ITERATIVE HESSIAN SKETCH WITH MOMENTUM

Ibrahim Kurban Ozaslan[†] *Mert Pilanci*[⋆] *Orhan Arikan*[†]

[†]EEE Department, Bilkent University, Ankara, Turkey
[⋆]EE Department, Stanford University, CA, USA
{iozaslan, oarikan}@ee.bilkent.edu.tr, pilanci@stanford.edu

ABSTRACT

We propose a novel randomized linear least squares solver which is an improvement of Iterative Hessian Sketch and randomized preconditioning. In the proposed Momentum-IHS technique (M-IHS), Heavy Ball Method is used to accelerate the convergence of iterations. It is shown that for any full rank data matrix rate of convergence depends on the ratio between the feature size and the sketch size. Unlike the Conjugate Gradient technique, the rate of convergence is unaffected by either the condition number or the eigenvalue spectrum of the data matrix. As demonstrated over many examples, the proposed M-IHS provides compatible performance with the state of the art randomized preconditioning methods such as LSRN or Blendenpik and yet, it provides a completely different perspective in the area of iterative solvers which can pave the way for future developments.

Index Terms— Iterative Hessian Sketch, Momentum, Randomized Preconditioning, Ill Condition, First Order Iterative Solvers

1. INTRODUCTION

Least squares(LS) problem has ever increasing applications in the era of data science. For a given full rank data matrix $A \in \mathbf{R}^{n\times d}$ and a measurement vector $b \in \mathbf{R}^n$, in the least squares setting, solution to the following optimization problem yields $x^{\text{LS}} \in \mathbf{R}^d$:

$$x^{\text{LS}} = \underset{x\in\mathbf{R}^d}{\text{argmin}} \frac{1}{2}\|Ax-b\|_2^2 = (A^TA)^{-1}A^Tb. \tag{1}$$

The case of $n \geq d$ is of central importance in big data applications. Most efficient way of obtaining the x^{LS} is to solve a triangular system obtained through QR decomposition requiring $O(nd^2)$ floating operations, which is prohibitively large in big data applications. The main cause of this high complexity is due to the calculation of the Hessian Matrix (A^TA) in eq. (1) and calculation of the R factor in the QR decomposition of A. One remedy for reducing the required computation is to use the first order iterative techniques which require only matrix-vector calculations at each iteration avoiding order nd^2 computations [**?**]. However, the required number of iterations are highly sensitive to the condition number of matrix A. If the largest singular value of A is known, then the optimal and unimprovable convergence rate of $O(1/k^2)$ belongs to the Nesterov's Accelerated Gradient Descent [**?**]. In addition to the largest singular value, if the smallest singular value is also known, then the optimal rate is achieved by the Polyak's Heavy Ball Method(HBM) [**?**]. Unfortunately, such information on A is rarely available in practice. In the absence of this information, the Conjugate Gradient(CG) technique can be used to have the same convergence rate of the HBM by tuning the required parameters adaptively through additional calculations at each iteration [**?**]. Similarly, Saunder's LSQR [**?**] which utilizes bidiagonalization by using Givens rotations shares the same convergence rate as well. Additionally, if one knows the ellipsoid containing eigenvalues, the Chehbyshev Semi-iterative (CS) technique has a similar convergence rate with a significant advantage over the CG and the LSQR. In the CS, there is no need for inner products which allows parallelization in distributed systems. Although convergence of the CS is slower than those of CG and LSQR, in the distributed systems with high communication cost, where data is stored in clusters, the convergence time of CS can be much significantly less [**?**],[**?**]. Many other techniques can also be added to this list [**?**],[**?**]. The common rate of convergence of these techniques is:

$$\|x^{\text{k}} - x^{\text{LS}}\| \leq \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k \|x^0 - x^{\text{LS}}\|, \tag{2}$$

where κ is the condition number of A^TA, which is defined as the ratio of the largest singular value to the smallest singular value of A^TA [**?**], [**?**]. However, for ill conditioned A matrices with excessively large κ, this rate of convergence becomes extremely slow.

