Stat260/CS294: Randomized Algorithms for Matrices and Data

Lecture 20 - 11/13/2013

Lecture 20: Low-rank Approximation with Element-wise Sampling

Lecturer: Michael Mahoney *Scribe: Michael Mahoney*

Warning: these notes are still very rough. They provide more details on what we discussed in class, but there may still be some errors, incomplete/imprecise statements, etc. in them.

20 Low-rank Matrix Approximation with Element-wise Sampling

Today, we will switch gears and start to discuss a different way to construct low-rank matrix approximations: randomly sample elements, rather than rows/columns, from the input matrix. Here is the reading for today and next time.

- Achlioptas and McSherry, "Fast Computation of Low-Rank Matrix Approximations" (the JACM version)

In particular, today we will cover the following topics.

- Review of general approaches to low-rank matrix approximation.
- An introduction to the basic ideas of element-wise sampling.
- A deterministic structural result that is useful for analyzing element-wise sampling algorithms.
- An introduction to a specific element-wise sampling result.

20.1 Review of Some General Themes

So far, we have been talking about sampling/projection of rows/columns—i.e., we have been working with the actual columns/rows or linear combinations of the columns/rows of an input matrix A. Formally, this means that we are pre- or post-multiplying the input matrix A with a sampling/projection/sketching operator (that itself can be represented as a matrix) to construct another matrix A' (with different dimensions) that is similar to A in some way—e.g., the eigenvalues, subspaces, the fraction of norm it captures, etc. are similar to the original matrix A.

This approach makes sense; and if access to full columns and/or rows is possible, then it is probably the best approach. After all, matrices are "about" their columns/rows, in the sense that if you have control over the column/row space and the various null spaces, then you basically have control over the entire matrix. As we will see, this is reflected in the stronger results that exist for column/row sampling than for element-wise sampling. In many data applications, e.g., the DNA SNPs, astronomy, etc., as well as data correlation matrices, etc., the actual columns/rows "mean" something and/or we have relatively-easy access to most or all of the elements in a given column/row. Alternatively, in other applications, e.g., in scientific computing and HPC, one often just wants some basis that is "good" in some well-defined sense, and it doesn't matter what that basis "means" or how it is constructed, but access to entire columns/rows is relatively-straightforward, e.g., by performing matrix-vector multiplications.

In other cases, however, one might want to access the matrix in different ways, e.g., across groups of columns, which might correspond to a cluster in a graph or in some underlying geometry; or access submatrix blocks, e.g., the intersection of sets of rows and columns, since one might want to do bi-clustering; or access individual elements, since an individual element might be meaningful, which is the case in many internet and social media applications. For example, in a prototypical recommendation system application, individual elements correspond to the rating that a given user gave to a given movie.

We are now going to switch to talking about random sampling algorithms that access *elements* of the input matrix. Here are several reasons why this might be of interest.

- It is *algorithmically interesting*, since it corresponds to another way to access the data that will have similarities and differences with the column/row sampling algorithms for sampling/projection that we have been discussing.
- It is a semi-plausible model of *data access*, e.g., in movie recommendation systems, where individual entries arguably "mean" something more than the full columns/rows in terms of how the data are generated and accessed. As an idealization, this leads to the so-called "matrix completion problem," which has received a lot of interest recently.
- It is a plausible model for *interactive analytics.* E.g., if large column/row leverage scores correspond to important/interesting/outlying data—and for many applications they are very non-uniform—then it stands to reason (and is true) that often their non-uniformity is not uniform along the other direction. E.g., high leverage SNPs might not be high leverage uniformly in all subpopulations or in all individuals.

As before, there are two quite different ways of thinking about this problem, which parallels the algorithmic-statistical perspective we had before.

- **Algorithmic perspective.** In this approach, one formulates the problem roughly as follows. Given an input data matrix that is arbitrary or worst-case, but explicitly or implicitly given as input, we sample a small number of elements from the input matrix, and we try to do it in such a way that we get a good approximation to the best low-rank approximation of that data matrix. Most of our previous discussion has adopted this algorithmic perspective.
- **Statistical perspective.** In this approach, one formulates the problem roughly as follows. Given a model for the unobserved data matrix, we assume that we observe a part of that data matrix according to some rule, and then we try to develop a procedure such that we can compute/predict the original unobserved data matrix exactly or approximately. Relatedly, one tries to establish sufficient conditions such that this computation is successful. For example, with rows and columns, if I know that the original matrix is exactly rank k, then if I have *any* set of exactly k linearly independent rows, then I can reconstruct the entire matrix. That last statement is obvious from a linear algebraic perspective, but the assumption of being

exactly rank k can viewed as a (very strong) statistical model (in which case one can ask about relaxing it or using a different procedure, e.g., sampling elements, that is less trivial).

