Sketching Meets Random Projection in the Dual: A Provable Recovery Algorithm for Big and High-dimensional Data

Jialei Wang*, Jason D. Lee‡, Mehrdad Mahdavi†, Mladen Kolar‡, and Nathan Srebro†

*Department of Computer Science, University of Chicago
‡Marshall School of Business, University of Southern California
†Toyota Technological Institute at Chicago
‡Booth School of Business, University of Chicago

Abstract

Sketching techniques have become popular for scaling up machine learning algorithms by reducing the sample size or dimensionality of massive data sets, while still maintaining the statistical power of big data. 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 accelerated iterative Hessian sketch via the searching the conjugate direction; we then establish primal-dual connections between the Hessian sketch and dual random projection, and apply the preconditioned conjugate gradient approach on the dual problem, which leads to the accelerated iterative dual random projection methods. Finally to tackle the challenges from both large sample size and high-dimensionality, we propose the primal-dual sketch, which iteratively sketches the primal and dual formulations. We show that using a logarithmic number of calls to solvers of small scaled problem, primal-dual sketch is able to recover the optimum of the original problem up to arbitrary precision. The proposed algorithms are validated via extensive experiments on synthetic and real data sets which complements our theoretical results.

Keywords: Iterative Hessian Sketch, Dual Random Projection, Preconditioning, Primal-Dual Conversion, Acceleration, Conjugate Gradient, Primal-Dual Sketch

1 Introduction

Machine Learning has gained great empirical success from the massive data sets collected from various domains. Among them a major challenge is to utilize existing computational resources to build predictive and inferential models from such huge data sets, while maintaining the statistical power of big data. One remedy for the big data challenge is to build distributed computer systems and design distributed learning algorithms to make big data learning possible, however, distributed systems may not always available, and the cost of running distributed system can be much higher than one can afford, which makes distributed learning not suitable for all scenarios. An alternative remedy is to use the state-of-the-art randomized optimization algorithms to accelerate the training
process, for example, researchers have proposed optimization algorithms for regularized empirical risk minimization problem, with provable fast convergence and low computational cost per iteration (see (Johnson and Zhang, 2013; Shalev-Shwartz and Zhang, 2013; Defazio et al., 2014) for examples), however, the speed of these optimization methods still heavily depends on the condition number of problem at hand, which can be undesirable for many real world problems.

Sketching (Woodruff, 2014), which approximates the solution via constructing some sketched, usually of smaller scale problem from the original data, has become an emerging technique for big data analytics. With the sketching technique, we can find solutions which approximately solve various forms of original large-scale problem, such as least square regression, robust regression, low-rank approximation, singular value decomposition, just to name a few. For survey and recent advances about sketching, we refer the readers to (Halko et al., 2011; Mahoney, 2011; Lu et al., 2013; Alaoui and Mahoney, 2014; Woodruff, 2014; Raskutti and Mahoney, 2015; Yang et al., 2015a; Oymak et al., 2015; Oymak and Tropp, 2015; Drineas and Mahoney, 2016) and references therein. However, one major drawback of sketching is that typically it’s not suitable for the case if we want high accurate solution: to obtain a solution with exponentially smaller approximation error, we often need to increase the sketching dimension also exponentially.

The situation has become better with recent work on “iterative sketch”, e.g. iterative Hessian sketch (IHS) (Pilanci and Wainwright, 2016) and iterative dual random projection (IDRP) (Zhang et al., 2014). These methods are able to refine their approximate solution by iteratively solving some small scale sketched problem. Among these innovations, Hessian sketch (Pilanci and Wainwright, 2016) is designed by reducing the sample size of the original problem, while dual random projection (Zhang et al., 2014) is proposed by reducing the dimension. 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 make the following improvement upon previous work: we first propose an accelerated version of IHS which requires the same computational cost to solve the IHS subproblem at each sketching iteration, while with provably fewer number of sketching iterations to reach certain accuracy; we then reveal the primal-dual connections between IHS (Pilanci and Wainwright, 2016) and IDRP (Zhang et al., 2014), which are independently proposed by two different groups of researchers. In particular, we show that these two methods are equivalent in the sense that dual random projection is performing Hessian sketch in the dual space. Finally, to 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 sketched sub-problem, with provable convergence guarantees.

**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 Hessian sketch, and propose the corresponding accelerated dual random projection as well. In Section 4 we combine the sketching from both primal and dual perspectives, and propose iterative algorithms by reduc-
ing both sample size and problem dimension. We provide several theoretical analysis in Section 5, though we defer few technical results to the appendices, and conduct extensive experiments in Section 6. Finally we summarize and discuss several future directions in Section 7.

**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. Given a matrix \( \mathbf{X} \in \mathbb{R}^{n \times p} \), we define the following matrix induced norm for any vector \( \mathbf{w} \in \mathbb{R}^p \),

\[
\|\mathbf{w}\|_X = \sqrt{\frac{\mathbf{w}^\top \mathbf{X}^\top \mathbf{X} \mathbf{w}}{n}}.
\]

We use \( \mathcal{N}(\mu, \Sigma) \) to denote the multivariate normal distribution with mean \( \mu \) and covariance \( \Sigma \). We use \( \mathbf{I}_n \) and \( \mathbf{I}_p \) to denote the identity matrices of size \( n \times n \) and \( p \times p \), and \( \lambda_{\text{max}}(\mathbf{H}) \) and \( \lambda_{\text{min}}(\mathbf{H}) \) to denote the maximum and minimum eigenvalue of \( \mathbf{H} \), respectively. For two sequences \( \{a_n\}_{n=1}^\infty \) and \( \{b_n\}_{n=1}^\infty \), we denote \( a_n \preceq b_n \) if \( a_n \leq C b_n \) always holds for \( n \) large enough with some constant \( C \), and denote \( a_n \succeq b_n \) if \( b_n \preceq a_n \). We also use the notation \( a_n = \mathcal{O}(b_n) \) if \( a_n \preceq b_n \), and use \( \tilde{\mathcal{O}}(\cdot) \) 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 (Pilanci and Wainwright, 2016) as an iterative preconditioned optimization process, and then propose a faster iterative algorithm by constructing better sketched problem to solve.

For ease of discussion, consider the following \( \ell_2 \) regularized least-squares (a.k.a. ridge regression) problem:

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

(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. Let \( \mathbf{w}^* \) denote the optimum of problem (2.1).

In real applications both \( n \) and \( p \) can be very large, thus sketching has become a widely used technique for finding an approximate solution of problem (2.1) efficiently (Drineas et al., 2011; Mahoney, 2011; Woodruff, 2014). In particular, to avoid solving a problem of huge sample size, the traditional sketching techniques (e.g. (Sarlos, 2006; Pilanci and Wainwright, 2015b)) were proposed to reduce the sample size from \( n \) to \( m \), where \( m \ll n \), by solving the following sketched problem:

\[
\min_{\mathbf{w} \in \mathbb{R}^p} P(\Pi^\top \mathbf{X}, \Pi^\top \mathbf{y}; \mathbf{w}) = \min_{\mathbf{w} \in \mathbb{R}^p} \frac{1}{2m} \|\Pi^\top \mathbf{y} - \Pi^\top \mathbf{X} \mathbf{w}\|_2^2 + \frac{\lambda}{2} \|\mathbf{w}\|_2^2,
\]

(2.2)

where \( \Pi \in \mathbb{R}^{n \times m} \) is a sketching matrix, and typical choice of \( \Pi \) can be random Gaussian matrix, matrix with Rademacher entries, Sub-sampled Randomized Hadamard Transform (Boutsidis and Gittens, 2013) and Sub-sampled Randomized Fourier Transform (Rokhlin and Tygert, 2008), see discussions in Section 2.1 of (Pilanci and Wainwright, 2016) for details.

Though the classical sketching has been successful in various problems with provable guarantees, as shown in (Pilanci and Wainwright, 2016) there exists a approximation limit for the classical
sketching methods to be practically useful, that is, to obtain an approximate solution with high precision, the sketching dimension \( m \) needs to grow exponentially, which is impractical if we want a high accuracy approximation to the original problem, as the main purpose of sketching is to speed up the algorithms via reducing the sample size.

The main idea of Hessian sketch (Pilanci and Wainwright, 2016) is based on the following equivalent formulation of (2.1):

\[
\min_{w \in \mathbb{R}^p} P(X, y; w) = \min_{w \in \mathbb{R}^p} \frac{1}{2n} \|y\|^2 + \frac{1}{2n} \|Xw\|^2 - \frac{1}{n} \langle y, Xw \rangle + \frac{\lambda}{2} \|w\|^2, \tag{2.3}
\]

and (Pilanci and Wainwright, 2016) proposed to only sketch the quadratic part \( \|Xw\|^2 \) with respect to \( X \), but not the linear part \( \langle y, Xw \rangle \). So the Hessian sketch considers solving the following sketched problem:

\[
\min_{w \in \mathbb{R}^p} P_{\text{HS}}(X, y; \Pi, w) = \min_{w \in \mathbb{R}^p} \frac{1}{2n} \|y\|^2 + \frac{1}{2n} \|\Pi^T Xw\|^2 - \frac{1}{n} \langle y, Xw \rangle + \frac{\lambda}{2} \|w\|^2. \tag{2.4}
\]

It is not hard to see that (2.4) has the following closed form solution:

\[
\hat{w}_{\text{HS}} = \left( \lambda I_p + \frac{X^T \Pi \Pi^T X}{n} \right)^{-1} \frac{X^T y}{n}. \tag{2.5}
\]

We see that different from classical sketching where both data matrix \( X \) and the response vector \( y \) are both sketched, in Hessian sketch the only sketched part is the Hessian matrix, through the following transform:

\[
X^T X \to X^T \Pi \Pi^T X.
\]

Though the Hessian sketch suffers from the same approximation limit as the classical sketch, one notable feature of Hessian sketch is that one can implement an iterative extension to refine the approximation to higher precision. To this end, define the initial Hessian sketch approximation as \( \hat{w}_{\text{HS}}^{(1)} \):

\[
\hat{w}_{\text{HS}}^{(1)} = \arg \min_{w} \left( \frac{X^T \Pi \Pi^T X}{2n} + \frac{\lambda}{2} I_p \right) w - \frac{1}{n} \langle y, Xw \rangle.
\]

After obtaining \( \hat{w}_{\text{HS}}^{(1)} \), we can consider the following optimization problem:

\[
\arg \min_{u} \frac{1}{2n} \left\| y - X(u + \hat{w}_{\text{HS}}^{(1)}) \right\|^2 + \frac{\lambda}{2} \left\| u + \hat{w}_{\text{HS}}^{(1)} \right\|^2
\]

\[
= \arg \min_{u} u^T \left( \frac{X^T X}{2n} + \frac{\lambda}{2} I_p \right) u - \left\langle \frac{X^T (y - X \hat{w}_{\text{HS}}^{(1)})}{n} - \lambda \hat{w}_{\text{HS}}^{(1)}, u \right\rangle.
\]

It is clear that \( w^* - \hat{w}_{\text{HS}}^{(1)} \) is the optimum for above problem. The main idea of iterative Hessian sketch (IHS) is to approximate the residual \( w^* - \hat{w}_{\text{HS}}^{(1)} \) by Hessian sketch again. At time \( t \), let \( u^{(t)} \) be the approximate of \( w^* - \hat{w}_{\text{HS}}^{(1)} \) via solving the following sketched problem:

\[
\hat{u}^{(t)} = \arg \min_{u} u^T \left( \frac{X^T \Pi \Pi^T X}{2n} + \frac{\lambda}{2} I_p \right) u - \left\langle \frac{X^T (y - X \hat{w}_{\text{HS}}^{(t)})}{n} - \lambda \hat{w}_{\text{HS}}^{(t)}, u \right\rangle, \tag{2.6}
\]
Algorithm 1: Iterative Hessian Sketch (IHS).

1 Input: Data $X, y$, sketching matrix $\Pi$.
2 Initialization: $\hat{w}_{\text{HS}}^{(0)} = 0$.
3 for $t = 0, 1, 2, \ldots$ do
4 \hspace{1em} Update the approximation by $\hat{w}_{\text{HS}}^{(t+1)} = \hat{w}_{\text{HS}}^{(t)} + \hat{u}^{(t)}$, where $\hat{u}^{(t)}$ is obtained by solving the sketched problem (2.6).
5 end

and we then update $\hat{w}^{(t+1)}$ by $\hat{w}_{\text{HS}}^{(t+1)} = \hat{w}_{\text{HS}}^{(t)} + \hat{u}^{(t)}$.

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). Besides, we can reuse the previous sketched data $\Pi^T X$ without constructing any new random sketching matrix. Moreover, (Pilanci and Wainwright, 2016) showed that the approximation error of IHS is exponentially decreasing when we increase the number of sketching iterations, thus IHS could find an approximated solution with $\epsilon$-approximation error within $O(\log(1/\epsilon))$ iterations, as long as the sketching dimension $m$ is large enough. Moreover, though this powerful technique is originally focused on the least-squares problem (2.1), the idea of IHS can be extended to solve more general problems, such as constrained least-squares (Pilanci and Wainwright, 2016), optimization with self-concordant loss (Pilanci and Wainwright, 2015a), as well as non-parametric methods (Yang et al., 2015b).

Though IHS improved the classical sketching by enabling us to find an high quality approximation more efficiently, it is imperfect due to the following reasons:

- The “exponentially approximation error decreasing” guarantee relies on the basis that the sketching dimension $m$ is large enough. The necessary sketching dimension depends on the intrinsic complexity of the problem, however, if the “sufficient sketching dimension” condition is violated, as we will show in experiments, IHS can even diverge, i.e. we obtain arbitrary worse approximation after applying IHS.

- As we will show later, even when the “sufficient sketching dimension” condition is satisfied, the decreasing speed of the approximation error in IHS can be significantly improved.

Here, we show that the iterative Hessian sketch is in fact an optimization process with preconditioning. For notation simplicity let

$$H = \frac{X^T X}{n} + \lambda I_p \quad \text{and} \quad \tilde{H} = \frac{X^T \Pi \Pi^T X}{n} + \lambda I_p,$$

and we use the following gradient notion on $P(X, y; w)$:

$$\nabla P(w) = -\frac{X^T (y - Xw)}{n} + \lambda w.$$
Then it is not hard to see that the IHS in Algorithm 1 is performing the following iterative update:

\[
\hat{w}_{HS}^{(t+1)} = \hat{w}_{HS}^{(t)} - \tilde{H}^{-1} \nabla P(\hat{w}_{HS}^{(t)}),
\]

which is like a Newton update where we replace the true Hessian \(H\) with the sketched Hessian \(\tilde{H}\).

