CPSC 536N: Randomized Algorithms 2014-15 Term 2

Lecture 8

Prof. Nick Harvey *University of British Columbia*

1 Fast Johnson-Lindenstrauss

In the previous lecture we saw the Johnson-Lindenstrauss theorem on dimensionality reduction. Suppose we have n points in $\mathbb{R}^d$ and we map them to $\mathbb{R}^t$, where $t = O(\log(n)/\epsilon^2)$, simply by applying a $t \times d$ matrix of Gaussians (scaled appropriately). Then all lengths and pairwise distances are preserved up to a factor $1+\epsilon$, with high probability. This is a very useful tool in algorithm design.

Today's lecture is concerned with the efficiency of such a mapping. Directly applying matrix-vector multiplication, the time to map a single point from $\mathbb{R}^d$ to $\mathbb{R}^t$ is $O(td)$. There has been much work on trying to make this faster.

One direction of research considered using a slightly *sparse* matrix instead of a dense matrix of Gaussians. The state of the art (Kane-Nelson JACM 2014) allows the matrix to have only an ϵ fraction of non-zero entries, so the time to map a single point becomes $O(\epsilon td)$. Their result is optimal to with a factor of $O(\log(1/\epsilon))$.

Today we discuss a different line of research. Instead of using sparse matrices, we will use *structured* matrices, for which multiplication can be done faster than the naive algorithm. (As a trivial example, consider the matrix of all ones. It is dense, but multiplying by it is very easy.) Such matrices are called "Fast Johnson-Lindenstrauss Transforms", and they have been used extensively in the algorithms, compressed sensing, machine learning and numerical linear algebra communities.

Two of the important results in this area are stated below. They are of the Distributional JL type, in that they

Theorem 1 (Ailon-Chazelle STOC 2006) *There is a $t \times d$ random matrix that satisfies the DJL lemma and for which matrix-vector multiplication takes time $O(d \log d + t^3)$.*

To understand if this runtime is good, consider the scenario where we are applying dimensionality reduction to n data points. The Ailon-Chazelle result is only interesting for certain parameters n, d and t. First, suppose n is very small, say $d = n$, so $t \approx \log n = \log d$. Then the original JL theorem takes time roughly $O(td) = O(d \log d)$ per vector, which is the same as Ailon-Chazelle. Second, suppose n is very large, say $n = 2^{\sqrt{d}}$, so $t \approx \log n = d^{1/2}$. Then the original JL theorem takes time $O(td) = O(d^{3/2})$ per vector and Ailon-Chazelle also takes time $O(d \log d + t^3) = O(d^{3/2})$. The Ailon-Chazelle result is interesting in the intermediate range, where $d \ll n \ll 2^{d^{1/2}}$

We will prove the following theorem, which is a slightly simplified form of the Ailon-Chazelle result.

Theorem 2 *There is a random matrix L of size $t \times d$ with $t = O(\log(d/\delta)^2 \log(1/\delta)/\epsilon^2)$ such that, for each $x \in \mathbb{R}^d$,*

$$\|Lx\| \in [1-\epsilon, 1+\epsilon] \cdot \|x\|$$

holds with probability at least $1-\delta$. Matrix-vector multiplication with L takes time $O(d \log d + t)$.

2 A Simple Start: Super-Sparse Sampling

Let's start with simple idea: given a vector x, we will sample it using a very sparse sampling matrix. This matrix is denoted S and it has size $t \times d$. Each row of S has a single non-zero entry of value $\sqrt{d/t}$ in a uniformly random location.

For any vector $x \in \mathbb{R}^d$, we have

$$\begin{aligned} \mathrm{E}\left[\,(Sx)_i^2\,\right] &= \sum_{j=1}^{d} \underbrace{\Pr\left[\,\text{random location is } j\,\right]}_{=1/d} \cdot (d/t) \cdot x_j^2 \;=\; (1/t) \cdot \|x\|_2^2 \\ \Longrightarrow \qquad \mathrm{E}\left[\,\|Sx\|^2\,\right] &= \mathrm{E}\left[\sum_{i=1}^{t} (Sx)_i^2\right] \;=\; \|x\|_2^2 \end{aligned} \tag{1}$$

So this works well in expectation, even if we have $t = 1$. Unfortunately the variance can be terrible. For example, if x has just one non-zero coordinate, then t needs to be $\Theta(d)$ in order to have a reasonable chance of sampling it. Typically we would want $t \ll d$. In generally if one coordinate is much larger than the others in absolute value then super-sparse sampling needs t to be very large. It is convenient to state this problematic scenario mathematically using norms.