Importantly, similar ideas appear in both perspectives, but they are handled differently. In particular, in the algorithmic approach, one needs to identify important or influential or outlying things, whether columns/rows or elements or something else, e.g., by biasing the sample toward high-leverage components; while in the statistical approach, one needs to make some sort of niceness assumption which typically amounts to assuming that there don't exist any very high leverage components. (That is, some underlying structure is important, but in one case it must be found, while in the other case it must be assumed not to exist.)

20.2 Introduction to Element-wise Sampling

We will be covering the AM07 paper, which is one of the earliest element-wise sampling results in the area. In particular, we will review here their motivation, since it is very nice pedagogically, and since it complements our discussion so far. The paper was introduced in TCS, and so it adopts the algorithmic approach, but many of the more recent developments in element-wise sampling that adopt a statistical approach can be understood in terms of it. Although the motivation for their work was less general than these methods have come to be viewed and used, it is good to know their motivation, since it is related to some of the iterative algorithms we discussed, and since it informed some of the design decisions they made in developing the algorithm.

The motivation for their work was that they were interested in accelerating the computation of good low-rank approximations to an arbitrary matrix $A \in \mathbb{R}^{m \times n}$ when A has strong spectral structure, i.e., when the singular values of interest $\gg$ those of a random matrix of similar size. In particular, recall that orthogonal iteration and Lanczos iteration, two common algorithms for computing low-rank matrix approximation, operate by performing repeated matrix-vector multiplications. To get around the memory requirements, etc., of exactly-optimal low-rank approximation, one might find it acceptable to work with a nearly-optimal low-rank approximation.

To have an efficient method for computing near-optimal rank-k approximations with an iterative algorithm like orthogonal/Lanczos iteration, the rough idea they propose is to do the following.

- Randomly sample or quantize the entries of the input matrix A to get a matrix $\hat{A}$.
- Use Lanczos/orthogonal iteration to get a best rank k approximation $\hat{A}_k$ to $\hat{A}$.
- Show that $\|A - \hat{A}_k\| \leq \|A - A_k\| + ADDL$, where $\|\cdot\|$ is some matrix norm and $ADDL$ is some additional (additive) error term.

So, this approach speeds up the computation of a good low-rank approximation to A (in theory, at least and so far, since their algorithm hasn't been implemented except as a proof of principle) by reducing the number of non-zero entries in the matrix and/or the representation size of those entries. In particular, recall that iterative algorithms require time that is $O\left(\mathrm{nnz}(A)\right)$ multiplied by the number of iterations. (We will treat the iterative algorithm as a black box, and most of our effort will be to show that the best rank k approximation of the sparsified matrix is not much worse than the best rank k approximation of the original matrix.) The analysis of this procedure is based on the idea that sampling/quantizing the entries of a matrix can be viewed as adding a random matrix (albeit, a specially-structured and data-dependent random matrix) to the input and then exploiting that the random matrix has weak spectral structure (in which case it only substantially affects the bottom part of the spectrum of the original matrix).

Before getting into the details, here is a thought experiment to make some of these ideas somewhat more precise. Suppose we want to get $\|A-\hat{A}_k\| \leq \|A-A_k\|+ADDL$, and we will use randomization in the following way to do this. Say that we have an allotment of $ADDL$ and we use it up by adding to A (which is a matrix that is reasonably-well approximated by a low-rank matrix) a matrix G that consists of i.i.d. $N(0,\sigma)$ random variables. This won't "help" computationally, at least with the motivation of speeding up iterative algorithms, since the matrix $A+G$ is no less dense than the matrix A, but it shouldn't "hurt" us "too much." By that, we mean that if the original matrix A had signal in the top part of the spectrum and noise in the bottom part of the spectrum, then we primarily added noise to the bottom part of the spectrum. More precisely, the reason is that if σ is not too large, then $\hat{A}_k = (A+G)_k$ well-approximates A nearly as well as does A_k. The reason for this latter observation has to do with the stability/robustness w.r.t. Gaussian noise that is well-understood, and it stems from the observation that no low-dimensional subspace "describes well" the matrix G. That is, if k is small, then $\|G_k\|$ is small and $\|G-G_k\|$ is large. (If this latter claim isn't "obvious" by now, then recall that, in terms of statistical modeling, low-rank approximations are often used precisely to remove such Gaussian noise in the hypothesized statistical model.)

