

# On Truncated-SVD-like Sparse Solutions to Least-Squares Problems of Arbitrary Dimensions <sup>\*</sup>

Christos Boutsidis

Mathematical Sciences Department  
IBM T.J. Watson Research Center  
cboutsi@us.ibm.com

December 30, 2011

## Abstract

We describe two algorithms for computing a sparse solution to a least-squares problem where the coefficient matrix can have arbitrary dimensions. We show that the solution vector obtained by our algorithms is close to the solution vector obtained via the truncated SVD approach.

## 1 Introduction

Fix inputs  $A \in \mathbb{R}^{m \times n}$  and  $\mathbf{b} \in \mathbb{R}^m$ . We study the following minimization least-squares problem,

$$\min_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2.$$

Since there is no assumption on  $m$  and  $n$ , the above problem might have more than one solutions. It is well known though that the solution vector which minimizes both  $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2$  and  $\|\mathbf{x}\|_2$  can be found using the pseudo-inverse of  $A$ ,

$$\mathbf{x}^* = A^+ \mathbf{b}.$$

When  $A$  is ill-conditioned,  $A^+$  becomes unstable to perturbations and overfitting can become a serious problem. Practitioners deal with such situations by regularizing the above least-squares problem. Popular regularization techniques include the Lasso [10], the Tikhonov regularization [6], as well as the truncated SVD regularization [8]. In this work, we focus on the later approach and develop a new regularization tool which returns a *sparse* solution vector that has comparable performance to the *dense* solution vector which is obtained via the truncated SVD. In some more details, for  $k < \text{rank}(A)$ , let  $A_k \in \mathbb{R}^{m \times n}$  of rank  $k$  denotes the rank- $k$  SVD of  $A$ ; then, the truncated SVD regularized solution  $\mathbf{x}_k^*$  is given by

$$\mathbf{x}_k^* = A_k^+ \mathbf{b}.$$

In the present article, we describe a deterministic and a randomized algorithm that compute a solution vector  $\hat{\mathbf{x}}_r \in \mathbb{R}^n$  with  $r$  non-zero entries such that  $r \approx k$  and

$$\|\mathbf{A}\hat{\mathbf{x}}_r - \mathbf{b}\|_2 \approx \|\mathbf{A}\mathbf{x}_k^* - \mathbf{b}\|_2.$$

Our main motivation is interpretability: a sparse vector  $\hat{\mathbf{x}}_r$  implies that  $\mathbf{b}$  can be (approximately) expressed as a linear combination of a small set of columns from  $A$ . On the other hand, any dense solution vector  $\mathbf{x}$ , such as  $\mathbf{x}_k^*$ , expresses  $\mathbf{b}$  as a linear combination of (up to) all the columns of  $A$ .

---

<sup>\*</sup>Similar results to the main results of this article appeared previously in [1].

## 1.1 Preliminaries

The Singular Value Decomposition (SVD) of a matrix  $A \in \mathbb{R}^{m \times n}$  of rank  $\rho$  is

$$A = \underbrace{\begin{pmatrix} U_k & U_{\rho-k} \end{pmatrix}}_{U_A \in \mathbb{R}^{m \times \rho}} \underbrace{\begin{pmatrix} \Sigma_k & \mathbf{0} \\ \mathbf{0} & \Sigma_{\rho-k} \end{pmatrix}}_{\Sigma_A \in \mathbb{R}^{\rho \times \rho}} \underbrace{\begin{pmatrix} V_k^T \\ V_{\rho-k}^T \end{pmatrix}}_{V_A^T \in \mathbb{R}^{\rho \times n}},$$