Aside on norms. The ℓ_p norm is defined to be $\|x\|_p = (\sum_i |x_i|^p)^{1/p}$ for $p \in [1, \infty)$. Taking $p \to \infty$ we obtain $\|x\|_\infty = \max_i |x_i|$. The usual Euclidean norm is $\|\cdot\|_2$. A standard inequality for norms in $\mathbb{R}^d$ is given by

$$\|x\|_p \;\leq\; \|x\|_r \;\leq\; d^{1/r - 1/p} \cdot \|x\|_p \qquad \forall 1 \leq r \leq p.$$

Today we are only interested in $\|\cdot\|_2$ and $\|\cdot\|_\infty$, for which we have $\|x\|_2 \leq \sqrt{d} \cdot \|x\|_\infty$.

The problematic scenario for super-sparse sampling is when one coordinate is much larger than the others, i.e., $\|x\|_\infty / \|x\|_2 \approx 1$. It turns out that super-sparse sampling actually works very well in the opposite scenario, when

$$\frac{\|x\|_\infty}{\|x\|_2} \;\approx\; 1/\sqrt{d}, \tag{2}$$

which we have just argued is its smallest possible value. Claim 6 formally proves that super-sparse sampling works when (2) holds.

3 Idea: Rotating the basis

Stating the problematic scenario in these terms leads to a useful idea. We said that super-sparse sampling will require large t when $\|x\|_\infty / \|x\|_2 \approx 1$. Now we recall an important property of the Euclidean norm: it is invariant under rotations and reflections. Formally, $\|Mx\|_2 = \|x\|_2$ for any orthogonal matrix M. Another way to say this is that $\|\cdot\|_2$ is invariant under an orthogonal change of basis, and indeed $\|\cdot\|_2$ can be defined without reference to any basis (using an inner product). On the other hand, the infinity norm $\|\cdot\|_\infty$ is *heavily* dependent on the choice of basis. For example,

$$\begin{aligned} u = (1, 0, 0, \ldots, 0) \in \mathbb{R}^d \qquad &\text{has} \qquad \|u\|_2 = \|u\|_\infty = 1 \\ v = (1, 1, \ldots, 1)/\sqrt{d} \in \mathbb{R}^d \qquad &\text{has} \qquad \|v\|_2 = 1 \text{ and } \|v\|_\infty = 1/\sqrt{d} \end{aligned}$$

even though v is a rotation of u. Intuitively, the quantity $\|x\|_\infty / \|x\|_2$ is telling us how well the vector x "aligns" with the standard basis.

In order for our super-sparse sampling to work we need (2) to hold, which means that x is not aligned with the standard basis. However we have no control over x because our theorem needs to work for all x. The key idea is to control our basis instead. Why not choose a random basis that is unlikely to align with the given vector x? Indeed, that is basically what is accomplished by the dense matrix of Gaussians in the previous lecture.

The only issue with the dense matrix of Gaussians is that matrix-vector multiplications are too slow. So let's think if there is a quicker way to randomly rotate the basis. Whenever I think of vectors that "disagree" with the standard basis, the first object that comes to mind is a Hadamard matrix.

Definition 3 *A **Hadamard matrix** is a $d \times d$ real matrix H that is orthogonal (meaning $H^{\mathsf{T}}H = I$) and has all entries in $\left\{-1/\sqrt{d}, 1/\sqrt{d}\right\}$.*

It is not a priori clear that Hadamard matrices exist, and indeed it is not fully known for what values of d they exist. In the case $d = 2$, we have the Hadamard matrix

$$H_2 \;=\; \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} / \sqrt{2},$$

which amounts to a 45-degree rotation of the standard basis. In fact, we can build on this recursively to obtain a Hadamard matrix whenever d is a power of two.

$$H_d \;=\; \begin{pmatrix} H_{d/2} & H_{d/2} \\ H_{d/2} & -H_{d/2} \end{pmatrix} / \sqrt{2}. \tag{3}$$

This is called a Walsh-Hadamard matrix, and it has many nice properties.