This update can be derived by using the change of variable \(z = \tilde{H}^{1/2}w\), and then applying gradient descent in the \(z\) space:

\[
\hat{z}^{(t+1)} = \hat{z}^{(t)} - \nabla_z P(\tilde{H}^{-1/2}z) = \hat{z}^{(t)} - \tilde{H}^{-1/2} \nabla_x P(\tilde{H}^{-1/2}\hat{z}^{(t)}).
\]

By changing the update back to the original space by multiplying \(\tilde{H}^{-1/2}\), we get back the IHS update:

\[
\hat{w}_{HS}^{(t+1)} = \hat{w}_{HS}^{(t)} - \tilde{H}^{-1} \nabla P(\hat{w}_{HS}^{(t)}).
\]

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 (Hestenes and Stiefel, 1952; Nocedal and Wright, 2006). Since iterative Hessian sketch is gradient descent in the transformed space \(z = \tilde{H}^{1/2}w\), this suggests accelerating by performing conjugate gradient in the transformed space \(^1\), instead of gradient descent.

This leads to the algorithm Acc-IHS as detailed in Algorithm 2, where at each iteration we call the solver for the following sketched linear system:

\[
\hat{u}^{(t)} = \arg \min_u \left( \frac{X^\top \Pi \Pi^\top X}{2n} + \frac{\lambda}{2} I_p \right) u - \left\langle r^{(t)}, u \right\rangle.
\]

(2.7)

Unlike IHS which uses \(\tilde{H}^{-1} \nabla P(\hat{w}_{HS}^{(t)})\) as the update direction at time \(t\), for Acc-IHS we use \(p^{(t)}\) as the update direction at time \(t\), and \(p^{(t)}\) is chosen to satisfy the conjugate condition:

\[
\left( p^{(t_1)} \right)^\top \left( \frac{X^\top \Pi \Pi^\top X}{2n} + \frac{\lambda}{2} I_p \right)^{-1} \left( \frac{X^\top X}{2n} + \frac{\lambda}{2} I_p \right) p^{(t_2)} = 0, \forall t_1, t_2 \geq 0, t_1 \neq t_2.
\]

Since the updating direction is conjugate to the previous directions, it is guaranteed that after \(p\) iterations, we reach the exact minimizer, i.e.,

\[
\hat{w}_{HS}^{(t)} = w^*, \forall t \geq p.
\]

Moreover, Acc-IHS shares the same computational cost with standard IHS in solving each sketched subproblem, but as we will show in the theoretical analysis part, the convergence rate of Algorithm 2 is much faster than IHS, i.e., we need to solve much smaller number of sketched subproblems than IHS to reach the same approximation accuracy.

---

\(^1\)Equivalently, we can implicitly transform the space by defining inner product as \(\langle x, y \rangle = x^\top \tilde{H} y\).
Algorithm 2: Accelerated Iterative Hessian Sketch (Acc-IHS).

1. **Input:** Data $X, y$, sketching matrix $\Pi$.
2. **Initialization:** $\hat{w}_{HS}(0) = 0, r(0) = -\frac{X^T y}{n}$.
3. Compute $\hat{u}(0)$ by solving (2.7), and update $p^{(0)} = -\hat{u}(0)$, calculate $v^{(0)} = \left( \frac{X^T X}{n} + \lambda I_p \right) p^{(0)}$.
4. for $t = 0, 1, 2, \ldots$ do
   5. Calculate $\alpha^{(t)} = \langle r^{(t)}, u^{(t)} \rangle / \langle p^{(t)}, v^{(t)} \rangle$.
   6. Update the approximation by $\hat{w}_{HS}^{(t+1)} = \hat{w}_{HS}^{(t)} + \alpha^{(t)} p^{(t)}$.
   7. Update $r^{(t+1)} = r^{(t)} + \alpha^{(t)} v^{(t)}$.
   8. Update $u^{(t+1)}$ by solving (2.7).
   9. Update $\beta^{(t+1)} = \langle r^{(t+1)}, u^{(t)} \rangle / \langle r^{(t)}, r^{(t)} \rangle$.
   10. Update $p^{(t+1)} = -u^{(t+1)} + \beta^{(t+1)} p^{(t)}$.
   11. Update $v^{(t+1)} = \left( \frac{X^T X}{n} + \lambda I_p \right) p^{(t+1)}$.
5. end

3. **Equivalence between Dual Random Projection and Hessian Sketch**

While Hessian sketch (Pilanci and Wainwright, 2016) tries to resolve the issue of huge sample size, Dual Random Projection (Zhang et al., 2013, 2014) is aimed to resolve the issue of high-dimensionality, where random projection as a standard technique is used to significantly reduce the dimension of data points. Again consider the standard ridge regression problem in (2.1), but now random projection is used to transform the original problem (2.1) to a low-dimensional problem:

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

(3.1)

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

Let $\tilde{z} = \arg \min_z P_{RP}(XR, y; z)$. If we want to recover the original high-dimensional solution, (Zhang et al., 2014) observed that the naive recovered solution $\hat{w}_{RP} = R\tilde{z}$ is a bad approximation, and propose to recover $w^*$ from the dual solution, which leads to the dual random projection (DRP) approach. To see this, consider the dual problem of the optimization problem in (2.1) as

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

(3.2)

Let $\alpha^* = \arg \max_{\alpha \in \mathbb{R}^n} D(X, y; \alpha)$ be the dual optimal solution. By standard primal-dual theory (Boyd and Vandenberghe, 2004), we have the following connection between the optimal primal and dual solutions:

$$
\alpha^* = y - Xw^* \quad \text{and} \quad w^* = \frac{1}{\lambda n} X^T \alpha^*.
$$

(3.3)

The dual random projection procedure works as follows: first, we construct solve the low-dimensional, randomly projected problem (3.1) and obtain the solution $\tilde{z}$, and then calculate the approximated
dual variables by:

\[ \hat{\alpha}_{\text{DRP}} = y - XR\hat{z}, \]  
(3.4)

based on the approximated dual solution \( \hat{\alpha}_{\text{DRP}} \). Then we recover the primal solution as:

\[ \hat{w}_{\text{DRP}} = \frac{1}{\lambda n} X^\top \hat{\alpha}_{\text{DRP}}. \]  
(3.5)

By combining above derivations, it is not hard to see that dual random projection for ridge regression has the following closed form solution:

\[ \hat{w}_{\text{DRP}} = \frac{1}{\lambda n} \left( \lambda I_n + \frac{XX^\top}{n} \right)^{-1} y. \]  
(3.6)

In (Zhang et al., 2014) it has been shown that the recovered solution from dual, i.e. \( \hat{w}_{\text{DRP}} \), is a much better approximation than the recovered solution directly from primal \( \hat{w}_{\text{RP}} \). More specifically, they showed that \( \hat{w}_{\text{RP}} \) is always a poor approximation of \( w^\ast \), because \( \hat{w}_{\text{RP}} \) lives in a random subspace spanned by the random projection matrix \( R \). For \( \hat{w}_{\text{DRP}} \), (Zhang et al., 2014) proved that as long as the projected dimension \( d \) is large enough, \( \hat{w}_{\text{DRP}} \) can be a good approximation of \( w^\ast \).

Moreover, (Zhang et al., 2014) proposed an iterative extension of DRP which can exponentially reducing the approximation error. To do so, suppose at iteration \( t \) we have the approximate solution \( \hat{w}_{\text{DRP}}^{(t)} \), and consider the following optimization problem:

\[ \min_{u \in \mathbb{R}^p} \frac{1}{2n} \left\| y - X(u + \hat{w}_{\text{DRP}}^{(t)}) \right\|_2^2 + \frac{\lambda}{2} \left\| u + \hat{w}_{\text{DRP}}^{(t)} \right\|_2^2. \]  
(3.7)

It is clear to see \( w^\ast - \hat{w}_{\text{DRP}}^{(t)} \) is the optimum solution of above optimization problem. The idea of iterative dual random projection (IDRP) is to approximate the residual \( w^\ast - \hat{w}_{\text{DRP}}^{(t)} \) by applying dual random projection again. That is, we construct the following randomly projected problem given \( \hat{w}_{\text{DRP}}^{(t)} \):

\[ \min_{z \in \mathbb{R}^p} \frac{1}{2n} \left\| y - Xw_{\text{DRP}}^{(t)} - XRz \right\|_2^2 + \frac{\lambda}{2} \left\| z + R\hat{w}_{\text{DRP}}^{(t)} \right\|_2^2. \]  
(3.8)

Let \( \hat{z}^{(t)} \) to be the solution of (3.8), then we update the refined approximation of dual variables:

\[ \hat{\alpha}_{\text{DRP}}^{(t+1)} = y - Xw_{\text{DRP}}^{(t)} - XR\hat{z}, \]

as well as the primal variables

\[ \hat{w}_{\text{DRP}}^{(t+1)} = \frac{1}{\lambda n} X^\top \hat{\alpha}_{\text{DRP}}^{(t+1)}. \]

The iterative dual random projection (IDRP) algorithm is shown in Algorithm 3. More generally, (Zhang et al., 2014) showed the iterative dual random projection can be used to solve any \( \ell_2 \) regularized empirical loss minimization problem as long as the loss function is smooth, typical examples include logistic regression, support vector machines with smoothed hinge loss, etc.

Though a powerful technique to cope with high-dimensionality issue, IDRP suffers from the same limitations as IHS: i) it requires the “large projection dimension” condition to make the approximation error decreasing, ii) the convergence speed of IDRP is not optimal. As will shown later, actually the dual random projection is equivalent to apply the Hessian sketch procedure on the dual problem, and we propose an accelerated IDRP approach to overcome above discussed limitations.
Algorithm 3: Iterative Dual Random Projection (IDRP).

1. **Input:** Data $X, y$, projection matrix $R$.
2. **Initialization:** $\hat{w}_{DRP}^{(0)} = 0$.
3. for $t = 0, 1, 2, \ldots$ do
4.   Solve the projected problem in (3.8) and obtain solution $\hat{z}_{DRP}^{(t)}$.
5.   Update dual approximation:
6.     $\hat{\alpha}_{DRP}^{(t+1)} = \lambda_n \hat{\alpha}_{DRP}^{(t+1)} - X \hat{w}_{DRP}^{(t)} - X R \hat{z}_{DRP}^{(t)}$.
7.   Update primal approximation:
8.     $\hat{w}_{DRP}^{(t+1)} = \frac{1}{\lambda_n} X^T \hat{\alpha}_{DRP}^{(t+1)}$.
9. end

3.1 Dual Random Projection is Hessian Sketch in Dual Space

In this section we present one of the key observation, i.e., the equivalence 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 a quadratic with respect to $\alpha$, and we can write the equivalent minimization objective in the following form:

$$
\min_{\alpha \in \mathbb{R}^n} \alpha^T \left( \frac{XX^T}{2\lambda n} + \frac{1}{2} I_n \right) \alpha - \langle y, \alpha \rangle.
$$

We can treat (3.9) as our primal problem and applying the Hessian sketch with sketching matrix $R \in \mathbb{R}^{p \times d}$ to find an approximated solution for $\alpha^*$:

$$
\hat{\alpha}_{HS} = \arg \min_{\alpha \in \mathbb{R}^n} \alpha^T \left( \frac{XX^T}{2\lambda n} + \frac{1}{2} I_n \right) \alpha - \langle y, \alpha \rangle,
$$

which has the closed form solution as

$$
\hat{\alpha}_{HS} = \lambda \left( \lambda I_n + \frac{XX^T}{n} \right)^{-1} y.
$$

If we substitute $\hat{\alpha}_{HS}$ to the primal-dual connection (3.3), we obtained an approximated primal solution:

$$
\hat{w} = \frac{X^T}{n} \left( \lambda I_n + \frac{XX^T}{n} \right)^{-1} y.
$$

Compared with the DRP approximation (3.6), we see these two approximations coincident, thus we see that *Dual Random Projection is Hessian sketch applied in dual space*. For ridge regression
problem (2.1) one have closed form solutions for various sketching techniques as:

\begin{align*}
\text{Original:} & \quad \hat{w}^* = \left( \lambda I_p + \frac{X^\top X}{n} \right)^{-1} X^\top y = \frac{X^\top}{n} \left( \lambda I_n + \frac{XX^\top}{n} \right)^{-1} y \\
\text{Classical Sketch:} & \quad \tilde{w}_{CS} = \left( \lambda I_p + \frac{X^\top \Pi \Pi^\top X}{n} \right)^{-1} X^\top \Pi \Pi^\top y \\
\text{Random Projection:} & \quad \tilde{w}_{RP} = R \left( \lambda I_d + \frac{R^\top X^\top XR}{n} \right)^{-1} R^\top X^\top y \\
\text{Hessian Sketch:} & \quad \tilde{w}_{HS} = \left( \lambda I_p + \frac{X^\top \Pi \Pi^\top X}{n} \right)^{-1} X^\top y \\
\text{Dual Random Projection:} & \quad \tilde{w}_{DRP} = \frac{X^\top}{n} \left( \lambda I_n + \frac{XRR^\top X^\top}{n} \right)^{-1} y
\end{align*}

As we can see above, Hessian sketch is essentially sketching the covariance matrix:
\[ X^\top X \rightarrow X^\top \Pi \Pi^\top X, \]
while DRP is essentially sketching the Gram matrix:
\[ XX^\top \rightarrow XRR^\top X^\top. \]

### 3.2 Accelerated Iterative Dual Random Projection

Based on the equivalence between dual random projection and Hessian sketch established in Section 3.1, we proposed an accelerated iterative dual random projection algorithms which improves the
convergence speed of standard iterative DRP procedure (Zhang et al., 2014). The algorithm is shown in Algorithm 4, in which at each iteration $t$, we call the solver for the following randomly projected problem based on the residual $r(t)$:

$$\hat{z}^{(t)} = \arg \min_{z \in \mathbb{R}^d} z^T \left( \frac{R^T X^T X R}{2n} + \frac{\lambda}{2} I_d \right) z - \langle R^T X^T r^{(t)}, z \rangle. \quad (3.11)$$

The accelerated IDRP algorithms simulate the accelerated IHS algorithm (2), but run Acc-IHS in the dual space. However, Acc-IDRP is still an primal algorithm since it updates the corresponding dual variables back from solving the randomly projected primal problem (3.11). Algorithm 5 summarized the accelerated IDRP algorithms directly from the dual problem. We note that it is not a practical algorithm as it requires solving the relatively more expensive dual problem, however it is easier to understand as it directly borrows the ideas of Acc-IHS described in Section 25. For the dual version of Acc-IDRP algorithm, at each iteration it is required to solve the following dual optimization problem given the dual residual $r(t)$:

$$\hat{u}^{(t)} = \arg \min_{u \in \mathbb{R}^n} u^T \left( \frac{X R R^T X^T}{2n} + \frac{\lambda}{2} I_n \right) u - \langle r^{(t)}, u \rangle. \quad (3.12)$$

As we will show later, though the computational cost per iteration of Acc-IDRP and standard IDRP is the same, Acc-IDRP has the following advantages over IDRP:

- As a preconditioned conjugate gradient procedure, Acc-IDRP is guaranteed to converge, and reach the optimum $w^*$ within $n$ iterations, even when the projection dimension $d$ is very small.

