Fast Computation of Low Rank Matrix Approximations

Dimitris Achlioptas
Department of Computer Science
University of California at Santa Cruz
Santa Cruz, CA 95060
optas@cs.ucsc.edu

Frank McSherry
Microsoft Research
1065 La Avenida
Mountain View, CA 94043
mcsherry@microsoft.com

Abstract

Given a matrix A, it is often desirable to find a good approximation to A that has low rank. We introduce a simple technique for accelerating the computation of such approximations when A has strong spectral features, i.e., when the singular values of interest are significantly greater than those of a random matrix with size and entries similar to A. Our technique amounts to independently sampling and/or quantizing the entries of A, thus speeding up computation by reducing the number of non-zero entries and/or the length of their representation. Our analysis is based on observing that the acts of sampling and quantization can be viewed as adding a random matrix N to A, whose entries are independent random variables with zero-mean and bounded variance. Since, with high probability, N has very weak spectral features, we can prove that the effect of sampling and quantization nearly vanishes when a low rank approximation to $A + N$ is computed. We give high probability bounds on the quality of our approximation both in the Frobenius and the 2-norm.

1 Introduction

It is a fundamental result of linear algebra that for any matrix A and positive integer k, there exists a matrix which simultaneously minimizes $\|A - D\|$ over rank k matrices D, for all norms that are invariant under rotation, e.g., the Frobenius norm and the 2-norm. The existence of such an optimal rank k approximation, denoted by A_k, and its efficient computation, follow from the *Singular Value Decomposition* of A, a manner of writing A as a sum of decreasingly significant rank one matrices[1].

Long in the purview of numerical analysts, low rank approximations have recently gained broad popularity in computer science. For example, in areas such as computer vision, information retrieval, and machine learning they are used as a basic tool for extracting correlations and removing noise from matrix-structured data. However, applications of this technique to massive matrices, such as those arising from web corpora and extended video sequences, quickly run against practical computational limits. Specifically, orthogonal iteration and Lanczos iteration, the two most common algorithms for computing optimal low rank approximations, operate through repeated matrix-vector multiplication, thereby requiring superlinear time and large working sets. (For an excellent discussion of these approaches see [10].)

[1] In general, there can be more than one rank k matrix minimizing $\|A - D\|$. The SVD reveals all of these minimizers and we will use the notation A_k to refer to any one of them.

One approach to overcoming the limitations associated with computing optimal low rank approximations is to consider *near-optimal* low rank approximations. Indeed, in many applications, it is reasonable to substitute for A_k some other rank k matrix C that satisfies

$$\|A - C\| \quad \leq \quad \|A - A_k\| + \delta \ ,$$

where δ represents a tolerable level of error for the given application. This approach has gained increasing popularity in recent years and we describe some of the related work [8, 6, 7] in Section 1.2.

Our main contribution is a simple and efficient method for computing near-optimal rank k approximations using orthogonal iteration or Lanczos iteration: given an input matrix A, randomly sparsify and/or quantize it to get a matrix $\widehat{A}$; compute $\widehat{A}_k$ using either method, and return it as the rank k approximation to A. Sparsifying A speeds up the aforementioned matrix-vector multiplications by decreasing the number of required arithmetic operations, while quantizing the entries allows each such operation to be performed faster. Moreover, both sparsification and quantization reduce the memory footprint of the working matrix, something that can have dramatic performance implications by itself. Naturally, given A, k, and $\delta > 0$ we need to be able to produce $\widehat{A}$ quickly and guarantee that $\|A - \widehat{A}_k\| \leq \|A - A_k\| + \delta$. Our main ally in achieving both of these goals will be randomization, as suggested by the following thought experiment.

Imagine that we squander our error allotment δ by obliviously adding to A a matrix G whose entries are independent Gaussian random variables with mean 0 and standard deviation σ. While this process is unlikely to yield a computationally advantageous matrix, we will see that at least it is rather harmless. That is, as long as σ is not too big, the optimal rank k approximation to $\widehat{A} = A + G$ will approximate A nearly as well as A_k. This stability of low rank approximations with respect to Gaussian noise is well-understood and stems from the fact that no low dimensional subspace accommodates G well, i.e., $\|G_k\|$ is small for small k. Indeed, low rank approximations are frequently used with the explicit purpose of removing Gaussian noise.

A fundamental result in random matrix theory is that the G_{ij} being Gaussian is *not* essential in the above example. Rather, G is innocuous by virtue of the following three properties of its entries: independence, mean zero, and small variance. If N is a random matrix whose entries N_{ij} are zero-mean, independent random variables with variance bounded by σ^2, then $\|N_k\| \sim \|G_k\|$. As we will prove, the quantity $\|N_k\|$ bounds the influence that N may exert over the optimal rank k approximation to $A+N$. Specifically, to the extent that $\|A_k\| \gg \|N_k\|$, the matrix $(A+N)_k$ will be largely determined by A.

Our contribution is to exploit the above phenomenon for computational gain. We do this by designing random matrices N that *depend on the input matrix* A, yet whose entries N_{ij} still enjoy independence, mean zero, and small variance. In particular, we will be able to choose N so that $\widehat{A} = A + N$ has computationally useful properties, such as sparsity, while being sufficiently random for $\|N_k\|$ to be small with high probability.

Example: Set $N_{ij} = \pm A_{ij}$ with equal probability, independently for all i, j. Observe that $\mathbf{E}[N_{ij}] = 0$ and that $\widehat{A} = A + N$ has about half as many non-zero entries as A.

As we will see shortly, the above example can be generalized so that we keep any desired fraction $p > 0$ of entries with the error δ growing as $1/\sqrt{p}$. This sparsification will be further refined so that the probability of retaining an entry A_{ij} is proportional to A_{ij}^2. This focuses attention on the significant entries of A and yields greater sparsification when entry magnitudes vary, without increasing the error. We will also see that N can also be designed to yield quantized entries, e.g., to ensure that all entries of $\widehat{A}$ belong in $\{-1, +1\}$.

1.1 Statement of Results

We begin by recalling the definitions of the Frobenius norm and of the 2-norm,

$$\|M\|_F \;=\; \left(\sum_{i,j} M_{ij}^2\right)^{1/2} \qquad \text{and} \qquad \|M\|_2 \;=\; \max_{\|x\|=1} \|Mx\| \;.$$

It is worth noting that for all matrices M and any k, $\|M_k\|_F \leq \sqrt{k}\|M\|_2$, and $\|M_k\|_2 = \|M\|_2$.

We next state a lemma formalizing our earlier notion that perturbation matrices which are poorly approximable in k dimensions have little influence on the optimal rank k approximation.

Lemma 1. *Let A and N be any matrices and write $\widehat{A} = A + N$. Then*

$$\|A - \widehat{A}_k\|_2 \;\leq\; \|A - A_k\|_2 + 2\|N_k\|_2 \quad \textit{and}$$

$$\|A - \widehat{A}_k\|_F \;\leq\; \|A - A_k\|_F + \|N_k\|_F + 2\sqrt{\|N_k\|_F\|A_k\|_F} \;.$$

Notice that all error terms above scale with $\|N_k\|$ and thus whenever N is poorly approximated in k dimensions, i.e., $\|N_k\|$ is small, the error caused by adding N to A must be small.

Equipped with Lemma 1, let us revisit our motivating example of a Gaussian perturbation matrix. This will provide a sense of scale for our results, stated in Theorems 1–3 below.