While more typical in terms of statistical modeling, being Gaussian is *not* essential for the above line of reasoning, and by now it should be clear that this would also hold for any of a wide range of random projection matrices. That is well-known in random matrix theory, and it was elucidated most clearly in RandNLA by AM07, but it holds more generally (Gaussian, Rademacher, sub-Gaussian, Hadamard, other sparsity-respecting constructions, with appropriate choices of parameters, etc.). In particular, to get these results to generalize, the following is sufficient for the random variable.

- Independence
- Mean zero
- Small variance

If N is any random matrix with entries N_{ij} satisfying these three conditions, then $\|N_k\| \sim \|G_k\|$. (In that case, we can ask about finding matrices N that have better algorithmic properties, in a manner analogous to how there are a range of different JL-like constructions that have better algorithmic property than the original JL construction while still obtaining similar quarantees. Indeed, AM07 shows that $\|N_k\|$ bounds the influence that N has on the optimal rank k approximation to $A+N$, and so if $\|A_k\| \gg \|N_k\|$, then $(A+N)_k$ will be well-described by A.) In particular, AM07 does the following.

- Design a random matrix N—*that depends on the input matrix* A—but that still satisfies these three conditions.
- Choose N such that $A+N$ has better sparsity, etc. properties.
- Exploit this phenomenon for computational gain by decreasing the time that each matrix-vector multiplication takes in traditional iterative algorithms.

Here is a toy example illustrating this approach. Let N be a random matrix such that $N_{ij} = \pm A_{ij}$ with equal probability, $\forall i, j$. Then, $\mathbf{E}\left[N_{ij}\right] = 0$ and $\hat{A} = A + N$ has half the number of non-zero entries as A, in expectation. (This is similar to what we saw before, when we observed that with $\{\pm 1\}$ random variables in a random projection matrix, we could set $2/3$ of the entries to zero, in expectation, and still obtain the same concentration results.) This basic idea can be extended to keeping any $p > 0$ fraction of the entries with *ADDL* error growing as $1/\sqrt{p}$.

It turns out that one can get even better sparsification/variance properties if we choose the probability of keeping an entry to depend on the magnitude of that entry. (We say sparsification/variance together, since we will be zeroing out entries to sparsify the matrix, and the bottleneck to getting even sparser is typically that the variance in the relevant estimators is not small enough. So, if we can reduce the variance then we can get sparser.) In particular, let's keep entries i.i.d. with the following probability.

- $\mathbf{Pr}\left[\text{ keeping } A_{ij}\right] \sim A_{ij}^2$.

If we do this, then we focus attention on the larger entries of A (i.e., those which contribute more to the variance). This should do particularly well when entries vary a lot in magnitude. Using the same reasoning, we can also quantize entries to be in $\{-1, 1\}$, which has the advantage that we can represent the entry with a single bit. Alternatively, we can both sample and quantize.

20.3 A Deterministic Structural Result

We will start with a *deterministic structural result* formalizing the idea that perturbation matrices that are poorly-approximated in $\mathbb{R}^k$ have little influence on the optimal rank-k approximation. (That is worth thinking about for a minute, as it is a different intuition than what has motivated most of our previous algorithms, but it is a helpful intuition to have.) We'll use their notation for simplicity of comparison with the paper.

Lemma 1 *Let $A, N \in \mathbb{R}^{m \times n}$, and let $\hat{A} = A + N$. Then,*

$$\begin{aligned} \left\|A - \hat{A}_k\right\|_2 &\leq \|A - A_k\|_2 + 2\|N_k\|_2 \\ \left\|A - \hat{A}_k\right\|_F &\leq \|A - A_k\|_F + \|N_k\|_F + 2\sqrt{\|N_k\|_F \|A_k\|_F} \end{aligned}$$

Remark. As with our other structural results, there is no randomness here in the statement of this lemma. That is, it is a deterministic structural result that holds for any worst-case matrix A and any matrix N. In our RandNLA application, we will apply it to the case where A is reasonably well-approximated by a low-rank matrix and N is one of the sparsifying/quantizing matrices we described above.

