PRINCETON COS 521: ADVANCED ALGORITHM DESIGN

Dimension Reduction and the Johnson-Lindenstrauss Lemma

Very high-dimensional vectors are ubiquitous in science, engineering, and machine learning. They give a simple way of representing data: for each object we want to study, we collect a very large set of numerical parameters, often with no inherent order or structure. We use these parameters to compare, analyze, and make inferences about those objects.

High-dimensional data comes from genetic data sets, time series (e.g. audio or seismographic data), image data, etc. It is also a common output of feature generation algorithms.

Feature generation algorithms are commonly used to pre-process image and audio data as well. For example, Shazam and other "song matching" services preprocess audio by computing a spectrogram, which essentially computes many Fourier transforms of different sections of the signal, shifted to start at different time points. More on this example later.

What do we want to do with such high dimensional vectors? Cluster them, use them in regression analysis, feed them into machine learning algorithms. As an even more basic goal, all of these tasks require being able to determine if one vector is similar to another. Even this simple task becomes an unwiedly in high-dimensions.

Dimensionality Reduction

The goal of dimensionality reduction is to reduce the cost of working with high-dimensional data by representing it more compactly. Instead of working with an entire vector, can we find a more compact "fingerprint" – i.e. a shorter vector – that at least allows us to quickly compare vectors? Or maybe the fingerprint preserves certain properties of the original vector that allows it to be used in other downstream tasks.

Computer scientists have developed a remarkably general purpose toolkit of dimensionality reduction methods for constructing compact representations that can be used effectively in a huge variety of downstream tasks. In this section of the course, we will study some of those methods.

The Johnson-Lindenstrauss Lemma

We start with a particular powerful and influential result in high-dimensional geometry. It applies to problems involving the ℓ_2 norm:

$$\|x\|_2 = \sqrt{\sum_{i=1}^{m} x_i^2}$$

For two vectors x and y, $\|x-y\|_2$ is the Euclidean distance.

Problem 1. *Given n points $v^1, v^2, ..., v^n \in \mathbb{R}^d$, we want to find a function $f : \mathbb{R}^d \to \mathbb{R}^m$ such that m is much smaller than d and for all i, j,*

$$(1-\epsilon)\|v^i - v^j\|_2 \leq \|f(v^i) - f(v^j)\|_2 \leq (1+\epsilon)\|v^j - v^j\|_2. \tag{1}$$

In other words, the distance between all pairs of points in preserved.

The following main result (Lemma in their words) is by Johnson & Lindenstrauss [1]:

Theorem 1 (Johnson-Lindenstrauss Lemma). *There is a function f satisfying* (1) *that maps vectors to $m = O(\frac{\log n}{\epsilon^2})$ dimensions. In fact, f is a* linear mapping *and can be applied in a computationally efficient way!*

The following ideas do not work to prove this theorem: (a) take a random sample of m coordinates out of d. (b) Partition the d coordinates into m subsets of size about n/m and *add* up the values in each subset to get a new coordinate.[2]

We're going to choose f randomly. In particular, let G be a $m \times d$ *random matrix* with each entry a normal random variable, $G_{i,j} \sim \mathcal{N}(0,1)$. Let $\Pi = \frac{1}{\sqrt{m}} G$:

$$f(x) = \Pi x.$$

So each entry in $u = f(v)$ equals $v \cdot g$ for some vector g filled with scaled Gaussian random variables. Other choices for G work: for example, we can use random signs or a random orthonormal matrix (used in the original proof).

We're going to prove a slightly stronger statement for this map:

Theorem 2 ((ϵ,δ)-JL property). *If $m = O(\log(1/\delta)/\epsilon^2)$, then for any vector x,*

$$(1-\epsilon)\|x\|_2^2 \leq \|\Pi x\|_2^2 \leq (1+\epsilon)\|x\|_2^2 \tag{2}$$

with probability $(1-\delta)$.

Note that, while stated with the squared Euclidean norm, (2) immediately implies that $(1-\epsilon)\|x\|_2 \leq \|\Pi x\|_2 \leq (1+\epsilon)\|x\|_2$ (just

[1]

[2] To see why these approaches fail whp, consider the case of two vectors: $(1,0,\ldots,0)$ and $(0,1,0,\ldots,0)$. Then the first approach succeeds iff we happen to pick coordinate one or two as one of the coordinates, which is unlikely. To see why the second approach fails, consider two vectors $(1,\ldots,1,0,\ldots,0)$ and $(0,\ldots,0,1,\ldots,1)$. Then the second approach whp generates nearly-identical vectors even though the initial two vectors are far apart.

by taking a square root of all sides, and observing that this brings the constants closer to 1). Then, to prove Theorem 1 from this stronger statement, we use the linearity of f to see that:

$$\|f(v^i) - f(v^j)\|_2 = \|\Pi v^i - \Pi v^j\|_2 = \|\Pi(v^i - v^j)\|_2.$$

So, with probability $(1-\delta)$ we preserve one distance. We have $\binom{n}{2} = O(n^2)$ distances total. By a union bound, we preserve all of them with probability $1-\delta$ as long as we reduce δ to $\delta/\binom{n}{2}$, which means that $m = O(\log(n/\delta)/\epsilon^2)$. This gives Theorem 1. So, we can focus our attention on proving Theorem 2.