Here,  $U_k \in \mathbb{R}^{m \times k}$  and  $U_{\rho-k} \in \mathbb{R}^{m \times (\rho-k)}$  contain the left singular vectors of  $A$ . Similarly,  $V_k \in \mathbb{R}^{n \times k}$  and  $V_{\rho-k} \in \mathbb{R}^{n \times (\rho-k)}$  contain the right singular vectors. The singular values of  $A$ , which we denote as  $\sigma_1(A) \geq \sigma_2(A) \geq \dots \geq \sigma_\rho(A) > 0$  are contained in  $\Sigma_k \in \mathbb{R}^{k \times k}$  and  $\Sigma_{\rho-k} \in \mathbb{R}^{(\rho-k) \times (\rho-k)}$ . One can compute the SVD of  $A$  in  $O(mn \min\{m, n\})$  time. We use  $A^+ = V_A \Sigma_A^{-1} U_A^T \in \mathbb{R}^{n \times m}$  to denote the Moore-Penrose pseudo-inverse of  $A$  with  $\Sigma_A^{-1}$  denoting the inverse of  $\Sigma_A$ . Let  $A_k = U_k \Sigma_k V_k^T \in \mathbb{R}^{m \times n}$  and  $A_{\rho-k} = A - A_k = V_{\rho-k} \Sigma_{\rho-k} V_{\rho-k}^T \in \mathbb{R}^{m \times n}$ . For  $k < \text{rank}(A)$ , the SVD gives the best rank  $k$  approximation to  $A$  in both the spectral and the Frobenius norm: for  $\tilde{A} \in \mathbb{R}^{m \times n}$ , let  $\text{rank}(\tilde{A}) \leq k$ ; then, for  $\xi = 2, F$ ,  $\|A - A_k\|_\xi \leq \|A - \tilde{A}\|_\xi$ . Also,  $\|A - A_k\|_2 = \|\Sigma_{\rho-k}\|_2 = \sigma_{k+1}(A)$ , and  $\|A - A_k\|_F^2 = \|\Sigma_{\rho-k}\|_F^2 = \sum_{i=k+1}^\rho \sigma_i^2(A)$ . The Frobenius and the spectral norm of  $A$  are defined as:  $\|A\|_F^2 = \sum_{i,j} A_{ij}^2 = \sum_{i=1}^\rho \sigma_i^2(A)$ ; and  $\|A\|_2 = \sigma_1(A)$ . Let  $X$  and  $Y$  are matrices of appropriate dimensions; then,  $\|XY\|_F \leq \min\{\|X\|_F \|Y\|_2, \|X\|_2 \|Y\|_F\}$ . This is a stronger version of the standard submultiplicativity property  $\|XY\|_F \leq \|X\|_F \|Y\|_F$ , which we will refer to as spectral submultiplicativity. Recall the least-squares problem  $\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2$ . Given  $k < \rho = \text{rank}(A)$ , the  $k$  truncated SVD regularized weights are  $\mathbf{x}_k^* = A_k^+ \mathbf{b} = V_k \Sigma_k^{-1} U_k^T \mathbf{b} \in \mathbb{R}^n$ . Note also that  $\|\mathbf{b} - U_k U_k^T \mathbf{b}\|_2 = \|\mathbf{b} - A_k A_k^+ \mathbf{b}\|_2$ . Finally, for  $r < n$ , let  $C \in \mathbb{R}^{m \times r}$  contains  $r$  columns of  $A$ . We can equivalently write  $C = A\Omega$ , where the *sampling matrix* is  $\Omega = [\mathbf{e}_{i_1}, \dots, \mathbf{e}_{i_r}] \in \mathbb{R}^{n \times r}$  and  $\mathbf{e}_i \in \mathbb{R}^n$  are appropriate vectors from the standard basis. Let  $S \in \mathbb{R}^{r \times r}$  be a diagonal *rescaling matrix* with positive entries; then,  $C = A\Omega S$  contains a subset of  $r$  columns from  $A$  rescaled with the corresponding diagonal elements in  $S$ .

## 1.2 Main Results

**Theorem 1.** Let  $A \in \mathbb{R}^{m \times n}$ ,  $\mathbf{b} \in \mathbb{R}^m$ ,  $0 < k < \text{rank}(A)$ , and  $0 < \epsilon < 1/2$ . Algorithm 1 runs in time  $O(mn \min\{m, n\} + nk^3/\epsilon^2)$  and returns  $\hat{\mathbf{x}}_r \in \mathbb{R}^n$  with  $r = \lceil 9k/\epsilon^2 \rceil$  non-zero entries such that,

$$\|A\hat{\mathbf{x}}_r - \mathbf{b}\|_2 \leq \|A\mathbf{x}_k^* - \mathbf{b}\|_2 + (1 + \epsilon)\|\mathbf{b}\|_2 \frac{\|A - A_k\|_F}{\sigma_k(A)}.$$

**Theorem 2.** Let  $A \in \mathbb{R}^{m \times n}$ ,  $\mathbf{b} \in \mathbb{R}^m$ ,  $0 < k < \text{rank}(A)$ , and  $0 < \epsilon < 1/2$ . Algorithm 3 runs in time  $O(mn \min\{m, n\} + k \log k \log(k/\epsilon)/\epsilon^2)$  and returns  $\hat{\mathbf{x}}_r \in \mathbb{R}^n$  with  $r = \lceil 36k \ln(20k)/\epsilon^2 \rceil$  non-zero entries such that with probability at least 0.7,

$$\|A\hat{\mathbf{x}}_r - \mathbf{b}\|_2 \leq \|A\mathbf{x}_k^* - \mathbf{b}\|_2 + \epsilon\|\mathbf{b}\|_2 \frac{\|A - A_k\|_F}{\sigma_k(A)}.$$

