Randomized Algorithms 2021A – Lecture 7 (second part) Fast JL (cont'd) and the JL Transform*

Robert Krauthgamer

1 Fast JL (cont'd)

Recall we wanted to prove the following.

Theorem 7: For every $d \geq 1$ and $0 < \delta < 1$, there is a random matrix $L \in \mathbb{R}^{k \times d}$ for $k = O(\varepsilon^{-2} \log^2(d/\delta) \log(1/\delta))$, such that

$$\forall v \in \mathbb{R}^d, \qquad \Pr\Big[\|Lv\| \notin (1 \pm \varepsilon)\|v\|\Big] \leq 1/\delta,$$

and multiplying L with a vector v takes time $O(d \log d + k)$.

Definition: A *Hadamard matrix* is a matrix $H \in \mathbb{R}^{d \times d}$ that is orthogonal, i.e., $H^T H = I$ and all its entries are in $\{\pm 1/\sqrt{d}\}$.

Observe that by definition $\|Hv\|_2^2 = (Hv)^T(Hv) = v^T v = \|v\|_2^2$.

When d is a power of 2, such a matrix exists, and can be constructed by induction as follows (called a Walsh-Hadamard matrix).

$$H_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} / \sqrt{2},$$
$$H_d = \begin{pmatrix} H_{d/2} & H_{d/2} \\ H_{d/2} & -H_{d/2} \end{pmatrix} / \sqrt{2}.$$

It is easy to verify it is indeed a Hadamard matrix, i.e., that all entries are $\pm 1/\sqrt{d}$ and $H_d^T H_d = I$.

Lemma 8: Multiplying H_d by a vector can be performed in time $O(d \log d)$.

Exer: Prove this lemma, using divide and conquer.

Randomized Hadamard matrix: Let $D \in \mathbb{R}^{d \times d}$ be a diagonal matrix whose ith diagonal entry is an independent random sign $r_i \in \{\pm 1\}$. Observe that HD is a random Hadamard matrix, because its entries are still $\pm 1/\sqrt{d}$ and $(HD)^T(HD) = D^T H^T H D = D^T D = I$.

*These notes summarize the material covered in class, usually skipping proofs, details, examples and so forth, and possibly adding some remarks, or pointers. The exercises are for self-practice and need not be handed in. In the interest of brevity, most references and credits were omitted.

Lemma 9: Let HD be a random Hadamard matrix as above, and let $\delta \in (0,1)$. Then

$$\forall 0 \neq v \in \mathbb{R}^d, \qquad \Pr_D\left[\frac{\|HDv\|_\infty}{\|HDv\|_2} \geq \sqrt{\frac{2\ln(4d/\delta)}{d}}\right] \leq \delta/2.$$

Exer: Prove Lemma 9 (as discussed in class) using the following concentration bound.

Hoeffding's (generalized) inequality: Let $X_1, \ldots, X_n$ be independent random variables where $X_i \in [a_i, b_i]$. Then $X = \sum_i X_i$ satisfies

$$\forall t \geq 0, \qquad \Pr\Big[|X - \mathbb{E}[X]| \geq t\Big] \leq 2e^{-2t^2/\sum_i (b_i - a_i)^2}.$$

Lemma 10: Let $S \in \mathbb{R}^{k \times d}$ be a super-sparse sampling matrix (i.e., each row has a single nonzero entry of value $\sqrt{d/k}$ in a uniformly random location). Then

$$\forall y \in \mathbb{R}^d, \|y\|_2 = 1, \|y\|_\infty \leq \lambda, \qquad \Pr_S[\|Sy\|_2^2 \notin (1 \pm \varepsilon)] \leq 2e^{-2\varepsilon^2 k/(d^2\lambda^4)}.$$

Exer: Prove this lemma using Hoeffding's inequality. Would you get the same bound using Chebyshev's inequality?

Proof of Theorem 7: Was discussed in class and basically follows from Lemmas 9 and 10.

2 The JL Transform

JL dimension reduction: We saw the JL lemma which reduces the dimension of n points in $\mathbb{R}^d$. Recall that it uses a random linear map that is drawn obliviously of the data and works with high probability.

Next, we abstract its performance guarantee (ignoring the implementation), because algorithms may have different tradeoffs, e.g., between the target dimension and the runtime. We also change some of the letters (e.g., use $\mathbb{R}^n$ instead of $\mathbb{R}^d$).

Here is a good way to think about the next definition. A matrix $S \in \mathbb{R}^{s \times n}$ is just a linear map $S : \mathbb{R}^n \to \mathbb{R}^s$. It will represent a dimension reduction operation, where b unknown points in $\mathbb{R}^n$ are reduced to points in dimension $s = s(n, b, \varepsilon, \delta)$, and we want this s (the number of rows in S) to be as small as possible. But instead of a single map S, we consider a probability distribution.

Throughout, all vector norms are ℓ_2-norms.

Definition: A random matrix $S \in \mathbb{R}^{s \times n}$ is called an (ε, δ, b)-*Johnson-Lindenstrauss Transform (JLT)* if

$$\forall B \subset \mathbb{R}^n, |B| \leq b, \qquad \Pr_S\Big[\forall x \in B,\ \|Sx\| \in (1 \pm \varepsilon)\|x\|\Big] \geq 1 - \delta.$$

We saw in class that a matrix of independent Gaussians (scaled appropriately) attains this guarantee, with a suitable $s = O(\varepsilon^{-2}\log(b/\delta))$. More precisely, we saw it only for $b = 1$, but general b follows easily by applying that result with smaller $\delta' = \delta/b$ and taking a union bound over B.

Notice that the target dimension s does not depend on the ambient dimension n.

We saw also another construction, with bigger target dimension s, but faster matrix-vector multiplication (back then we called it $L = SHD$).

3 Approximate Matrix Multiplication

Definition: The *Frobenius norm* of a real matrix A is defined as

$$\|A\|_F := (\sum_{i,j} A_{ij}^2)^{1/2}.$$

Problem definition: In *Approximate Matrix Multiplication (AMM)*, the input is $\varepsilon > 0$ and two matrices $A, B \in \mathbb{R}^{n \times m}$, and the goal is to compute a matrix $C \in \mathbb{R}^{m \times m}$ such that

$$\|A^\top B - C\|_F \leq \varepsilon \|A\|_F \|B\|_F.$$