Remark. The error in this lemma scales with $\|N_k\|$, and so if N is poorly-approximated in $\mathbb{R}^k$, i.e.,if $\|N_k\|$ is small, then the additional error caused by adding N to A is bounded, compared with the error of the best rank k approximation to A.

Proof:[of Lemma 1] To do the proof, we'll prove two claims relating $\|A - B_k\|$ to $\|A - A_k\|$ for arbitrary matrices, for the spectral and Frobenius norm, as well as an intermediate claim.

Here is the first claim.

Claim 1 *For all matrices A and B, we have that*

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

Proof:[of claim]

$$\begin{aligned}
\|A - B_k\|_2 &\leq \|A - B\|_2 + \|B - B_k\|_2 \quad \text{(by the triangle inequality)} \\
&\leq \|A - B\|_2 + \|B - A_k\|_2 \quad \text{(since } B_k \text{ is the ``best'' rank } k \text{ approximation to } B\text{)} \\
&\leq \|A - B\|_2 + \|B - A\|_2 + \|A - A_k\|_2 \quad \text{(by the triangle inequality)} \\
&= \|A - A_k\|_2 + 2\left\|(A - B)_k\right\|_2 \quad \text{(since } \|B - A\|_2 = \|A - B\|_2 = \left\|(A - B)_k\right\|_2\text{)}
\end{aligned}$$

⋄

Here is the second claim.

Claim 2 *For all matrices A and B, we have that*

$$\left\|P_{B_k} A\right\|_F \geq \left\|P_{A_k} A\right\|_F - 2\left\|(A - B)_k\right\|_F .$$

Here, P_{B_k} is the projection onto the space spanned by the columns of B_k.

Proof:[of claim] The idea is: for all matrices A and B, if $\left\|(A - B)_k\right\|_F$ is small, then projecting A onto P_{B_k} is almost as good as projecting A onto P_{A_k}. Here are the details.

$$\begin{aligned}
\left\|P_{B_k} A\right\|_F &\geq \left\|P_{B_k} B\right\|_F - \left\|P_{B_k}(A - B)\right\|_F \quad \text{(triangle inequality)} \\
&\geq \left\|P_{A_k} B\right\|_F - \left\|P_{B_k}(A - B)\right\|_F \quad \text{(since projecting onto } B_k \text{ is the ``best'')} \\
&\geq \left\|P_{A_k} A\right\|_F - \left\|P_{A_k}(B - A)\right\|_F - \left\|P_{B_k}(B - A)\right\|_F \quad \text{(triangle inequality)} \\
&\geq \left\|P_{A_k} A\right\|_F - 2\left\|P_{(B-A)_k}(B - A)\right\|_F \quad \text{(since } (B - A)_k \text{ is the best)} \\
&= \left\|P_{A_k} A\right\|_F - 2\left\|(B - A)_k\right\|_F
\end{aligned}$$

Above we used that $\left\|P(B - A)\right\|_F \leq \left\|P_{B-A}(B - A)\right\|_F$.

⋄

Now, we will use this intermediate claim to prove that if $\left\|(A - B)_k\right\|_F$ is small, then $\|A - B_k\|_F$ is not much worse than $\|A - A_k\|_F$, and so we can use B_k as a surrogate for A_k w.r.t., $\|\cdot\|_F$—*even if $\|B - A\|_F$ is large*, as long as $\left\|(A - B)_k\right\|_F$ is small.

Here is the third claim.

Claim 3 *For all matrices A and B, we have that*

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

Proof:[of claim] First, observe the following.

$$\begin{aligned}
\|A - B_k\|_F &\leq \|A - P_{B_k}A\|_F + \|P_{B_k}A - B\|_F \quad \text{(triangle inequality)} \\
&\leq \|A - P_{B_k}A\|_F + \|P_{B_k}(A - B)\|_F \quad \text{(since } P_{B_k}B = B_k\text{)} \\
&= \left(\|A\|_F^2 - \|P_{B_k}A\|_F^2 \right)^{1/2} + \|P_{B_k}(A - B)\|_F \quad \text{(by the Pythagorean theorem)} \\
&\leq \left(\|A\|_F^2 - \|P_{B_k}A\|_F^2 \right)^{1/2} + \left\|P_{(A-B)_k}(A - B)\right\|_F \quad \text{(since } (A-B)_k \text{ is the best)} \\
&\leq \left(\|A\|_F^2 - \|P_{B_k}A\|_F^2 \right)^{1/2} + \|(A - B)_k\|_F .
\end{aligned}$$

The comment about "by the Pythagorean theorem" above is that

$$\|A - P_{B_k}A\|_F^2 = \|A\|_F^2 - \|P_{B_k}A\|_F^2 ,$$

in which case we can apply the Pythagorean theorem to each column of A.

