch and dual random projection, from a primal-dual point of view. We proposed accelerated methods for IHS and IDRP, from the preconditioned optimization perspective. By combining the primal and dual sketching techniques, we proposed a novel iterative primal-dual sketching approach, which substantially reduces the computational cost when



(10000,100,20)

(20000,20,20)

(20000,50,20)

(20000,50,200)

20000,500,200)

(50000,500,500)

Fig 7. *Comparion of various approaches for distributed optimization under the partition by sample scenario, with different settings of* (n, p, m).

solving sketched subproblems. We demonstrated applications of the iterative sketching techniques for distributed optimization when the data is partitioned by features or by samples.

The proposed approach can be extended to solve more general problems. For example, by sketching the Newton step in a second-order optimization method, as done in [32], we will be able to solve regularized risk minimization problems with self-concordant losses. It will be interesting to examine its empirical performance compared with existing approaches. More generally, Hessian sketch and dual random projection are designed for solving convex problems. It will be interesting to extend them for some structured non-convex problems, such as principle component analysis.

Appendix A: Appendix

The appendix contains proofs of theorems stated in the main paper.

A.1. Proof of Theorem 2

Proof. Based on the optimality condition for $\mathbf{w}^*$ and $\widehat{\mathbf{w}}_{\text{HS}}$, we have

$$\left(\frac{\mathbf{X}^\top\mathbf{X}}{n} + \lambda\mathbf{I}_p\right)\mathbf{w}^* = \frac{\mathbf{X}^\top\mathbf{y}}{n} \quad \text{and} \quad \left(\frac{\mathbf{X}^\top\mathbf{\Pi}\mathbf{\Pi}^\top\mathbf{X}}{n} + \lambda\mathbf{I}_p\right)\widehat{\mathbf{w}}_{\text{HS}} = \frac{\mathbf{X}^\top\mathbf{y}}{n}.$$

Therefore

$$\left(\frac{\mathbf{X}^\top\mathbf{X}}{n}+\lambda\mathbf{I}_p\right)\mathbf{w}^*-\left(\frac{\mathbf{X}^\top\mathbf{\Pi}\mathbf{\Pi}^\top\mathbf{X}}{n}+\lambda\mathbf{I}_p\right)\widehat{\mathbf{w}}_{\text{HS}}=\mathbf{0},$$

and

$$\left\langle\left(\frac{\mathbf{X}^\top\mathbf{X}}{n}+\lambda\mathbf{I}_p\right)\mathbf{w}^*-\left(\frac{\mathbf{X}^\top\mathbf{\Pi}\mathbf{\Pi}^\top\mathbf{X}}{n}+\lambda\mathbf{I}_p\right)\widehat{\mathbf{w}}_{\text{HS}},\mathbf{w}^*-\widehat{\mathbf{w}}_{\text{HS}}\right\rangle=0.$$

By adding and subtracting $\left\langle\mathbf{w}^*-\widehat{\mathbf{w}}_{\text{HS}},\left(\frac{\mathbf{X}^\top\mathbf{\Pi}\mathbf{\Pi}^\top\mathbf{X}}{n}+\lambda\mathbf{I}_p\right)\mathbf{w}^*\right\rangle$, we have

$$\begin{aligned}&\left\langle\left(\frac{\mathbf{X}^\top\mathbf{\Pi}\mathbf{\Pi}^\top\mathbf{X}}{n}-\frac{\mathbf{X}^\top\mathbf{X}}{n}\right)\mathbf{w}^*,\widehat{\mathbf{w}}_{\text{HS}}-\mathbf{w}^*\right\rangle\\&=(\mathbf{w}^*-\widehat{\mathbf{w}}_{\text{HS}})^\top\left(\frac{\mathbf{X}^\top\mathbf{\Pi}\mathbf{\Pi}^\top\mathbf{X}}{n}+\lambda\mathbf{I}_p\right)(\mathbf{w}^*-\widehat{\mathbf{w}}_{\text{HS}})\end{aligned}$$

The term on right hand side is lower bounded as

$$\begin{aligned}(\mathbf{w}^*-\widehat{\mathbf{w}}_{\text{HS}})^\top\left(\frac{\mathbf{X}^\top\mathbf{\Pi}\mathbf{\Pi}^\top\mathbf{X}}{n}\right)&(\mathbf{w}^*-\widehat{\mathbf{w}}_{\text{HS}})+\lambda\left\|\mathbf{w}^*-\widehat{\mathbf{w}}_{\text{HS}}\right\|_2^2\\&\geq\rho_1(\mathbf{X}\mathbb{R}^p,\mathbf{\Pi})\left\|\mathbf{w}^*-\widehat{\mathbf{w}}_{\text{HS}}\right\|_{\mathbf{X}}^2.\end{aligned}\tag{A.1}$$

For the left hand side, we have the following upper bound

$$\begin{aligned}&\left\langle\left(\mathbf{\Pi}\mathbf{\Pi}^\top-\mathbf{I}_n\right)\frac{\mathbf{X}\mathbf{w}^*}{\sqrt{n}},\frac{\mathbf{X}}{\sqrt{n}}(\widehat{\mathbf{w}}_{\text{HS}}-\mathbf{w}^*)\right\rangle\\&\quad\leq\rho_2(\mathbf{X}\mathbb{R}^p,\mathbf{\Pi},\mathbf{w}^*)\left\|\mathbf{w}^*\right\|_{\mathbf{X}}\left\|\widehat{\mathbf{w}}_{\text{HS}}-\mathbf{w}^*\right\|_{\mathbf{X}}.\end{aligned}\tag{A.2}$$

Combining (A.1) and (A.2) we have

$$\left\|\widehat{\mathbf{w}}_{\text{HS}}-\mathbf{w}^*\right\|_{\mathbf{X}}\leq\frac{\rho_2(\mathbf{X}\mathbb{R}^p,\mathbf{\Pi},\mathbf{w}^*)}{\rho_1(\mathbf{X}\mathbb{R}^p,\mathbf{\Pi})}\left\|\mathbf{w}^*\right\|_{\mathbf{X}}.$$

For the recovery of dual variables, we have

$$\begin{aligned}\left\|\widehat{\boldsymbol{\alpha}}_{\text{HS}}-\boldsymbol{\alpha}^*\right\|_2&=\left\|\mathbf{y}-\mathbf{X}\widehat{\mathbf{w}}_{\text{HS}}-(\mathbf{y}-\mathbf{X}\mathbf{w}^*)\right\|_2\\&=\sqrt{n}\left\|\widehat{\mathbf{w}}_{\text{HS}}-\mathbf{w}^*\right\|_{\mathbf{X}}\\&\leq\sqrt{n}\frac{\rho_2(\mathbf{X}\mathbb{R}^p,\mathbf{\Pi},\mathbf{w}^*)}{\rho_1(\mathbf{X}\mathbb{R}^p,\mathbf{\Pi})}\left\|\mathbf{w}^*\right\|_{\mathbf{X}}.\end{aligned}$$

This completes the proof for the Hessian sketch.