To see that H_d is indeed a Hadamard matrix, we must make two observations. First, induction shows that every entry of H_d is $\pm(1/\sqrt{2})^{\log_2 d} = \pm 1/\sqrt{d}$. Second, we have

$$H_d^{\mathsf{T}} H_d \;=\; \begin{pmatrix} H_{d/2}^{\mathsf{T}} & H_{d/2}^{\mathsf{T}} \\ H_{d/2}^{\mathsf{T}} & -H_{d/2}^{\mathsf{T}} \end{pmatrix} \cdot \begin{pmatrix} H_{d/2} & H_{d/2} \\ H_{d/2} & -H_{d/2} \end{pmatrix} / 2 \;=\; \begin{pmatrix} 2H_{d/2}^{\mathsf{T}} H_{d/2} & 0 \\ 0 & 2H_{d/2}^{\mathsf{T}} H_{d/2} \end{pmatrix} / 2 \;=\; I,$$

so H_d is orthogonal.

Another nice property of H_d is that matrix-vector multiplication can be performed in $O(d \log d)$ time. This follows from an easy divide-and-conquer algorithm.

4 Randomized Hadamard Matrix

Using H_d to change our basis will not ensure that (2) holds. It is obvious that, for *any* fixed orthogonal matrix M, there exist vectors x for which $\|Mx\|_\infty / \|Mx\|_2 = 1$: simply let x be any row of M. This is why the randomization is necessary.

Instead, we must pick a *random* change of basis M and argue that

$$\text{each } x \in \mathbb{R}^d \text{ satisfies} \qquad \left(\; \frac{\|Mx\|_\infty}{\|Mx\|_2} \approx \frac{1}{\sqrt{d}} \quad \text{with high probability} \; \right). \tag{4}$$

So far our Hadamard matrix $H = H_d$ has no randomness. How can we "randomize" it? Well, in the recursive definition (3) we quite arbitrarily put the minus sign in the lower-right quadrant. The construction works just as well if we put the minus sign in any quadrant, so this suggests that we should be able to randomize the construction using random signs.

We will introduce random signs in a slightly more convenient way. Let D be a diagonal matrix whose i^{th} diagonal entry is a random sign $\xi_i \in \{-1, 1\}$, and these are independent. Our random Hadamard matrix is the product $M = HD$. This is indeed a Hadamard matrix: its entries are still $\pm 1/\sqrt{d}$, and M is orthogonal because $M^{\mathsf{T}} M = D H^{\mathsf{T}} H D = D^2 = I$.

The following claim shows that, with this randomized Hadamard matrix M, every vector x satisfies (4).

Claim 4 *Let $x \in \mathbb{R}^d$ be non-zero. Let $y = HDx$, where HD is the random Hadamard matrix. Then*

$$\Pr_D \left[\frac{\|y\|_\infty}{\|y\|_2} \geq \underbrace{\sqrt{\frac{2\ln(4d/\delta)}{d}}}_{\lambda} \right] \;\leq\; \delta/2.$$

PROOF: It suffices to consider the case $\|x\|_2 = 1$, because $\|HDx\|_\infty / \|HDx\|_2$ is invariant under rescaling x. Note that $\|HDx\|_2 = \|x\|_2 = 1$ as HD is orthogonal, so it suffices to show that all coordinates of HDx are likely small. We will follow our usual approach of showing that each coordinate is very likely to be small, then union bounding over all coordinates.

Consider y_1, the first coordinate of $y = HDx$. It is obtained by multiplying each coordinate of x by a random sign, then taking the dot-product with the first row of H. So

$$y_1 \;=\; \sum_j \xi_j H_{1,j} x_j,$$

Note that the terms of this sum are independent random variables. It is tempting to apply the Chernoff bound to analyze y_1, but the Chernoff bound that we have used so far is only valid for sums of non-negative random variables. Let us state the following generalized Chernoff bound, which is very useful in our scenario.

Theorem 5 (Hoeffding's General Inequality) *Let $X_1, \ldots, X_n$ be independent random variables where $X_i \in [a_i, b_i]$. Let $X = \sum_i X_i$. Then*

$$\Pr\left[\left| \sum_{i=1}^n X_i - \mathrm{E}[\,X\,] \right| \geq s \right] \;\leq\; 2\exp\Big(-2 \frac{s^2}{\sum_{i=1}^n (b_i - a_i)^2} \Big). \tag{5}$$

In particular, for any desired $q \in (0,1)$, setting $s = \sqrt{\ln(2/q) \sum_{i=1}^n (b_i - a_i)^2 / 2}$ gives

$$\Pr\left[\, |\textstyle\sum_i X_i - \mathrm{E}[\,X\,]| \geq s \,\right] \;\leq\; q. \tag{6}$$

References: McDiarmid Theorem 2.5, Dubhashi-Panconesi Problem 1.9, Wikipedia.