Preconditioning is a linear mapping of the solution domain which aims to transform an ill conditioned problem to a well conditioned problem. In the deterministic setting, finding an appropriate preconditioning matrix has always been a challenging task until the introduction of the Random Projection(RP) techniques [**?**]. To the best of our knowledge, the RP techniques developed in [**?**] is utilized first by Rokhlin to construct a preconditioning matrix for a CG-like iterative solver [**?**]. By using the R factor in QR decomposition of a sketched matrix which is obtained through a series of randomized Givens Rotations, randomized diagonal scaling and

a Fourier Transform. Implementation of a similar idea resulted in Blendenpik which is superior to some deterministic LAPACK solvers [**?**]. Also, LSRN uses RP to construct a preconditioner, but it uses the right singular vectors instead of the R factor in the QR decomposition and utilizes the Chebyshev technique as an iterative solver for parallelization [**?**]. Analysis of these algorithms, especially derivation of the the statistical bounds are only accessible by specialists in the field. In this work we propose a new technique, M-IHS, which has comparable performance with the state-of-the-art techniques while its derivation is highly accessible by a large audience of practitioners.

Besides, instead of using randomization in the preconditioning for the first order solvers, RP can be utilized directly to solve the least squares problem. In naive randomized least squares techniques, both data matrix and measurements are projected to lower dimensions in order to decrease the computational complexity(see [**?**] and references therein). However, Pilanci showed that projection of both A and b is suboptimal and he proposed a novel method, called Iterative Hessian Sketch (IHS), which approximates only the Hessian term in eq. (1) [**?**]. Very recently an accelerated version of IHS is proposed by using CG-like iterations [**?**]. Some efforts, also, has been made to use RP in Nesterov's Accelerated Gradient Method which is known also as FISTA [**?**][**?**].

In the proposed M-IHS technique, IHS and HBM are jointly used to improve the rate of convergence. Our main contribution is to determine the momentum weights through Marchenko-Pastur (MP) Law instead of an adaptive approach as proposed in [**?**] which increases the computational complexity of the iterations. Further, for a sketch size of m, we proved that the convergence rate of the proposed M-IHS is $\sqrt{d/m}$, which is completely independent of the data matrix A. The computational complexity of the proposed M-IHS technique is $O(nd\log(m) + md^2 + (nd + d^2)\log(1/\epsilon))$ where ϵ is the desired accuracy. Furthermore, as presented in the Section 2.2, for the regularized schemes, squared dependencies like md^2 and d^2 can be avoided by using inexact solvers for the subproblem.

2. SKETCH BASED ITERATIVE LS SOLVERS

We are interested in sketch matrices that satisfy $\mathbf{E}[S^TS] = I_d$ where $S \in \mathbf{R}^{m\times d}$. Amongst many, we specifically use sketch matrices that are based on Randomized Orthonormal Systems(ROS) which are constructed as follows:

- Choose an n-dimensional orthonormal transformation matrix $H \in \mathbf{R}^{n\times n}$ such as the Hadamard, Fourier, Hartley or Cosine transformation matrix which can realize matrix-vector products in $n\log(n)$ operations.
- Construct a diagonal matrix $D \in \mathbf{R}^{n\times n}$ whose diagonal elements are i.i.d. Rademacher random variables.
- Row vectors are $\tilde{s}^T = \sqrt{n}e_iHD$ with probability $1/n$, $i = 1,\dots,n$ and $e_i \in \mathbf{R}^n$ is i$^{\text{th}}$canonical basis.

2.1. The Naive IHS Technique

The least squares problem can be formulated by the Newton Method as a combination of the Hessian and the Jacobian term:

$$x^{\text{LS}} = \underset{x\in\mathbf{R}^d}{\text{argmin}}\ \frac{1}{2}\|A(x-x^0)\|_2^2 - < A^T(b-Ax^0), x >, \quad (3)$$

where x^0 is any initial vector. Newton Method converges in exactly one iteration for any x^0[**?**]. In IHS only the Hessian term is approximated and the solution is improved recursively by solving the following minimization problem:

$$\begin{aligned} x^{\text{k+1}} &= \underset{x\in\mathcal{C}}{\text{argmin}}\ \frac{1}{2}\|SA(x-x^{\text{k}})\|_2^2 - < A^T(y-Ax^{\text{k}}), x > \\ &= x^{\text{k}} + (A^TS^TSA)^{-1}A^T(y-Ax^{\text{k}}). \end{aligned}$$