For the dual random projection, the proof is mostly analogous. Based on the optimality condition for $\boldsymbol{\alpha}^*$ and $\widehat{\boldsymbol{\alpha}}_{\text{DRP}}$, we have

$$\left(\frac{\mathbf{X}\mathbf{X}^\top}{n}+\lambda\mathbf{I}_n\right)\boldsymbol{\alpha}^*=\lambda\mathbf{y}\quad\text{and}\quad\left(\frac{\mathbf{X}\mathbf{R}\mathbf{R}^\top\mathbf{X}^\top}{n}+\lambda\mathbf{I}_n\right)\widehat{\boldsymbol{\alpha}}_{\text{DRP}}=\lambda\mathbf{y}.$$

Therefore

$$\left(\frac{\mathbf{X}\mathbf{X}^\top}{n} + \lambda \mathbf{I}_n\right)\boldsymbol{\alpha}^* - \left(\frac{\mathbf{X}\mathbf{R}\mathbf{R}^\top\mathbf{X}^\top}{n} + \lambda \mathbf{I}_n\right)\widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}} = \mathbf{0},$$

and

$$\left\langle \left(\frac{\mathbf{X}\mathbf{X}^\top}{n} + \lambda \mathbf{I}_n\right)\boldsymbol{\alpha}^* - \left(\frac{\mathbf{X}\mathbf{R}\mathbf{R}^\top\mathbf{X}^\top}{n} + \lambda \mathbf{I}_n\right)\widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}, \boldsymbol{\alpha}^* - \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}} \right\rangle = 0.$$

Simple algebra gives us

$$\begin{aligned}&\left\langle \left(\frac{\mathbf{X}\mathbf{R}\mathbf{R}^\top\mathbf{X}^\top}{n} - \frac{\mathbf{X}\mathbf{X}^\top}{n}\right)\boldsymbol{\alpha}^*, \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}} - \boldsymbol{\alpha}^* \right\rangle \\ &= (\boldsymbol{\alpha}^* - \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}})^\top \left(\frac{\mathbf{X}\mathbf{R}\mathbf{R}^\top\mathbf{X}^\top}{n} + \lambda \mathbf{I}_n\right)(\boldsymbol{\alpha}^* - \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}).\end{aligned}$$

The term on right hand side is lower bounded as

$$\begin{aligned}(\boldsymbol{\alpha}^* - \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}})^\top \left(\frac{\mathbf{X}\mathbf{R}\mathbf{R}^\top\mathbf{X}^\top}{n}\right)(\boldsymbol{\alpha}^* - \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}) + \lambda \left\|\boldsymbol{\alpha}^* - \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}\right\|_2^2 \\ \geq \rho_1(\mathbf{X}^\top\mathbb{R}^n, \mathbf{R}) \left\|\boldsymbol{\alpha}^* - \widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}}\right\|_{\mathbf{X}^\top}^2.\end{aligned} \tag{A.3}$$

The term on the left hand side is upper bounded as

$$\begin{aligned}&\left\langle (\mathbf{R}\mathbf{R}^\top - \mathbf{I}_p)\frac{\mathbf{X}^\top\boldsymbol{\alpha}^*}{\sqrt{n}}, \frac{\mathbf{X}^\top}{\sqrt{n}}(\widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}} - \boldsymbol{\alpha}^*) \right\rangle \\ &\quad \leq \rho_2(\mathbf{X}^\top\mathbb{R}^n, \mathbf{R}, \boldsymbol{\alpha}^*) \left\|\boldsymbol{\alpha}^*\right\|_{\mathbf{X}^\top} \left\|\widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}} - \boldsymbol{\alpha}^*\right\|_{\mathbf{X}^\top}.\end{aligned} \tag{A.4}$$

Combining (A.3) and (A.4) we have

$$\left\|\widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}} - \boldsymbol{\alpha}^*\right\|_{\mathbf{X}^\top} \leq \frac{\rho_2(\mathbf{X}^\top\mathbb{R}^n, \mathbf{R}, \boldsymbol{\alpha}^*)}{\rho_1(\mathbf{X}^\top\mathbb{R}^n, \mathbf{R})} \left\|\boldsymbol{\alpha}^*\right\|_{\mathbf{X}^\top}.$$

For the recovery of primal variables, we have

$$\begin{aligned}\left\|\widehat{\mathbf{w}}_{\mathrm{DRP}} - \mathbf{w}^*\right\|_2 =& \frac{1}{\lambda\sqrt{n}} \left\|\widehat{\boldsymbol{\alpha}}_{\mathrm{DRP}} - \boldsymbol{\alpha}^*\right\|_{\mathbf{X}^\top} \leq \frac{1}{\lambda\sqrt{n}} \frac{\rho_2(\mathbf{X}^\top\mathbb{R}^n, \mathbf{R}, \boldsymbol{\alpha}^*)}{\rho_1(\mathbf{X}^\top\mathbb{R}^n, \mathbf{R})} \left\|\boldsymbol{\alpha}^*\right\|_{\mathbf{X}^\top} \\ =& \frac{\rho_2(\mathbf{X}^\top\mathbb{R}^n, \mathbf{R}, \boldsymbol{\alpha}^*)}{\rho_1(\mathbf{X}^\top\mathbb{R}^n, \mathbf{R})} \left\|\mathbf{w}^*\right\|_2.\end{aligned}$$

An application of Lemma 1 concludes the proof. □

A.2. Proof of Theorem 3

We only prove the result for the Hessian sketch here as the proof for the dual random projection is analogous. We will make usage of the following concentration result for sub-Gaussian random matrices.

Lemma 14 (Lemma 3 in [53])**.** *Let $\mathbf{B} \in \mathbb{R}^{r \times m}$ be a random matrix with entries sampled i.i.d. from zero-mean sub-Gaussian distribution with variance $1/m$, then*

$$\left\|\mathbf{B}\mathbf{B}^\top - \mathbf{I}_r\right\|_2 \leq 2\sqrt{\frac{2(r+1)}{m}\log\frac{2r}{\delta}} := \epsilon_1$$

with probability at least $1-\delta$.

Lemma 15 (Theorem 3.2 in [35])**.** *Let $\mathbf{B} \in \mathbb{R}^{r \times m}$, $\mathbf{A} \in \mathbb{R}^{(n-r)\times m}$ be two random matrices with entries sampled i.i.d. from a zero-mean sub-Gaussian distribution with variance $1/m$, then*

$$\left\|\mathbf{A}\mathbf{B}^\top\right\|_2 \leq \frac{7}{3}\sqrt{\frac{2(n-r)}{m}\log\frac{n}{\delta}} := \tau_1$$

with probability at least $1-\delta$.