**Lemma 3.** Fix  $A \in \mathbb{R}^{m \times n}$ ,  $\mathbf{b} \in \mathbb{R}^m$ , rank parameter  $k < \text{rank}(A)$ , and sparsity parameter  $r \geq k$ . Let  $\mathbf{x}_k^* = A_k^+ \mathbf{b} \in \mathbb{R}^n$ , where  $A_k \in \mathbb{R}^{m \times n}$  is the rank  $k$  matrix from the SVD of  $A$ . Let  $\Omega \in \mathbb{R}^{n \times r}$  and  $S \in \mathbb{R}^{r \times r}$  be any sampling and rescaling matrices with  $\text{rank}(V_k^T \Omega S) = k$ . Let  $\hat{\mathbf{x}}_r \in \mathbb{R}^n$  be a vector with  $r$  non-zero entries which is obtained as follows: let  $C = A\Omega S \in \mathbb{R}^{m \times r}$  and  $\mathbf{x}_r = C^+ \mathbf{b} \in \mathbb{R}^r$ ; construct  $\hat{\mathbf{x}}_r \in \mathbb{R}^n$  from  $\mathbf{x}_r$  at the indices corresponding to the selected columns of  $C$  and 0 elsewhere. Then,

$$\|A\hat{\mathbf{x}}_r - \mathbf{b}\|_2 \leq \|A\mathbf{x}_k^* - \mathbf{b}\|_2 + \|(A - A_k)\Omega S(V_k^T \Omega S)^+ \Sigma_k^{-1} U_k^T \mathbf{b}\|_2.$$

This lemma is the main technical contribution of our work. Combining this lemma with two existing algorithms that satisfy its requirements gives our main theorems. More specifically, to design our deterministic algorithm (Algorithm 1) we used a method from [2] (Algorithm 2); to design our randomized algorithm (Algorithm 3) we used a method from [9] (Algorithm 4).

- 1: **Input:**  $A \in \mathbb{R}^{m \times n}$ , target rank  $k < \text{rank}(A)$ , and accuracy parameter  $0 < \epsilon < 1/2$ .
- 2: Compute  $V_k \in \mathbb{R}^{n \times k}$  and  $E = A - A_k = A - AV_kV_k^T \in \mathbb{R}^{m \times n}$  from the SVD of  $A$ .
- 3: Let  $r = \lceil 9k/\epsilon^2 \rceil$ ,  $[\Omega, S] = \text{DeterministicSampling}(V_k^T, E, r)$ , and  $C = A\Omega S \in \mathbb{R}^{m \times r}$ .
- 4: Let  $\mathbf{x}_r = C^+ \mathbf{b} \in \mathbb{R}^r$ ; and construct  $\hat{\mathbf{x}}_r \in \mathbb{R}^n$  from  $\mathbf{x}_r$  at the indices corresponding to the selected columns  $C$  and 0 elsewhere.
- 5: **Return**  $\hat{\mathbf{x}}_r \in \mathbb{R}^n$  with at most  $r$  non-zero entries.

**Algorithm 1:** A Deterministic Sparse Solver for Least-Squares

- 1: **Input:**  $V^T = [\mathbf{v}_1, \dots, \mathbf{v}_n] \in \mathbb{R}^{k \times n}$  with  $\mathbf{v}_i \in \mathbb{R}^k$ ,  $E = [\mathbf{e}_1, \dots, \mathbf{e}_n] \in \mathbb{R}^{m \times n}$  with  $\mathbf{e}_i \in \mathbb{R}^m$ , and  $r > k$ .
- 2: Initialize  $B_0 = \mathbf{0}_{k \times k}$ ,  $\Omega = \mathbf{0}_{n \times r}$ , and  $S = \mathbf{0}_{r \times r}$ .
- 3: **for**  $\tau = 0$  **to**  $r - 1$  **do**
- 4:   Set  $L_\tau = \tau - \sqrt{rk}$ .
- 5:   Pick index  $i \in \{1, 2, \dots, n\}$  and  $t$  such that  $U(\mathbf{e}_i) \leq \frac{1}{t} \leq L(\mathbf{v}_i, B_{\tau-1}, L_\tau)$ .
- 6:   Update  $B_\tau = B_{\tau-1} + t\mathbf{v}_i\mathbf{v}_i^T$ . Set  $\Omega_{i,\tau+1} = 1$  and  $S_{\tau+1,\tau+1} = 1/\sqrt{t}$ .
- 7: **end for**
- 8: **Return:** Sampling and rescaling matrices  $\Omega \in \mathbb{R}^{n \times r}$ ,  $S \in \mathbb{R}^{r \times r}$ .

**Algorithm 2:** DeterministicSampling [2]

## 2 Algorithms

**Our deterministic algorithm (Theorem 1).** Algorithm 1 deterministically selects  $r$  columns of  $A$  to form  $C$  and the corresponding sparse vector  $\hat{\mathbf{x}}_r$ . The meat of this method is the subroutine `DeterministicSampling`, which is an algorithm to simultaneously sample the columns of two matrices, while controlling their spectral and Frobenius norms. `DeterministicSampling` takes input two matrices  $V^T \in \mathbb{R}^{k \times n}$  and  $E \in \mathbb{R}^{m \times n}$ . We assume that  $V$  is orthonormal, so  $V^T V = I_k$ . To describe the algorithm, it is convenient to view these two matrices as two sets of  $n$  column vectors,  $V^T = [\mathbf{v}_1, \dots, \mathbf{v}_n]$  and  $E = [\mathbf{e}_1, \dots, \mathbf{e}_n]$ . In our application,  $V^T = V_k^T$  and  $E = A - A_k$ .