Fact 1. *Let G be an $m \times n$ matrix, where $m \leq n$, whose entries are independent Gaussian random variables with mean 0 and variance σ^2. With probability $1 - e^{-\Theta(n)}$,*

$$\|G_k\|_2 \;\leq\; 4\,\sigma\sqrt{n} \quad \textit{and} \quad \|G_k\|_F \;\leq\; 4\,\sigma\sqrt{kn} \;.$$

To put these two bounds in perspective consider the trivial rank k approximation, D, that results from zeroing-out all but the first k rows of G. With high probability we have $\|D\|_F \approx \sigma\sqrt{kn}$. Moreover, since D has rank at most k, $\|D\|_2 \geq \|D\|_F/\sqrt{k}$. Fact 1 asserts that the optimal rank k approximation of G only improves upon this trivial approximation by at most a factor of 4, attesting to the near-orthogonality of the rows of G. In contrast, for general $m \times n$ matrices A with $|A_{ij}| = \sigma$, $\|A_k\|$ can easily be as large as $\sigma\sqrt{mn}$, in either norm.

Our main results, Theorems 1, 2, and 3, assert that the effect of random quantization and random sparsification is qualitatively the same as that of adding Gaussian noise. In stating each result, we use the value $b = \max_{ij} |A_{ij}|$ to represent the scale of the entries in the matrix, roughly corresponding to the standard deviation σ used above. Also, to simplify notation, all theorems are stated for matrices where $m \leq n$.

Our first result asserts that it is possible to find a good low rank approximation to A even after randomly quantizing its entries. In Theorem 1 below we quantize each entry to a single bit, representing a 32 to 64 factor of compression over standard floating point numbers. Naturally, one can generalize the quantization process to a larger set of numbers, trading representation length for error.

Theorem 1. *Let A be any $m \times n$ matrix where $m \leq n$, and let $b = \max_{ij} |A_{ij}|$. Let $\widehat{A}$ be a random $m \times n$ matrix whose entries are independently distributed as*

$$\widehat{A}_{ij} \quad = \quad \begin{cases} +b & \textit{with probability } \dfrac{1}{2} + \dfrac{A_{ij}}{2b} \ , \\ \\ -b & \textit{with probability } \dfrac{1}{2} - \dfrac{A_{ij}}{2b} \ . \end{cases} \tag{1}$$

For all sufficiently large n, with probability at least $1 - \exp(-19(\log n)^4)$, the matrix $N = A - \widehat{A}$ satisfies

$$\|N_k\|_2 \ < \ 4\,b\sqrt{n} \qquad \textit{and} \qquad \|N_k\|_F \ < \ 4\,b\sqrt{kn} \ .$$

Our second result asserts that it is possible to find a good low rank approximation to A even after randomly omitting many of its entries. In particular, the stronger the spectral features of A the more of its entries we can afford to omit.

Theorem 2. *Let A be any $m \times n$ matrix where $76 \leq m \leq n$, and let $b = \max_{ij} |A_{ij}|$. For $p \geq (8 \log n)^4/n$, let $\widehat{A}$ be a random $m \times n$ matrix whose entries are independently distributed as*

$$\widehat{A}_{ij} \quad = \quad \begin{cases} A_{ij}/p & \textit{with probability } p, \\ \\ 0 & \textit{otherwise.} \end{cases} \tag{2}$$

With probability at least $1 - \exp(-19(\log n)^4)$, the matrix $N = A - \widehat{A}$ satisfies

$$\|N_k\|_2 \ < \ 4\,b\sqrt{n/p} \qquad \textit{and} \qquad \|N_k\|_F \ < \ 4\,b\sqrt{kn/p} \ .$$

As mentioned earlier, we can improve upon the uniform sparsification process by retaining entries with probability that depends on their magnitude. This yields greater sparsification when entry magnitudes vary, without increasing the error bounds.

Theorem 3. *Let A be any $m \times n$ matrix where $76 \leq m \leq n$, and let $b = \max_{ij} |A_{ij}|$. For any $p > 0$, define $\tau_{ij} = p(A_{ij}/b)^2$ and let*

$$p_{ij} = \max\left\{\tau_{ij} \ , \ \sqrt{\tau_{ij} \times (8 \log n)^4/n}\right\} \ .$$

Let $\widehat{A}$ be a random $m \times n$ matrix whose entries are independently distributed as

$$\widehat{A}_{ij} \quad = \quad \begin{cases} A_{ij}/p_{ij} & \textit{with probability } p_{ij}, \\ \\ 0 & \textit{otherwise .} \end{cases} \tag{3}$$

1. *With probability at least $1 - \exp(-19(\log n)^4)$, the matrix $N = A - \widehat{A}$ satisfies*

$$\|N_k\|_2 \ < \ 4\,b\sqrt{n/p} \quad \textit{and} \quad \|N_k\|_F \ < \ 4\,b\sqrt{kn/p} \ .$$

2. *The expected number of non-zero entries in $\widehat{A}$ is bounded by $(pmn) \times \mathrm{Avg}[(A_{ij}/b)^2] + m(8 \log n)^4$.*

Remark 1. *In Section 4.1 we describe an algorithm which given n and any $s > 0$ produces a matrix $\widehat{A}$ from the distribution of Theorem 3 with $p = sb^2/\|A\|_F^2$ in one pass over A. As a result, the expected number of non-zero entries in $\widehat{A}$ is bounded by $s + m(8 \log n)^4$.*

Intuitively, our results are useful for k such that A_k has spectral features, i.e., singular values, stronger than those in a random matrix of comparable scale. The presence of such notable spectral features is the typical reason for considering low rank approximations, as they imply the existence of notable linear correlations. We note that there are several occasions when one might seek particularly *weak* correlations. Vectors in the null space of a matrix, for example, can make good starting points for quadratic minimization. Our results are not useful in these domains.

1.2 Related Work

In a pioneering paper [11], Papadimitriou et al. introduced and analyzed a generative model for text corpora (represented as term-document matrices), in order to give a mathematical explanation for the success of low rank approximations in the information retrieval domain [5, 4, 3]. Azar et al. [2], extended the model of [11] to arbitrary matrices that result from adding a random noise matrix N to a data matrix A. (The idea being that A represents perfect but unobservable data, while N models the vagaries of the measurement process.) The key observation in [2] is that if A has strong spectral features, then computing a low rank approximation to $A+N$ allows one to largely recover the original matrix A. This denoising occurs precisely because A has strong spectral features, while N does not. Our work is inspired by the ideas in [2] and our results can be viewed as turning the random noise process acting on the data on its head: rather than viewing the random matrix N as a foe corrupting the data, we co-opt the random process and choose a distribution for N so that with high probability $A+N$ has computationally useful properties.

The first mathematically rigorous approach to speeding up the computation of low rank approximations by employing randomization was given in a seminal paper by Frieze, Kannan, and Vempala [8], where it was shown that one can efficiently approximate A from a submatrix whose size is independent of m, n. Specifically, the authors show how to compute a rank k matrix C such that $\|A-C\|_F^2 \leq \|A-A_k\|_F^2 + \epsilon\|A\|_F^2$ from a square submatrix of A having dimension $d \geq 10^7 k^4/\epsilon^3$. (In [8] an analysis is given for probability of failure 1/10, resulting in the explicit constant 10^7 in the lower bound for d). The approach of [8] was refined and extended in subsequent works of Drineas et al. [6] and Drineas and Kannan [7] motivated by practical considerations. Specifically, by sampling entire columns of A, the algorithms in [6, 7] greatly improve the dependence of d on k, ϵ at the cost of introducing a linear dependence of d on $m \leq n$.

The main conceptual difference between the approach of [8, 6, 7] and our sampling process is that [8, 6, 7] sample random submatrices of A, whereas we independently sample individual *entries* of A. By breaking the correlation between elements of the same row/column in the sampling process, we immediately gain access to a wealth of results from the theory of random matrices. These enable the derivation of the very sharp matrix perturbation bounds underlying our work. Moreover, this entry-wise independence allows us to perform the sampling in a single pass over the matrix. In contrast, the correlation among rows/columns requires the algorithms in [8, 6, 7] to make two passes over the matrix A: the first to sample a set of indices, and the second to collect the contents of the corresponding submatrix. Such a second pass can be impractical or even impossible for massive data sets, where one often has only streaming access to the data.