Let $\Delta\mathbf{w} = \mathbf{w}^* - \widehat{\mathbf{w}}_{\mathrm{HS}}$. Then

$$\begin{aligned}\|\Delta\mathbf{w}\|_{\mathbf{X}}^2 &= \left\|\frac{\mathbf{X}\Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2 \\ &= \left\|\frac{(\mathbf{U}\mathbf{\Sigma}_r\mathbf{V}^\top + \mathbf{U}\mathbf{\Sigma}_{\bar{r}}\mathbf{V}^\top)\Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2 \\ &= \left\|\frac{\mathbf{\Sigma}_r\mathbf{V}^\top\Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2 + \left\|\frac{\mathbf{\Sigma}_{\bar{r}}\mathbf{V}^\top\Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2\end{aligned}$$

Consider the term $\Delta\mathbf{w}^\top\left(\frac{\mathbf{X}^\top\mathbf{\Pi}\mathbf{\Pi}^\top\mathbf{X}}{n} + \lambda\mathbf{I}_p\right)\Delta\mathbf{w}$. We have

$$\begin{aligned}&\Delta\mathbf{w}^\top\left(\frac{\mathbf{X}^\top\mathbf{\Pi}\mathbf{\Pi}^\top\mathbf{X}}{n} + \lambda\mathbf{I}_p\right)\Delta\mathbf{w} \\ &\quad\geq \Delta\mathbf{w}^\top\left(\frac{\mathbf{V}^\top\mathbf{\Sigma}_r\mathbf{U}^\top\mathbf{\Pi}\mathbf{\Pi}^\top\mathbf{U}\mathbf{\Sigma}_r\mathbf{V}^\top}{n}\right)\Delta\mathbf{w} + \lambda\|\Delta\mathbf{w}\|_2^2 \\ &\qquad + 2\Delta\mathbf{w}^\top\left(\frac{\mathbf{V}^\top\mathbf{\Sigma}_{\bar{r}}\mathbf{U}^\top\mathbf{\Pi}\mathbf{\Pi}^\top\mathbf{U}\mathbf{\Sigma}_r\mathbf{V}^\top}{n}\right)\Delta\mathbf{w}.\end{aligned}$$

Since

$$\Delta\mathbf{w}^\top\left(\frac{\mathbf{V}^\top\mathbf{\Sigma}_r\mathbf{U}^\top\mathbf{\Pi}\mathbf{\Pi}^\top\mathbf{U}\mathbf{\Sigma}_r\mathbf{V}^\top}{n}\right)\Delta\mathbf{w} \geq (1-\epsilon_1)\left\|\frac{\mathbf{\Sigma}_r\mathbf{V}^\top\Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2,$$

and

$$\lambda\|\Delta\mathbf{w}\|_2^2 \geq \frac{\lambda}{\boldsymbol{\sigma}_{r+1}^2}\left\|\mathbf{\Sigma}_{\bar{r}}\mathbf{V}^\top\Delta\mathbf{w}\right\|_2^2 = \frac{\lambda n}{\boldsymbol{\sigma}_{r+1}^2}\left\|\frac{\mathbf{\Sigma}_{\bar{r}}\mathbf{V}^\top\Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2,$$

where

$$
\begin{aligned}
&2\Delta\mathbf{w}^\top\left(\frac{\mathbf{V}^\top\boldsymbol{\Sigma}_{\bar{r}}\mathbf{U}^\top\boldsymbol{\Pi}\boldsymbol{\Pi}^\top\mathbf{U}\boldsymbol{\Sigma}_r\mathbf{V}^\top}{n}\right)\Delta\mathbf{w}\\
&\quad= 2\Delta\mathbf{w}^\top\left(\frac{\mathbf{V}^\top\boldsymbol{\Sigma}_{\bar{r}}\mathbf{U}_{\bar{r}}^\top\boldsymbol{\Pi}\boldsymbol{\Pi}^\top\mathbf{U}_r\boldsymbol{\Sigma}_r\mathbf{V}^\top}{n}\right)\Delta\mathbf{w}\\
&\quad\geq -\tau_1\left\|\frac{\boldsymbol{\Sigma}_r\mathbf{V}^\top\Delta\mathbf{w}}{\sqrt{n}}\right\|_2\left\|\frac{\boldsymbol{\Sigma}_{\bar{r}}\mathbf{V}^\top\Delta\mathbf{w}}{\sqrt{n}}\right\|_2,
\end{aligned}
$$

we have

$$
\begin{aligned}
&\Delta\mathbf{w}^\top\left(\frac{\mathbf{X}^\top\boldsymbol{\Pi}\boldsymbol{\Pi}^\top\mathbf{X}}{n}+\lambda\mathbf{I}_p\right)\Delta\mathbf{w}\\
&\geq (1-\epsilon_1)\left\|\frac{\boldsymbol{\Sigma}_r\mathbf{V}^\top\Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2+\frac{\lambda n}{\boldsymbol{\sigma}_{r+1}^2}\left\|\frac{\boldsymbol{\Sigma}_{\bar{r}}\mathbf{V}^\top\Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2\\
&\quad-2\tau_1\left\|\frac{\boldsymbol{\Sigma}_r\mathbf{V}^\top\Delta\mathbf{w}}{\sqrt{n}}\right\|_2\left\|\frac{\boldsymbol{\Sigma}_{\bar{r}}\mathbf{V}^\top\Delta\mathbf{w}}{\sqrt{n}}\right\|_2\\
&\geq\left(\frac{1}{2}-\frac{\epsilon_1}{2}\right)\left\|\frac{\boldsymbol{\Sigma}_r\mathbf{V}^\top\Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2+\frac{\lambda n}{2\sigma_{r+1}^2}\left\|\frac{\boldsymbol{\Sigma}_{\bar{r}}\mathbf{V}^\top\Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2.
\end{aligned}
$$