Given  $k$  and  $r$ , introduce the iterator  $\tau = 0, 1, 2, \dots, r - 1$ , and define the parameter  $L_\tau = \tau - \sqrt{rk}$ . For a square symmetric matrix  $B \in \mathbb{R}^{k \times k}$  with eigenvalues  $\lambda_1, \dots, \lambda_k$  and  $L \in \mathbb{R}$ , define functions  $\phi(L, B) = \sum_{i=1}^k \frac{1}{\lambda_i - L}$ , and  $L(\mathbf{v}, B, L) = \frac{\mathbf{v}^T (B - L'I_k)^{-2} \mathbf{v}}{\phi(L', B) - \phi(L, B)} - \mathbf{v}^T (B - L'I_k)^{-1} \mathbf{v}$ , where  $L' = L + 1$ . Also, for a vector  $\mathbf{e}$ , define function  $U(\mathbf{e}) = \frac{\mathbf{e}^T \mathbf{e}}{\|A\|_F^2} \left(1 - \sqrt{k/r}\right)$ . At every step  $\tau$ , the algorithm selects a column with index  $i$  for which  $U(\mathbf{e}_i) \leq L(\mathbf{v}_i, B, L_\tau)$ . The running time of the method is dominated by the search for a column which satisfies  $U \leq L$ . To compute  $L$ , one needs  $\phi(L, B)$ , and hence the eigenvalues of  $B$ , and  $(B - L'I_k)^{-1}$ . This takes  $O(k^3)$  time once per iteration, for a total of  $O(rk^3)$ . Then, for  $i = 1, \dots, n$ , we need to compute  $L$  for every  $\mathbf{v}_i$ . This takes  $O(nk^2)$  per iteration, for a total of  $O(nrk^2)$ . To compute  $U$ , we need  $\mathbf{e}_i^T \mathbf{e}_i$  for  $i = 1, \dots, n$  which takes  $O(mn)$ . So, in total `DeterministicSampling` takes  $O(nrk^2 + mn)$ , hence Algorithm 1 needs  $O(mn \min\{m, n\} + nk^3/\epsilon^2)$ .

`DeterministicSampling` selects vectors using a greedy procedure such that the sampled vectors satisfy the bounds of the Lemma 4 below. The bounds of Lemma 4 along with the structural bound of Lemma 3 immediately give the result in Theorem 1.

**Lemma 4** ([2]). `DeterministicSampling` constructs matrices  $\Omega, S$  such that,

$$\|(V^T \Omega S)^+\|_2 \leq 1 - \sqrt{\frac{k}{r}} \qquad \|E \Omega S\|_F \leq \|E\|_F.$$

- 1: **Input:**  $A \in \mathbb{R}^{m \times n}$ , target rank  $k < \text{rank}(A)$ , and accuracy parameter  $0 < \epsilon < 1/2$ .
- 2: Compute  $V_k \in \mathbb{R}^{n \times k}$  from the SVD of  $A$ .
- 3: Let  $r = \lceil 36k \ln(20k)/\epsilon^2 \rceil$ ,  $[\Omega, S] = \text{RandomSampling}(V_k^T, r)$ , and  $C = X\Omega S \in \mathbb{R}^{m \times r}$ .
- 4: Let  $\mathbf{x}_r = C^+ \mathbf{b} \in \mathbb{R}^r$ ; and construct  $\hat{\mathbf{x}}_r \in \mathbb{R}^n$  from  $\mathbf{x}_r$  at the indices corresponding to the selected columns  $C$  and 0 elsewhere.
- 5: **return**  $\hat{\mathbf{x}}_r \in \mathbb{R}^n$  with at most  $r$  non-zero entries.

**Algorithm 3:** A Randomized Sparse Solver for Least-Squares

**Our randomized algorithm (Theorem 2).** Algorithm 3 is similar to Algorithm 1. The main qualitative differences are that it only needs  $V_k^T$  and it uses a randomized algorithm to sample the columns. The basic idea in `RandomSampling` is to compute probabilities  $p_i = \|\mathbf{v}_i\|_2^2/k$  for  $i = 1, \dots, n$ , and then sample  $r$  columns in  $r$  i.i.d. trials. In each trial, a column is sampled according to these probabilities. The running time of `RandomSampling` is  $O(nk + r \log(r))$ , so the total running time of Algorithm 3 is  $O(mn \min\{m, n\} + k \log k \log(k/\epsilon)/\epsilon^2)$ .

As with Lemma 4, we are going to need some properties of the sampling and rescaling matrices that `RandomSampling` delivers. This random sampling algorithm was introduced in [9]. Lemma 5 is an application of this algorithm for sampling columns from matrices of orthonormal rows, while Lemma 6 is a simple corollary of Lemma 5. Lemma 8 is a simple fact that proved recently in [5]. Finally, Lemma 9 is a direct application from a matrix multiplication result in [5].