So, to establish the claim, we just need to bound the first term. To do so, use Claim 2, from which it follows that

$$\begin{aligned}
\|P_{B_k}A\|_F^2 &\geq \|P_{A_k}A\|_F^2 + 4\,\|(A - B)_k\|_F^2 - 4\,\|P_{A_k}A\|_F\, \|(A - B)_k\|_F \\
&\geq \|P_{A_k}A\|_F^2 - 4\,\|P_{A_k}A\|_F\, \|(A - B)_k\|_F .
\end{aligned}$$

Thus, it follows that

$$\begin{aligned}
\left(\|A\|_F^2 - \|P_{B_k}A\|_F^2 \right)^{1/2} &\leq \left(\|A\|_F^2 - \left\|P_{A_k}A + 4\,\|P_{A_k}A\|_F\, \|(A - B)_k\|_F\right\|_F^2 \right)^{1/2} \\
&= \left(\|A - A_k\|_F^2 + 4\,\|P_{A_k}A\|_F\, \|(A - B)_k\|_F \right)^{1/2} \quad \text{(by the Pythagorean result above)} \\
&\leq \|A - A_k\|_F + 2\sqrt{\|P_{A_k}A\|_F\, \|(A - B)_k\|_F}.
\end{aligned}$$

From this the claim follows.

◇

From this the lemma follows.

◇

20.4 Introduction to the Element-wise Sampling Algorithm

To apply this deterministic structural result to develop a provably-good element-wise random sampling algorithm, we will need a result from random matrix theory. (Actually, we can do it more simply with a matrix Chernoff bound, and this has been developed in subsequent work, but for ease of comparison we will follow AM07 here.).

Fact. Let $G \in \mathbb{R}^{m \times n}$, with $m < n$, and with entries i.i.d., $N(0, \sigma^2)$ r.v. Then, w.p. $\geq 1 - e^{-\Theta(n)}$, we have that

$$\begin{aligned}
\|G_k\|_2 &\leq 4\sigma\sqrt{n} \\
\|G_k\|_F &\leq 4\sigma\sqrt{kn}.
\end{aligned}$$

The first result is somewhat like Wigner's semicircle law (but the details are importantly different, and in particular it is *not* an asymptotic result); and the second result is since $\|A\|_F \le \sqrt{k}\,\|A\|_2$, for all A.

To get a sense of "scale," by which I mean "how big is big and how small is small," note also that the trivial rank k approximation obtained by keeping just the first k rows of G. Call that matrix D, i.e., it is just the first k rows from G. This gives w.h.p. that $\|D\|_F \sim \sigma\sqrt{kn}$. Since $\text{rank}(D) \le k$, we have also that $\|D\|_2 \ge \|D\|_F / \sqrt{k}$.

So, the above fact says that the optimal rank-k approximation improves this trivial approximation by only a factor of ≤ 4. The reason for this is the near orthogonality of the rows of G. By contrast, for a general matrix $A \in \mathbb{R}^{m\times n}$, with $\sigma = |A_{ij}|$ we can have that $\|A_k\|$ can be $\sim \sigma\sqrt{mn}$, in either norm. The main results below say that the effect of random quantization and random sparsification, as described above, is qualitatively the same as adding Gaussian random noise.

Let's start with a more rigorous statement of the above fact. A very brief history is the following.

- Wigner's semicircle law, which makes a similar claim, but asymptotically in convergence.
- Furedi-Komos, which is what AM07 originally used.
- Vu's improvement, using a result of Alon that is due to Talagrand.

Also, there has been a lot of work in recent years improving these results; many of them are simplified with the matrix concentration results we discussed.

Here is a theorem making precise the above discussion.