Consider the term $\left\langle\left(\boldsymbol{\Pi}\boldsymbol{\Pi}^\top-\mathbf{I}_n\right)\frac{\mathbf{X}\mathbf{w}^*}{\sqrt{n}},-\frac{\mathbf{X}\Delta\mathbf{w}}{\sqrt{n}}\right\rangle$, we have

$$
\begin{aligned}
\left\langle\left(\boldsymbol{\Pi}\boldsymbol{\Pi}^\top-\mathbf{I}_n\right)\frac{\mathbf{X}\mathbf{w}^*}{\sqrt{n}},\frac{\mathbf{X}}{\sqrt{n}}(\widehat{\mathbf{w}}_{\mathrm{HS}}-\mathbf{w}^*)\right\rangle &= \left\langle\left(\boldsymbol{\Pi}\boldsymbol{\Pi}^\top-\mathbf{I}_n\right)\frac{\mathbf{X}_r\mathbf{w}^*}{\sqrt{n}},-\frac{\mathbf{X}_r\Delta\mathbf{w}}{\sqrt{n}}\right\rangle\\
&\quad+\left\langle\left(\boldsymbol{\Pi}\boldsymbol{\Pi}^\top-\mathbf{I}_n\right)\frac{\mathbf{X}_{\bar{r}}\mathbf{w}^*}{\sqrt{n}},-\frac{\mathbf{X}_r\Delta\mathbf{w}}{\sqrt{n}}\right\rangle\\
&\quad+\left\langle\left(\boldsymbol{\Pi}\boldsymbol{\Pi}^\top-\mathbf{I}_n\right)\frac{\mathbf{X}_r\mathbf{w}^*}{\sqrt{n}},-\frac{\mathbf{X}_{\bar{r}}\Delta\mathbf{w}}{\sqrt{n}}\right\rangle\\
&\quad+\left\langle\left(\boldsymbol{\Pi}\boldsymbol{\Pi}^\top-\mathbf{I}_n\right)\frac{\mathbf{X}_{\bar{r}}\mathbf{w}^*}{\sqrt{n}},-\frac{\mathbf{X}_{\bar{r}}\Delta\mathbf{w}}{\sqrt{n}}\right\rangle.
\end{aligned}
$$

Notice that the random matrix $\boldsymbol{\Pi}^\top\mathbf{U}_r$ and $\boldsymbol{\Pi}^\top\mathbf{U}_r$ can be treated as two Gaussian random matrices with entries sampled i.i.d from $\mathcal{N}(0,1/m)$. Applying Lemma 14 and Lemma 15, we can bound above terms separately:

$$
\begin{aligned}
\left\langle\left(\boldsymbol{\Pi}\boldsymbol{\Pi}^\top-\mathbf{I}_n\right)\frac{\mathbf{X}_r\mathbf{w}^*}{\sqrt{n}},-\frac{\mathbf{X}_r\Delta\mathbf{w}}{\sqrt{n}}\right\rangle &\leq \epsilon_1\left\|\frac{\boldsymbol{\Sigma}_r\mathbf{V}^\top\mathbf{w}^*}{\sqrt{n}}\right\|_2\left\|\frac{\boldsymbol{\Sigma}_r\mathbf{V}^\top\Delta\mathbf{w}}{\sqrt{n}}\right\|_2,\\
\left\langle\left(\boldsymbol{\Pi}\boldsymbol{\Pi}^\top-\mathbf{I}_n\right)\frac{\mathbf{X}_{\bar{r}}\mathbf{w}^*}{\sqrt{n}},-\frac{\mathbf{X}_r\Delta\mathbf{w}}{\sqrt{n}}\right\rangle &\leq \tau_1\left\|\frac{\boldsymbol{\Sigma}_{\bar{r}}\mathbf{V}^\top\mathbf{w}^*}{\sqrt{n}}\right\|_2\left\|\frac{\boldsymbol{\Sigma}_r\mathbf{V}^\top\Delta\mathbf{w}}{\sqrt{n}}\right\|_2,\\
\left\langle\left(\boldsymbol{\Pi}\boldsymbol{\Pi}^\top-\mathbf{I}_n\right)\frac{\mathbf{X}_r\mathbf{w}^*}{\sqrt{n}},-\frac{\mathbf{X}_{\bar{r}}\Delta\mathbf{w}}{\sqrt{n}}\right\rangle &\leq \tau_1\left\|\frac{\boldsymbol{\Sigma}_r\mathbf{V}^\top\mathbf{w}^*}{\sqrt{n}}\right\|_2\left\|\frac{\boldsymbol{\Sigma}_{\bar{r}}\mathbf{V}^\top\Delta\mathbf{w}}{\sqrt{n}}\right\|_2,\\
\left\langle\left(\boldsymbol{\Pi}\boldsymbol{\Pi}^\top-\mathbf{I}_n\right)\frac{\mathbf{X}_{\bar{r}}\mathbf{w}^*}{\sqrt{n}},-\frac{\mathbf{X}_{\bar{r}}\Delta\mathbf{w}}{\sqrt{n}}\right\rangle &\leq \upsilon_1\left\|\frac{\boldsymbol{\Sigma}_{\bar{r}}\mathbf{V}^\top\mathbf{w}^*}{\sqrt{n}}\right\|_2\left\|\frac{\boldsymbol{\Sigma}_{\bar{r}}\mathbf{V}^\top\Delta\mathbf{w}}{\sqrt{n}}\right\|_2.
\end{aligned}
$$

By Cauchy-Schwarz inequality, we have

$$\begin{aligned}
&\left\langle \left(\mathbf{\Pi}\mathbf{\Pi}^\top - \mathbf{I}_n\right) \frac{\mathbf{X}\mathbf{w}^*}{\sqrt{n}}, \frac{\mathbf{X}}{\sqrt{n}}(\widehat{\mathbf{w}}_{\mathrm{HS}} - \mathbf{w}^*) \right\rangle \\
\leq{}& \epsilon_1 \left\|\frac{\mathbf{\Sigma}_r \mathbf{V}^\top \mathbf{w}^*}{\sqrt{n}}\right\|_2 \left\|\frac{\mathbf{\Sigma}_r \mathbf{V}^\top \Delta\mathbf{w}}{\sqrt{n}}\right\|_2 + \tau_1 \left\|\frac{\mathbf{\Sigma}_{\bar{r}} \mathbf{V}^\top \mathbf{w}^*}{\sqrt{n}}\right\|_2 \left\|\frac{\mathbf{\Sigma}_r \mathbf{V}^\top \Delta\mathbf{w}}{\sqrt{n}}\right\|_2 \\
&+\tau_1 \left\|\frac{\mathbf{\Sigma}_r \mathbf{V}^\top \mathbf{w}^*}{\sqrt{n}}\right\|_2 \left\|\frac{\mathbf{\Sigma}_{\bar{r}} \mathbf{V}^\top \Delta\mathbf{w}}{\sqrt{n}}\right\|_2 + \upsilon_1 \left\|\frac{\mathbf{\Sigma}_{\bar{r}} \mathbf{V}^\top \mathbf{w}^*}{\sqrt{n}}\right\|_2 \left\|\frac{\mathbf{\Sigma}_{\bar{r}} \mathbf{V}^\top \Delta\mathbf{w}}{\sqrt{n}}\right\|_2 \\
\leq{}& \frac{4\epsilon_1^2}{1-\epsilon_1} \left\|\frac{\mathbf{\Sigma}_r \mathbf{V}^\top \mathbf{w}^*}{\sqrt{n}}\right\|_2^2 + \frac{1-\epsilon_1}{8} \left\|\frac{\mathbf{\Sigma}_r \mathbf{V}^\top \Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2 \\
&+ \frac{4\tau_1^2}{1-\epsilon_1} \left\|\frac{\mathbf{\Sigma}_{\bar{r}} \mathbf{V}^\top \mathbf{w}^*}{\sqrt{n}}\right\|_2^2 + \frac{1-\epsilon_1}{8} \left\|\frac{\mathbf{\Sigma}_r \mathbf{V}^\top \Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2 \\
&+ \frac{4\tau_1^2\sigma_{r+1}^2}{\lambda n} \left\|\frac{\mathbf{\Sigma}_r \mathbf{V}^\top \mathbf{w}^*}{\sqrt{n}}\right\|_2^2 + \frac{\lambda n}{8\sigma_{r+1}^2} \left\|\frac{\mathbf{\Sigma}_{\bar{r}} \mathbf{V}^\top \Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2 \\
&+ \frac{4\upsilon_1^2\sigma_{r+1}^2}{\lambda n} \left\|\frac{\mathbf{\Sigma}_{\bar{r}} \mathbf{V}^\top \mathbf{w}^*}{\sqrt{n}}\right\|_2^2 + \frac{\lambda n}{8\sigma_{r+1}^2} \left\|\frac{\mathbf{\Sigma}_{\bar{r}} \mathbf{V}^\top \Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2.
\end{aligned}$$