**Lemma 5** ([9]). *If  $r \geq 4k \ln(2k/\delta)/\epsilon^2$ , then, w.p. at least  $1 - \delta$ :  $\|V^T \Omega S S^T \Omega^T V - I_k\|_2 \leq \epsilon$ .*

**Lemma 6** (Corollary of Lemma 5). *If  $r = 4k \ln(2k/\delta)/\epsilon^2$ , then w.p. at least  $1 - \delta$ ,  $\|(V^T \Omega S)^+ - (V^T \Omega S)^T\|_2 \leq \frac{\epsilon}{\sqrt{1-\epsilon}}$ .*

**Lemma 7** ([5], eq. (36)). *For any matrix  $E \in \mathbb{R}^{m \times n}$ ,  $\mathbf{E} [\|E \Omega S\|_F^2] = \|E\|_F^2$ .*

**Corollary 8** (By Markov's inequality). *For any  $E \in \mathbb{R}^{m \times n}$ , w.p.  $1 - \delta$   $\|E \Omega S\|_F^2 \leq \frac{1}{\delta} \|E\|_F^2$ .*

**Lemma 9** ([5]). *For  $1 \leq r \leq n$  and any  $E$  with  $EV = \mathbf{0}_{m \times k}$ , w.p.  $1 - \delta$ ,  $\|E \Omega S S^T \Omega^T V\|_F^2 \leq \frac{k}{\delta r} \|E\|_F^2$ .*

*Proof.* We apply eqn. (4) of Lemma 4 in [5], with  $E$ ,  $V$ , and  $EV = \mathbf{0}_{m \times k}$ :

$$\mathbf{E} [\|E \Omega S S^T \Omega^T V\|_F^2] \leq \sum_{i=1}^n \frac{\|E^{(i)}\|_2^2 \|V_{(i)}\|_2^2}{r p_i}.$$

After substituting  $p_i = \|V_{(i)}\|_2^2/k$ , and using Markov's inequality, the result follows. ■

- 1: **Input:**  $V^T = [\mathbf{v}_1, \dots, \mathbf{v}_n] \in \mathbb{R}^{k \times n}$  with  $\mathbf{v}_i \in \mathbb{R}^k$  and  $r \geq 4k \ln k$ .
- 2: For  $i = 1, \dots, n$  compute probabilities  $p_i = \frac{1}{k} \|\mathbf{v}_i\|_2^2$ .
- 3: Initialize  $\Omega = \mathbf{0}_{n \times r}$  and  $S = \mathbf{0}_{r \times r}$ .
- 4: **for**  $\tau = 1$  **to**  $r$  **do**
- 5: Select index  $i \in \{1, 2, \dots, n\}$  i.i.d. with the probability of selecting index  $i$  being  $p_i$ .
- 6: Set  $\Omega_{i,\tau} = 1$  and  $S_{\tau,\tau} = 1/\sqrt{p_i r}$ .
- 7: **end for**
- 8: **Return:** Sampling and rescaling matrices  $\Omega \in \mathbb{R}^{n \times r}$ ,  $S \in \mathbb{R}^{r \times r}$

**Algorithm 4:** RandomSampling [9]

## 3 Proofs

### 3.1 Proof of Theorem 1

By Lemma 4,  $\text{rank}(\mathbf{V}_k^T \Omega \mathbf{S}) = k$  (to see this, use  $\|(\mathbf{V}_k^T \Omega \mathbf{S})^+\|_2 = 1/\sigma_k(\mathbf{V}_k^T \Omega \mathbf{S})$  and  $r > k$ ), so we can apply Lemma 3:

$$\|\mathbf{A} \hat{\mathbf{x}}_r - \mathbf{b}\|_2 \leq \|\mathbf{A} \mathbf{x}_k^* - \mathbf{b}\|_2 + \|(\mathbf{A} - \mathbf{A}_k) \Omega \mathbf{S} (\mathbf{V}_k^T \Omega \mathbf{S})^+ \Sigma_k^{-1} \mathbf{U}_k^T \mathbf{b}\|_2.$$

We manipulate the second term of this equation as follows. First, notice that

$$\|(\mathbf{A} - \mathbf{A}_k) \Omega \mathbf{S} (\mathbf{V}_k^T \Omega \mathbf{S})^+ \Sigma_k^{-1} \mathbf{U}_k^T \mathbf{b}\|_2 = \|(\mathbf{A} - \mathbf{A}_k) \Omega \mathbf{S} (\mathbf{V}_k^T \Omega \mathbf{S})^+ \Sigma_k^{-1} \mathbf{U}_k^T \mathbf{b}\|_{\mathbb{F}}.$$