- When the projection dimension $d$ is large enough to make 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 combines the idea of iterative Hessian sketch and iterative dual random projection from the primal-dual point of view, and propose a more efficient sketching technique named Iterative Primal-Dual Sketch (IPDS) which simultaneously reduce the sample size and dimensionality of the problem, while still recovering the original solution to a high precision. Hessian sketch is particular suitable for the case where sample size is much larger than problem dimension, where the computational bottleneck is big $n$, and it is possible to use Hessian sketch to reduce the sample size significantly thus speed up the computation, and uses the iterative extension to reducing approximation error further to recover the original solution to a high precision; while dual random projection goes to another direction: it is mostly suitable for the case of high-dimensional data but relatively small sample size, where the computational bottleneck is “large $p$”, and we could like to use random projection to perform dimension reduction thus gain speedup. Table 1 summarized these characteristics.

As shown in Table 1, Hessian sketch and dual random projection are suitable for case when the problem scales are not balanced, i.e. for the sample size and dimensions, one is much larger than
Algorithm 5: Accelerated Iterative Dual Random Projection (Acc-IDRP)—Dual Version.

1. **Input**: Data \(X, y\), projection matrix \(R\).
2. **Initialization**: \(\hat{w}_{\text{DRP}}^{(0)} = \mathbf{0}, \hat{\alpha}_{\text{DRP}}^{(0)} = \mathbf{0}, r^{(0)} = -y\).
3. Compute \(u^{(0)}\) by solving (3.12), and update \(p^{(0)} = -u^{(0)}\), \(v^{(0)} = \left(\frac{XX^T}{n} + \lambda I_n\right)p^{(0)}\).
4. **for** \(t = 0, 1, 2, \ldots \) **do**
   5. Calculate \(a^{(t)} = \frac{\langle r^{(t)}, u^{(t)} \rangle}{\langle p^{(t)}, v^{(t)} \rangle}\).
   6. Update the dual approximation by \(\hat{\alpha}^{(t+1)}_{\text{DRP}} = \hat{\alpha}^{(t)}_{\text{DRP}} + a^{(t)}p^{(t)}\).
   7. Update primal approximation: \(\hat{w}^{(t+1)}_{\text{DRP}} = \frac{1}{\lambda n}X^T\hat{\alpha}^{(t+1)}_{\text{DRP}}\).
   8. Update \(r^{(t+1)} = r^{(t)} + a^{(t)}v^{(t)}\).
   9. Update \(u^{(t+1)}\) by solving (3.12).
   10. Update \(\beta^{(t+1)} = \frac{\langle r^{(t+1)}, u^{(t)} \rangle}{\langle r^{(t+1)}, r^{(t)} \rangle}\).
   11. Update \(p^{(t+1)} = -u^{(t+1)} + \beta^{(t+1)}p^{(t)}\).
   12. Update \(v^{(t+1)} = \left(\frac{XX^T}{n} + \lambda I_n\right)p^{(t+1)}\).
4. **end**

| Approach       | Suitable Situation | Reduced Dimension | Recovery | Iterative |
|----------------|--------------------|-------------------|----------|-----------|
| Classical Sketch | large \(n\), small \(p\) | sample reduction | \(\times\) | \(\times\) |
| Random Projection | small \(n\), large \(p\) | dimension reduction | \(\times\) | \(\times\) |
| Hessian Sketch  | large \(n\), small \(p\) | sample reduction | \(\checkmark\) | \(\checkmark\) |
| DRP             | small \(n\), large \(p\) | dimension reduction | \(\checkmark\) | \(\checkmark\) |

Table 1: Comparison of various algorithms for data sketching in solving large-scale problems.

the other. For modern massive datasets, it is usually the case where both \(n\) and \(p\) are very large, for example, the click-through rate (CTR) prediction data sets provided by Criteo\(^2\) has \(n \geq 4 \times 10^9\) and \(p \geq 8 \times 10^8\). Thus it is desirable to have a sketching method to simultaneously reduce the huge sample size and dimensionality.

Inspired by the primal-dual view described in Section 3.1, we propose the iterative Primal-Dual Sketch, which only involves solving small scale problems. For the original problem (2.1) with data \(\{X, y\}\), we first construct the randomly projected data, as well as the doubly sketched data, as follows:

\[
X \rightarrow XR, \quad XR \rightarrow \Pi^T XR,
\]

where \(XR\) is the randomly projected data, and \(\Pi^T XR\) is doubly sketched data by sketching \(XR\) via sample reduction. We initialize the Primal-Dual Sketch solution as \(\hat{w}_{\text{DS}}^{(0)} = (0)\), and at every iteration, we first apply random projection on the primal problem (which is equivalent to Hessian Sketch on the dual problem), and obtain the following problem:

\[
\min_{z \in \mathbb{R}^d} \frac{1}{2n} \|y - X\hat{w}_{\text{DS}}^{(t)} - XRz\|_2^2 + \frac{\lambda}{2} \|z\|_2 + R^T\hat{w}_{\text{DS}}^{(t)}\|_2^2, \quad (4.1)
\]

\(^2\)http://labs.criteo.com/downloads/download-terabyte-click-logs/
Algorithm 6: Iterative Primal-Dual Sketch (IPDS).

1. **Input**: Data $X \in \mathbb{R}^{n \times p}$, $y \in \mathbb{R}^n$, sketching matrix $R \in \mathbb{R}^{p \times d}$, $\Pi \in \mathbb{R}^{n \times m}$.

2. **Initialization**: $\hat{w}_{DS}^{(0)} = 0$.

3. **for** $t = 0, 1, 2, \ldots$ **do**

   4. **Initialization**: $\tilde{z}^{(0)} = 0$, $k = 0$

   5. **if** Not converged **then**

      6. Solve the sketched problem in (4.2) and obtain solution $\Delta z^{(k)}$.

      7. Update $\tilde{z}^{(k+1)} = \tilde{z}^{(k)} + \Delta z^{(k)}$.

      8. Update $k = k + 1$.

   **end**

10. Update dual approximation: $\hat{\alpha}_{DS}^{(t+1)} = y - X\hat{w}_{DS}^{(t)} - X\tilde{z}^{(k+1)}$.

11. Update primal approximation: $\hat{w}_{DS}^{(t+1)} = \frac{1}{\lambda n} X^\top \hat{\alpha}_{DS}^{(t+1)}$.

**end**

which is the same as the iterative dual random projection subproblem (3.8). However, different from IDRP, we don’t directly solve (4.1), but apply the iterative Hessian sketch to find an approximate solution. Note that the expanded form of (4.1) is

$$\min_{z \in \mathbb{R}^d} z^\top \left( \frac{R^\top X^\top X R}{2n} + \frac{\lambda}{2} I_d \right) z - \frac{1}{n} \left( y - X\hat{w}_{DS}^{(t)} - X\tilde{z}^{(k+1)} \right)^\top X R \hat{\alpha}_{DS}^{(t)}.$$

We apply Hessian sketch to above problem, and iteratively refine the solution. That is, we first initialize an approximate solution of (4.1) of $\tilde{z}^{(0)}$ as $0$, then at inner loop iteration $k$ find a solution for the following sketched problem:

$$\Delta z^{(k)} = \arg \min_{\Delta z} \Delta z^\top \left( \frac{R^\top X^\top X R}{2n} + \frac{\lambda}{2} I_d \right) z - \frac{1}{n} \left( y - X\hat{w}_{DS}^{(t)} - X\tilde{z}^{(k+1)} \right)^\top X R \hat{\alpha}_{DS}^{(t)} - \lambda R^\top \hat{w}_{DS}^{(t)} - \lambda \tilde{z}^{(k)}$$.

(4.2)

then update $\tilde{z}^{(k+1)}$ as

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

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 need to solve sub-problems of size $m \times p$, while IDRP need to solve sub-problems of size $n \times d$. As will show in the theoretical analysis, 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:

$$X \rightarrow \Pi^\top X, \quad \Pi^\top X \rightarrow \Pi^\top X R.$$
Algorithm 7: Accelerated Iterative Primal-Dual Sketch (Acc-IPDS).

1 Input: Data $X \in \mathbb{R}^{n \times p}$, $y \in \mathbb{R}^n$, sketching matrix $R \in \mathbb{R}^{p \times d}$, $\Pi \in \mathbb{R}^{n \times m}$.

2 Initialization: $\hat{w}_{\text{DS}}^{(0)} = 0, \hat{\alpha}_{\text{DS}}^{(0)} = 0, r_{\text{Dual}}^{(0)} = -y$.

3 Initialization: $k = 0, \hat{z}^{(k)} = 0, r_p^{(0)} = R^T X^T r_D^{(0)}$.

4 Compute $\hat{u}_p^{(0)}$ by solving (4.3), and update $p_p^{(0)} = -u_p^{(0)}$, calculate $v_p^{(0)} = \left(\frac{R^T X R}{n} + \lambda I_d\right) p_p^{(0)}$.

5 if Not converged then

6 Calculate $a_p^{(k)} = \langle r_p^{(k)}, u_p^{(k)} \rangle / \langle p_p^{(k)}, v_p^{(k)} \rangle$, and update the approximation by $\hat{z}^{(k+1)} = \hat{z}^{(k)} + a_p^{(k)} p_p^{(k)}$.

7 Update $r_p^{(k+1)} = r_p^{(k)} + a_p^{(k)} v^{(k)}$, and update $u_p^{(k+1)}$ by solving (4.3).

8 Update $\beta_p^{(k+1)} = \langle r_p^{(k+1)}, u_p^{(k+1)} \rangle / \langle r_p^{(k+1)}, r_p^{(k+1)} \rangle$, and update $p_p^{(k+1)} = -u_p^{(k+1)} + \beta_p^{(k+1)} p_p^{(k)}$.

9 Update $v_p^{(k+1)} = \left(\frac{R^T X R}{n} + \lambda I_p\right) p_p^{(t+1)}$, and update $k = k + 1$.

10 end

11 Compute $u_D^{(0)} = r_D^{(0)} - X R \hat{z}^{(k+1)}$, $p_D^{(0)} = -u_D^{(0)}$, $v_D^{(0)} = \left(\frac{X^T X}{n} + \lambda I_n\right) p_D^{(0)}$.

12 for $t = 0, 1, 2, \ldots$ do

13 Calculate $a_D^{(t)} = \langle r_D^{(t)}, u_D^{(t)} \rangle / \langle p_D^{(t)}, v_D^{(t)} \rangle$ and update the dual approximation by $\hat{\alpha}_{\text{DS}}^{(t+1)} = \hat{\alpha}_{\text{DS}}^{(t)} + a_D^{(t)} p_D^{(t)}$.

14 Update primal approximation: $\hat{w}_{\text{DS}}^{(t+1)} = \frac{1}{\lambda n} X^T \hat{\alpha}_{\text{DS}}^{(t+1)}$, and update $r_D^{(t+1)} = r_D^{(t)} + a_D^{(t)} v_D^{(t)}$.

15 Initialization: $k = 0, \hat{z}^{(k)} = 0, r_p^{(0)} = R^T X^T r_D^{(t+1)}$.

16 Compute $\hat{u}_p^{(0)}$ by solving (4.3), and update $p_p^{(0)} = -u_p^{(0)}$, calculate $v_p^{(0)} = \left(\frac{R^T X R}{n} + \lambda I_p\right) p_p^{(0)}$.

17 if Not converged then

18 Calculate $a_p^{(k)} = \langle r_p^{(k)}, u_p^{(k)} \rangle / \langle p_p^{(k)}, v_p^{(k)} \rangle$, and update the approximation by $\hat{z}^{(k+1)} = \hat{z}^{(k)} + a_p^{(k)} p_p^{(k)}$.

19 Update $r_p^{(k+1)} = r_p^{(k)} + a_p^{(k)} v^{(k)}$, and update $u_p^{(k+1)}$ by solving (4.3).

20 Update $\beta_p^{(k+1)} = \langle r_p^{(k+1)}, u_p^{(k+1)} \rangle / \langle r_p^{(k+1)}, r_p^{(k+1)} \rangle$, and update $p_p^{(k+1)} = -u_p^{(k+1)} + \beta_p^{(k+1)} p_p^{(k)}$.

21 Update $v_p^{(k+1)} = \left(\frac{R^T X R}{n} + \lambda I_p\right) p_p^{(t+1)}$, and update $k = k + 1$.

22 end

23 Update $u_D^{(t+1)} = r_D^{(t+1)} - X R \hat{z}^{(k+1)}$, and update $\beta_D^{(t+1)} = \langle r_D^{(t+1)}, u_D^{(t+1)} \rangle / \langle r_D^{(t+1)}, r_D^{(t+1)} \rangle$.

24 Update $p_D^{(t+1)} = -u_D^{(t+1)} + \beta_D^{(t+1)} p_D^{(t)}$, and update $v_D^{(t+1)} = \left(\frac{X^T X}{n} + \lambda I_n\right) p_D^{(t+1)}$.

25 end
Also, the idea presented in Section 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 (Summarized in Algorithm 7). In Acc-IPDS, we maintains the both vectors in primal space $u_P, v_P, r_P$ and vectors in dual space $u_D, v_D, r_D$, to make sure the updating directions for both primal variables and dual variables are conjugate with previous updating directions. Moreover, based on the residual vector $r_P$, Acc-IPDS iteratively calls the solver to find solution of the following sketched linear system of scale $m \hat{d}$:

$$
\hat{u}_p^{(k)} = \arg\min_u u^T \left( \frac{R^T X^T \Pi \Pi^T X R}{2n} + \frac{\lambda}{2} I_d \right) u - \langle r_p^{(k)}, u \rangle. \tag{4.3}
$$

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

In this section we present the theoretical analysis of various iterative sketching procedures, and defer the omitted proofs to Appendix A. First we will provide a unified analysis of Hessian sketch as dual random projection. The unified analysis basically follows the analysis of (Zhang et al., 2014) and (Pilanci and Wainwright, 2016), but simultaneously provide recovering guarantees for both primal and dual variables of interest. Then we move to the convergence analysis of the proposed accelerated IHS and IDRP algorithms, where we will show improved convergence speed over standard IHS and IDRP. Finally, we prove the iterative primal-dual sketch will converge to the optimum within iterations only grow logarithmically with the target approximation accuracy.

### 5.1 A unified analysis of Hessian Sketch and Dual Random Projection

In this section we provide a simple, unified analysis for the recovery performance of Hessian Sketch and Dual random projection. As in (Pilanci and Wainwright, 2016), we use the following notion of Gaussian width for any set $K \subseteq \mathbb{R}^p$:

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

where $g$ is a random vector drawn from normal distribution $\mathcal{N}(0, I_p)$. Intuitively speaking, if the set $K$ is restrictive to certain directions, then $\mathbb{W}(K)$ should be small as well (Vershynin, 2015). Given a set $K$ and a random matrix $R \in \mathbb{R}^{p \times d}$, the following quantities will be important for further analysis:

$$
\rho_1(K, R) = \inf_{u \in K \cap S^{p-1}} u^T R R^T u \quad \text{and} \quad \rho_2(K, R, v) = \sup_{u \in K \cap S^{p-1}} |u^T (R R^T - I_p) v|,
$$

where $S^{p-1}$ is the $p$-dimensional unit-sphere. Firstly we could like the sketching matrix $R$ to satisfy

$$
\mathbb{E}[RR^T] = I_p,
$$

and moreover, the matrix $RR^T$ becomes closer to $I_p$ as sketching dimension $d$ increases. Thus we would like to push $\rho_1(K, R)$ to be close to 1, and $\rho_2(K, R, v)$ to be close to 0.
For the sake of simplicity, we just assume the random matrix $R$ is sampled i.i.d. from some $\frac{1}{\sqrt{d}}$-sub-Gaussian distributions, this can be done by first sample a matrix $\tilde{R}$ where entries are sampled i.i.d. from 1-sub-Gaussian distribution, then perform the normalization as:

$$R = \frac{\tilde{R}}{\sqrt{d}}.$$  

The following lemma, from Pilanci and Wainwright (2015b), states how large the sketching dimension $d$ should be to make $\rho_1(\mathcal{K}, R), \rho_2(\mathcal{K}, R, v)$ to be close to 1 and 0, respectively.