From the proof of Theorem 2, we know

$$\begin{aligned}
&\frac{1-\epsilon_1}{2} \left\|\frac{\mathbf{\Sigma}_r \mathbf{V}^\top \Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2 + \frac{\lambda n}{2\sigma_{r+1}^2} \left\|\frac{\mathbf{\Sigma}_{\bar{r}} \mathbf{V}^\top \Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2 \\
&\leq \Delta\mathbf{w}^\top \left(\frac{\mathbf{X}^\top \mathbf{\Pi}\mathbf{\Pi}^\top \mathbf{X}}{n} + \lambda \mathbf{I}_p\right) \Delta\mathbf{w} \\
&= \left\langle \left(\mathbf{\Pi}\mathbf{\Pi}^\top - \mathbf{I}_n\right) \frac{\mathbf{X}\mathbf{w}^*}{\sqrt{n}}, \frac{\mathbf{X}}{\sqrt{n}}(\widehat{\mathbf{w}}_{\mathrm{HS}} - \mathbf{w}^*) \right\rangle.
\end{aligned}$$

Combining the above, we have

$$\begin{aligned}
&\frac{1-\epsilon_1}{4} \left\|\frac{\mathbf{\Sigma}_r \mathbf{V}^\top \Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2 + \frac{\lambda n}{4\sigma_{r+1}^2} \left\|\frac{\mathbf{\Sigma}_{\bar{r}} \mathbf{V}^\top \Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2 \\
&\leq \left(\frac{4\epsilon_1^2}{1-\epsilon_1} + \frac{4\tau_1^2\sigma_{r+1}^2}{\lambda n}\right) \left\|\frac{\mathbf{\Sigma}_r \mathbf{V}^\top \mathbf{w}^*}{\sqrt{n}}\right\|_2^2 \\
&\quad + \left(\frac{4\tau_1^2}{1-\epsilon_1} + \frac{4\upsilon_1^2\sigma_{r+1}^2}{\lambda n}\right) \left\|\frac{\mathbf{\Sigma}_{\bar{r}} \mathbf{V}^\top \mathbf{w}^*}{\sqrt{n}}\right\|_2^2 \\
&\leq \left(\frac{4\epsilon_1^2}{1-\epsilon_1} + \frac{4\tau_1^2\sigma_{r+1}^2}{\lambda n} + \frac{4\tau_1^2\rho^2}{1-\epsilon} + \frac{4\rho^2\upsilon_1^2\sigma_{r+1}^2}{\lambda n}\right) \|\mathbf{w}^*\|_{\mathbf{X}}^2.
\end{aligned}$$

Thus

$$\|\Delta\mathbf{w}\|_{\mathbf{X}}^2 = \left\|\frac{\mathbf{\Sigma}_r \mathbf{V}^\top \Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2 + \left\|\frac{\mathbf{\Sigma}_{\bar{r}} \mathbf{V}^\top \Delta\mathbf{w}}{\sqrt{n}}\right\|_2^2$$

$$
\begin{aligned}
&\leq \left(\frac{4}{1-\epsilon_1} + \frac{4\sigma_{r+1}^2}{\lambda n}\right)\left(\frac{1-\epsilon_1}{4}\left\|\frac{\mathbf{\Sigma}_r \mathbf{V}^\top \Delta \mathbf{w}}{\sqrt{n}}\right\|_2^2 + \frac{\lambda n}{4\sigma_{r+1}^2}\left\|\frac{\mathbf{\Sigma}_{\bar{r}} \mathbf{V}^\top \Delta \mathbf{w}}{\sqrt{n}}\right\|_2^2\right) \\
&\leq \left(\frac{4}{1-\epsilon_1} + \frac{4\sigma_{r+1}^2}{\lambda n}\right)\left(\frac{4\epsilon_1^2}{1-\epsilon_1} + \frac{4\tau_1^2\sigma_{r+1}^2}{\lambda n} + \frac{4\tau_1^2\rho^2}{1-\epsilon_1} + \frac{4\rho^2\upsilon_1^2\sigma_{r+1}^2}{\lambda n}\right)\|\mathbf{w}^*\|_{\mathbf{X}}^2,
\end{aligned}
$$

which concludes the proof.

A.3. Proof of Theorem 6

For notation simplicity, let

$$
\widetilde{\mathbf{H}} = \frac{\mathbf{X}^\top \mathbf{\Pi}\mathbf{\Pi}^\top \mathbf{X}}{n} + \lambda \mathbf{I}_p \quad \text{and} \quad \mathbf{H} = \frac{\mathbf{X}^\top \mathbf{X}}{n} + \lambda \mathbf{I}_p.
$$

Based on the property of similarity matrices, we have

$$
\kappa(\widetilde{\mathbf{H}}^{-1}\mathbf{H}) = \kappa(\widetilde{\mathbf{H}}^{-1/2}\mathbf{H}\widetilde{\mathbf{H}}^{-1/2}) = \frac{\max_{\mathbf{w}} \mathbf{w}^\top \widetilde{\mathbf{H}}^{-1/2}\mathbf{H}\widetilde{\mathbf{H}}^{-1/2}\mathbf{w}}{\min_{\mathbf{w}} \mathbf{w}^\top \widetilde{\mathbf{H}}^{-1/2}\mathbf{H}\widetilde{\mathbf{H}}^{-1/2}\mathbf{w}}.
$$