Then,

$$\begin{aligned} \|(\mathbf{A} - \mathbf{A}_k) \Omega \mathbf{S} (\mathbf{V}_k^T \Omega \mathbf{S})^+ \Sigma_k^{-1} \mathbf{U}_k^T \mathbf{b}\|_{\mathbb{F}} &\leq \|(\mathbf{A} - \mathbf{A}_k) \Omega \mathbf{S} (\mathbf{V}_k^T \Omega \mathbf{S})^+\|_{\mathbb{F}} \|\Sigma_k^{-1} \mathbf{U}_k^T \mathbf{b}\|_2 & (a) \\ &\leq \|(\mathbf{A} - \mathbf{A}_k) \Omega \mathbf{S}\|_{\mathbb{F}} \|(\mathbf{V}_k^T \Omega \mathbf{S})^+\|_2 \sigma_k^{-1}(\mathbf{A}) \|\mathbf{b}\|_2^2 & (b) \\ &\leq \|\mathbf{A} - \mathbf{A}_k\|_{\mathbb{F}} \left(1 - \sqrt{k/r}\right)^{-2} \sigma_k^{-1}(\mathbf{A}) \|\mathbf{b}\|_2 & (c) \\ &\leq (1 + \epsilon) \|\mathbf{A} - \mathbf{A}_k\|_{\mathbb{F}} \sigma_k^{-1}(\mathbf{A}) \|\mathbf{b}\|_2. & (d) \end{aligned}$$

(a) follows by spectral submultiplicativity; (b) follows by submultiplicativity and using  $\|\mathbf{U}_k^T\|_2 = 1$ ; (c) follows by the bounds in Lemma 4; finally, (d) follows after some algebra, because  $r = \lceil 9k/\epsilon^2 \rceil$ .

### 3.2 Proof of Theorem 2

The basic idea of the proof is similar, except that we now use the lemmas corresponding to **Random-Sampling**. Let  $\delta = 1/10$  and assume for the moment that  $r = \lceil 4k \ln(2k/\delta)/\epsilon^2 \rceil$  (rescaling  $\epsilon$  below will give the value of  $r$  in the theorem); the sampling and rescaling matrices  $\Omega, \mathbf{S}$  are returned by **RandomSampling** $(\mathbf{V}_k^T, r)$  (see also Section 2). Then,  $\mathbf{C} = \mathbf{A}\Omega$ . By the union bound, with probability at least  $1 - 3\delta = 0.7$ , the bounds in Lemmas 5, 9 and Corollary 8 all hold. Since Lemma 5 implies Lemma 6 and 9, all four of these bounds hold. The remainder of the proof assumes that we are in this 0.7 probability event when all four bounds hold.

From Lemma 5,  $\text{rank}(\mathbf{V}_k^T \Omega \mathbf{S}) = k$ , so Lemma 3 gives (recall that  $\mathbf{E} = \mathbf{A} - \mathbf{A}_k$ ):

$$\|\mathbf{A} \hat{\mathbf{x}}_r - \mathbf{b}\|_2 \leq \|\mathbf{A} \mathbf{x}_k^* - \mathbf{b}\|_2 + \|\mathbf{E} \Omega \mathbf{S} (\mathbf{V}_k^T \Omega \mathbf{S})^+ \Sigma_k^{-1} \mathbf{U}_k^T \mathbf{b}\|_{\mathbb{F}}.$$

As with the proof of Theorem 1, we manipulate the second term as follows:

$$\begin{aligned} \|\mathbf{E} \Omega \mathbf{S} (\mathbf{V}_k^T \Omega \mathbf{S})^+ \Sigma_k^{-1} \mathbf{U}_k^T \mathbf{b}\|_{\mathbb{F}} &\leq \|\mathbf{E} \Omega \mathbf{S} (\mathbf{V}_k^T \Omega \mathbf{S})^+\|_{\mathbb{F}} \|\Sigma_k^{-1} \mathbf{U}_k^T \mathbf{b}\|_2 \\ &\leq \|\mathbf{E} \Omega \mathbf{S} (\mathbf{V}_k^T \Omega \mathbf{S})^+\|_{\mathbb{F}} \cdot \sigma_k^{-1}(\mathbf{A}) \cdot \|\mathbf{b}\|_2. \end{aligned}$$

We now bound  $\|\mathbf{E} \Omega \mathbf{S} (\mathbf{V}_k^T \Omega \mathbf{S})^+\|_{\mathbb{F}}$  as follows.

$$\begin{aligned} \|\mathbf{E} \Omega \mathbf{S} (\mathbf{V}_k^T \Omega \mathbf{S})^+\|_{\mathbb{F}} &\leq \|\mathbf{E} \Omega \mathbf{S} \mathbf{S}^T \mathbf{V}_k\|_{\mathbb{F}} + \|\mathbf{E} \Omega \mathbf{S} ((\mathbf{V}_k^T \Omega \mathbf{S})^+ - (\mathbf{V}_k^T \Omega \mathbf{S})^T)\|_{\mathbb{F}} \\ &\leq \|\mathbf{E}\|_{\mathbb{F}} \sqrt{k/r\delta} + \|\mathbf{E}\|_{\mathbb{F}} \epsilon / \sqrt{\delta(1-\epsilon)} \\ &\leq 3\epsilon \|\mathbf{E}\|_{\mathbb{F}}. \end{aligned}$$

The first inequality follows from the triangle inequality, and the second follows after applying Lemma 9 to the first term, spectral submultiplicativity, Lemma 6, and Corollary 8; the last inequality follows from elementary algebra, because  $\epsilon < 1/2$ . Rescale  $\epsilon \rightarrow \epsilon/3$ , to wrap up.  $\blacksquare$

### 3.3 Proof of Lemma 3

We will prove a considerably more general result, of which Lemma 3 is a simple corollary. Below, we first introduce a general matrix approximation problem and present an algorithm and an approximation result for this problem (Lemma 10). Lemma 3 is a simple corollary of Lemma 10.