The important point here is that instead of changing the sketch matrix S at each iteration as described in [**?**], we can use a single sketch matrix in all iterations. However, we should use a tunable step size to prevent divergent behaviour stemming from single sketch scheme as follows:.

$$x^{\text{k+1}} = x^{\text{k}} + t_k(A^TS^TSA)^{-1}A^T(y-Ax^{\text{k}}), \quad (4)$$

The convergence rate of IHS can be investigated through the same approach in [**?**], by finding the transformation matrix between the current and the previous error vectors. The l_2-norm of the transformation matrix serves as a lower bound for the convergence rate.

For this purpose, consider the following transformation and recall that $A^T(Ax^{\text{LS}} - y) = 0$:

$$\begin{aligned} \|x^{\text{k+1}} - x^{\text{LS}}\|_2 &= \|x^{\text{k}} + t_k(A^TS^TSA)^{-1}A^T(y-Ax^{\text{k}}) - x^{\text{LS}}\|_2 \\ &= \|(I_d - t_k(A^TS^TSA)^{-1}A^TA)(x^{\text{k}} - x^{\text{LS}})\|_2 \\ &\le \|\underbrace{I_d - t_k(A^TS^TSA)^{-1}A^TA}_{T}\|_2\|x^{\text{k}} - x^{\text{LS}}\|_2 \quad (5) \end{aligned}$$

Therefore, we can write following improvement by using the Gelfand Formula:

$$\begin{aligned} \|x^{\text{k}} - x^{\text{LS}}\|_2 &\le \|T^k\|_2\|x^0 - x^{\text{LS}}\|_2 \\ &\le \left(\rho(T)^k + \epsilon_k\right)\|x^0 - x^{\text{LS}}\|_2, \end{aligned}$$

where $\lim_{k\to\infty}\epsilon_k = 0$ and $\rho(T)$ is the spectral radius of T. Thus, if the spectral radius of T is bounded, then contraction ratio (or the norm of transformation) can be bounded as well. To find $\rho(T)$, the largest and the smallest eigenvalues of matrix $(A^TS^TSA)^{-1}A^TA$ should be determined. Changing basis by using $(A^TA)^{-1/2}$yields $(A^TA)^{1/2}(A^TS^TSA)^{-1}(A^TA)^{1/2}$ which is a symmetric matrix similar to $(A^TS^TSA)^{-1}A^TA$. By using compact SVD of $A = U\Sigma V^T$, where $U \in \mathbf{R}^{n\times d}$, $V, \Sigma \in \mathbf{R}^{d\times d}$, $U^TU = V^TV = VV^T = I_d$ and $\Sigma = \mathbf{diag}(\sigma_1,\dots,\sigma_d)$, $\sigma_1 \ge \dots \ge \sigma_d \ge 0$, we obtain:

$$\begin{aligned} &(A^TA)^{1/2}(A^TS^TSA)^{-1}(A^TA)^{1/2} \\ &\quad = V\Sigma V^T(V\Sigma U^TS^TSU\Sigma V^T)^{-1}V\Sigma V^T \\ &\quad = V(U^TS^TSU)^{-1}V^T. \end{aligned}$$

Note that in the last step, we have used the fact that V is full rank. Since V is a unitary matrix, spectral properties depends only on the eigenvalues of $(U^T S^T SU)^{-1}$. Since the columns of U is an orthonormal set of vectors and entries of S are zero mean, unit variance i.i.d. random variables, the entries of SU have the same probability distribution as the entries of S. Hence, if we generate a sketch matrix $\tilde{S} \in \mathbf{R}^{m\times d}$ with the same techniques used for S, then SU will be statistically equivalent to $\tilde{S}$.