**Lemma 5.1.** When $R$ is sampled i.i.d. from $1/\sqrt{d}$-sub-Gaussian distributions, then there exists universal constants $C_0$ such that, we have with probability at least $1 - \delta$, we have

$$\rho_1(\mathcal{K}, R) \geq 1 - C_0 \sqrt{\frac{\mathbb{W}^2(\mathcal{K})}{d}} \log \left( \frac{1}{\delta} \right) \quad \text{and} \quad \rho_2(\mathcal{K}, R, v) \leq C_0 \sqrt{\frac{\mathbb{W}^2(\mathcal{K})}{d}} \log \left( \frac{1}{\delta} \right).$$

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

$$X\mathcal{K} = \{ u \in \mathbb{R}^n | u = Xv, v \in \mathcal{K} \}.$$  

To present the main results about the unified analysis. Let’s recall the main reductions in Hessian sketch and dual random projection. For Hessian sketch, we perform sample reduction with the transformation $X \rightarrow \Pi^\top X$; for dual random projection, we perform dimension reduction with the transformation $X \rightarrow XR$, where $\Pi \in \mathbb{R}^{n \times m}$, $R \in \mathbb{R}^{p \times d}$. Let $\hat{w}_{HS}$ be the approximated solution via Hessian sketch by solving (2.4), and the corresponding dual variables by the following transform

$$\hat{\alpha}_{HS} = y - X\hat{w}_{HS}.$$  

Likewise, let $\hat{\alpha}_{DRP}$ and $\hat{w}_{DRP}$ be the approximated dual variables and primal variables obtained by dual random projection. The following theorem established the recovery bound for $\hat{\alpha}_{HS}, \hat{\alpha}_{DRP}$ and $w_{HS}, \hat{w}_{DRP}$ simultaneously.

**Theorem 5.2.** Suppose we perform Hessian sketch or dual random projection for problem (2.1) with sub-Gaussian sketching matrix $\Pi \in \mathbb{R}^{n \times m}$ (for HS) or $R \in \mathbb{R}^{p \times d}$ (for DRP). Then there exists universal constants $C_0$ such that with probability at least $1 - \delta$, the following approximation error bounds for HS or DRP holds:

For Hessian sketch:

$$\| \hat{w}_{HS} - w^* \|_X \leq \frac{C_0 \sqrt{\frac{\mathbb{W}^2(\mathbb{R}^{p \times p})}{m}} \log \left( \frac{1}{\delta} \right)}{1 - C_0 \sqrt{\frac{\mathbb{W}^2(\mathbb{R}^{p \times p})}{m}} \log \left( \frac{1}{\delta} \right)} \| w^* \|_X ,$$  

$$\| \hat{\alpha}_{HS} - \alpha^* \|_2 \leq \sqrt{nC_0 \sqrt{\frac{\mathbb{W}^2(\mathbb{R}^{p \times p})}{m}} \log \left( \frac{1}{\delta} \right)} \| w^* \|_X ,$$

16
For dual random projection:

\[
\|\mathbf{\hat{w}}_{\text{DRP}} - \mathbf{w}^*\|_2 \leq \frac{C_0 \sqrt{\frac{W^2(X^\top R^n)}{d}} \log \left(\frac{1}{\varepsilon}\right)}{1 - C_0 \sqrt{\frac{W^2(X^\top R^n)}{d}} \log \left(\frac{1}{\varepsilon}\right)} \|\mathbf{w}^*\|_2, \\
\|\mathbf{\hat{\alpha}}_{\text{DRP}} - \mathbf{\alpha}^*\|_{X^\top} \leq \frac{C_0 \sqrt{\frac{W^2(X^\top R^n)}{d}} \log \left(\frac{1}{\varepsilon}\right)}{1 - C_0 \sqrt{\frac{W^2(X^\top R^n)}{d}} \log \left(\frac{1}{\varepsilon}\right)} \|\mathbf{\alpha}^*\|_{X^\top},
\]

where the norm \(\|\cdot\|_X\) is defined as \(\|\mathbf{w}\|_X = \sqrt{\frac{\mathbf{w}^\top X \mathbf{w}}{n}}\).

Remark. We have the following remarks for Theorem 5.2.

- For general low-dimensional problems where \(n \gg p\), \(W^2(X^\top R^p) = p\), thus we have \(\|\mathbf{\hat{w}}_{\text{HS}} - \mathbf{w}^*\|_X \leq \sqrt{\frac{p}{n}} \log \left(\frac{1}{\varepsilon}\right) \|\mathbf{w}^*\|_X\), which is the recovery bound proved in (Pilanci and Wainwright, 2016) (Proposition 1 in their paper).
- For high-dimensional problems when \(p\) is large, \(W^2(X^\top R^n) = n\), thus we have \(\|\mathbf{\hat{w}}_{\text{DRP}} - \mathbf{w}^*\|_2 \leq \sqrt{\frac{n}{d}} \log \left(\frac{1}{\varepsilon}\right) \|\mathbf{w}^*\|_2\). Moreover, when \(X\) is low-rank, i.e. rank\((X) = r\) and \(r \ll \min(n, p)\), we have \(W^2(X^\top R^n) = r\), thus we have \(\|\mathbf{\hat{w}}_{\text{DRP}} - \mathbf{w}^*\|_2 \leq \sqrt{\frac{r}{d}} \log \left(\frac{1}{\varepsilon}\right) \|\mathbf{w}^*\|_2\), which is the recovery bound obtained in Theorem 1 of (Zhang et al., 2014), in fact the bound established in Theorem 5.2 improves Theorem 1 of (Zhang et al., 2014) by removing an additional \(\sqrt{\log r}\) factor.

5.1.1 Analysis of IHS and DRP when \(X\) is approximately low-rank

In this section we provide recovery guarantees for the case when the data matrix \(X\) is \textit{approximately} low rank. To make \(X\) can we well approximated by a rank \(r\) matrix where \(r \ll \min(n, p)\), we assume \(\sigma_{r+1}\), the \(r + 1\)-th singular value of \(X\), is small enough. Suppose \(X\) admits the following singular value decomposition:

\[X := U \Sigma V^\top = U \Sigma_r V^\top + U \Sigma_{\bar{r}} V^\top = X_r + X_{\bar{r}},\]

where \(\bar{r}\) denotes the index \(\{r + 1, \ldots, \max(n, p)\}\). We also requires \(\mathbf{w}^*\) can be well approximated by the a linear combination of top \(r\) left singular vectors of \(X\), i.e. the remaining singular vectors are almost orthogonal with \(\mathbf{w}^*\), depends on the method (Hessian sketch or dual random projection), we require the following notion of orthogonality holds for \(\rho\) and \(\varrho\) which are small:

\[\|X_r \mathbf{w}^*\|_2 \leq \rho \|X \mathbf{w}^*\|_2, \quad \|V_{\bar{r}}^\top \mathbf{w}^*\|_2 \leq \varrho \|\mathbf{w}^*\|_2,\]

where \(V_{\bar{r}} \in \mathbb{R}^{p \times r}\) is the remaining right singular vectors of \(X\). Also, to simplify the results, let 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 zero-mean Gaussian distributions, with variance \(\frac{1}{m}\) and \(\frac{1}{n}\), respectively. We have the following recovery bounds for Hessian sketch and dual random projection:
Theorem 5.3. With probability at least $1 - \delta$, the following approximation error bounds for HS
and DRP hold:

**For Hessian sketch:** if

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

then

$$\| \hat{w}_{\text{HS}} - w^* \|_X \leq 4 \sqrt{\frac{1}{1 - \epsilon_1} + \frac{\sigma_{r+1}^2}{\lambda n}} \cdot \sqrt{\epsilon_1^2 + \tau_1^2\rho^2 + \frac{\tau_1^2\sigma_{r+1}^2 + \rho^2v_1^2\sigma_{r+1}^2}{\lambda n}} \| w^* \|_X.$$  

**For 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

$$\| \hat{w}_{\text{DRP}} - w^* \|_2 \leq 4 \sqrt{\frac{1}{1 - \epsilon_2} + \frac{\sigma_{r+1}^2}{\lambda n}} \cdot \sqrt{\epsilon_2^2 + \tau_2^2\sigma^2 + \frac{\tau_2^2\sigma_{r+1}^2 + \sigma^2v_2^2\sigma_{r+1}^2}{\lambda n}} \| w^* \|_2. \quad (5.6)$$

where $\epsilon_1, \epsilon_2, \tau_1, \tau_2, v_1, v_2$ are defined as

$$\epsilon_1 = 2\sqrt{\frac{2(r + 1)}{m} \log \frac{2r}{\delta}}, \quad \epsilon_2 = 2\sqrt{\frac{2(r + 1)}{d} \log \frac{2r}{\delta}},$$

$$\tau_1 = 7 \sqrt{\frac{2(n - r)}{m} \log \frac{n}{\delta}}, \quad \tau_2 = 7 \sqrt{\frac{2(p - r)}{d} \log \frac{p}{\delta}},$$

$$v_1 = 2\sqrt{\frac{2(n - r + 1)}{m} \log \frac{2(n - r)}{\delta}}, \quad v_2 = 2\sqrt{\frac{2(p - r + 1)}{d} \log \frac{2(p - r)}{\delta}}.$$

**Remark.** We make the following comments on Theorem 5.3:

- When $\sigma_{r+1} = 0$, i.e. $X$ is exactly rank $r$, above results becomes

$$\| \hat{w}_{\text{HS}} - w^* \|_X \leq \sqrt{\frac{r}{m}} \| w^* \|_X, \quad \| \hat{w}_{\text{DRP}} - w^* \|_2 \leq \sqrt{\frac{r}{d}} \| w^* \|_2 \quad (5.7)$$

which reduces to the results in Theorem 5.2.

- We see that if we have $\sigma_{r+1}, \rho, \sigma$ sufficiently small in the following order, i.e. for Hessian sketch:

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

for DRP:

$$\sigma_{r+1} \lesssim \sqrt{\frac{\lambda n}{p}}, \quad \sigma \lesssim \sqrt{\frac{r}{p}},$$

the guarantees (5.7) still hold.
5.2 Analysis of the accelerated IHS and IDRP methods

In this section we provide convergence analysis for the proposed Acc-IHS and Acc-IDRP approaches. As discussed before, since Acc-IHS and Acc-IDRP are preconditioned conjugate gradient methods on primal and dual problems, respectively, with a sketched Hessian as a preconditioner. By classical analysis of preconditioned conjugate gradient (Luenberger), we have the following convergence guarantees:

**Proposition 5.4.** For Acc-IHS, we have

\[
\left\| \hat{w}_\text{HS}^{(t)} - w^* \right\|_X \leq 2 \left( \frac{\sqrt{\kappa_\text{HS}(X, \Pi, \lambda)} - 1}{\sqrt{\kappa_\text{HS}(X, \Pi, \lambda)} + 1} \right)^t \|w^*\|_X,
\]

and Acc-IDRP, we have

\[
\left\| \hat{\alpha}_\text{DRP}^{(t)} - \alpha^* \right\|_{X^T} \leq 2 \left( \frac{\sqrt{\kappa_\text{DRP}(X, R, \lambda)} - 1}{\sqrt{\kappa_\text{DRP}(X, R, \lambda)} + 1} \right)^t \|\alpha^*\|_{X^T}.
\]

where

\[
\kappa_\text{HS}(X, \Pi, \lambda) = \kappa \left( \left( \frac{X^T \Pi \Pi^T X}{n} + \lambda I_p \right)^{-1} \left( \frac{X^T X}{n} + \lambda I_p \right) \right),
\]

\[
\kappa_\text{DRP}(X, R, \lambda) = \kappa \left( \left( \frac{X R R^T X^T}{n} + \lambda I_n \right)^{-1} \left( \frac{X X^T}{n} + \lambda I_n \right) \right).
\]

From Proposition 5.4, we know the convergence of Acc-IHS and Acc-IDRP heavily depends on the condition number \( \kappa_\text{HS}(X, \Pi, \lambda) \) and \( \kappa_\text{DRP}(X, R, \lambda) \). Thus the key of the rest of the analysis is to upper bound the condition numbers. To analyze the condition numbers, we make use of the following result in (Mendelson et al., 2007).

**Lemma 5.5.** If the elements in \( \Pi \in \mathbb{R}^{n \times m} \) are i.i.d. sampled from a zero-mean \( \frac{1}{m} \)-sub-Gaussian distribution, then there exists universal constants \( C_0 \) such that, for any subset \( K \subseteq \mathbb{R}^n \), with probability at least \( 1 - \delta \), we have

\[
\sup_{u \in K \cap S^{n-1}} |u^T (\Pi \Pi^T - I_n) u| \leq C_0 \sqrt{\frac{\|\Pi\|^2(K)}{m} \log \left( \frac{1}{\delta} \right)}.
\]

Based on above lemma, we have the following bounds on the condition numbers \( \kappa_\text{HS}(X, \Pi, \lambda) \) and \( \kappa_\text{DRP}(X, R, \lambda) \):

**Theorem 5.6.** If the sketching matrix \( \Pi \in \mathbb{R}^{n \times m} \) and \( R \in \mathbb{R}^{p \times d} \) are sampled from \( \frac{1}{\sqrt{m}} \)-sub-Gaussian and \( \frac{1}{\sqrt{d}} \)-sub-Gaussian distributions, respectively, then with probability at least \( 1 - \delta \), the
following upper bounds hold:

\[
\kappa_{\text{HS}}(X, \Pi, \lambda) \leq \frac{1}{1 - 2C_0 \sqrt{\frac{W^2(X \Pi)}{m}}} \log \left(\frac{1}{\delta}\right).
\]

\[
\kappa_{\text{DRP}}(X, R, \lambda) \leq \frac{1}{1 - 2C_0 \sqrt{\frac{W^2(X' R)}{d}}} \log \left(\frac{1}{\delta}\right).
\]

With Theorem 5.6, we immediately obtain the following corollary which states the overall convergence for Acc-IHS and Acc-IDRP:

**Corollary 5.7.** Suppose the sketching matrix \(\Pi \in \mathbb{R}^{n \times m}\) and \(R \in \mathbb{R}^{p \times d}\) are sub-Gaussian, if \(t\), the number of iterations of Acc-IHS satisfies

\[
t \geq \left\lceil \left( \frac{1}{1 - 2C_0 \sqrt{\frac{W^2(X \Pi)}{m}}} \right) \log \left(\frac{2 \|w^*\|_X}{\epsilon}\right) \right\rceil
\]

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

\[
\|\hat{w}^{(t)} - w^*\|_X \leq \epsilon.
\]

If the number of iterations of Acc-IDRP satisfies

\[
t \geq \left\lceil \left( \frac{1}{1 - 2C_0 \sqrt{\frac{W^2(X' R)}{d}}} \right) \log \left(\frac{2 \|w^*\|_2}{\epsilon}\right) \right\rceil
\]

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

\[
\|\hat{w}^{(t)}_{\text{DRP}} - w^*\|_2 \leq \epsilon.
\]

**Remark.** To compare the convergence rate of Acc-IHS, and Acc-IDRP with standard IHS (Pilanci and Wainwright, 2016) and IDRP (Zhang et al., 2014), we observe that

- For IHS, the number of iterations to reach \(\epsilon\)-accuracy (Corollary 1 in (Pilanci and Wainwright, 2016)) is

\[
t \geq \left\lceil \left( \frac{1+\rho}{1-\rho} \right) \log \left(\frac{2\|w^*\|_X}{\epsilon}\right) \right\rceil
\]

where \(\rho = C_0 \sqrt{\frac{W^2(X \Pi)}{m}}} \log \left(\frac{1}{\delta}\right)\). Acc-IHS reduces the number of iterations to \(t \geq \left\lceil \left( \frac{1+\rho}{1-\rho} \right) \log \left(\frac{2\|w^*\|_X}{\epsilon}\right) \right\rceil\), which is significant when \(\rho\) relatively large. Moreover, IHS requires \(m \gtrsim W^2(X \Pi)\) to holds to converge, while Acc-IHS is always convergent.