1.3 Paper Outline

We begin by proving Lemma 1 in Section 2, and then derive a bound on the 2-norm of random matrices in Section 3. Combined, these two results establish both the Frobenius and the 2-norm bounds of Theorems 1 and 2. In Section 4 we first discuss how to use non-uniform sampling to achieve further sparsification, yielding Theorem 3, and then address Remark 1 with an algorithm for adaptive non-uniform sampling. In Section 5 we compare the bounds of our non-uniform sampling approach with those of [6, 7]. In Section 6, we consider several modifications and enhancements to our algorithms motivated by practical concerns. Finally, in Section 7 we experimentally evaluate our algorithms along with various others on data sets drawn from machine learning and information retrieval domains.

2 Proof of Lemma 1

We now prove two lemmas relating $\|A - B_k\|$ to $\|A - A_k\|$ for arbitrary matrices A, B, in the Frobenius and 2-norm. Specifically, Lemma 2 compares $\|A - B_k\|_2$ to $\|A - A_k\|_2$, while Lemma 4 compares $\|A - B_k\|_F$ to $\|A - A_k\|_F$.

Lemma 2. *For any matrices A and B*

$$\|A - B_k\|_2 \quad \leq \quad \|A - A_k\|_2 + 2\|(A - B)_k\|_2 \ .$$

Proof. Starting with $\|A - B_k\|_2$ and applying the triangle inequality we get (4). Using that for any rank k matrix D, $\|B - B_k\|_2 \leq \|B - D\|_2$ we get (5). Applying the triangle inequality again gives (6).

$$\begin{aligned} \|A - B_k\|_2 \quad &\leq \quad \|A - B\|_2 + \|B - B_k\|_2 && (4) \\ &\leq \quad \|A - B\|_2 + \|B - A_k\|_2 && (5) \\ &\leq \quad \|A - B\|_2 + \|B - A\|_2 + \|A - A_k\|_2 \ . && (6) \end{aligned}$$

Finally, we note that $\|B - A\|_2 = \|A - B\|_2 = \|(A - B)_k\|_2$, which concludes the proof. □

In order to prove the Frobenius norm bound we need to introduce the following notion. Given a matrix M, let P_M denote the projection matrix onto the space spanned by the columns of M_k (we suppress the dependence of P_M on k to simplify notation). An important consequence of the Singular Value Decomposition is that $M_k = P_M M$ and, as a result, that for any matrices A and B,

$$\|P_A A\|_F \quad \geq \quad \|P_B A\|_F \ . \tag{7}$$

To prove our stated bounds for the Frobenius norm we will first show that for any matrices A, B, if $\|(A - B)_k\|_F$ is small, then projecting A onto P_B is almost as good as projecting it onto P_A in terms of capturing Frobenius norm.

Lemma 3. *For any matrices A and B*

$$\|P_B A\|_F \quad \geq \quad \|P_A A\|_F - 2\|(A - B)_k\|_F \ .$$

Proof. Starting with $\|P_B A\|_F$ and applying the triangle inequality we get (8). Applying (7) yields (9). Applying the triangle inequality again gives (10).

$$\begin{aligned} \|P_B A\|_F \quad &\geq \quad \|P_B B\|_F - \|P_B(A - B)\|_F && (8) \\ &\geq \quad \|P_A B\|_F - \|P_B(A - B)\|_F && (9) \\ &\geq \quad \|P_A A\|_F - \|P_A(B - A)\|_F - \|P_B(A - B)\|_F \ . && (10) \end{aligned}$$

Finally, we apply (7) to bound the $\|P_X(A - B)\|_F$ terms in (10) by $\|(A - B)_k\|_F$. □

We now use Lemma 3 to prove that if $\|(A-B)_k\|_F$ is small, then $\|A-B_k\|_F$ is not much larger than $\|A-A_k\|_F$. In other words, B_k can be a good surrogate for A_k with respect to the Frobenius norm even when $\|A-B\|_F$ is large, so long as $\|(A-B)_k\|_F$ is small.

Lemma 4. *For any matrices A and B,*

$$\|A-B_k\|_F \quad \leq \quad \|A-A_k\|_F + 2\sqrt{\|(A-B)_k\|_F\|A_k\|_F} + \|(A-B)_k\|_F \ .$$

Proof. By the fact $P_B B = B_k$ and the triangle inequality we get

$$\|A-B_k\|_F \quad \leq \quad \|A-P_B A\|_F + \|P_B(A-B)\|_F \ . \tag{11}$$

Applying the Pythagorean equality to each column of A implies that for any projection matrix P_B,

$$\|A-P_B A\|_F^2 \quad = \quad \|A\|_F^2 - \|P_B A\|_F^2 \ . \tag{12}$$

Inserting (12) in (11), we get

$$\|A-B_k\|_F \quad \leq \quad (\|A\|_F^2 - \|P_B A\|_F^2)^{1/2} + \|P_B(A-B)\|_F \ . \tag{13}$$

To bound the right hand side of (13) we first invoke the lower bound for $\|P_B A\|_F$ provided by Lemma 3 to get (14). We then use (12) to get (15) to which we apply the inequality $\sqrt{a+b} \leq \sqrt{a}+\sqrt{b}$ to get (16).

$$\begin{aligned}
\|A-B_k\|_F &\leq \left(\|A\|_F^2 - \|P_A A\|_F^2 + 4\,\|(A-B)_k\|_F\,\|P_A A\|_F\right)^{1/2} + \|P_B(A-B)\|_F & (14)\\
&= \left(\|A-A_k\|_F^2 + 4\,\|(A-B)_k\|_F\,\|A_k\|_F\right)^{1/2} + \|P_B(A-B)\|_F & (15)\\
&\leq \|A-A_k\|_F + (4\,\|(A-B)_k\|_F\,\|A_k\|_F)^{1/2} + \|P_B(A-B)\|_F & (16)
\end{aligned}$$

To wrap up, we use (7) again to bound $\|P_B(A-B)\|_F$ by $\|(A-B)_k\|_F$. □

3 Random Matrices

Our methods are based on the fact that if B is a matrix whose entries are independent random variables, then with high probability the spectrum of B will be close to the spectrum of $\mathbf{E}[B]$. In particular, the matrix $B-\mathbf{E}[B]$ with high probability will have small 2-norm. To understand why this is so, observe that each row of $N = B - \mathbf{E}[B]$ is a vector of zero-mean, independent random variables, so that the inner product of any two rows is tightly concentrated around its expectation, i.e., 0. In other words, the rows of N are effectively orthogonal making it impossible for a single vector to have non-trivial inner product with many of them simultaneously.

Theorem 4 formalizes this notion by combining a very recent improvement by Vu [12] of the main result in Füredi and Komlós [9] to bound Median($\|N\|_2$), with a concentration result for $\|N\|_2$ by Alon, Krivelevich, and Vu [1], based on Talagrand's inequality.

Theorem 4. *Given any $m \times n$ matrix A with $m \leq n$ and any fixed $\epsilon > 0$, let $\widehat{A}$ be a random matrix whose entries are independent random variables such that for all i, j: $\mathbf{E}[\widehat{A}_{ij}] = A_{ij}$, $\mathrm{Var}(\widehat{A}_{ij}) \leq \sigma^2$, and $\widehat{A}_{ij}$ takes values in an interval of length K, where*

$$K \quad = \quad \left(\frac{\log(1+\epsilon)}{2\log(m+n)}\right)^2 \times \sigma\sqrt{m+n} \ .$$

For any $\theta > 0$ and $m+n \geq 152$,

$$\Pr\left[\|A-\widehat{A}\|_2 \geq 2(1+\epsilon+\theta)\sigma\sqrt{m+n}\right] \ < \ 2\exp\left(-\frac{16\theta^2}{\epsilon^4}(\log n)^4\right) \ .$$

Proof. Let $d = m+n$ and observe that $\|A-\widehat{A}\|_2 = \|M\|_2$, where M is the $d \times d$ symmetric matrix

$$M \quad = \quad \left[\begin{array}{cc} 0 & (A-\widehat{A})^T \\ (A-\widehat{A}) & 0 \end{array}\right] \ .$$

Recall also that for every symmetric matrix B and even positive integer t, $\|B\|_2 \leq \mathrm{Trace}(B^t)^{1/t}$.