Based on this observation, we need to know the largest and the smallest eigenvalues of a sample covariance matrix of $\tilde{S} \in \mathbf{R}^{m\times d}$ which is called as the Wishart matrix in statistics [**?**]. By the MP Law, the largest and the smallest eigenvalues of Wishart matrices converge to $(1 \pm \sqrt{d/m})^2$, as $m \to \infty$ while the ratio d/m remains the same [**?**], [**?**]). Therefore, the largest and the smallest eigenvalues of $(A^T S^T SA)^{-1} A^T A$ asymptotically converge to $1/(1 \mp \sqrt{d/m})^2$. The spectral radius $\rho(T)$ is:

$$\rho(T) = \max\left\{\left|1 - \frac{t_k}{(1+\sqrt{r})^2}\right|, \left|1 - \frac{t_k}{(1-\sqrt{r})^2}\right|\right\},$$

where $r = d/m$. Here, the following choice for t_k yields the minimum spectral radius

$$t_k = \frac{2\cdot(1+\sqrt{r})^2(1-\sqrt{r})^2}{(1+\sqrt{r})^2 + (1-\sqrt{r})^2} = \frac{(1-r)^2}{1+r}, \tag{6}$$

which remains constant during the iterations:

$$\rho(T) = \left|1 - \frac{(1-r)^2}{1+r} \Big/ \left(1+\sqrt{r}\right)^2\right| = \frac{2\sqrt{r}}{1+r}. \tag{7}$$

In conclusion, the damped IHS converges with the following exponentially decaying upper bound:

$$\|x^{\mathrm{k}} - x^{\mathrm{LS}}\|_2 \le \left(\frac{2\sqrt{r}}{1+r}\right)^{k} \|x^0 - x^{\mathrm{LS}}\|_2. \tag{8}$$

2.2. The Proposed Momentum based M-IHS Technique

Momentum effect in iterations can be realized by taking a step in the direction of a linear combination of the gradients of both the objective function and the solution trajectory:

$$x^{\mathrm{k+1}} = x^{\mathrm{k}} - \alpha_k \nabla f(x^{\mathrm{k}}) + \beta_k (x^{\mathrm{k}} - x^{\mathrm{k\text{-}1}}),$$

where $\alpha_k$and $\beta_k$are respective momentum weights. For the objective function of the IHS, the momentum update becomes:

$$\begin{aligned} x^{\mathrm{k+1}} &= x^{\mathrm{k}} + \alpha_k (A^T S^T SA)^{-1} A^T (b - Ax^{\mathrm{k}}) + \beta_k (x^{\mathrm{k}} - x^{\mathrm{k\text{-}1}}) \\ &= x^{\mathrm{k}} - \alpha_k (A^T S^T SA)^{-1} A^T A (x^{\mathrm{k}} - x^{\mathrm{LS}}) + \beta_k (x^{\mathrm{k}} - x^{\mathrm{k\text{-}1}}). \end{aligned}$$

Now, consider the following bipartite transformation:

$$\begin{bmatrix} x^{\mathrm{k+1}} - x^{\mathrm{LS}} \\ x^{\mathrm{k}} - x^{\mathrm{LS}} \end{bmatrix} = T \begin{bmatrix} x^{\mathrm{k}} - x^{\mathrm{LS}} \\ x^{\mathrm{k\text{-}1}} - x^{\mathrm{LS}} \end{bmatrix}$$

$$T = \begin{bmatrix} (1+\beta) I_d - \alpha (A^T S^T SA)^{-1} A^T A & -\beta I_d \\ I_d & 0 \end{bmatrix}$$

where momentum weights are kept constant during the iterations. By using the same similarity transformation in [**?**],[**?**], we can find a block diagonal form for the transformation matrix T, so that we can determine its eigenvalues easily. For this purpose, the following change of basis will be used:

$$T = P^{-1}\,\mathbf{diag}(T_1, \ldots, T_d) P, \quad T_i := \begin{bmatrix} 1+\beta-\alpha\lambda_i & \beta \\ 1 & 0 \end{bmatrix}$$

$$P = \begin{bmatrix} U & 0 \\ 0 & U \end{bmatrix} \Pi, \qquad \Pi_{i,j} = \begin{cases} 1 & \text{i odd} j = i, \\ 1 & \text{i even} j = n+i, \\ 0 & \text{otherwise} \end{cases}$$

where $U\Lambda U^T$is the eigenvalue decomposition of $(A^T S^T SA)^{-1} A^T A$ and λ_i is the i^{th} eigenvalue. The characteristic polynomial of each block is $u^2 - (1+\beta-\alpha\lambda_i)u + \beta = 0$. If $\beta \ge (1-\sqrt{\alpha\lambda_i})^2$, then both of the roots will be imaginary and both will have a magnitude $\sqrt{\beta}$ which will be the contraction ratio of the transformation. Note that β can be selected to ensure this upper bound for all eigenvalues. For this purpose, checking only the largest and the smallest λ_i values, which is determined by using the MP Law in the previous section, is sufficient:

$$\beta \ge \max\left\{\left|1 - \frac{\sqrt{\alpha}}{1+\sqrt{r}}\right|, \left|1 - \frac{\sqrt{\alpha}}{1-\sqrt{r}}\right|\right\}^2. \tag{9}$$