Consider the quantity $|\mathbf{w}^\top \widetilde{\mathbf{H}}^{-1/2}(\mathbf{H} - \widetilde{\mathbf{H}})\widetilde{\mathbf{H}}^{-1/2}\mathbf{w}|$. We have

$$
\begin{aligned}
|\mathbf{w}^\top \widetilde{\mathbf{H}}^{-1/2}(\mathbf{H} - \widetilde{\mathbf{H}})\widetilde{\mathbf{H}}^{-1/2}\mathbf{w}| &= \left\langle \left(\mathbf{H} - \widetilde{\mathbf{H}}\right)\widetilde{\mathbf{H}}^{-1/2}\mathbf{w}, \widetilde{\mathbf{H}}^{-1/2}\mathbf{w}\right\rangle \\
&= \left\langle \left(\mathbf{\Pi}\mathbf{\Pi}^\top - \mathbf{I}_n\right)\frac{\mathbf{X}}{\sqrt{n}}\widetilde{\mathbf{H}}^{-1/2}\mathbf{w}, \frac{\mathbf{X}}{\sqrt{n}}\widetilde{\mathbf{H}}^{-1/2}\mathbf{w}\right\rangle \\
&\leq \rho_2\left(\mathbf{X}\mathbb{R}^p, \mathbf{\Pi}, \frac{\mathbf{X}}{\sqrt{n}}\widetilde{\mathbf{H}}^{-1/2}\mathbf{w}\right)\left\|\widetilde{\mathbf{H}}^{-1/2}\mathbf{w}\right\|_{\mathbf{X}}^2 \\
&\leq C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p \cap \mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}\left\|\widetilde{\mathbf{H}}^{-1/2}\mathbf{w}\right\|_{\mathbf{X}}^2.
\end{aligned}
$$

For any vector $\mathbf{u} \in \mathbb{R}^p$, we have

$$
\begin{aligned}
\left\|\widetilde{\mathbf{H}}^{1/2}\mathbf{u}\right\|_2^2 &= \mathbf{u}^\top\left(\frac{\mathbf{X}^\top \mathbf{\Pi}\mathbf{\Pi}^\top \mathbf{X}}{n} + \lambda \mathbf{I}_p\right)\mathbf{u} \\
&= \mathbf{u}^\top\left(\frac{\mathbf{X}^\top \mathbf{\Pi}\mathbf{\Pi}^\top \mathbf{X}}{n}\right)\mathbf{u} + \lambda\|\mathbf{u}\|_2^2 \\
&\geq \rho_1(\mathbf{X}\mathbb{R}^p, \mathbf{\Pi})\|\mathbf{u}\|_{\mathbf{X}}^2 \\
&\geq \left(1 - C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p \cap \mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}\right)\|\mathbf{u}\|_{\mathbf{X}}^2.
\end{aligned}
$$

Let $\mathbf{u} = \widetilde{\mathbf{H}}^{-1/2}\mathbf{w}$, we have

$$
\left\|\widetilde{\mathbf{H}}^{-1/2}\mathbf{w}\right\|_{\mathbf{X}}^2 \leq \frac{1}{1 - C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p \cap \mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}}\|\mathbf{w}\|_2^2.
$$

Combining, we get

$$|\mathbf{w}^\top \widetilde{\mathbf{H}}^{-1/2}(\mathbf{H}-\widetilde{\mathbf{H}})\widetilde{\mathbf{H}}^{-1/2}\mathbf{w}| \le \frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}}\|\mathbf{w}\|_2^2,$$

which implies

$$\begin{aligned}\max_{\mathbf{w}} \mathbf{w}^\top \widetilde{\mathbf{H}}^{-1/2}\mathbf{H}\widetilde{\mathbf{H}}^{-1/2}\mathbf{w} \le& \|\mathbf{w}\|_2^2 + \frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}}\|\mathbf{w}\|_2^2\\ =&\frac{1}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}}\|\mathbf{w}\|_2^2,\end{aligned}$$

and

$$\begin{aligned}\min_{\mathbf{w}} \mathbf{w}^\top \widetilde{\mathbf{H}}^{-1/2}\mathbf{H}\widetilde{\mathbf{H}}^{-1/2}\mathbf{w} \ge& \|\mathbf{w}\|_2^2 - \frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}}\|\mathbf{w}\|_2^2\\ =&\frac{1-2C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}}\|\mathbf{w}\|_2^2.\end{aligned}$$

Thus we know

$$\kappa(\widetilde{\mathbf{H}}^{-1}\mathbf{H}) \le \frac{1}{1-2C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}}.$$

The proof for $\kappa_{\mathrm{DRP}}(\mathbf{X},\mathbf{R},\lambda)$ is analogous.

A.4. Proof of Lemma 8

Note that (2.6) is sketching the following problem

$$\arg\min_{\mathbf{u}} \mathbf{u}^\top\left(\frac{\mathbf{X}^\top\mathbf{X}}{2n}+\frac{\lambda}{2}\mathbf{I}_p\right)\mathbf{u} - \left\langle \frac{\mathbf{X}^\top(\mathbf{y}-\mathbf{X}\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)})}{n} - \lambda\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)}, \mathbf{u}\right\rangle,$$

where $\mathbf{w}^* - \widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)}$ is the optimal solution. Thus applying Theorem 2, We have

$$\left\|\widehat{\mathbf{u}}^{(t)} - (\mathbf{w}^* - \widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)})\right\|_{\mathbf{X}} \le \frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}\log\left(\frac{1}{\delta}\right)}}\left\|\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)} - \mathbf{w}^*\right\|_{\mathbf{X}}.$$

Using the definition that $\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)} = \widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)} + \widehat{\mathbf{u}}^{(t)}$, we obtain the desired result.

A.5. Proof of Theorem 9

By triangle inequality we have the following decomposition:

$$\begin{aligned}
\left\|\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}-\mathbf{w}^*\right\|_{\mathbf{X}} &\leq \left\|\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}-\mathbf{w}^*\right\|_{\mathbf{X}}+\left\|\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}-\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}\right\|_{\mathbf{X}}\\
&\leq \frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}}\log\left(\frac{1}{\delta}\right)}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}}\log\left(\frac{1}{\delta}\right)}\left\|\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)}-\mathbf{w}^*\right\|_{\mathbf{X}}+\left\|\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}-\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}\right\|_{\mathbf{X}}\\
&\leq \left(\frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}}\log\left(\frac{1}{\delta}\right)}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}}\log\left(\frac{1}{\delta}\right)}\right)^t\|\mathbf{w}^*\|_{\mathbf{X}}+\left\|\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}-\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}\right\|_{\mathbf{X}}.
\end{aligned}$$

For the term $\left\|\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}-\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}\right\|_{\mathbf{X}}$, we can further bridge $\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}$ and $\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}$ by $\bar{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}$, which is the result of one exact step of IHS initialized at $\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t)}$. Thus we have the following decomposition

$$\left\|\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}-\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}\right\|_{\mathbf{X}}\leq\left\|\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}-\bar{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}\right\|_{\mathbf{X}}+\left\|\bar{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}-\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}\right\|_{\mathbf{X}}.$$