- For IDRP, Theorem 7 in (Zhang et al., 2014) considered low-rank data, which requires IDRP to reach \(\epsilon\)-accuracy when \(t \geq \left\lceil \left( \frac{1+\rho}{1-\rho} \right) \log \left(\frac{2\|w^*\|_2}{\epsilon}\right) \right\rceil\), where \(\rho = C_0 \sqrt{\frac{r}{d}}} \log \left(\frac{1}{\delta}\right)\). Acc-IDRP reduces the number of iterations to \(t \geq \left\lceil \left( \frac{1+\rho}{1-\rho} \right) \log \left(\frac{2\|w^*\|_2}{\epsilon}\right) \right\rceil\). Moreover, IDRP requires \(d \gtrsim r \log r\) to holds to converge, while Acc-IDRP is always convergent.
5.3 Analysis for the primal-dual sketch methods

In this section, we provide runtime theoretical analysis for the proposed primal-dual sketch methods, where the sketched dual problem is not solved exactly, but approximately solved via sketching the primal problem again. At outer loop iteration \( t \), the standard analysis of iterative Hessian sketch (Pilanci and Wainwright, 2016) and Theorem 5.2), we have the following lemma:

**Lemma 5.8.** Let \( \tilde{\mathbf{w}}_{\text{HS}}^{(t+1)} \) be the iterate defined in Algorithm 1, then we have the following inequality:

\[
\left\| \tilde{\mathbf{w}}_{\text{HS}}^{(t+1)} - \mathbf{w}^* \right\|_\mathbf{X} \leq \frac{C_0 \sqrt{\frac{W^2(X \mathbb{R}^p)}{m}} \log \left( \frac{1}{\delta} \right)}{1 - C_0 \sqrt{\frac{W^2(X \mathbb{R}^p)}{m}} \log \left( \frac{1}{\delta} \right)} \left\| \mathbf{w}^{(t)}_{\text{HS}} - \mathbf{w}^* \right\|_\mathbf{X}.
\]

However, in iterative primal-dual sketch, we don’t have access to the exact minimizer \( \tilde{\mathbf{w}}_{\text{HS}}^{(t+1)} \), instead an approximate minimizer \( \tilde{\mathbf{w}}_{\text{HS}}^{(t+1)} \) which is close to \( \tilde{\mathbf{w}}_{\text{HS}}^{(t+1)} \). The key is the analyze the iteration complexities of inner loops.

**Theorem 5.9.** With probability at least \( 1 - \delta \), we have the following approximation error bound for \( \tilde{\mathbf{w}}_{\text{HS}}^{(t+1)} \) in iterative primal-dual sketch:

\[
\left\| \tilde{\mathbf{w}}_{\text{HS}}^{(t+1)} - \mathbf{w}^* \right\|_\mathbf{X} \leq \left( \frac{C_0 \sqrt{\frac{W^2(X \mathbb{R}^p)}{m}} \log \left( \frac{1}{\delta} \right)}{1 - C_0 \sqrt{\frac{W^2(X \mathbb{R}^p)}{m}} \log \left( \frac{1}{\delta} \right)} \right)^t \| \mathbf{w}^* \|_\mathbf{X} + \frac{10 \lambda^2_{\text{max}} \left( \frac{X^T X}{n} \right)}{\lambda^2} \left( \frac{C_0 \sqrt{\frac{W^2(X \mathbb{R}^p)}{m}} \log \left( \frac{1}{\delta} \right)}{1 - C_0 \sqrt{\frac{W^2(X \mathbb{R}^p)}{m}} \log \left( \frac{1}{\delta} \right)} \right)^k \| \mathbf{w}^* \|_2.
\]

With Theorem 5.9, we have the following iterative complexity for the proposed IPDS approach.

**Corollary 5.10.** If the number of outer loops \( t \) and number of inner loops \( k \) in IPDS satisfying the following:

\[
t \geq \left[ \frac{1 + C_0 \sqrt{\frac{W^2(X \mathbb{R}^p)}{m}} \log \left( \frac{1}{\delta} \right)}{1 - C_0 \sqrt{\frac{W^2(X \mathbb{R}^p)}{m}} \log \left( \frac{1}{\delta} \right)} \right] \log \left( \frac{4 \| \mathbf{w}^* \|_\mathbf{X}}{\epsilon} \right)
\]

\[
k \geq \left[ \frac{1 + C_0 \sqrt{\frac{W^2(X \mathbb{R}^p)}{d}} \log \left( \frac{1}{\delta} \right)}{1 - C_0 \sqrt{\frac{W^2(X \mathbb{R}^p)}{d}} \log \left( \frac{1}{\delta} \right)} \right] \log \left( \frac{40 \lambda^2_{\text{max}} \left( \frac{X^T X}{n} \right) \| \mathbf{w}^* \|_2}{\lambda \epsilon} \right)
\]

Then with probability at least \( 1 - \delta \):

\[
\left\| \tilde{\mathbf{w}}_{\text{IPDS}}^{(t+1)} - \mathbf{w}^* \right\|_\mathbf{X} \leq \epsilon.
\]

**Proof.** Apply Theorem 5.9 and substitute above inequalites for \( t \) and \( k \) we get \( \left\| \tilde{\mathbf{w}}_{\text{IPDS}}^{(t+1)} - \mathbf{w}^* \right\|_\mathbf{X} \leq \frac{\epsilon}{2} + \frac{\epsilon}{2} = \epsilon. \)
Remark. Since the total number of sketched subproblem to solve in iterative primal-dual sketch is $tk$. To obtain $\epsilon$ approximation error, we have the total number of subproblems is

$$tk \leq \left[ 1 + \sqrt{\frac{\mathcal{W}^2(X \mathbb{R}^p)}{m}} \cdot \frac{1 + \sqrt{\mathcal{W}^2(X^T \mathbb{R}^n)}}{d} \right] \log^2 \left( \frac{1}{\epsilon} \right).$$

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

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

**Corollary 5.11.** If the number of outer loops $t$ and number of inner loops $k$ in IPDS satisfying the following:

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

$$k \geq \left\lceil \frac{1}{\sqrt{1 - 2C_0 \sqrt{\frac{\mathcal{W}^2(X^T \mathbb{R}^n)}{d}}} \log \left( \frac{1}{\delta} \right)} \right\rceil \log \left( \frac{40 \lambda^2_{\max} \left( \frac{X^T X}{n} \right) \| w^* \|_2}{\lambda \epsilon} \right)$$

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

$$\left\| w_{\text{IPDS}}^{(t+1)} - w^* \right\|_X \leq \epsilon.$$

**Proof.** The proof is similar to the proof of Theorem 5.9 and then substitute the lower bounds for $t$ and $k$ to obtain the result. \qed

### 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: sample size $n$, problem dimension $p$ as well as the problem condition. To make the comparison simpler, we simply assume $X$ is rank $r$, note that $r$ might be much smaller than $n, p$: $r \ll n, p$. For (2.1) generally the regularization parameter $\lambda$ is chosen at the order of $O(1/\sqrt{n})$ to $O(1/n)$ (Sridharan et al., 2009; Dhillon et al., 2013), here in favor of the iterative optimization algorithms we simply choose the large $\lambda$, i.e. of order $O(1/\sqrt{n})$. For iterative optimization algorithms, the convergence usually depend on the smoothness of the problem. In (2.1), the smoothness parameter is $\lambda_{\max} \left( \frac{X^T X}{n} + \lambda I_p \right)$, which is often of the order $O(p)$ (e.g. random sub-Gaussian design). To compare the runtime for solving (2.1), we consider the following methods:

- **Solving Linear System:** which solves the problem exactly using matrix inversion, which requires $O(np^2 + p^3)$. 

22
| Approach / Runtime | $\mathcal{O}(\cdot)$ | $\mathcal{O}(\cdot)$ | Comment |
|--------------------|----------------------|----------------------|---------|
| Linear System      | $np^2 + p^3$         | $np^2 + p^3$         |         |
| LS with Low-rank SVD| $npr + r^3$          | $npr + r^3$          |         |
| Gradient Descent   | $(n^{1.5}p^5) \log \left( \frac{1}{\lambda} \right)$ | $n^{1.5}p^5$     |         |
| Acc. Gradient Descent| $(n^{1.25}p^{1.5}) \log \left( \frac{1}{\epsilon} \right)$ | $n^{1.25}p^{1.5}$ |         |
| Coordinate Descent | $(n^{1.5}p) \log \left( \frac{1}{\epsilon} \right)$ | $n^{1.5}p$         |         |
| SVRG,SDCA,SAG      | $(np + n^{0.75}p^2) \log \left( \frac{1}{\epsilon} \right)$ | $np + n^{0.75}p^2$ |         |
| Catalyst,APPA      | $(np + n^{0.75}p^{1.5}) \log \left( \frac{1}{\epsilon} \right)$ | $np + n^{0.75}p^{1.5}$ |         |
| DSPDC               | $npr + (nr + n^{0.75}p^{1.5}r) \log \left( \frac{1}{\epsilon} \right)$ | $npr + n^{0.75}p^{1.5}r$ |         |
| IHS + Catalyst     | $np \log p + n^{0.25}p^{1.5}r \log^2 \left( \frac{1}{\epsilon} \right)$ | $np + n^{0.25}p^{1.5}r$ | Fast when $p \ll n$ |
| DRP + Exact        | $np \log n + (nr^2 + r^3) \log \left( \frac{1}{\epsilon} \right)$ | $np + nr^2 + r^3$ | Fast when $n \ll p$ |
| Iter.primal-dual   | $np \log p + (n + r^3) \log^2 \left( \frac{1}{\epsilon} \right)$ | $np + r^3$         | Fast when $r \ll \max(p,n)$ |

Table 2: Comparison of various approaches for solving large scale problems (2.1), the runtime depend on $n, p, r, \varepsilon$.

- **Linear System with Low-rank SVD**: if we have the factorization $X = U^T V$, where $U \in \mathbb{R}^{n \times r}$, $V \in \mathbb{R}^{p \times r}$. Then we can solve the matrix inversion efficiently with the Sherman-Morrison-Woodbury formula: 
  \[
  \left( \lambda I_p + \frac{X^T X}{n} \right)^{-1} = \frac{1}{\lambda} I_p - \frac{1}{\lambda^2} V U^T (I_r + V^T V U) \lambda U V^T \right)^{-1} V^T.
  \]
  Which can be done in $\mathcal{O}(npr + r^3)$.

- **Gradient Descent**: standard analysis (Nesterov, 2013) shows gradient descent requires $\mathcal{O}\left(\left(\frac{1}{\lambda}\right)^2 \log \left(\frac{1}{\epsilon}\right)\right)$ iterations, with each iteration $\mathcal{O}(np)$, Since $L = \mathcal{O}(p), \lambda = \mathcal{O}(1/\sqrt{n})$. We have the overall runtime is $\mathcal{O}\left((n^{1.5}p^2) \log \left(\frac{1}{\epsilon}\right)\right)$.

- **Accelerated Gradient Descent** (Nesterov, 2013): which requires $\mathcal{O}\left(\sqrt{\left(\frac{1}{\lambda}\right)} \log \left(\frac{1}{\epsilon}\right)\right)$ iterations, with each iteration $\mathcal{O}(np)$, Since $L = \mathcal{O}(p), \lambda = \mathcal{O}(1/\sqrt{n})$. We have the overall runtime is $\mathcal{O}\left((n^{1.25}p^{1.5}) \log \left(\frac{1}{\epsilon}\right)\right)$.

- **Randomized Coordinate Descent** (Nesterov, 2012): which requires $\mathcal{O}\left(\left(\frac{1}{\lambda}\right) \log \left(\frac{1}{\epsilon}\right)\right)$ iterations, with each iteration $\mathcal{O}(n)$, Since $\lambda = \mathcal{O}(1/\sqrt{n})$. We have the overall runtime is $\mathcal{O}\left((n^{1.5}p) \log \left(\frac{1}{\epsilon}\right)\right)$.

- **SVRG,SDCA,SAG** (Johnson and Zhang, 2013; Shalev-Shwartz and Zhang, 2013; Roux et al., 2012): which requires $\mathcal{O}\left(\left(n + \frac{1}{\lambda}\right) \log \left(\frac{1}{\epsilon}\right)\right)$ iterations, with each iteration $\mathcal{O}(p)$. Since $L = \mathcal{O}(p), \lambda = \mathcal{O}(1/\sqrt{n})$. We have the overall runtime is $\mathcal{O}\left((np + n^{0.5}p^2) \log \left(\frac{1}{\epsilon}\right)\right)$.

- **Accelerated SVRG**: Catalyst,APPA,SPDC,RPDG (Lin et al., 2015; Frostig et al., 2015; Zhang and Lin, 2015; Lan and Zhou, 2015): which requires $\mathcal{O}\left(\left(n + \sqrt{n\frac{1}{\lambda}}\right) \log \left(\frac{1}{\epsilon}\right)\right)$ iterations, with each iteration $\mathcal{O}(p)$. Since $L = \mathcal{O}(p), \lambda = \mathcal{O}(1/\sqrt{n})$. We have the overall runtime is $\mathcal{O}\left((np + n^{0.75}p^{1.5}) \log \left(\frac{1}{\epsilon}\right)\right)$.