Let  $B \in \mathbb{R}^{m \times \omega}$  be a matrix which we would like to approximate; let  $A \in \mathbb{R}^{m \times n}$  be the matrix which we will use to approximate  $B$ . Specifically we want a *sparse* approximation of  $B$  from  $A$ , which means that we would like to choose  $C \in \mathbb{R}^{m \times r}$  consisting of  $r < n$  columns from  $A$  such that  $\|B - CC^+B\|_F$  is small. If  $A = B$  (approximating  $B$  using the columns of  $B$ ), then, this is the column based matrix approximation problem, which has received much interest recently [3, 2]. The more general problem which we study here, with  $A \neq B$ , takes on a surprisingly more difficult flavor. Our motivation is regression, but the problem could be of more general interest. We will approach the problem through the use of matrix factorizations. For  $Z \in \mathbb{R}^{n \times k}$ , with  $Z^T Z = I_k$ , let  $A = HZ^T + E$ , where  $H \in \mathbb{R}^{m \times k}$ ; and,  $E \in \mathbb{R}^{m \times n}$  is the residual error of the factorization. For fixed  $A$  and  $Z$ ,  $\|E\|_\xi$  ( $\xi = 2, F$ ) is minimized when  $H = AZ$ . Let  $\Omega \in \mathbb{R}^{n \times r}$ ,  $S \in \mathbb{R}^{r \times r}$  be sampling and rescaling matrices, respectively, and let  $C = A\Omega S \in \mathbb{R}^{m \times r}$ .

**Lemma 10** (Generalized Column-based Matrix Approximation). *If  $\text{rank}(Z^T \Omega S) = k$ , then,*

$$\|B - CC^+B\|_\xi \leq \|B - HH^+B\|_\xi + \|E\Omega(Z^T \Omega)^+ H^+ B\|_\xi.$$

*Proof.*

$$\|B - CC^+B\|_\xi \leq \|B - C(Z^T \Omega S)^+ H^+ B\|_\xi \tag{a}$$

$$= \|B - A\Omega(Z^T \Omega S)^+ H^+ B\|_\xi$$

$$= \|B - (HZ^T + E)\Omega S(Z^T \Omega S)^+ H^+ B\|_\xi$$

$$= \|B - H(Z^T \Omega S)(Z^T \Omega S)^+ H^+ B + E\Omega(Z^T \Omega S)^+ H^+ B\|_\xi$$

$$= \|B - HH^+B + E\Omega S(Z^T \Omega S)^+ H^+ B\|_\xi \tag{b}$$

$$\leq \|B - HH^+B\|_\xi + \|E\Omega S(Z^T \Omega S)^+ H^+ B\|_\xi. \tag{c}$$

(a) follows by the optimality of  $C^+B$ ; (b) follows because  $\text{rank}(Z^T \Omega S) = k$  and so  $Z^T \Omega S(Z^T \Omega S)^+ = I_k$ ; finally, (c) follows by the triangle inequality of matrix norms.  $\blacksquare$

This lemma is a general tool for the general matrix approximation problem. It is worth parsing this lemma carefully, to understand its implications. The left hand side is the matrix approximation of  $B$  using the dimensionally reduced  $C$ . The right hand side has two terms which highlight some tradeoffs: the first term is the approximation of  $B$  using  $H$  ( $H$  is used in the factorization to approximate  $A$ ); the second term is related to  $E$ , the residual error in approximating  $A$ . Ideally, one should choose  $H$  and  $Z$  to simultaneously approximate  $B$  with  $H$  and have small residual error  $E$ . In general, these are two competing goals, and a balance should be struck. For the remainder of this work, we focus on the Frobenius norm, and will consider only one extreme of this tradeoff, namely choosing the factorization to minimize  $\|E\|_F$ . Specifically, since  $Z$  has rank  $k$ , the best choice for  $HZ^T$  which minimizes  $\|E\|_F$  is  $A_k$ . In this case,  $E = A - A_k$ . Via the SVD,  $A_k = U_k \Sigma_k V_k^T$ , and so  $A = (U_k \Sigma_k) V_k^T + A - A_k$ . We can apply Lemma 10, with  $B = \mathbf{b}$ ,  $H = U_k \Sigma_k$ ,  $Z = V_k$  and  $E = A - A_k$ , giving as a corollary the next lemma.