Applying the Theorem 2 for DRP we have the following bound for $\Big\|\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}-\bar{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}Big\|_{\mathbf{X}}$:

$$\begin{aligned}
\left\|\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}-\bar{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}\right\|_{\mathbf{X}} &\leq \lambda_{\max}\left(\frac{\mathbf{X}^\top\mathbf{X}}{n}\right)\left\|\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}-\bar{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}\right\|_2\\
&\leq \lambda_{\max}\left(\frac{\mathbf{X}^\top\mathbf{X}}{n}\right)\left(\frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}\right)^k\left\|\bar{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}\right\|_2\\
&\leq \lambda_{\max}\left(\frac{\mathbf{X}^\top\mathbf{X}}{n}\right)\left(\frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}\right)^k\\
&\quad\times\left(\left\|\bar{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}-\mathbf{w}^*\right\|_2+\|\mathbf{w}^*\|_2\right)\\
&\leq 2\lambda_{\max}\left(\frac{\mathbf{X}^\top\mathbf{X}}{n}\right)\left(\frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}\right)^k\|\mathbf{w}^*\|_2.
\end{aligned}$$

We can relate the error $\left\|\bar{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}-\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}\right\|_{\mathbf{X}}$ to the error term at t-th outer loop iteration: $\left\|\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t)}-\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)}\right\|_{\mathbf{X}}$:

$$\begin{aligned}
&\left\|\bar{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}-\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}\right\|_{\mathbf{X}}\\
&\quad=\left\|\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t)}-\widetilde{\mathbf{H}}^{-1}\nabla P(\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t)})-\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)}-\widetilde{\mathbf{H}}^{-1}\nabla P(\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)})\right\|_{\mathbf{X}}\\
&\quad=\left\|\widetilde{\mathbf{H}}^{-1}(\widetilde{\mathbf{H}}-\mathbf{H})(\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t)}-\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)})\right\|_{\mathbf{X}}\\
&\quad\leq\left\|\widetilde{\mathbf{H}}^{-1}\right\|_2\left\|\widetilde{\mathbf{H}}-\mathbf{H}\right\|_2\left\|\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t)}-\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)}\right\|_{\mathbf{X}}
\end{aligned}$$

$$
\begin{aligned}
&\leq \frac{4\lambda_{\max}\left(\frac{\mathbf{X}^\top\mathbf{X}}{n}\right)}{\lambda}\left\|\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t)}-\widehat{\mathbf{w}}_{\mathrm{HS}}^{(t)}\right\|_{\mathbf{X}} \\
&\leq \frac{8\lambda_{\max}^2\left(\frac{\mathbf{X}^\top\mathbf{X}}{n}\right)}{\lambda}\left(\frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}\right)^k\|\mathbf{w}^*\|_2\,.
\end{aligned}
$$

Combining above inequalities we obtained the following iterative error bound for $\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}$:

$$
\begin{aligned}
\left\|\widetilde{\mathbf{w}}_{\mathrm{HS}}^{(t+1)}-\mathbf{w}^*\right\|_{\mathbf{X}} \leq &\left(\frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}}\log\left(\frac{1}{\delta}\right)}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}\mathbb{R}^p\cap\mathcal{S}^{n-1})}{m}}\log\left(\frac{1}{\delta}\right)}\right)^t\|\mathbf{w}^*\|_{\mathbf{X}} \\
&+\frac{10\lambda_{\max}^2\left(\frac{\mathbf{X}^\top\mathbf{X}}{n}\right)}{\lambda}\left(\frac{C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}{1-C_0\sqrt{\frac{\mathbb{W}^2(\mathbf{X}^\top\mathbb{R}^n\cap\mathcal{S}^{p-1})}{d}}\log\left(\frac{1}{\delta}\right)}\right)^k\|\mathbf{w}^*\|_2.
\end{aligned}
$$

Acknowledgments

This work is partially supported by an IBM Corporation Faculty Research Fund at the University of Chicago Booth School of Business. This work was completed in part with resources provided by the University of Chicago Research Computing Center.

References

[1] Ailon, N. and Chazelle, B. (2009). The fast Johnson-Lindenstrauss transform and approximate nearest neighbors. *SIAM Journal on Computing* **39** 302–322. MR2506527

[2] Alaoui, A. E. and Mahoney, M. W. (2014). Fast randomized kernel methods with statistical guarantees. *arXiv preprint arXiv:1411.0306*.

[3] Avron, H., Maymounkov, P. and Toledo, S. (2010). Blendenpik: Supercharging LAPACK's least-squares solver. *SIAM Journal on Scientific Computing* **32** 1217–1236. MR2639236

[4] Boutsidis, C. and Gittens, A. (2013). Improved matrix algorithms via the subsampled randomized Hadamard transform. *SIAM Journal on Matrix Analysis and Applications* **34** 1301–1340. MR3101094

[5] Boyd, S. P., Parikh, N., Chu, E., Peleato, B. and Eckstein, J. (2011). Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Found. Trends Mach. Learn.* **3** 1–122.

[6] Boyd, S. P. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press, Cambridge. MR2061575 (2005d:90002) MR2061575

[7] Defazio, A., Bach, F. and Lacoste-Julien, S. (2014). Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems* 1646–1654.

[8] Dhillon, P. S., Foster, D. P., Kakade, S. M. and Ungar, L. H. (2013). A risk comparison of ordinary least squares vs ridge regression. *Journal of Machine Learning Research* **14** 1505–1511. MR3092989

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

[10] Drineas, P., Mahoney, M. W., Muthukrishnan, S. and Sarlós, T. (2011). Faster least squares approximation. *Numerische Mathematik* **117** 219–249. MR2754850

[11] Frostig, R., Ge, R., Kakade, S. and Sidford, A. (2015). Unregularizing: approximate proximal point and faster stochastic algorithms for empirical risk minimization. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)* 2540–2548.

[12] Halko, N., Martinsson, P.-G. and Tropp, J. A. (2011). Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review* **53** 217–288. MR2806637

[13] Heinze, C., McWilliams, B. and Meinshausen, N. (2015). DUAL-LOCO: Distributing Statistical Estimation Using Random Projections. *arXiv preprint arXiv:1506.02554*.

[14] Heinze, C., McWilliams, B., Meinshausen, N. and Krummenacher, G. (2014). LOCO: Distributing Ridge Regression with Random Projections. *arXiv preprint arXiv:1406.3469*.

[15] Hestenes, M. R. and Stiefel, E. (1952). *Methods of conjugate gradients for solving linear systems* **49**. MR0060307

[16] Johnson, R. and Zhang, T. (2013). Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems* 315–323.

[17] Jordan, M. I., Lee, J. D. and Yang, Y. (2016). Communication-Efficient Distributed Statistical Inference. *arXiv preprint arXiv:1605.07689*.

[18] Lan, G. and Zhou, Y. (2015). An optimal randomized incremental gradient method. *arXiv preprint arXiv:1507.02000*.