Clearly, for $i \leq j$, $\{M_{ij}\}$ are independent random variables with $\mathbf{E}[M_{ij}] = 0$, $\mathrm{Var}(M_{ij}) \leq \sigma^2$ and $|M_{ij}| \leq K$. In [9] it was shown that for any such matrix M and all even $t \leq t_0 = \frac{1}{\sqrt{2}}\left(\sigma/K\right)^{1/2} d^{1/4}$,

$$\mathbf{E}\left[\mathrm{Trace}(M^t)\right] < 4d\left(2\sigma\sqrt{d}\right)^t \ . \tag{17}$$

Since for any non-negative random variable X, $\mathrm{Median}(X) \leq 2\mathbf{E}[X]$ and $\mathrm{Median}(X^t) = \mathrm{Median}(X)^t$, we see that for every even $t \leq t_0$,

$$\mathrm{Median}\left(\|M\|_2\right) \leq \mathrm{Median}\left(\mathrm{Trace}(M^t)^{1/t}\right) = \left(\mathrm{Median}\left(\mathrm{Trace}(M^t)\right)\right)^{1/t} \leq \left(2\mathbf{E}[\mathrm{Trace}(M^t)]\right)^{1/t} \ .$$

Setting $t = t_0$ in (17) we thus get $\mathrm{Median}\left(\|M\|_2\right) < 2\sigma\sqrt{d} \times (8d)^{1/t_0}$. Requiring $(8d)^{1/t_0} < 1+\epsilon$ is equivalent to $K < \frac{1}{2}\left(\frac{\log(1+\epsilon)}{\log(8d)}\right)^2 \sigma\sqrt{d}$, which for all $d \geq 152$ is implied by

$$K < \left(\frac{\log(1+\epsilon)}{2\log d}\right)^2 \sigma\sqrt{d} \ .$$

To get a large deviations inequality we adapt Theorem 1 of [1] to non-symmetric matrices noting that in our context their bound on the probability that $\|A-\widehat{A}\|_2$ exceeds its median value by tK can be sharpened to $2\exp(-t^2/4)$. Taking $tK = 2\theta\sigma\sqrt{d}$ we thus get that for all $d \geq 152$,

$$\begin{aligned}
\Pr\left[\|A-\widehat{A}\|_2 > 2(1+\epsilon+\theta)\sigma\sqrt{m+n}\right] \quad &< \quad 2\exp\left(-\frac{\theta^2\sigma^2 d}{K^2}\right) \\
&\leq \quad 2\exp\left(-\theta^2\left(\frac{2\log d}{\log(1+\epsilon)}\right)^4\right) \\
&< \quad 2\exp\left(-\frac{16\theta^2}{\epsilon^4}(\log n)^4\right) \ .
\end{aligned}$$

□

To prove Theorems 1 and 2 we first bound $\|N\|_2$ by applying Theorem 4 with $\epsilon = 3/10$, $\theta = 1/10$. The equivalence $\|N_k\|_2 = \|N\|_2$ yields the 2-norm bounds, while the Frobenius bounds follow from the inequality $\|N_k\|_F \leq \sqrt{k}\|N_k\|_2$. Specifically, using $\sqrt{m+n} \leq \sqrt{2n}$ and compiling the constants we get $2(1 + 3/10 + 1/10)\sqrt{2} = 3.95... < 4$, while $2\exp\left(-\frac{16\theta^2(\log n)^4}{\epsilon^4}\right) < \exp(-19(\log n)^4)$.

The range constraints for the entries of $\widehat{A}$ give the remaining conditions in the theorems. Recall that $K = \left(\frac{\log(1+\epsilon)}{2\log(m+n)}\right)^2 \sigma\sqrt{m+n}$. For Theorem 1, instantiating ϵ and requiring $2b \leq K$ imposes the lower bound $m + n \geq 3.07.. \times 10^9$, which we write as $n \geq n_0$ since $n \geq m$. The lower bound on p in Theorem 2 is derived likewise and amounts to $p \geq \left(\frac{2\log(m+n)}{\log(1+\epsilon)}\right)^4 \frac{1}{m+n}$ which, by instantiating ϵ and requiring $m \geq 17$, we simplify to $p \geq (8\log n)^4/n$. The proof of Theorem 3 is effectively identical to that of Theorem 2 but we defer it until we discuss the choice of p_{ij} in Section 4.

It should be noted that while conditions such as $n \geq n_0$ and $p \geq (8\log n)^4/n$ require n to be unrealistically large, they are only the result of choosing ϵ small enough to reflect the limiting behavior of $\|N_k\|$. We will see in Section 7 that even for matrices of modest size, e.g., $m+n = 1000$, our approaches yield useful approximations.

4 Non-Uniform Sampling

When we sparsify A by keeping every entry with the same probability p, we see that for every i, j

$$\mathrm{Var}\left(\widehat{A}_{ij}\right) = \frac{1-p}{p} A_{ij}^2 .$$

That is, small entries of A give rise to random variables with comparatively small variance in $\widehat{A}$. Nevertheless, our bound for $\|A - \widehat{A}\|_2$ via Theorem 4 relies only on the *maximum* variance of the entries in $\widehat{A}$. This suggests decreasing the probability of keeping each entry A_{ij} to some $p_{ij} \leq p$, so that entries in $\widehat{A}$ have essentially the same variance. This will increase the number of omitted entries when entry magnitudes vary without affecting our bound on $\|A - \widehat{A}\|_2$.

Concretely, if we let $\widehat{A}_{ij} = A_{ij}/p_{ij}$ with probability p_{ij} and 0 otherwise, then $\mathbf{E}[\widehat{A}_{ij}] = A_{ij}$ and $\mathrm{Var}(\widehat{A}_{ij}) = A_{ij}^2(1/p_{ij} - 1)$. In particular, if we take $p_{ij} = p\,(A_{ij}/b)^2$, where $p \in (0, 1]$ and $b = \max_{ij} |A_{ij}|$, then $\mathrm{Var}(\widehat{A}_{ij}) = b^2/p - A_{ij}^2 \approx b^2/p$ for all i, j. With this choice of p_{ij} the expected number of retained entries is $\sum_{i,j} p(A_{ij}/b)^2 = p\|A\|_F^2/b^2$. In other words, we retain $(pmn) \times \mathrm{Avg}[(A_{ij}/b)^2]$ entries in expectation, compared with pmn entries for uniform sampling.

One technical point we need to address in performing non-uniform sampling is to respect the range constraint of Theorem 4. Note that if the probability of keeping entry (i, j) is $p(A_{ij}/b)^2 \equiv \tau_{ij}$, then in the unlikely event that a very small entry A_{ij} is kept, the resulting entry $\widehat{A}_{ij} = A_{ij}/\tau_{ij} = b^2/(pA_{ij})$ will be very large and may violate the range constraint. To address this issue it suffices to slightly increase the probability associated with very small entries so that for those elements A_{ij} below a certain threshold we take $p_{ij} \propto |A_{ij}|$, rather than $p_{ij} \propto A_{ij}^2$. Specifically, if

$$p_{ij} = \max\left\{\tau_{ij}\ ,\ \sqrt{\tau_{ij} \times (8\log n)^4/n}\right\} , \tag{18}$$

then the range constraints are never violated. Notice that $p_{ij} \leq \tau_{ij} + (8\log n)^4/n$ for all i, j, and therefore using p_{ij} in place of τ_{ij} adds no more than $mn \times (8\log n)^4/n$ elements to $\widehat{A}$ in expectation. Theorem 3 now follows from Theorem 4, by taking p_{ij} as defined in (18).