We apply this with $X_j = \xi_j H_{1,j} x_j$. Since $\xi_j \in \{-1, +1\}$, we have $X_j = \pm H_{1,j} x_j$. So X_j lies in the interval $[a_j, b_j]$ where $-a_j = b_j = H_{1,j} x_j$. Note that $\mathrm{E}[\,X_j\,] = 0$ and

$$\sum_{j=1}^d (b_j - a_j)^2 \;=\; 4 \sum_{j=1}^d H_{1,j}^2 x_j^2 \;=\; 4 \sum_{j=1}^d (1/d) x_j^2 \;=\; 4\,\|x\|_2^2 / d \;=\; 4/d.$$

Defining $q = \delta/2d$, the quantity s in Theorem 5 becomes $s = \sqrt{2\ln(4d/\delta)/d}$, which equals the value of λ defined in the statement of Claim 4. Consequently, (6) yields

$$\Pr\left[\,|y_1| \geq \lambda\,\right] \;=\; \Pr\left[\,\left|\sum_j X_j - \underbrace{\mathrm{E}\left[\sum_j X_j\right]}_{=0}\right| \geq s\,\right] \;\leq\; \delta/2d.$$

A union bound over all coordinates of y shows that

$$\Pr\left[\,\|y\|_\infty \geq \lambda\,\right] \;\leq\; \sum_{j=1}^{d} \Pr\left[\,|y_j| \geq \lambda\,\right] \;\leq\; d \cdot (\delta/2d) \;=\; \delta/2.$$

□

5 Putting it all together

Earlier we said that super-sparse sampling should work as long as (2) holds. We have just shown (2) will hold after applying the randomized Hadamard matrix; more precisely, (4) holds with $M = HD$. The final step is to apply the super-sparse sampling matrix S to Mx. To summarize, the overall linear map is $L = SHD$ where S is a super-sparse sampling matrix of size $t \times d$, H is a $d \times d$ Hadamard matrix, and D is a diagonal matrix of random signs.

We now formalize the claim that super-sparse sampling works.

Claim 6 *Let y be a fixed vector in $\mathbb{R}^d$ with $\|y\|_2 = 1$ and*

$$\|y\|_\infty \;\leq\; \lambda \;=\; \sqrt{\frac{2\ln(4d/\delta)}{d}}.$$

(This is the value of λ used in Claim 4.) Let S be a $t \times d$ super-sparse sampling matrix with $t = 2\ln(4d/\delta)^2\ln(4/\delta)/\epsilon^2$. Then $\Pr\left[\,\|Sy\|_2^2 \notin (1-\epsilon, 1+\epsilon)\,\right] \leq \delta/2$.

Proof (of Theorem 2). Fix any vector x with $\|x\|_2 = 1$. Construct the random matrix $L = SHD$ as explained above and let $y = HDx$. Define the events

$$\begin{aligned}
\mathcal{E}_1 &= \{\|y\|_\infty \geq \lambda\} \\
\mathcal{E}_2 &= \{\|Sy\|_2 \notin (1-\epsilon, 1+\epsilon)\}
\end{aligned}$$

Claim 4 shows that $\Pr\left[\,\mathcal{E}_1\,\right] \leq \delta/2$. Claim 6 shows that $\Pr\left[\,\mathcal{E}_2 \mid \mathcal{E}_1^c\,\right] \leq \delta/2$. Combining those bounds we obtain

$$\begin{aligned}
\Pr\left[\,\|Lx\| \notin (1-\epsilon, 1+\epsilon)\,\right] &= \Pr\left[\,\mathcal{E}_2\,\right] \\
&= \Pr\left[\,\mathcal{E}_1 \cap \mathcal{E}_2\,\right] + \Pr\left[\,\mathcal{E}_1^c \cap \mathcal{E}_2\,\right] \\
&\leq \Pr\left[\,\mathcal{E}_1\,\right] + \Pr\left[\,\mathcal{E}_2 \mid \mathcal{E}_1^c\,\right] \\
&\leq \delta/2 + \delta/2 \;=\; \delta.
\end{aligned}$$

Finally, let us check that matrix-vector multiplication with $L = SHD$ is efficient.

- Multiplication by D is trivial and takes $O(d)$ time.

- Multiplication by H requires $O(d \log d)$ time as explained above.
- Multiplication by S is trivial and takes $O(t)$ time.

So the total time is $O(d \log d + t)$. ■