Proof. Let $w = Gx$ be a scaling of our dimension reduced vector. Our goal is to show that $\|x\|_2^2$ is approximated by:

$$\|\Pi x\|_2^2 = \|\frac{1}{\sqrt{m}}Gx\|_2^2 = \frac{1}{m}\sum_{i=1}^{m} w_i^2.$$

Consider one term of the sum, w_i^2, which is a random variable since G is chosen randomly. We will start by showing that each term is equal to $\|x\|_2^2$ in expectation. We have:

$$w_i = \sum_{j=1}^{d} x_j g_j$$

where each $g_j \sim \mathcal{N}(0,1)$. So $\mathbb{E}[w_i] = \sum_{j=1}^{d} x_j \mathbb{E}[g_j] = 0$ and thus $\mathbf{Var}[w_i] = \mathbb{E}[w_i^2]$. It follows that:

$$\mathbb{E}[w_i^2] = \mathbf{Var}[w_i] = \sum_{j=1}^{d} \mathbf{Var}[x_j g_j] = \sum_{j=1}^{d} x_j^2 \mathbf{Var}[g_j] = \sum_{j=1}^{d} x_j^2 = \|x\|_2^2.$$

Thus $\mathbb{E}[w_i^2] = \|x\|_2^2$ and our estimate is correct in expectation:

$$\mathbb{E}\left[\frac{1}{m}\sum_{i=1}^{m} w_i^2\right] = \|x\|_2^2.$$

How do we know that it's close to this expectation with high probability? We actually know that w_i is a *normal random variable*.

Fact 1 (Stability of Gaussian random variables)**.** *If X and Y are independent and $X \sim \mathcal{N}(0, a^2)$ and $Y \sim \mathcal{N}(0, b^2)$, then $X + Y \sim \mathcal{N}(0, a^2 + b^2)$. The property that the sum of Gaussian's remains Gaussian is known as "stability"*[3].

So each $w_i \sim \mathcal{N}(0, \|x\|^2) = \|x\|_2 \cdot \mathcal{N}(0,1)$. We can prove the concentration via a similar approach to the proof of Chernoff bound.

[3] There are other classes of stable distributions, but the normal distribution is the only stable distribution with bounded variance, which gives some intuition for why the central limit theorem holds for random variables with bounded variance.

Let $t > 0$ be a parameter to be determined later, we have

$$\begin{aligned}\Pr\left[\sum_{i=1}^m w_i^2 > (1+\varepsilon)m\|x\|_2^2\right] &= \Pr\left[e^{t\sum_{i=1}^m w_i^2} > e^{t(1+\varepsilon)m\|x\|_2^2}\right] \\ &\le \frac{\mathbb{E}\left[e^{t\sum_{i=1}^m w_i^2}\right]}{e^{t(1+\varepsilon)m\|x\|_2^2}} \\ &= \frac{\mathbb{E}\left[e^{tw_1^2}\right]^m}{e^{t(1+\varepsilon)m\|x\|_2^2}}.\end{aligned}$$

Now note that

$$\begin{aligned}\mathbb{E}\left[e^{tw_i^2}\right] &= \int_{\mathbb{R}} \frac{1}{\sqrt{2\pi}} e^{-\frac{g^2}{2}} \cdot e^{t\|x\|_2^2 g^2}\,\mathrm{d}g \\ &= \int_{\mathbb{R}} \frac{1}{\sqrt{2\pi}} e^{-(1-2t\|x\|_2^2)g^2/2}\,\mathrm{d}g \\ &= \frac{1}{(1-2t\|x\|_2^2)^{1/2}}.\end{aligned}$$

Thus,

$$\begin{aligned}\Pr\left[\sum_{i=1}^m w_i^2 > (1+\varepsilon)m\|x\|_2^2\right] &\le \frac{1}{(1-2t\|x\|_2^2)^{m/2}\cdot e^{t(1+\varepsilon)m\|x\|_2^2}} \\ &\le ((1-2t\|x\|_2^2)(1+2t(1+\varepsilon)\|x\|_2^2))^{-m/2} \\ &= (1+2\varepsilon t\|x\|_2^2 - 4t^2(1+\varepsilon)\|x\|_2^4)^{-m/2},\end{aligned}$$

setting $t = \varepsilon/(4(1+\varepsilon)\|x\|_2^2)$ minimizes the RHS, and we obtain that

$$\Pr\left[\sum_{i=1}^m w_i^2 > (1+\varepsilon)m\|x\|_2^2\right] \le e^{-\Theta(\varepsilon^2 m)}.$$

A similar bound on the probability that $\sum_{i=1}^m w_i^2 < (1-\varepsilon)m\|x\|_2^2$ can also be proved.