4.1 Adaptive Non-Uniform Sampling in a Single Pass

Given n, b and a fixed sparsification value p, it is easy to perform non-uniform sampling in a single pass over A as for every observed entry A_{ij} we can readily compute and use the correct probability p_{ij} from (18). Such sampling will, in expectation, yield at most $p\|A\|_F^2/b^2 + m(8\log n)^4$ entries. Unfortunately, the first term in this expression represents an amount of data unknown at the outset, which may be more than we can afford to keep or less than we would like to keep. To make this term equal to some predetermined number s we need to chose $p = sb^2/\|A\|_F^2$, but unfortunately we only arrive at the correct values of b and $\|A\|_F$ after a complete pass through the matrix.

Below we give an algorithm **Sample**(s, n) that properly produces a matrix $\widehat{A}$ according to the distribution of Theorem 3 with $p = sb^2/\|A\|_F^2$, thereby yielding $s + m(8\log n)^4$ non-zero entries in expectation. **Sample** uses standard techniques from adaptive sampling, exploiting the fact that the probabilities p_{ij} only decrease with each additional observed element, since $\|A\|_F^2$ only grows.

```
Sample$(s, n)$

1. Let $Q$ be an empty priority queue and let $Z = 0$.
2. For each element $A_{ij}$
    (a) Set $Z \leftarrow Z + A_{ij}^2$.
    (b) Draw $r_{ij}$ uniformly at random from $[0, 1]$.
    (c) Insert $A_{ij}$ in $Q$ with key $k_{ij} = \max\{sA_{ij}^2/r_{ij}\ ,\ sA_{ij}^2/r_{ij}^2 \times (8\log n)^4/n\}$.
    (d) Remove from $Q$ all elements with key smaller than $Z$.
3. Return the contents of $Q$.
```

Lemma 5. *Let A be any $m \times n$ matrix where $76 \leq m \leq n$. For every $s > 0$,* **Sample**(s, n) *yields a matrix $\widehat{A}$ such that:*

1. *With probability at least $1 - \exp(-19(\log n)^4)$, the matrix $N = A - \widehat{A}$ satisfies*

$$\|N_k\|_2 \ \leq\ 4\sqrt{n/s} \times \|A\|_F \quad \text{and} \quad \|N_k\|_F \ \leq\ 4\sqrt{kn/s} \times \|A\|_F \ . \tag{19}$$

2. *The expected number of non-zero entries in $\widehat{A}$ is bounded by $s + m(8\log n)^4$.*

Proof. Let $p = sb^2/\|A\|_F^2$ and define $\tau_{ij} = p(A_{ij}/b)^2 = s(A_{ij}/\|A\|_F)^2$. By Theorem 3, it suffices to prove that each entry A_{ij} is kept by **Sample** independently, with probability

$$p_{ij} \quad = \quad \max\left\{\tau_{ij}\ ,\ \sqrt{\tau_{ij} \times (8\log n)^4/n}\right\} \ . \tag{20}$$

We observe that an element A_{ij} is in Q when **Sample** terminates if and only if

$$sA_{ij}^2/r_{ij} \ \geq\ \|A\|_F^2 \quad \text{or} \quad sA_{ij}^2/r_{ij}^2 \times (8\log n)^4/n \ \geq\ \|A\|_F^2 \ . \tag{21}$$

But (21) is equivalent to $r_{ij} \leq p_{ij}$ and each r_{ij} is chosen uniformly and independently in $[0, 1]$. □

5 Comparison with Related Work

We focus our comparison to the work of Drineas et al. [6, 7], as it gives the best known bounds for computing near-optimal low rank matrix approximations by column/row sampling. The approach in [6, 7] is fairly simple to describe: i) sample c columns from A, selecting each column with probability proportional to its squared 2-norm, ii) determine the optimal k-dimensional subspace for the sample, and iii) project the original matrix A onto this subspace to get a rank k matrix D. Intuitively, as the number of columns sampled grows, the sample approaches the distribution of columns of A. In particular, [6, 7] show that if $c = O(1/\epsilon^2)$ columns are drawn, then with constant probability the resulting rank k matrix D satisfies

$$\|A - D\|_2^2 \quad \leq \quad \|A - A_k\|_2^2 + \epsilon \, \|A\|_F^2 \tag{22}$$

$$\|A - D\|_F^2 \quad \leq \quad \|A - A_k\|_F^2 + \epsilon \sqrt{k} \, \|A\|_F^2 \ . \tag{23}$$

More generally, if $c = O((\log z)^2/\epsilon^2)$ columns are drawn then (22),(23) hold with probability $1-1/z$. We note that, in the case of the Frobenius norm, the bound in (23) is meaningful only if $\epsilon\sqrt{k} < 1$.

By comparison, given the same matrix A, invoking **Sample**$(16n/\epsilon^2, n)$ yields an $m \times n$ matrix $\widehat{A}$ with $O(n/\epsilon^2 + m(\log n)^4)$ non-zero entries, sampled from the distribution of Theorem 3 with $p = 16nb^2/(\epsilon\|A\|_F)^2$. Thus, by Lemma 1 and Theorem 3, with probability $1 - \exp(-19(\log n)^4)$,

$$\|A - \widehat{A}_k\|_2 \quad \leq \quad \|A - A_k\|_2 + \epsilon \, \|A\|_F \tag{24}$$

$$\|A - \widehat{A}_k\|_F \quad \leq \quad \|A - A_k\|_F + 3\sqrt{\epsilon}\, k^{1/4} \|A\|_F \ , \tag{25}$$

where in deriving (25) we assumed that $\epsilon\sqrt{k} < 1$ so that the bound in (23) is meaningful.

Unfortunately, making a direct comparison of the two results at this point is hindered by the fact that (22), (23) bound $\|A - D\|^2 - \|A - A_k\|^2$ while (24), (25) bound $\|A - \widehat{A}_k\| - \|A - A_k\|$. If we bound the right hand side of (22), (23) using the inequality $a^2 + b^2 < (a + b)^2$, we see that the Frobenius bounds are comparable, while our 2-norm decays at the square of the rate of [6, 7]. Such a comparison is oversimplified, to be sure: using $a^2 + b^2 < (a + b)^2$ is inappropriate when $\|A - A_k\| \gg \epsilon\|A\|_F$. Also, equating $O(1/\epsilon^2)$ columns of A with $O(n/\epsilon^2 + m(\log n)^4)$ non-zero entries is justified only when each column of A contributes $\Theta(n)$ non-zero entries and n/ϵ^2 is not dominated by $m(\log n)^4$. (We ignore the $O((\log z)^2)$ factor related to the success probability.)

There are two other important considerations regarding the two approaches. As mentioned in Section 1.2, the approach of [6, 7] requires two passes over the data to produce D, while our approach can be implemented in a single pass. In fact, since our methods treat all matrix entries independently, sampling and quantization can be performed before organizing the data into matrix form, which means that these operations can be performed directly by the data collection agents. On the other hand, by sampling c columns, [6, 7] reduce the eigenproblem at hand to that of a $m \times c$ matrix, which is more casually solved than an eigenproblem of an $m \times n$ matrix having an equal number of non-zero entries.

6 Practical Considerations and Modifications

In this section we discuss a few practically motivated issues regarding our approach, and suggest corresponding modifications to the algorithms of the previous sections.

6.1 Combining Sampling and Quantization

Mathematically, the result of Theorem 1 for quantization and the results of Theorems 2 and 3 for sampling are orthogonal; one could first sample the entries of the input matrix A and then proceed to quantize the remaining non-zero entries. Practically, however, a significant fraction of the representation of a sparse matrix lies in representing its structure, not its entries. The typical representation of a sparse matrix is a list of triples $\langle row, col, val \rangle$, each representing a non-zero entry in the matrix. Compressing *val* to one bit results in relatively minor compression when we still have to retain *row* and *col*.