**Lemma 11.** *If  $\text{rank}(V_k^T \Omega S) = k$ , then,*

$$\|\mathbf{b} - CC^+\mathbf{b}\|_2 \leq \|\mathbf{b} - U_k U_k^T \mathbf{b}\|_2 + \|(A - A_k)\Omega S(V_k^T \Omega S)^+ \Sigma_k^{-1} U_k^T \mathbf{b}\|_2.$$

Note that

$$\|\mathbf{b} - CC^+\mathbf{b}\|_2 = \|A \hat{\mathbf{x}}_r - \mathbf{b}\|_2,$$

for  $\hat{\mathbf{x}}_r \in \mathbb{R}^n$  constructed from this  $C$  and  $\|\mathbf{b} - U_k U_k^T \mathbf{b}\|_2 = \|\mathbf{b} - A \mathbf{x}_k^*\|$ , which give Lemma 3.

## 4 Related work

Our results can be viewed as extensions of similar results obtained before using the so-called Rank-Revealing QR (RRQR) factorization [4]. For fixed  $\mathbf{A}$ ,  $\mathbf{b}$ , and  $k$ , the authors of [4] use a QR-like decomposition to select exactly  $k$  columns of  $\mathbf{A}$  and compare their sparse solution vector  $\hat{\mathbf{x}}_k$  ( $r = k$  in this case) with  $\mathbf{x}_k^*$ ; notice that we compare the corresponding values  $\|\mathbf{A}\hat{\mathbf{x}}_k - \mathbf{b}\|_2$  and  $\|\mathbf{A}\mathbf{x}_k^* - \mathbf{b}\|_2$ . More specifically, Eqn. (12) of [4] along with the Strong RRQR results of [7] imply that

$$\begin{aligned} \|\mathbf{x}_k^* - \hat{\mathbf{x}}_k\|_2 &\leq \frac{\|\mathbf{A} - \mathbf{C}\mathbf{C}^+\mathbf{A}\|_2}{\sigma_k(\mathbf{A})} \cdot \left( 2\|\mathbf{A}_k^+\mathbf{b}\|_2 + \frac{\|\mathbf{b} - \mathbf{A}\mathbf{x}_k^*\|_2}{\sigma_k(\mathbf{A})} \right) \\ &\leq \frac{\sqrt{4k(n-k)+1}\|\mathbf{A} - \mathbf{A}_k\|_2}{\sigma_k(\mathbf{A})} \cdot \left( 2\|\mathbf{A}_k^+\mathbf{b}\|_2 + \frac{\|\mathbf{b} - \mathbf{A}\mathbf{x}_k^*\|_2}{\sigma_k(\mathbf{A})} \right) \\ &\leq \frac{\sqrt{4k(n-k)+1}\sigma_{k+1}(\mathbf{A})}{\sigma_k(\mathbf{A})} \cdot \left( 2\frac{\|\mathbf{b}\|_2}{\sigma_k(\mathbf{A})} + \frac{\|\mathbf{b} - \mathbf{A}\mathbf{x}_k^*\|_2}{\sigma_k(\mathbf{A})} \right) \\ &\leq \frac{\sqrt{4k(n-k)+1}}{\sigma_k(\mathbf{A})} \cdot (2\|\mathbf{b}\|_2 + \|\mathbf{b} - \mathbf{A}\mathbf{x}_k^*\|_2) \end{aligned}$$

This bound is interesting; however, it only applies to fixed sparsity  $r = k$ . Extending the Rank-Revealing QR approach for obtaining arbitrary  $r$ -sparse solutions is not obvious. Our algorithms though allow the user to set the sparsity parameter as large as she likes and trade the accuracy with the number of non-zero elements in the solution vector.

## References

- [1] C. Boutsidis, P. Drineas, and M. Magdon-Ismail. Sparse Features for PCA-Like Linear Regression. In *Neural Information Processing Systems (NIPS)*, 2011
- [2] C. Boutsidis, P. Drineas, and M. Magdon-Ismail. Near-optimal column based matrix reconstruction. In *Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2011.
- [3] C. Boutsidis, M. W. Mahoney, and P. Drineas. An improved approximation algorithm for the column subset selection problem. In *Proceedings of the 19th Annual ACM -SIAM Symposium on Discrete Algorithms (SODA)*, 2009.
- [4] T. F. Chan and P. C. Hansen. Some applications of the rank revealing QR factorization. *SIAM Journal on Scientific and Statistical Computing*, 13:727–741, 1992.
- [5] P. Drineas, R. Kannan, and M. Mahoney. Fast Monte Carlo algorithms for matrices I: Approximating matrix multiplication. *SIAM Journal of Computing*, 36(1):132–157, 2006.
- [6] G.H. Golub, P.C. Hansen, and D. O’Leary. Tikhonov regularization and total least squares. *SIAM Journal on Matrix Analysis and Applications*, 21(1):185–194, 2000.
- [7] M. Gu and S. Eisenstat. Efficient algorithms for computing a strong rank-revealing QR factorization. *SIAM Journal on Scientific Computing*, 17:848–869, 1996.
- [8] P.C. Hansen. The truncated svd as a method for regularization. *BIT Numerical Mathematics*, 27(4):534–553, 1987.
- [9] M. Rudelson and R. Vershynin. Sampling from large matrices: An approach through geometric functional analysis. *Journal of the ACM (JACM)*, 54, 2007.
- [10] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society*, pages 267–288, 1996.