- **DSPDC** (Yu et al., 2015): which requires $\mathcal{O}\left(\left(n + \sqrt{n\frac{1}{\lambda}}\right) \log \left(\frac{1}{\epsilon}\right)\right)$ iterations, with each iteration $\mathcal{O}(r)$. Since $L = \mathcal{O}(p), \lambda = \mathcal{O}(1/\sqrt{n})$. Also, to apply DSPDC, one should compute
the low-rank factorization as a preprocessing step which takes \( O(npr) \). Thus we have the overall runtime is \( O(npr + (nr + n^{0.75}p^{0.5}r) \log (\frac{1}{\varepsilon})) \).

- **Iterative Hessian Sketch + Accelerated SVRG (Pilanci and Wainwright, 2016):** compute sketched problem takes \( O(np \log p) \) (e.g. via fast Johnson-Lindenstrauss transforms (Ailon and Chazelle, 2009)), solve \( O(\log (\frac{1}{\varepsilon})) \) sketched problems use accelerated SVRG type algorithm takes \( O(n^{0.25}r^{1.5} \log (\frac{1}{\varepsilon})) \). This leads to the overall runtime: \( O(np \log p + n^{0.25}p^{1.5}r \log (\frac{1}{\varepsilon})) \).

- **DRP + Matrix inversion (Zhang et al., 2014):** compute sketched problem takes \( O(np \log n) \). Solve \( O(\log (\frac{1}{\varepsilon})) \) reduced problem with matrix inversion takes \( O(nr^2 + r^3) \). This leads to the overall runtime: \( O(np \log n + (nr^2 + r^3) \log (\frac{1}{\varepsilon})) \).

- **Iterative Primal-Dual Sketch:** compute sketched problem takes \( O(np \log p) \). Solve \( O(\log^2 (\frac{1}{\varepsilon})) \), and solve reduced problem exactly takes \( O(n + r^3) \). Overall runtime: \( O(np \log p + (n + r^3) \log^2 (\frac{1}{\varepsilon})) \).

### 6 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, and then show the proposed iterative primal-dual sketch procedure and its accelerated version could simultaneously reduce the sample size and dimensions of the problem, while still maintaining high approximation precision. Then we test these algorithms on some real world data sets.

#### 6.1 Simulations for Acc-IHS and Acc-IDRP

We first examine the effectiveness of the proposed Acc-IHS and Acc-IDRP algorithms on simulated data. The response variable \( \{y_i\}_{i \in [n]} \) are drawn from the following linear model:

\[
y_i = \langle x_i, \beta^* \rangle + \epsilon_i,
\]

where the noise \( \epsilon_{ji} \) is sampled from a standard normal distribution. The true model \( \beta^* \) 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 data. We generate \( \{x_i\}_{i \in [n]} \) from multivariate normal distribution with mean zero vector, and covariance matrix \( \Sigma \), which controls the condition number of the problem. We will varying \( \Sigma \) 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 varying the dimensions with \( p = 50, 100, 300 \). The results are shown in Figure 1, where 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 converges to the optimum, which is consistent with the theory, as also observed in (Pilanci and Wainwright, 2016) and (Zhang et al., 2014) for IHS and IDRP algorithm.
When compared with IHS and Acc-IHS, we observed Acc-IHS converges significantly faster than IHS. 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 cases we tested, Acc-IHS converges faster than IHS even when its sketching dimension is only 1/3 of the sketching dimension in IHS.

We then compare the proposed Acc-IDRP with the standard IDRP on high-dimensional, but relatively low-rank data. We generate \( \{x_i\}_{i=1}^{n} \) from a low-rank factorization: \( X = UV^T \), where the entries in \( U \in \mathbb{R}^{n \times r} \), \( V \in \mathbb{R}^{p \times r} \) are sampled i.i.d from standard normal distribution. We fix the sample size \( n = 10^4 \) and varying the dimensions with \( p = 2000, 5000, 20000 \), we also vary the rank \( r = 20, 50 \). The results are shown in Figure 2, where for each problem setting, we test 3 different sketching dimensions (number inside parentheses in legend). We have similar observations with the IHS case, i.e. Acc-IDRP always converges significantly faster than IDRP, even in the low sketching dimension case where IDRP diverge.

Above experiments validate the theoretical analysis which showed the accelerated procedures for IHS and IDRP could significantly boost the convergence of their standard counterpart. Since the computational cost per iteration of the standard iterative sketching techniques and their accelerated version is almost the same, thus Acc-IHS and Acc-IDRP will be useful techniques for iterative sketching with faster convergence speed.

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

\[
X \sim \mathcal{N}(0, \Sigma), \Sigma_{ij} = 0.5|i-j|/10.
\]
### 6.2 Simulations for IPDS and Acc-IPDS

In this section, we demonstrate how iterative primal-dual sketch and its accelerated version work for simulated data. We generated the data using the same procedure as the simulation for Acc-DRP, where we fix the low-rank data with rank 10, and vary the original sample size $n$ and dimension $p$. For primal-dual sketching, we reduce the sample size to $m$, and dimension to $d$, where $m \ll n, d \ll p$. We also compare with standard IHS and IDRP, where for IHS we only perform sample reduction from $n$ to $m$, for IDRP only data dimension is reduced from $p$ to $d$. Thus the subproblem size for IPDS (and Acc-IPDS), IHS, IDRP, are $m \times d$, $m \times p$, $n \times d$, respectively. For IPDS and Acc-IPDS, we terminate the inner loop when the $\ell_\infty$ distance between two inner iterations are 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:

- Though simultaneously reduce the sample size and data dimension, IPDS and Acc-IPDS are able to recover the optimum to a very high precision. However, they require generally more iterations to reach a certain approximation level compared with IHS and IDRP, where at each iteration we need to solve a substantially larger scale subproblem. Therefore, primal-dual sketching approach still enjoy more computational advantages. For example, on problem of scale $(n, p) = (10000, 20000)$, IHS and IDRP need to solve 5 sub-problems of scale $(m, p) = (500, 20000)$ and $(n, d) = (10000, 500)$, respectively, while for Acc-IPDS, it 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 verified the effectiveness of
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    |

the proposed acceleration procedure for these sketching techniques.

6.3 Experiments on real datasets

We also conduct experiments on some real-world data sets where the statistics of them are summarized in Table 3. Among all the data sets, the first 3 are cases where sample size is significantly larger than the data dimension, where we used to compare the IHS and Acc-IHS algorithms; the middle 3 data sets are high-dimensional datasets with small sample size, where we compare to DRP and Acc-DRP procedures; the last 3 datasets are cases where sample size and data dimension 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 fails (unless with very large sketching dimension), thus we compared with Acc-IHS and Acc-DRP. We follow the same experimental setup with the simulation study, and the convergence plots are summarized in Figure 4. We have the following observations:

- Acc-IHS and Acc-DRP converges significantly faster than IHS and DRP, respectively, where similar observation is drawn from simulation studies.

- For the last 3 data sets where \( n \) and \( p \) are both large, and the data is not exactly low-rank: IHS, DRP and IPDS often diverge because of the requirement of the sketching dimension to ensure convergence is high, while the accelerated versions still converges to the optimum. It is notable that the Acc-IPDS only requires solving several least squares problems where both sample size and dimension are relatively small.

7 Conclusion and Discussion

In this paper, we focused on sketching techniques for solving large-scale \( \ell_2 \) regularized least square problems, we established the equivalence between the recently proposed two emerging techniques (Hessian sketch and dual random projection) from a primal-dual point of view, we also proposed accelerated methods for IHS and IDRP, from the preconditioned optimization perspective, and
by combining the primal and dual sketching technique, we proposed a novel iterative primal-dual sketching approach which substantially reduced the computational cost in solving sketched sub-problems.

The proposed approach can be extended to solving more general problems, for example, by sketching the Newton step in second-order optimization methods, as done in the “Newton Sketch” paper (Pilanci and Wainwright, 2015a), we will be able to solve regularized risk minimization problem 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, e.g. principle component analysis. Last by not least, it will be interesting to apply iterative sketching techniques for distributed optimization and learning problems (Heinze et al., 2014, 2015).
Figure 3: Comparison of IPDS and Acc-IPDS versus with IHS and DRP various simulated datasets.
Figure 4: Comparison of various iterative sketching approaches on real-world datasets. Top row: Acc-IHS versus IHS, middle row: Acc-DRP versus DRP, bottom row: Acc-IPDS versus Acc-IHS and Acc-DRP.
A Appendix

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

A.1 Proof of Theorem 5.2

Proof. Based on the optimality condition for $w^*$ and $\hat{w}_{HS}$:

$$\left( \frac{X^TX}{n} + \lambda I_p \right) w^* = \frac{X^Ty}{n}.$$  \hfill (A.1)

$$\left( \frac{X^T\Pi \Pi^T X}{n} + \lambda I_p \right) \hat{w}_{HS} = \frac{X^Ty}{n}.$$  \hfill (A.2)

Thus we have

$$\left( \frac{X^TX}{n} + \lambda I_p \right) w^* - \left( \frac{X^T\Pi \Pi^T X}{n} + \lambda I_p \right) \hat{w}_{HS} = 0.$$

So

$$\left\langle \left( \frac{X^TX}{n} + \lambda I_p \right) w^* - \left( \frac{X^T\Pi \Pi^T X}{n} + \lambda I_p \right) \hat{w}_{HS}, w^* - \hat{w}_{HS} \right\rangle = 0.$$  \hfill (A.1)

By adding and subtracting $\left\langle w^* - \hat{w}_{HS}, \left( \frac{X^T\Pi \Pi^T X}{n} + \lambda I_p \right) w^* \right\rangle$ we have

$$\left\langle \left( \frac{X^T\Pi \Pi^T X}{n} - \frac{X^TX}{n} \right) w^*, \hat{w}_{HS} - w^* \right\rangle = (w^* - \hat{w}_{HS})^\top \left( \frac{X^T\Pi \Pi^T X}{n} + \lambda I_p \right) (w^* - \hat{w}_{HS})$$

For the term on right hand side, we have

$$(w^* - \hat{w}_{HS})^\top \left( \frac{X^T\Pi \Pi^T X}{n} + \lambda I_p \right) (w^* - \hat{w}_{HS}) \geq \rho_1(X_{\mathbb{R}^p,\Pi}) \| w^* - \hat{w}_{HS} \|^2_X.$$  \hfill (A.1)

For the term on the left hand side, we have

$$\left\langle \left( \frac{X^T\Pi \Pi^T X}{n} - \frac{X^TX}{n} \right) w^*, \hat{w}_{HS} - w^* \right\rangle \leq \rho_2(X_{\mathbb{R}^p,\Pi}, w^*) \| w^* \|_X \| \hat{w}_{HS} - w^* \|_X.$$  \hfill (A.2)

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

$$\| \hat{w}_{HS} - w^* \|_X \leq \frac{\rho_2(X_{\mathbb{R}^p,\Pi}, w^*)}{\rho_1(X_{\mathbb{R}^p,\Pi})} \| w^* \|_X$$

For the recovery of dual variables, we have

$$\| \hat{\alpha}_{HS} - \alpha^* \|_2 = \| y - X \hat{w}_{HS} - (y - Xw^*) \|_2 = \sqrt{n} \| \hat{w}_{HS} - w^* \|_X \leq \sqrt{n} \frac{\rho_2(X_{\mathbb{R}^p,\Pi}, w^*)}{\rho_1(X_{\mathbb{R}^p,\Pi})} \| w^* \|_X.$$  \hfill (31)
For dual random projection, the proof is mostly analogous to the proof for Hessian sketch. Based on the optimality condition for $\alpha^*$ and $\tilde{\alpha}_{\text{DRP}}$

\[
\begin{align*}
\left(XX^\top + \lambda I_n\right) \alpha^* &= \lambda y, \\
\left(\frac{XRR^\top X^\top}{n} + \lambda I_n\right) \tilde{\alpha}_{\text{DRP}} &= \lambda y.
\end{align*}
\]

Thus we have

\[
\left(XX^\top + \lambda I_n\right) \alpha^* - \left(\frac{XRR^\top X^\top}{n} + \lambda I_n\right) \tilde{\alpha}_{\text{DRP}} = 0.
\]

So

\[
\left(XX^\top + \lambda I_n\right) \alpha^* - \left(\frac{XRR^\top X^\top}{n} + \lambda I_n\right) \tilde{\alpha}_{\text{DRP}}, \alpha^* - \tilde{\alpha}_{\text{DRP}} = 0.
\]

By some algebraic manipulations we have

\[
\left(XX^\top + \lambda I_n\right) \alpha^* - \left(\frac{XRR^\top X^\top}{n} + \lambda I_n\right) \tilde{\alpha}_{\text{DRP}}, \alpha^* - \tilde{\alpha}_{\text{DRP}} = 0.
\]

For the term on right hand side, we have

\[
\left(XX^\top + \lambda I_n\right) \alpha^* - \left(\frac{XRR^\top X^\top}{n} + \lambda I_n\right) \tilde{\alpha}_{\text{DRP}}, \alpha^* - \tilde{\alpha}_{\text{DRP}} = \rho_1(X^\top \mathbb{R}^n, R) \|\alpha^* - \tilde{\alpha}_{\text{DRP}}\|^2_{X^\top}.
\]

(A.3)

For the term on the left hand side, we have

\[
\left(XX^\top + \lambda I_n\right) \alpha^* - \left(\frac{XRR^\top X^\top}{n} + \lambda I_n\right) \tilde{\alpha}_{\text{DRP}}, \alpha^* - \tilde{\alpha}_{\text{DRP}} = \rho_2(X^\top \mathbb{R}^n, R) \|\alpha^* - \tilde{\alpha}_{\text{DRP}}\|^2_{X^\top}.
\]

(A.4)

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

\[
\|\tilde{\alpha}_{\text{DRP}} - \alpha^*\|_{X^\top} \leq \frac{\rho_2(X^\top \mathbb{R}^n, R, \alpha^*)}{\rho_1(X^\top \mathbb{R}^n, R)} \|\alpha^*\|_{X^\top}.
\]

For the recovery of primal variables, we have

\[
\|\tilde{w}_{\text{DRP}} - w^*\|_2 = \frac{1}{\lambda \sqrt{n}} \|\tilde{\alpha}_{\text{DRP}} - \alpha^*\|_{X^\top} \leq \frac{1}{\lambda \sqrt{n}} \frac{\rho_2(X^\top \mathbb{R}^n, R, \alpha^*)}{\rho_1(X^\top \mathbb{R}^n, R)} \|\alpha^*\|_{X^\top} = \frac{\rho_2(X^\top \mathbb{R}^n, R, \alpha^*)}{\rho_1(X^\top \mathbb{R}^n, R)} \|w^*\|_2.
\]

Then applying Lemma 5.1 we conclude the proof.

\[\square\]
A.2 Proof of Theorem 5.3

Proof. We only prove the results for Hessian sketch here as the proof for dual random projection is analogous and similar results already appeared in (Zhang et al., 2014). We will make usage of the following concentration results of Gaussian random matrix:

Lemma A.1. (Lemma 3 in (Zhang et al., 2014)) Let \( B \in \mathbb{R}^{r \times m} \) be a random matrix with entries sampled i.i.d. from zero-mean Gaussian distribution with variance \( \frac{1}{m} \), then with probability at least \( 1 - \delta \), the following inequality holds:

\[
\|BB^\top - I_r\|_2 \leq 2 \sqrt{\frac{2(r + 1)}{m} \log \frac{2r \log 2}{\delta}} := \epsilon_1.
\]

Lemma A.2. (Theorem 3.2 in (Recht, 2011)) Let \( B \in \mathbb{R}^{r \times m}, A \in \mathbb{R}^{(n-r) \times m} \) be random matrices with entries sampled i.i.d. from zero-mean Gaussian distribution with variance \( \frac{1}{m} \), then with probability at least \( 1 - \delta \), the following inequality holds:

\[
\|AB^\top\|_2 \leq \frac{7}{3} \sqrt{\frac{2(n-r)}{m} \log \frac{n}{\delta}} := \tau_1.
\]

We are ready to prove, let \( \Delta w = w^* - \hat{w}_{HS} \), it is easy to see

\[
\|\Delta w\|^2_X = \left\| \left( \frac{\Delta w}{\sqrt{n}} \right) \right\|^2_2 = \left\| \left( \frac{\Sigma_r V^\top + U \Sigma_r V^\top}{{\sqrt{n}}} \right) \Delta w \right\|^2_2 = \left\| \frac{\Sigma_r V^\top \Delta w}{\sqrt{n}} \right\|^2_2 + \left\| \frac{\Sigma_p V^\top \Delta w}{\sqrt{n}} \right\|^2_2.
\]

Consider the term \( \Delta w^\top \left( \frac{X^\top \Pi \Pi^\top X}{n} + \lambda I_p \right) \Delta w \), we have

\[
\Delta w^\top \left( \frac{X^\top \Pi \Pi^\top X}{n} + \lambda I_p \right) \Delta w \geq \Delta w^\top \left( \frac{V^\top \Sigma_r U^\top \Pi \Pi^\top U \Sigma_r V^\top}{n} \right) \Delta w + \lambda \|\Delta w\|^2_2
\]

\[
+ 2 \Delta w^\top \left( \frac{V^\top \Sigma_r U^\top \Pi \Pi^\top U \Sigma_r V^\top}{n} \right) \Delta w.
\]

Since

\[
\Delta w^\top \left( \frac{V^\top \Sigma_r U^\top \Pi \Pi^\top U \Sigma_r V^\top}{n} \right) \Delta w \geq (1 - \epsilon_1) \left\| \frac{\Sigma_p V^\top \Delta w}{\sqrt{n}} \right\|^2_2,
\]

and

\[
\lambda \|\Delta w\|^2_2 \geq \frac{\lambda}{\sigma_{r+1}^2} \left\| \frac{\Sigma_p V^\top \Delta w}{\sqrt{n}} \right\|^2_2 = \frac{\lambda n}{\sigma_{r+1}^2} \left\| \frac{\Sigma_p V^\top \Delta w}{\sqrt{n}} \right\|^2_2,
\]

where

\[
2 \Delta w^\top \left( \frac{V^\top \Sigma_r U^\top \Pi \Pi^\top U \Sigma_r V^\top}{n} \right) \Delta w = 2 \Delta w^\top \left( \frac{V^\top \Sigma_r U^\top \Pi \Pi^\top U \Sigma_r V^\top}{n} \right) \Delta w
\]

\[
\geq - \tau_1 \left\| \frac{\Sigma_r V^\top \Delta w}{\sqrt{n}} \right\|^2_2 - \left\| \frac{\Sigma_p V^\top \Delta w}{\sqrt{n}} \right\|^2_2.
\]

33
Combining above we have
\[
\Delta w^T \left( \frac{X^T \Pi \Pi X}{n} + \lambda I_p \right) \Delta w \geq (1 - \epsilon_1) \left\| \frac{\Sigma_r V^T \Delta w}{\sqrt{n}} \right\|_2^2 + \frac{\lambda n}{\sigma_{r+1}^2} \left\| \frac{\Sigma_r V^T \Delta w}{\sqrt{n}} \right\|_2^2
\]
\[
- 2\tau_1 \left\| \frac{\Sigma_r V^T \Delta w}{\sqrt{n}} \right\|_2 \left\| \frac{\Sigma_r V^T \Delta w}{\sqrt{n}} \right\|_2
\]
\[
\geq \left( \frac{1}{2} - \frac{\epsilon_1}{2} \right) \left\| \frac{\Sigma_r V^T \Delta w}{\sqrt{n}} \right\|_2^2 + \frac{\lambda n}{2\sigma_{r+1}^2} \left\| \frac{\Sigma_r V^T \Delta w}{\sqrt{n}} \right\|_2^2
\]

Consider the term \( \left\langle (\Pi \Pi^T - I_n) \frac{Xw^*}{\sqrt{n}}, \frac{X\Delta w}{\sqrt{n}} \right\rangle \), we have
\[
\left\langle (\Pi \Pi^T - I_n) \frac{Xw^*}{\sqrt{n}}, \frac{X\Delta w}{\sqrt{n}} \right\rangle - \left\langle (\Pi \Pi^T - I_n) \frac{Xw^*}{\sqrt{n}}, \frac{X\Delta w}{\sqrt{n}} \right\rangle
= \left\langle (\Pi \Pi^T - I_n) \frac{Xw^*}{\sqrt{n}}, \frac{X\Delta w}{\sqrt{n}} \right\rangle
\]
\[
+ \left\langle (\Pi \Pi^T - I_n) \frac{Xw^*}{\sqrt{n}}, \frac{X\Delta w}{\sqrt{n}} \right\rangle
\]
\[
+ \left\langle (\Pi \Pi^T - I_n) \frac{Xw^*}{\sqrt{n}}, \frac{X\Delta w}{\sqrt{n}} \right\rangle
\]
\[
+ \left\langle (\Pi \Pi^T - I_n) \frac{Xw^*}{\sqrt{n}}, \frac{X\Delta w}{\sqrt{n}} \right\rangle
\]

Notice that the random matrix \( \Pi^T U_r \) and \( \Pi^T U_r \) can be treated as two Gaussian random matrices with entries sampled i.i.d from \( \mathcal{N}(0, 1/m) \), applying Lemma A.1 and A.2 we can bound above terms separately:
\[
\left\langle (\Pi \Pi^T - I_n) \frac{Xw^*}{\sqrt{n}}, \frac{X\Delta w}{\sqrt{n}} \right\rangle \leq \epsilon_1 \left\| \frac{\Sigma_r V^T w^*}{\sqrt{n}} \right\|_2 \left\| \frac{\Sigma_r V^T \Delta w}{\sqrt{n}} \right\|_2,
\]
\[
\left\langle (\Pi \Pi^T - I_n) \frac{Xw^*}{\sqrt{n}}, \frac{X\Delta w}{\sqrt{n}} \right\rangle \leq \tau_1 \left\| \frac{\Sigma_r V^T w^*}{\sqrt{n}} \right\|_2 \left\| \frac{\Sigma_r V^T \Delta w}{\sqrt{n}} \right\|_2,
\]
\[
\left\langle (\Pi \Pi^T - I_n) \frac{Xw^*}{\sqrt{n}}, \frac{X\Delta w}{\sqrt{n}} \right\rangle \leq \tau_1 \left\| \frac{\Sigma_r V^T \Delta w}{\sqrt{n}} \right\|_2 \left\| \frac{\Sigma_r V^T \Delta w}{\sqrt{n}} \right\|_2,
\]
\[
\left\langle (\Pi \Pi^T - I_n) \frac{Xw^*}{\sqrt{n}}, \frac{X\Delta w}{\sqrt{n}} \right\rangle \leq v_1 \left\| \frac{\Sigma_r V^T \Delta w}{\sqrt{n}} \right\|_2 \left\| \frac{\Sigma_r V^T \Delta w}{\sqrt{n}} \right\|_2.
\]
By Cauchy-Schwarz inequality, we have

\[
\left\langle (\Pi \Pi^\top - I_n) \frac{X w^*}{\sqrt{n}}, \frac{X}{\sqrt{n}} (\hat{w}_{HS} - w^*) \right\rangle \leq \epsilon_1 \left\| \frac{\Sigma_r V^\top w^*}{\sqrt{n}} \right\|_2^2 + \left\| \frac{\Sigma_r V^\top \Delta w}{\sqrt{n}} \right\|_2^2 + \tau_1 \left\| \frac{\Sigma_r V^\top w^*}{\sqrt{n}} \right\|_2^2 + \left\| \frac{\Sigma_r V^\top \Delta w}{\sqrt{n}} \right\|_2^2
\]

\[
\leq \frac{4 \epsilon_1^2}{1 - \epsilon_1} \left\| \frac{\Sigma_r V^\top w^*}{\sqrt{n}} \right\|_2^2 + \frac{1 - \epsilon_1}{8} \left\| \frac{\Sigma_r V^\top \Delta w}{\sqrt{n}} \right\|_2^2
\]

\[
+ \frac{4 \tau_1^2}{1 - \epsilon_1} \left\| \frac{\Sigma_r V^\top w^*}{\sqrt{n}} \right\|_2^2 + \frac{\lambda n}{8 \sigma^2_{r+1}} \left\| \frac{\Sigma_r V^\top \Delta w}{\sqrt{n}} \right\|_2^2
\]

\[
+ \frac{4 \tau_1^2 \sigma^2_{r+1}}{\lambda n} \left\| \frac{\Sigma_r V^\top w^*}{\sqrt{n}} \right\|_2^2 + \frac{4 \rho^2 \sigma^2_{r+1}}{\lambda n} \left\| \frac{\Sigma_r V^\top \Delta w}{\sqrt{n}} \right\|_2^2
\]

From the proof of 5.2 we know

\[
\frac{1 - \epsilon_1}{2} \left\| \frac{\Sigma_r V^\top \Delta w}{\sqrt{n}} \right\|_2^2 + \frac{\lambda n}{2 \sigma^2_{r+1}} \left\| \frac{\Sigma_r V^\top \Delta w}{\sqrt{n}} \right\|_2^2 \leq \Delta w^\top \left( \frac{X^\top \Pi \Pi^\top X}{n} + \lambda I \right) \Delta w
\]

\[
= \left\langle (\Pi \Pi^\top - I_n) \frac{X w^*}{\sqrt{n}}, \frac{X}{\sqrt{n}} (\hat{w}_{HS} - w^*) \right\rangle
\]

Combining above we have

\[
\frac{1 - \epsilon_1}{4} \left\| \frac{\Sigma_r V^\top \Delta w}{\sqrt{n}} \right\|_2^2 + \frac{\lambda n}{4 \sigma^2_{r+1}} \left\| \frac{\Sigma_r V^\top \Delta w}{\sqrt{n}} \right\|_2^2 \leq \left( \frac{4 \epsilon_1^2}{1 - \epsilon_1} + \frac{4 \tau_1^2 \sigma^2_{r+1}}{\lambda n} \right) \left\| \frac{\Sigma_r V^\top w^*}{\sqrt{n}} \right\|_2^2
\]

\[
+ \left( \frac{4 \tau_1^2}{1 - \epsilon_1} + \frac{4 \rho^2 \sigma^2_{r+1}}{\lambda n} \right) \left\| \frac{\Sigma_r V^\top \Delta w}{\sqrt{n}} \right\|_2^2
\]

\[
\leq \left( \frac{4 \epsilon_1^2}{1 - \epsilon_1} + \frac{4 \tau_1^2 \sigma^2_{r+1}}{\lambda n} + \frac{4 \rho^2 \sigma^2_{r+1}}{1 - \epsilon_1} + \frac{4 \rho^2 \nu^2 \sigma^2_{r+1}}{\lambda n} \right) \left\| w^* \right\|_X^2
\]

Thus

\[
\left\| \Delta w \right\|_X^2 = \left\| \frac{\Sigma_r V^\top \Delta w}{\sqrt{n}} \right\|_2^2 + \left\| \frac{\Sigma_r V^\top \Delta w}{\sqrt{n}} \right\|_2^2
\]

\[
\leq \left( \frac{4}{1 - \epsilon_1} + \frac{4 \sigma^2_{r+1}}{\lambda n} \right) \left( \frac{1 - \epsilon_1}{4} \left\| \frac{\Sigma_r V^\top \Delta w}{\sqrt{n}} \right\|_2^2 + \lambda n \left\| \frac{\Sigma_r V^\top \Delta w}{\sqrt{n}} \right\|_2^2 \right)
\]

\[
\leq \left( \frac{4}{1 - \epsilon_1} + \frac{4 \sigma^2_{r+1}}{\lambda n} \right) \left( \frac{4 \epsilon_1^2}{1 - \epsilon_1} + \frac{4 \tau_1^2 \sigma^2_{r+1}}{\lambda n} + \frac{4 \tau_1^2 \rho^2}{1 - \epsilon_1} + \frac{4 \rho^2 \nu^2 \sigma^2_{r+1}}{\lambda n} \right) \left\| w^* \right\|_X^2
\]

which concludes the proof.
A.3 Proof of Theorem 5.6

Proof. For notation simplicity let

\[ \tilde{H} = \frac{X^\top \Pi \Pi^\top X}{n} + \lambda \mathbf{I}_p \quad \text{and} \quad H = \frac{X^\top X}{n} + \lambda \mathbf{I}_p, \]

based on the property of similarity matrices, we have

\[ \kappa(\tilde{H}^{-1}H) = \kappa(\tilde{H}^{-1/2} \tilde{H} \tilde{H}^{-1/2}) = \frac{\max_w w^\top \tilde{H}^{-1/2} \tilde{H} \tilde{H}^{-1/2} \tilde{w}}{\min_w w^\top \tilde{H}^{-1/2} \tilde{H} \tilde{H}^{-1/2} \tilde{w}}. \]

Consider the quantity \( |w^\top \tilde{H}^{-1/2} (H - \tilde{H}) \tilde{H}^{-1/2} \tilde{w}| \), since

\[
|w^\top \tilde{H}^{-1/2} (H - \tilde{H}) \tilde{H}^{-1/2} \tilde{w}| = \left\langle (H - \tilde{H}) \tilde{H}^{-1/2} \tilde{w}, \tilde{H}^{-1/2} \tilde{w} \right\rangle \\
= \left\langle \left( \Pi \Pi^\top - I_n \right) \frac{X}{\sqrt{n}} \tilde{H}^{-1/2} \tilde{w}, \frac{X}{\sqrt{n}} \tilde{H}^{-1/2} \tilde{w} \right\rangle \\
\leq \rho_2 \left( X_{\mathbb{R}^p}, \Pi, \frac{X}{\sqrt{n}} \tilde{H}^{-1/2} \tilde{w} \right) \left\| \tilde{H}^{-1/2} \tilde{w} \right\|_2 \\
\leq C_0 \sqrt{\frac{\mathbb{W}^2(X_{\mathbb{R}^p})}{m} \log \left( \frac{1}{\delta} \right)} \left\| \tilde{H}^{-1/2} \tilde{w} \right\|_2.
\]