We can overcome this issue in the case of uniform sampling by exploiting the fact that in practice the sparsity structure of $\widehat{A}$ will be determined by a pseudorandom number generator. As a result, it suffices to store the *seed* of the generator in order to reconstruct the sparsity structure of $\widehat{A}$. Specifically, given A we use the generator to produce a series of indices $i, j \in [m] \times [n]$ corresponding to the kept locations, record the randomly quantized corresponding entries, and record the seed (alternatively, we can generate geometric random variables corresponding to the number of entries skipped). To perform matrix-vector multiplications by $\widehat{A}$ we regenerate the indices of the non-zero entries in $\widehat{A}$ on the fly and retrieve their associated quantized values.

6.2 One-pass vs. Two-pass Algorithms

Our quantization and sparsification techniques suffer from the fact that as k grows, our approximation $\widehat{A}_k$ does not converge to A. In each method, we have permanently discarded data from A that we do not expect to recover. In [8, 6] this problem is addressed by using the sampled matrix only to compute a rank k projection P_D, such that $D = P_D A$ is a good approximation to A. Of course, computing the projection of A onto P_D requires an additional pass over the input matrix, but this can greatly improve $\|A - D\|$ for large values of k.

We can easily adapt our techniques to this approach. From a sparse and/or quantized $\widehat{A}$, we compute and return $P_{\widehat{A}} A$ as our rank k approximation to A. Projection matrices have the property that $\|A - PX\|$ is minimized at $X = A$. Specifically, $\|A - P_{\widehat{A}} A\|$ is never more than $\|A - P_{\widehat{A}} \widehat{A}\| = \|A - \widehat{A}_k\|$, and can be significantly smaller. Indeed, it helps to write $\widehat{A}_k$ as

$$\widehat{A}_k \quad = \quad P_{\widehat{A}} A + P_{\widehat{A}}(\widehat{A} - A) \ .$$

From the above we see that in addition to the optimal approximation to A that lies in the span of $P_{\widehat{A}}$, namely $P_{\widehat{A}} A$, the matrix $\widehat{A}_k$ also contains the projection of the random noise $\widehat{A} - A$ onto $P_{\widehat{A}}$. While we have argued that this term has small norm with high probability, removing it can yield a significantly more accurate approximation, especially for large values of k.

6.3 Computing Optimal Low Rank Approximations

Algorithms such as orthogonal iteration and Lanczos iteration operate through iterative improvement of their approximations. At each step a non-optimal approximation is improved through matrix multiplication. When started from a random initial configuration, as they usually are, both methods spend a good deal of time wending their way through poor approximations, leading to their superlinear running time.

It appears natural to use non-optimal low rank approximation techniques as an initial step for the computation of *optimal* approximations. That is, finding a near-optimal approximation using a small fraction of the entries takes less time than discovering one using the whole matrix. One can generalize the above idea to a scheme that starts off with a highly sparsisfied/quantized version of the input matrix A and in successive iterations uses more and more accurate versions of A. The computational savings come from the fact that until we get quite close to the k dimensional invariant subspace of A, a rough approximation to A can be almost as good in terms of driving convergence. Understanding the behavior of such schemes remains an interesting open problem.

7 Experimental Observations

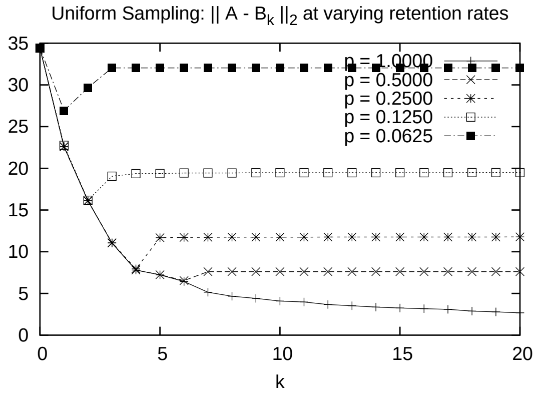
In this section we present several experimental results that shed light on the utility of our techniques when applied to data sets arising in practice. We note that our techniques perform much better than our bounds suggest, especially in the 2-norm. Moreover, we see that our algorithms perform well for m, n dramatically smaller than our theorems require.

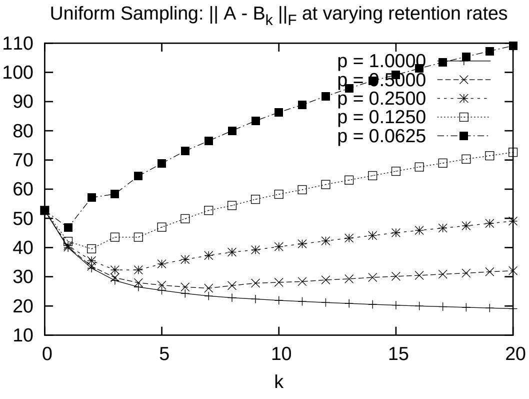
We consider two data sets: the first is a 500×500 dense Gaussian kernel matrix arising in kernel learning, the second is a 9603×22226 sparse document-term matrix drawn from Reuters newsfeed data. Throughout the presentation below we use B_k to denote the rank k matrix produced by each method as its approximation to the input matrix A. We write $\delta_2 \equiv \|A - B_k\|_2 - \|A - A_k\|_2$ for the excess error in the 2-norm and, similarly, $\delta_F \equiv \|A - B_k\|_F - \|A - A_k\|_F$, for the excess error in the Frobenius norm.

7.1 Dense Gaussian Kernel Matrix

The first data set we examine is a 500×500 dense matrix arising in kernel learning. This matrix is generated by drawing a sample of 500 faces from the Yale face database [13], each face represented by a vector $x_i \in \mathbb{R}^d$, and setting $A_{ij} = \exp(-\|x_i - x_j\|^2)$. Note that this matrix has substantial variance in the entry magnitude, ranging from 1.0 entries on the diagonal to many very small entries. It is worth noting that $m = n = 500$ is not even remotely covered by our theorems, so it is reassuring to see good performance in this setting.

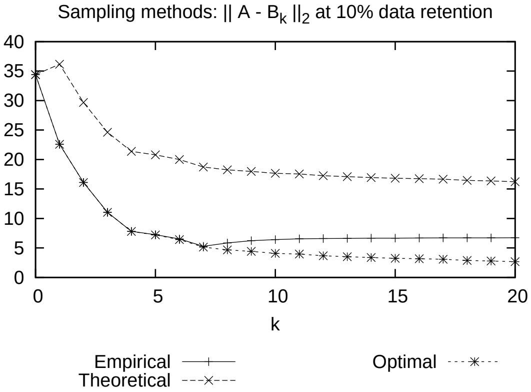
We first examine the performance of uniform sparsification for varying values of p.

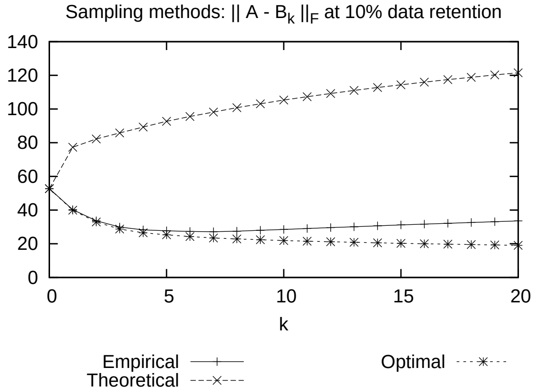




While from Lemma 2 we know that $\|A - B_k\|_2 \leq \|A - A_k\|_2 + 2\|A - B\|$ for all k, experimentally $\|A - B_k\|_2$ appears to exhibit a threshold phenomenon around some critical value k_p for each sparsification level p. Specifically, for $k \leq k_p$, sparsification introduces virtually no excess error, i.e., $\|A - B_k\|_2 \approx \|A - A_k\|_2$, while for $k > k_p$, $\|A - B_k\|_2$ takes the same value for all k. (This value, numerically, appears to be very close to $\|A-B\|_2$.) This behavior can be seen in the Frobenius norm measurements as well, though the behavior is muddled somewhat, due in part to the Frobenius norm's averaging nature. Understanding this remarkably good performance for $k \le k_p$ and the slight non-monotonicity of $\|A - B_k\|_2$ around k_p seems to us an interesting open problem.