Theorem 1 *Given a matrix $A \in \mathbb{R}^{m\times n}$, with $m \le n$, and fix an $\epsilon > 0$. Let $\hat{A}$ be a random matrix with entries independent random variables such that for all i, j:*

- $\mathbf{E}\left[\hat{A}_{ij}\right] = A_{ij}$
- $\mathbf{Var}\left[\hat{A}_{ij}\right] \le \sigma^2$
- *$\hat{A}_{ij}$ takes values in the interval of length κ, where $\kappa = \left(\frac{\log(1+\epsilon)}{2\log(m+n)}\right)^2 \cdot \sigma \cdot \sqrt{m+n}$.*

Then, for all $\theta > 0$, and for all $m + n \ge 152$, we have that

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

Remark. We have stated the above result as it appears in AM07. Satisfying the range constraint is awkward, but important, and so we will consider more-or-less awkward ways to do it. In recent years, there have been several improvements to that result which simplify it somewhat, and likely more sophisticated techniques could simplify it even more. I mention that as an FYI, but we won't have time to go into that in detail.

Given this result, we will state a simple sparsification result and a simple quantization result, as we discussed above, and then we will state a more complicated sparsification result that is more comparable with the previous additive error algorithms via column/row sampling.

Theorem 2 *Let* $A \in \mathbb{R}^{m \times n}$*, with* $m \leq n$*, and let* $b = \max_{ij} |A_{ij}|$*. (Think of* b *as analogous to the variance parameter.) Let* $\hat{A} \in \mathbb{R}^{m \times n}$ *be a random matrix with entries distributed i.i.d as*

$$\hat{A}_{ij} = \begin{cases} b & \textit{with probability } \frac{1}{2} + \frac{A_{ij}}{2b} \\ -b & \textit{with probability } \frac{1}{2} - \frac{A_{ij}}{2b} \end{cases} .$$

Then, $\forall$ *sufficiently large* n*, w.p.* $\geq 1 - \exp\left(-19 \left(\log(n)\right)^4\right)$*, the matrix* $\Delta = A - \hat{A}$ *satisfies*

$$\begin{aligned} \|\Delta_k\|_2 &< 4b\sqrt{n} \\ \|\Delta_k\|_F &< 4b\sqrt{kn}. \end{aligned}$$

Theorem 3 *Let* $A \in \mathbb{R}^{m \times n}$*, with* $76 \leq m \leq n$*, and let* $b = \max_{ij} |A_{ij}|$*. (Again, think of* b *as analogous to the variance parameter.) For* $p \geq \frac{(8 \log(n))^4}{n}$*, let* $\hat{A} \in \mathbb{R}^{m \times n}$ *be a random matrix with entries distributed i.i.d as*

$$\hat{A}_{ij} = \begin{cases} A_{ij}/p & \textit{with probability } p \\ 0 & \textit{with probability } 1 - p \end{cases} .$$

Then, w.p. $\geq 1 - \exp\left(-19 \left(\log(n)\right)^4\right)$*, the matrix* $\Delta = A - \hat{A}$ *satisfies*

$$\begin{aligned} \|\Delta_k\|_2 &< 4b\sqrt{n/p} \\ \|\Delta_k\|_F &< 4b\sqrt{kn/p}. \end{aligned}$$

Remark. These two results are complementary, and in particular they can be combined.

Remark. As stated, these results are *not* immediately-comparable to the additive-error or relative-error bounds that we provided before. We will get to that result in the next class.

Proof:[of both results] We just have to fit together all the pieces that we have been discussing.

We can apply the random matrix theorem to $\|N\|_2$ with $\epsilon = 3/10$ and $\theta = 1/10$. Since $\sqrt{m+n} \leq \sqrt{2n}$, we have that $2\left(1 + 3/10 + 1/10\right)\sqrt{2} < 4$ and also that

$$2 \exp\left(-\frac{16\theta^2 \left(\log(n)\right)^4}{\epsilon^4}\right) < \exp\left(-19 \left(\log(n)\right)^4\right).$$

Then, we can use the results that $\|N_k\|_2 = \|N\|_2$ and $\|N_k\|_F \leq \sqrt{k}\,\|N\|_2$ to get the spectral and Frobenius norm bounds, respectively. To deal with the range constraint, recall that $\kappa = \left(\frac{\log((1+\epsilon)}{2\log(m+n)}\right)^2 \sigma\sqrt{m+n}$.

For the quantization theorem, using that $\epsilon = 3/10$ and that $2b < \kappa$ gives that $m + n > 10^{10}$, which is "sufficiently large" in the theorem.

For the sampling theorem, the lower bound on $p \geq \left(\frac{2\log(m+n)}{\log(1+\epsilon)}\right)^4 \frac{1}{m+n}$, which simplifies to $p > \frac{8}{n}\left(\log(n)\right)^4$.

$\diamond$