The lower bound on β can be minimized over α by choosing $\alpha = (1-r)^2$, so that the contraction ratio reaches its smallest value of $\beta = r$. Consequently, the resulting convergence rate becomes:

$$\|x^{\mathrm{k+1}} - x^{\mathrm{LS}}\|_2 \le r^{k/2} \|x^0 - x^{\mathrm{LS}}\|_2. \tag{10}$$

If the convergence rates of the Naive IHS and the proposed M-IHS obtained in eq. (8) and eq. (10), respectively, are compared, then an improvement of factor $2/(1+r)$ can be observed. Recall that the above analysis is valid for use of a single sketch in all iterations. A pseudo-algorithm of M-IHS can be seen below. Iterations of the M-IHS do not require

Algorithm 1 M-IHS

Data: $SA \in \mathbf{R}^{m\times d}, x^0, A, b$
$\beta = d/m$ $\alpha = (1-\beta)^2$
while *stopping criteria* **do**
  1. $g^{\mathrm{k}} = A^T(b - Ax^{\mathrm{k}})$
  2. $(SA)^T SAz^{\mathrm{k}} = g^{\mathrm{k}}$ $(solve\ for\ z)$
  3. $x^{\mathrm{k+1}} = x^{\mathrm{k}} + \alpha z^{\mathrm{k}} + \beta(x^{\mathrm{k}} - x^{\mathrm{k\text{-}1}})$
end

any inner products or norm calculations, which avoids synchronization steps in parallel computing and results in overwhelming advantages over the CG or the GMRES like iterative solvers in distributed or hierarchical memory systems (we refer the reader to Section 2.4 of [**?**]). Indeed, M-IHS is equivalent to the CS with a preconditioner $(SA)^T SA$ except for one improvement: momentum parameters are determined more accurately by the MP Law in the M-IHS than adaptive approach in CS, which results in faster convergence as seen

on Figure 3. This suggests that the M-IHS can take CS's place in those applications where parallel computation is viable.

Moreover, due to the absence of inner products, the M-IHS, even in sequential systems, requires fewer operations than the CG or the LSQR as observed on Figure 3. Most importantly, computation of vector z^k in the second line of Algorithm 1 can be realized by utilizing a symmetric CG technique as a sub-solver, which avoids the md^2 term in the complexity. Avoiding md^2 term may not be possible for the CG-like techniques which use randomized preconditioning, because the R factor in the QR decomposition or the V factor in the SVD require $O(md^2)$ operations. Note that an inexact sub-solver strategy is more suitable if a regularization term is used. Otherwise, convergence of the sub-solver would be exorbitantly slow since SA is ill conditioned.

3. RESULTS AND COMPARISONS

In MATLAB simulations, we used the singular value profile extracted from *baart* function of Hansen's Toolbox [**?**]. After scaling and shifting into desired interval, the singular values have been placed into SVD of data matrix $A \in \mathbf{R}^{2^{16}\times 500}$ whose entries are sampled from the distribution $\mathcal{N}(0, 9)$. We did not inlude any noise in the simulations to focus on the convergence behaviour of the algorithms. Additionally, results of all randomized schemes were averaged over 20 Monte Carlo simulations. The obtained $2/(1 + r)$ improvement by the M-IHS over the Naive IHS can be seen on Figure 1. Furthermore, as shown in Figure 2, when the condition number κ increases, convergence rate of the CG degrades considerably while the performance of the proposed M-IHS technique remains unaffected. To solve the normal equations, LS version of CG implemented by Saunders was used [**?**]. Operation counts in the figures were obtained by Lightspeed Toolbox [**?**]. LS solution was obtained by using the QR Decomposition with the Householder transformation.