[19] Lee, J. D., Sun, Y., Liu, Q. and Taylor, J. E. (2015). Communication-efficient sparse regression: a one-shot approach. *arXiv preprint arXiv:1503.04337*. MR3625709

[20] Lin, H., Mairal, J. and Harchaoui, Z. (2015). A universal catalyst for first-order optimization. In *Advances in Neural Information Processing Systems* 3366–3374.

[21] Lu, Y., Dhillon, P., Foster, D. P. and Ungar, L. (2013). Faster ridge regression via the subsampled randomized hadamard transform. In *Advances in neural information processing systems* 369–377.

[22] Luenberger, D. G. *Introduction to linear and nonlinear programming* **28**.

[23] Mahoney, M. W. (2011). Randomized algorithms for matrices and data. *Foundations and Trends® in Machine Learning* **3** 123–224.

[24] Mendelson, S., Pajor, A. and Tomczak-Jaegermann, N. (2007). Reconstruction and subgaussian operators in asymptotic geometric analysis. *Geometric and Functional Analysis* **17** 1248–1282.

[25] Meng, X., Saunders, M. A. and Mahoney, M. W. (2014). LSRN: A parallel iterative solver for strongly over-or underdetermined systems. *SIAM Journal on Scientific Computing* **36** C95–C118.

[26] Nesterov, Y. (2012). Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization* **22** 341–362.

[27] Nesterov, Y. (2013). *Introductory lectures on convex optimization: A basic course* **87**. Springer Science & Business Media.

[28] Nocedal, J. and Wright, S. (2006). *Numerical optimization*. Springer Science & Business Media.

[29] Oymak, S., Recht, B. and Soltanolkotabi, M. (2015). Isometric sketching of any set via the Restricted Isometry Property. *arXiv preprint arXiv:1506.03521*.

[30] Oymak, S. and Tropp, J. A. (2015). Universality laws for randomized dimension reduction, with applications. *arXiv preprint arXiv:1511.09433*.

[31] Pilanci, M. and Wainwright, M. J. (2015). Randomized sketches of convex programs with sharp guarantees. *Information Theory, IEEE Transactions on* **61** 5096–5115.

[32] Pilanci, M. and Wainwright, M. J. (2015). Newton Sketch: A Linear-time Optimization Algorithm with Linear-Quadratic Convergence. *arXiv preprint arXiv:1505.02250*.

[33] Pilanci, M. and Wainwright, M. J. (2016). Iterative Hessian sketch: Fast and accurate solution approximation for constrained least-squares. *Journal of Machine Learning Research*.

[34] Raskutti, G. and Mahoney, M. (2015). A statistical perspective on randomized sketching for ordinary least-squares. *stat* **1050** 25.

[35] Recht, B. (2011). A simpler approach to matrix completion. *Journal of Machine Learning Research* **12** 3413–3430.

[36] Rokhlin, V. and Tygert, M. (2008). A fast randomized algorithm for overdetermined linear least-squares regression. *Proceedings of the National Academy of Sciences* **105** 13212–13217.

[37] Roux, N. L., Schmidt, M. and Bach, F. R. (2012). A stochastic gradient method with an exponential convergence _rate for finite training sets. In *Advances in Neural Information Processing Systems* 2663–2671.

[38] Sarlos, T. (2006). Improved approximation algorithms for large matrices via random projections. In *Foundations of Computer Science, 2006. FOCS'06. 47th Annual IEEE Symposium on* 143–152. IEEE.

[39] Shalev-Shwartz, S. and Zhang, T. (2013). Stochastic dual coordinate ascent methods for regularized loss. *The Journal of Machine Learning Research* **14** 567–599.

[40] Shamir, O., Srebro, N. and Zhang, T. (2014). Communication efficient distributed optimization using an approximate newton-type method. In *Proceedings of The 31st International Conference on Machine Learning* 1000–1008.

[41] Smith, V., Forte, S., Ma, C., Takac, M., Jordan, M. I. and Jaggi, M. (2016). CoCoA: A General Framework for Communication-Efficient Distributed Optimization. *arXiv preprint arXiv:1611.02189*.

[42] Sridharan, K., Shalev-Shwartz, S. and Srebro, N. (2009). Fast rates for regularized objectives. In *Advances in Neural Information Processing Systems* 1545–1552.

[43] Vershynin, R. (2015). Estimation in high dimensions: a geometric perspective. In *Sampling Theory, a Renaissance* 3–66. Springer. MR3467418

[44] Wang, J., Kolar, M., Srebro, N. and Zhang, T. (2016). Efficient distributed learning with sparsity. *arXiv preprint arXiv:1605.07991*.

[45] Wang, X., Dunson, D. B. and Leng, C. (2016). DECOrrelated feature space partitioning for distributed sparse regression. In *Advances in Neural Information Processing Systems* 802–810.

[46] Woodruff, D. P. (2014). Sketching as a tool for numerical linear algebra. *arXiv preprint arXiv:1411.4357*. MR3285427

[47] Yang, J., Mahoney, M. W., Saunders, M. and Sun, Y. (2016). Feature-distributed sparse regression: a screen-and-clean approach. In *Advances in Neural Information Processing Systems* 2712–2720.

[48] Yang, T., Zhang, L., Jin, R. and Zhu, S. (2015). Theory of Dual-sparse Regularized Randomized Reduction. In *Proceedings of The 32nd International Conference on Machine Learning* 305–314.

[49] Yang, Y., Pilanci, M. and Wainwright, M. J. (2015). Randomized sketches for kernels: Fast and optimal non-parametric regression. *arXiv preprint arXiv:1501.06195*. MR3662446

[50] Yu, A. W., Lin, Q. and Yang, T. (2015). Doubly Stochastic Primal-Dual Coordinate Method for Regularized Empirical Risk Minimization with Factorized Data. *arXiv preprint arXiv:1508.03390*.

[51] Zhang, L., Mahdavi, M. and Jin, R. (2013). Linear convergence with condition number independent access of full gradients. In *Advances in Neural Information Processing Systems* 980–988.

[52] Zhang, L., Mahdavi, M., Jin, R., Yang, T. and Zhu, S. (2013). Recovering the Optimal Solution by Dual Random Projection. In *Conference on Learning Theory* 135–157.

[53] Zhang, L., Mahdavi, M., Jin, R., Yang, T. and Zhu, S. (2014). Random projections for classification: A recovery approach. *Information Theory, IEEE Transactions on* **60** 7300–7316. MR3273056

[54] Zhang, Y. and Lin, X. (2015). Stochastic Primal-Dual Coordinate Method for Regularized Empirical Risk Minimization. In *Proceedings of The 32nd International Conference on Machine Learning* 353–361.

[55] Zhang, Y. and Lin, X. (2015). DiSCO: Distributed Optimization for Self-Concordant Empirical Loss. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)* 362–370.