Since for any vector \( u \in \mathbb{R}^p \), we have

\[
\left\| \tilde{H}^{1/2} u \right\|_2^2 = u^\top \left( \frac{X^\top \Pi \Pi^\top X}{n} + \lambda \mathbf{I}_p \right) u \\
= u^\top \left( \frac{X^\top \Pi \Pi^\top X}{n} \right) u + \lambda \left\| u \right\|_2^2 \\
\geq \rho_1 (X_{\mathbb{R}^p}, \Pi) \left\| u \right\|_2^2 \\
\geq \left( 1 - C_0 \sqrt{\frac{\mathbb{W}^2(X_{\mathbb{R}^p})}{m} \log \left( \frac{1}{\delta} \right)} \right) \left\| u \right\|_2^2.
\]

Let \( u = \tilde{H}^{-1/2} \tilde{w} \) we have

\[
\left\| \tilde{H}^{-1/2} \tilde{w} \right\|_2^2 \leq \frac{1}{1 - C_0 \sqrt{\frac{\mathbb{W}^2(X_{\mathbb{R}^p})}{m} \log \left( \frac{1}{\delta} \right)}} \left\| \tilde{w} \right\|_2^2.
\]

Combining above we get

\[
|w^\top \tilde{H}^{-1/2} (H - \tilde{H}) \tilde{H}^{-1/2} \tilde{w}| \leq C_0 \sqrt{\frac{\mathbb{W}^2(X_{\mathbb{R}^p})}{m} \log \left( \frac{1}{\delta} \right)} \left\| \tilde{w} \right\|_2, 
\]

which implies

\[
\max_w w^\top \tilde{H}^{-1/2} \tilde{H} \tilde{H}^{-1/2} \tilde{w} \leq \left\| \tilde{w} \right\|_2^2 + C_0 \sqrt{\frac{\mathbb{W}^2(X_{\mathbb{R}^p})}{m} \log \left( \frac{1}{\delta} \right)} \left\| \tilde{w} \right\|_2 \\
= \frac{1}{1 - C_0 \sqrt{\frac{\mathbb{W}^2(X_{\mathbb{R}^p})}{m} \log \left( \frac{1}{\delta} \right)}} \left\| \tilde{w} \right\|_2^2.
\]

36
and
\[
\min_w w^\top \tilde{H}^{-1/2} H \tilde{H}^{-1/2} w \geq \|w\|_2^2 - \frac{C_0 \sqrt{\frac{\|W^2(X,R)^p\|}{m}} \log \left( \frac{1}{\delta} \right)}{1 - C_0 \sqrt{\frac{\|W^2(X,R)^p\|}{m}} \log \left( \frac{1}{\delta} \right)} \|w\|_2^2
\]
\[
= \frac{1 - 2C_0 \sqrt{\frac{\|W^2(X,R)^p\|}{m}} \log \left( \frac{1}{\delta} \right)}{1 - C_0 \sqrt{\frac{\|W^2(X,R)^p\|}{m}} \log \left( \frac{1}{\delta} \right)} \|w\|_2^2.
\]
Thus we know
\[
\kappa(\tilde{H}^{-1} H) \leq \frac{1}{1 - 2C_0 \sqrt{\frac{\|W^2(X,R)^p\|}{m}} \log \left( \frac{1}{\delta} \right)}
\]
and proof for \( \kappa_{\text{DRP}}(X, R, \lambda) \) is analogous.

\[\square\]

**A.4 Proof of Lemma 5.8**

Proof. Note that (2.6) is sketching the following problem:

\[
\arg\min_u u^\top \left( \frac{X^\top X}{2n} + \frac{\lambda}{2} I_p \right) u - \left\langle \frac{X^\top (y - \tilde{w}_\text{HS}^{(t)})}{n} - \lambda \tilde{w}_H^{(t)}, u \right\rangle,
\]

where \( w^* - \tilde{w}_\text{HS}^{(t)} \) is the optimal solution, thus applying Theorem 5.2, we have

\[
\|\tilde{u}^{(t)} - (w^* - \tilde{w}_\text{HS}^{(t)})\|_X \leq \frac{C_0 \sqrt{\frac{\|W^2(X,R)^p\|}{m}} \log \left( \frac{1}{\delta} \right)}{1 - C_0 \sqrt{\frac{\|W^2(X,R)^p\|}{m}} \log \left( \frac{1}{\delta} \right)} \|\tilde{w}_\text{HS}^{(t)} - w^*\|_X,
\]

using the definition that \( \tilde{w}_\text{HS}^{(t+1)} = \tilde{w}_\text{HS}^{(t)} + \tilde{u}^{(t)} \), we obtain the desired result.

\[\square\]

**A.5 Proof of Theorem 5.9**

Proof. By triangle inequality we have the following decomposition:

\[
\|\tilde{w}_\text{HS}^{(t+1)} - w^*\|_X \leq \|\tilde{w}_\text{HS}^{(t+1)} - \tilde{w}_\text{HS}^{(t+1)}\|_X + \|\tilde{w}_\text{HS}^{(t+1)} - \tilde{w}_\text{HS}^{(t+1)}\|_X
\]
\[
\leq \frac{C_0 \sqrt{\frac{\|W^2(X,R)^p\|}{m}} \log \left( \frac{1}{\delta} \right)}{1 - C_0 \sqrt{\frac{\|W^2(X,R)^p\|}{m}} \log \left( \frac{1}{\delta} \right)} \|\tilde{w}_\text{HS}^{(t)} - w^*\|_X + \|\tilde{w}_\text{HS}^{(t+1)} - \tilde{w}_\text{HS}^{(t+1)}\|_X
\]
\[
\leq \left( \frac{C_0 \sqrt{\frac{\|W^2(X,R)^p\|}{m}} \log \left( \frac{1}{\delta} \right)}{1 - C_0 \sqrt{\frac{\|W^2(X,R)^p\|}{m}} \log \left( \frac{1}{\delta} \right)} \right)^t \|w^*\|_X + \|\tilde{w}_\text{HS}^{(t+1)} - \tilde{w}_\text{HS}^{(t+1)}\|_X
\]

For the term \( \|\tilde{w}_\text{HS}^{(t+1)} - \tilde{w}_\text{HS}^{(t+1)}\|_X \), we can further bridge \( \tilde{w}_\text{HS}^{(t+1)} \) and \( \tilde{w}_\text{HS}^{(t+1)} \) by \( w^{(t+1)} \), which is the result of one exact step of IHS initialized at \( \tilde{w}_\text{HS}^{(t)} \). Thus we have the following decomposition

\[
\|\tilde{w}_\text{HS}^{(t+1)} - \tilde{w}_\text{HS}^{(t+1)}\|_X \leq \|\tilde{w}_\text{HS}^{(t+1)} - \tilde{w}_\text{HS}^{(t+1)}\|_X + \|\tilde{w}_\text{HS}^{(t+1)} - \tilde{w}_\text{HS}^{(t+1)}\|_X
\]

37
Applying the Theorem 5.2 for DRP we have the following bound for \( \| \tilde{w}_{(t+1)} - \tilde{w}_{(t+1)} \|_X \):

\[
\| \tilde{w}_{(t+1)} - \tilde{w}_{(t+1)} \|_X \leq \lambda_{\max} \left( \frac{X^T X}{n} \right) \| \tilde{w}_{(t+1)} - \tilde{w}_{(t+1)} \|_2 \\
\leq \lambda_{\max} \left( \frac{X^T X}{n} \right) \left( \frac{C_0 \sqrt{\frac{W^2(X^T X)}{d}} \log \left( \frac{\lambda}{\delta} \right)}{1 - C_0 \sqrt{\frac{W^2(X^T X)}{d}} \log \left( \frac{\lambda}{\delta} \right)} \right)^k \| \tilde{w}_{(t+1)} - \tilde{w}_{(t+1)} \|_2 \\
\leq 2\lambda_{\max} \left( \frac{X^T X}{n} \right) \left( \frac{C_0 \sqrt{\frac{W^2(X^T X)}{d}} \log \left( \frac{\lambda}{\delta} \right)}{1 - C_0 \sqrt{\frac{W^2(X^T X)}{d}} \log \left( \frac{\lambda}{\delta} \right)} \right)^k \| \tilde{w}^* \|_2
\]

Also, we can relate the error \( \| \tilde{w}_{(t+1)} - \tilde{w}_{(t+1)} \|_X \) to the error term at \( t \)-th outer loop iteration:

\[
\| \tilde{w}_{(t)} - \tilde{w}_{(t)} \|_X \\
= \| \tilde{w}_{(t)} - \tilde{H}^{-1} \nabla P(\tilde{w}_{(t)}) - \tilde{w}_{(t)} \|_X \\
= \| \tilde{H}^{-1} (\tilde{H} - \tilde{H}) (\tilde{w}_{(t)} - \tilde{w}_{(t)}) \|_X \\
\leq \lambda_{\max} \left( \frac{X^T X}{n} \right) \| \tilde{w}_{(t)} - \tilde{w}_{(t)} \|_X \\
\leq 4\lambda_{\max} \left( \frac{X^T X}{n} \right) \left( \frac{C_0 \sqrt{\frac{W^2(X^T X)}{d}} \log \left( \frac{\lambda}{\delta} \right)}{1 - C_0 \sqrt{\frac{W^2(X^T X)}{d}} \log \left( \frac{\lambda}{\delta} \right)} \right)^k \| \tilde{w}^* \|_2
\]

Combining above inequalities we obtained the following iterative error bound for \( \tilde{w}_{(t+1)} \):

\[
\| \tilde{w}_{(t+1)} - \tilde{w}^* \|_X \leq \left( \frac{C_0 \sqrt{\frac{W^2(X^T X)}{d}} \log \left( \frac{\lambda}{\delta} \right)}{1 - C_0 \sqrt{\frac{W^2(X^T X)}{d}} \log \left( \frac{\lambda}{\delta} \right)} \right)^t \| \tilde{w}^* \|_X \\
+ 10\lambda_{\max} \left( \frac{X^T X}{n} \right) \left( \frac{C_0 \sqrt{\frac{W^2(X^T X)}{d}} \log \left( \frac{\lambda}{\delta} \right)}{1 - C_0 \sqrt{\frac{W^2(X^T X)}{d}} \log \left( \frac{\lambda}{\delta} \right)} \right)^k \| \tilde{w}^* \|_2
\]
References

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

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

C. Boutsidis and A. Gittens. Improved matrix algorithms via the subsampled randomized hadamard transform. *SIAM Journal on Matrix Analysis and Applications*, 34(3):1301–1340, 2013.

S. P. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, 2004.

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

P. S. Dhillon, D. P. Foster, S. M. Kakade, and L. H. Ungar. A risk comparison of ordinary least squares vs ridge regression. *Journal of Machine Learning Research*, 14(1):1505–1511, 2013.

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

P. Drineas, M. W. Mahoney, S. Muthukrishnan, and T. Sarlós. Faster least squares approximation. *Numerische Mathematik*, 117(2):219–249, 2011.

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

N. Halko, P.-G. Martinsson, and J. A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011.

C. Heinze, B. McWilliams, N. Meinshausen, and G. Krummenacher. Loco: Distributing ridge regression with random projections. *arXiv preprint arXiv:1406.3469*, 2014.

C. Heinze, B. McWilliams, and N. Meinshausen. Dual-loco: Distributing statistical estimation using random projections. *arXiv preprint arXiv:1506.02554*, 2015.

M. R. Hestenes and E. Stiefel. *Methods of conjugate gradients for solving linear systems*, volume 49. 1952.

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

G. Lan and Y. Zhou. An optimal randomized incremental gradient method. *arXiv preprint arXiv:1507.02000*, 2015.
H. Lin, J. Mairal, and Z. Harchaoui. A universal catalyst for first-order optimization. In *Advances in Neural Information Processing Systems*, pages 3366–3374, 2015.

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

D. G. Luenberger. *Introduction to linear and nonlinear programming*, volume 28.

M. W. Mahoney. Randomized algorithms for matrices and data. *Foundations and Trends® in Machine Learning*, 3(2):123–224, 2011.

S. Mendelson, A. Pajor, and N. Tomczak-Jaegermann. Reconstruction and subgaussian operators in asymptotic geometric analysis. *Geometric and Functional Analysis*, 17(4):1248–1282, 2007.

Y. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012.

Y. Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2013.

J. Nocedal and S. Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

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

S. Oymak, B. Recht, and M. Soltanolkotabi. Isometric sketching of any set via the restricted isometry property. *arXiv preprint arXiv:1506.03521*, 2015.

M. Pilanci and M. J. Wainwright. Newton sketch: A linear-time optimization algorithm with linear-quadratic convergence. *arXiv preprint arXiv:1505.02250*, 2015a.

M. Pilanci and M. J. Wainwright. Randomized sketches of convex programs with sharp guarantees. *Information Theory, IEEE Transactions on*, 61(9):5096–5115, 2015b.

M. Pilanci and M. J. Wainwright. Iterative hessian sketch: Fast and accurate solution approximation for constrained least-squares. *Journal of Machine Learning Research*, 2016.

G. Raskutti and M. Mahoney. A statistical perspective on randomized sketching for ordinary least-squares. *stat*, 1050:25, 2015.

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

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

N. L. Roux, M. Schmidt, and F. R. Bach. A stochastic gradient method with an exponential convergence rate for finite training sets. In *Advances in Neural Information Processing Systems*, pages 2663–2671, 2012.
T. Sarlos. Improved approximation algorithms for large matrices via random projections. In *Foundations of Computer Science, 2006. FOCS’06. 47th Annual IEEE Symposium on*, pages 143–152. IEEE, 2006.

S. Shalev-Shwartz and T. Zhang. Stochastic dual coordinate ascent methods for regularized loss. *The Journal of Machine Learning Research*, 14(1):567–599, 2013.

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

R. Vershynin. Estimation in high dimensions: a geometric perspective. In *Sampling Theory, a Renaissance*, pages 3–66. Springer, 2015.

D. P. Woodruff. Sketching as a tool for numerical linear algebra. *arXiv preprint arXiv:1411.4357*, 2014.

T. Yang, L. Zhang, R. Jin, and S. Zhu. Theory of dual-sparse regularized randomized reduction. In *Proceedings of The 32nd International Conference on Machine Learning*, pages 305–314, 2015.

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

A. W. Yu, Q. Lin, and T. Yang. Doubly stochastic primal-dual coordinate method for regularized empirical risk minimization with factorized data. *arXiv preprint arXiv:1508.03390*, 2015.

L. Zhang, M. Mahdavi, R. Jin, T. Yang, and S. Zhu. Recovering the optimal solution by dual random projection. In *Conference on Learning Theory*, pages 135–157, 2013.

L. Zhang, M. Mahdavi, R. Jin, T. Yang, and S. Zhu. Random projections for classification: A recovery approach. *Information Theory, IEEE Transactions on*, 60(11):7300–7316, 2014.

Y. Zhang and X. Lin. Stochastic primal-dual coordinate method for regularized empirical risk minimization. In *Proceedings of The 32nd International Conference on Machine Learning*, pages 353–361, 2015.