Performance comparison of the proposed M-IHS with Blendenpik and LSRN in MATLAB would not be fair, since their released packages are implemented in C language. Instead, we compared, in Figure 3, the M-IHS with the CG and the LSQR both of which use randomized preconditioning. The R factor in the QR decomposition of the sketched matrix was used as the preconditioner for the CG, LSQR and the CS techniques, whereas it was used for M-IHS to solve the linear system appeared in the second line of the Algorithm 1. The same sketched matrix SA with sketch size $m = 7d$ was used for all the techniques and SA was created by using the Discrete Cosine Transform. All randomization parts in the compared techniques are the same for a fair comparison. In all figures, the numbers between parenthesis in the legends indicates the number of total iterations made by the technique to obtain seen result.

4. CONCLUSION

By using the heavy ball method, a novel iterative solver for the least square problem is proposed. The proposed M-IHS technique converges significantly faster than the naive IHS technique. Furthermore, the computational complexity of the proposed method is lower than the CG-like techniques. Also, the M-IHS can easily be implemented in parallel and the complexity can be reduced further by using an iterative sub-solver in regularized cases. As a future work, the impact of regularization on the convergence rate will be investigated.

5. ACKNOWLEDGEMENTS

This work was partially supported by the National Science Foundation under grant IIS-1838179.

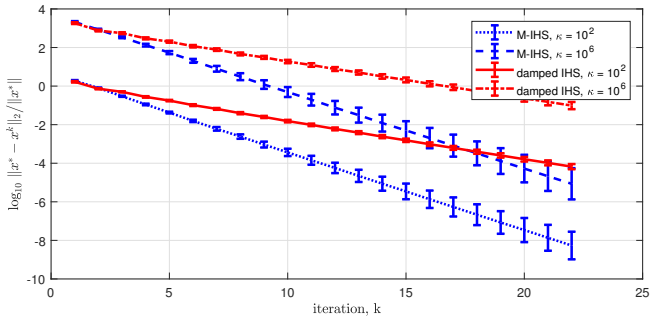

Fig. 1. Comparison with the damped IHS: the theoretical improvement rate $2/(1+r)$ can be observed by comparing iteration numbers for an accuracy.

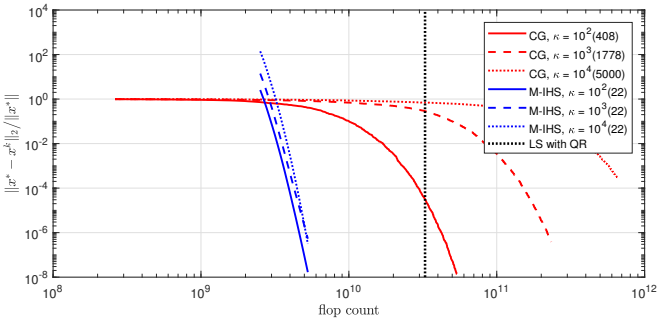

Fig. 2. Comparison with the CGLS: as κ increases, M-IHS remains unaffected and requires substantially less operations than both the CGLS which is unable to converge even in d iterations due to round-off errors and the full LS solution.

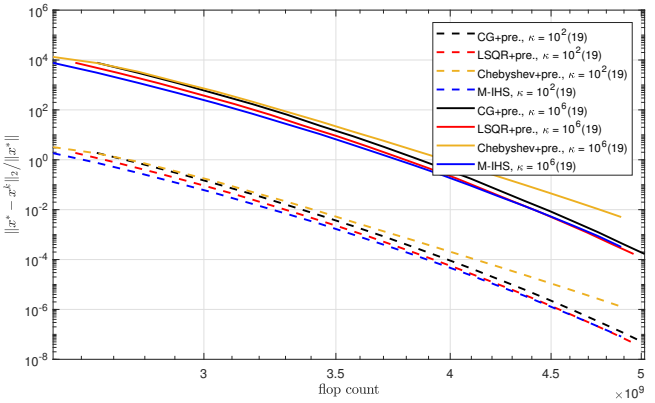

Fig. 3. Comparison with randomized preconditioning methods: the slopes of the curves demonstrates that CG-like methods using randomized preconditioners have similar convergence rate with the M-IHS which requires fewer flop counts per iteration.