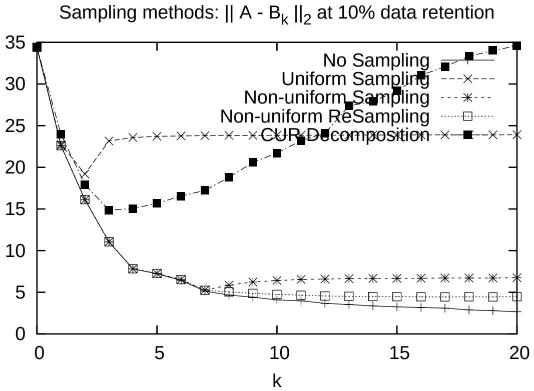
To highlight the comparison between the true empirical error and the theoretical prediction we plot both quantities below when we apply non-uniform sampling with $p = 0.1$, for different values of k. Specifically, we plot the empirical value of $\|A - B_k\|$ vs. the bound for $\|A - B_k\|$ given by Lemma 1 (which gives the sharpest form of our theoretical bounds for both norms). To evaluate the theoretical bound we explicitly computed the spectrum of the matrix $A-B$, allowing us access to the quantities $\|A - B\|_2$ and $\|(A - B)_k\|_F$ which enter in the theoretical bound. For ease of reference, we also plot $\|A - A_k\|$ in the same plot. We refer to the three plots as "empirical", "theoretical" and "optimal", respectively. Note that the very first point in all our charts is the $k = 0$ case of no approximation, which gives no error.

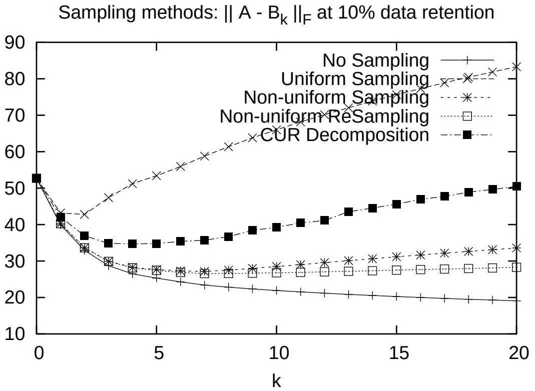




For all of the remaining experiments, we compare several methods at a fixed 10% level of data retention. We broadly classify these methods as "sampling methods" and "projection methods" to indicate that, given a sample of elements in A, the former produce an approximation B_k, whereas the latter produce a projection P_B that is used to compute the approximation $P_B A$.

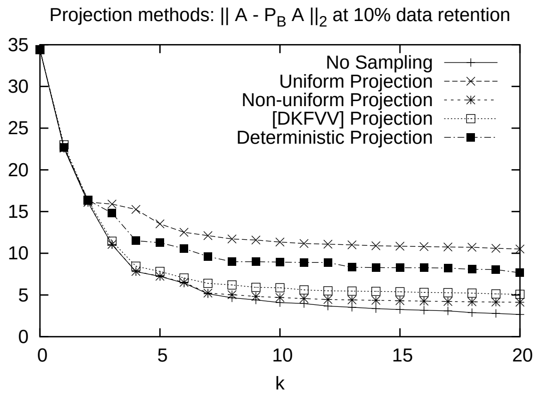
We start by comparing three sampling methods, namely uniform and non-uniform sparsification as well as the CUR decomposition of [7].

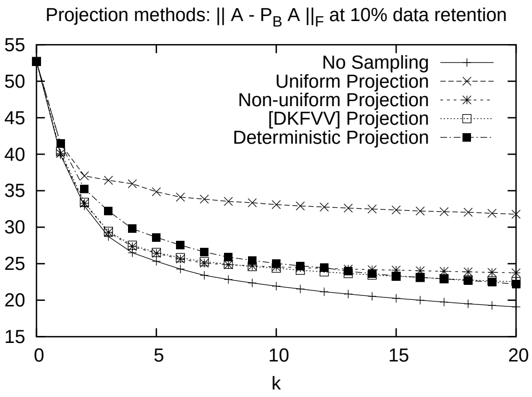




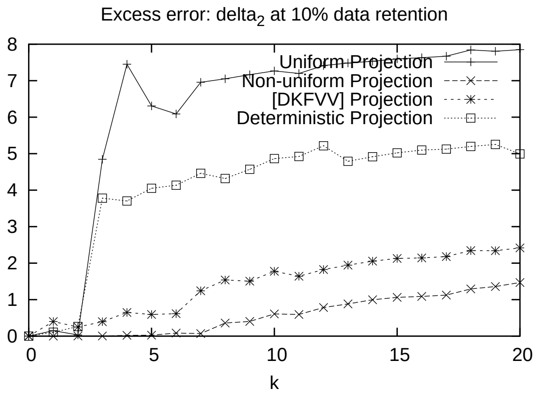
As might be expected, the uniform sampling technique performs quite poorly when compared to techniques that reflect the magnitudes of the entries. The CUR algorithm does not do too well on this data set either, possibly due to the data's strong diagonal dominance, a feature that row and column sampling techniques struggle with.

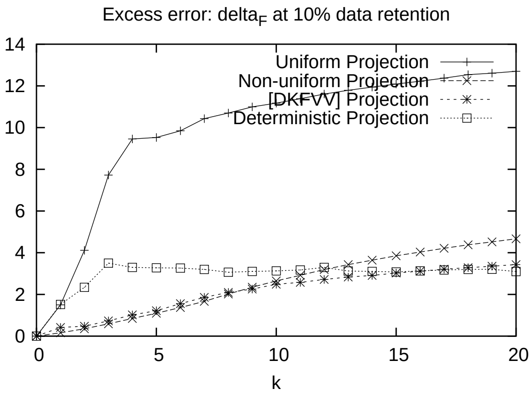
Moving along to projection methods, we look at multiple-pass algorithms that operate by computing a rank k projection P_B, and then approximating A with P_BA. Each of our sparsification techniques produce a projection matrix, as do the methods of Drineas et al. [6]. Additionally, here we look at the projection that results from keeping the largest 10% of the entries deterministically.


Projection methods: || A - P_B A $||_2$ at 10% data retention


Projection methods: || A - P_B A $||_F$ at 10% data retention

Naturally, since in these experiments the input matrix A is projected onto a nested sequence of subspaces of increasing dimension, the corresponding approximation converges to A as k goes to n, hence the absence of any upward tilting curves. To get a more precise view of the error we plot δ_2 and δ_F, highlighting the degree to which our non-uniform projection tracks $\|A - A_k\|_2$.

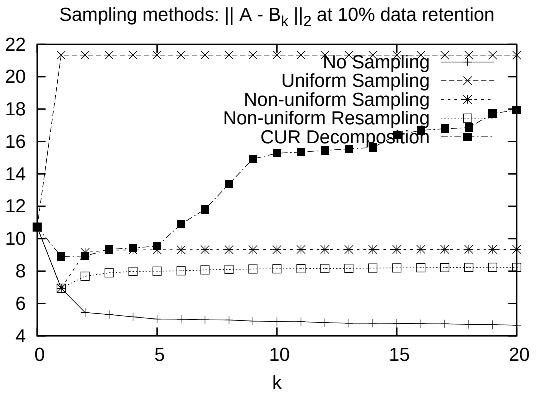

Excess error: delta_2 at 10% data retention

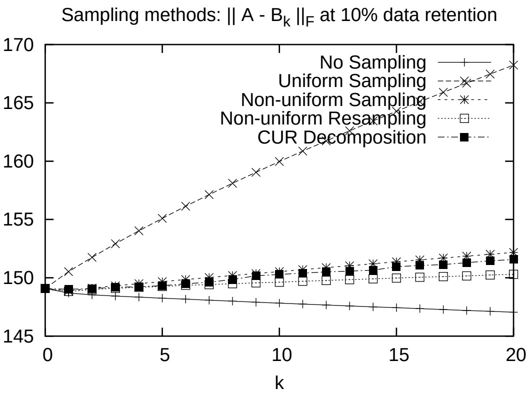

Excess error: delta_F at 10% data retention

7.2 Sparse Document-Term Matrix

For our second experiment, we use a 9603×22226 subset of the Reuters news data set with 503,607 non-zero entries. Here each entry A_{ij} indicates the presence of term j in document i, normalized by a tf-idf factor to discount the influence of frequent terms and wordy documents.

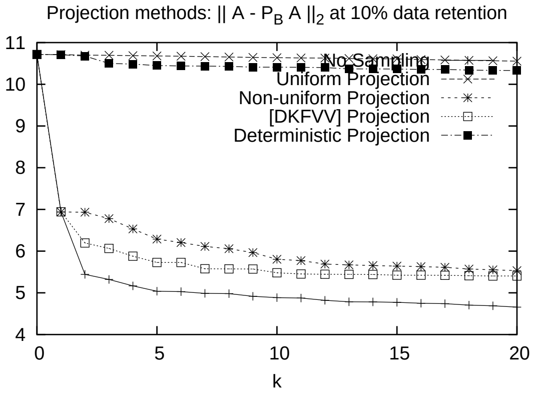
We start by examining the various sampling methods at 10% data retention.

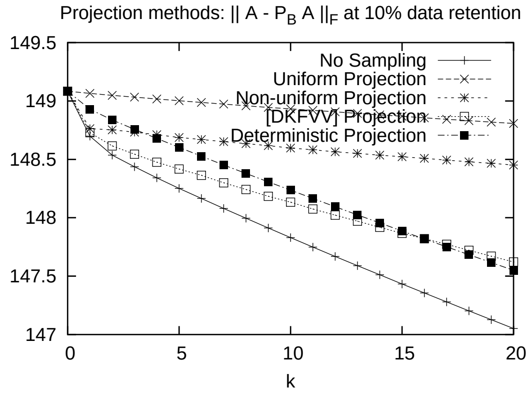




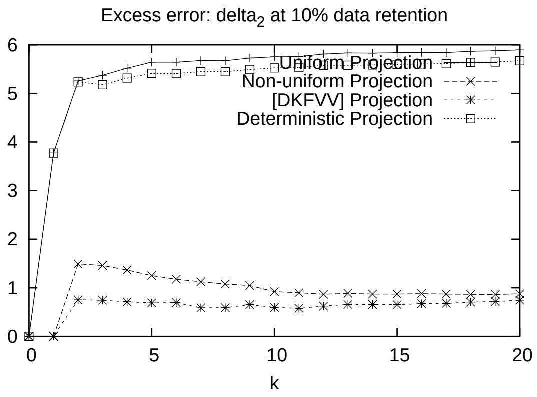
Notice that like in the dense Gaussian kernel matrix, we see something of a threshold behavior in the 2-norm, with the error tracking the optimal error briefly and then fixing itself at $\|A - B\|_2$. The Frobenius norm results are interesting, in that none of the approximations are noticeably better than simply using the "all-zeros" matrix as our approximation. Generally, the utility of sampling-based techniques relies on the assumption that the error introduced by sampling will be outweighed by the gain of extracting significant low rank matrices. This does not appear to be the case here, i.e., at 10% retention rate.

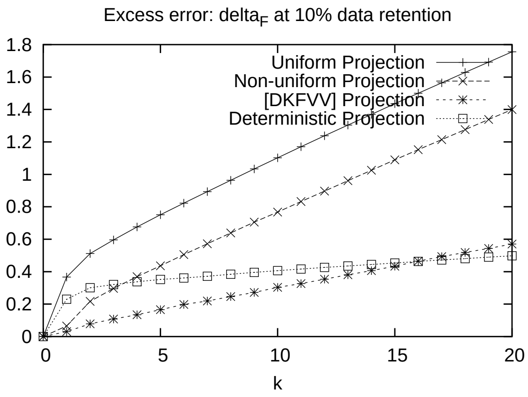
We now turn to the projective methods applied to the sparse data set.





Noteworthy is the dominance of the [DFKVV] algorithm. Taking a closer look by plotting the excess error, we see that the shape of the non-uniform projection curve resembles [DFKVV] in the 2-norm, and uniform projection in the Frobenius norm.





Acknowledgements

We are indebted to Anna Karlin and Charles Van Loan for valuable conversations and comments on earlier drafts. We are also grateful to Petros Drineas and Ravi Kannan for stimulating conversations and encouragement. Finally, we wish to thank Chris Burges, Kwang In Kim, John Platt, and Bernhard Schölkopf for providing us with the data sets used in our experimental evaluation.

References

[1] Noga Alon, Michael Krivelevich and Van H. Vu, *Concentration of eigenvalues of random matrices*, Israel Math. Journal **131** (2002), 259–267.

[2] Yossi Azar, Amos Fiat, Anna Karlin, Frank McSherry, and Jared Saia, *Data mining through spectral analysis*, Proceedings of the 33rd Annual Symposium on Theory of Computing, (Heraklion, Crete, Greece), 2001, pp. 619–626.

[3] Michael W. Berry, Zlatko Drmač, and Elizabeth R. Jessup, *Matrices, vector spaces, and information retrieval*, SIAM Rev. **41** (1999), no. 2, 335–362 (electronic).

[4] Michael W. Berry, Susan T. Dumais, and Gavin W. O'Brien, *Using linear algebra for intelligent information retrieval*, SIAM Rev. **37** (1995), no. 4, 573–595.

[5] Scott Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman, *Indexing by latent semantic analysis*, J. Soc. Inf. Sci. **41** (1990), no. 6, 391–407.

[6] Petros Drineas, Alan Frieze, Ravi Kannan, Santosh Vempala, and V. Vinay, *Clustering Large Graphs via the Singular Value Decomposition*, Machine Learning, **56** (2004), no. 1–3, 9–33.

[7] Petros Drineas, Ravi Kannan, *Pass Efficient Algorithms for Approximating Large Matrices*, Proceedings of the 14th Annual Symposium on Discrete Algorithms (Baltimore, MD), 2003, pp. 223–232.

[8] Alan Frieze, Ravi Kannan, and Santosh Vempala, *Fast monte-carlo algorithms for finding low-rank approximations*, J. ACM, **51** (2004), no. 6, 1025–1041.

[9] Zoltán Füredi and János Komlós, *The eigenvalues of random symmetric matrices*, Combinatorica **1** (1981), no. 3, 233–241.

[10] Gene H. Golub and Charles F. Van Loan, *Matrix computations*, third ed., Johns Hopkins University Press, Baltimore, MD, 1996.

[11] Christos H. Papadimitriou, Prabhakar Raghavan, Hisao Tamaki, and Santosh Vempala, *Latent semantic indexing: A probabilistic analysis*, J. Comput. Syst. Sci., **61** (2000), no. 2, 217–235.

[12] Van H. Vu, *Spectral norm of random matrices*, Proceedings of the 37th Annual Symposium on Theory of Computing, (Baltimore, MD), 2005, pp. 423–430.

[13] *Yale face database*, http://cvc.yale.edu/projects/yalefaces/yalefaces.html