So, if we set $m = O(\log(1/\delta)/\epsilon^2)$ then $\|\Pi x\|_2^2 = \frac{1}{m}\sum_{i=1}^m w_i^2$ satisfies:

$$\|x\|_2^2 - \epsilon\|x\|_2^2 \le \|\Pi x\|_2^2 \le \|x\|_2^2 + \epsilon\|x\|_2^2$$

with probability $1-\delta$. □

It's worth noting that Theorem 1 is tight – i.e. there are point sets that cannot be embedded into less than $O(\log n/\epsilon^2)$ dimensions if we want to preserve all pairwise distances. This was proven up to a $\log(1/\epsilon)$ factor by Noga Alon in [4]. The fully tight result was only obtained in 2017 [5]. The result was proven first for *linear embeddings* and then extended to a lower-bound for all possible functions f.

[4]

[5]

Faster running time

Given a vector x, for a random matrix Π, it takes $O(md)$ time to compute the matrix vector product Πx. This could potentially be very slow. There are two strategies to speed up this computation.

- The first is to use a sparse matrix Π (called sparse JL);
- the second is to use a structured matrix Π that allows fast matrix vector multiplication (called fast JL).

It turns out that for generic vectors x, the second approach usually has a better running time. However, when the input vector x itself is also sparse, sparse JL can utilize both the sparsity of Π and x, and may be more efficient.

Sparse JL

We will sample random Π such that

- every column of Π has exactly s non-zero entries,
- every non-zero entry is a random $\pm 1/\sqrt{s}$.

It was shown that we can set $s = O(\varepsilon m)$.

Theorem 3. *$\exists c_1, c_2 > 0$, we can set $m = c_1 \varepsilon^{-2} \log(1/\delta)$ and $s = c_2 \cdot \varepsilon m$ such that $\forall x \in \mathbb{R}^d$,*

$$(1-\varepsilon)\|x\|_2^2 \le \|\Pi x\|_2^2 \le (1+\varepsilon)\|x\|_2^2$$

with probability $\ge 1 - \delta$.

Remark 1. *Unlike standard JL, we cannot use $1/\sqrt{s} \cdot \mathcal{N}(0,1)$ in every nonzero entry.*

We skip the proofs.

Fast JL

We will use Π that has the form $\Pi = \frac{1}{\sqrt{m}} S \cdot H \cdot D$, where S is $m \times d$, and H, D are $d \times d$ [6].

S is a random matrix such that Sx samples m random coordinates of x. Note that which entry of x goes to which entry of Sx, S is linear. As we mentioned above, S itself is not a good matrix for dimensionality reduction, as x may have very few entries contributing to most of its norm. However, when x has small $\|x\|_\infty$, i.e., no entry has very large value, S turns out to be effective. Thus, we can view HD as a preprocessing step that maps x to some x' with small ℓ_∞-norm. Formally, one can prove that

[6]

Lemma 1. *If* $\|x\|_\infty \leq \frac{T}{\sqrt{d}} \cdot \|x\|_2$*, then* $\|\frac{1}{\sqrt{m}} Sx\|_2 = (1 \pm \varepsilon)\|x\|_2$ *with probability* $1-\delta$ *for* $m = O(T^4 \varepsilon^{-2} \log(1/\delta))$*.*

Next, H is the (deterministic) $d \times d$ Hadamard matrix (we assume d is a power of two without loss of generality). The $2^k \times 2^k$ Hadamard matrices H_{2^k} can be defined recursively.

$$\begin{pmatrix} H_{2^{k-1}} & H_{2^{k-1}} \\ H_{2^{k-1}} & -H_{2^{k-1}} \end{pmatrix}$$

and $H_{2^0} = (1)$. Due to the recursive structure of H, although H is a dense $d \times d$ matrix, it is not hard to verify that Hx can be computed in $O(d \log d)$ time using recursion.

Finally, D is a random ± 1 diagonal matrix: each diagonal entry is a random ± 1, and all other entries are zeroes.

Since each entry of H is ± 1, each entry of HDx is (marginally) an inner product between x and a random ± 1 vector. One can prove the following lemma using concentration inequalities (and union bound).

Lemma 2. *The probability that* $\|HDx\|_\infty > \sqrt{2 \ln d/\delta} \cdot \|x\|_2$ *is at most* $O(\delta)$*.*

We skip the proofs of the lemmas. Combining the lemmas, we obtain that one can set the output dimension m to $O(\varepsilon^{-2} \log^2(d/\delta) \log(1/\delta))$, which is close to optimal. Applying another round of optimal JL transform gives us the optimal m. Note that multiplying a vector with H is the bottleneck in computation, which takes $O(d \log d)$ time. The final round of JL is also efficient ($O(\varepsilon^{-4} \log^2(d/\delta) \log^2(1/\delta))$), as we already begin with a low-dimensional vector.

Bibliography

[1] William B. Johnson and Joram Lindenstrauss. Extensions of Lipschitz mappings into a Hilbert space. Contemporary Mathematics, 1984.

[2] Noga Alon. Problems and results in extremal combinatorics-I. Discrete Mathematics, 273(1-3):31-53, 2003.

[3] Kasper Green Larsen and Jelani Nelson. Optimality of the Johnson-Lindenstrauss lemma. FOCS, 2017.

[4] Daniel M. Kane, Jelani Nelson. Sparser Johnson-Lindenstrauss Transforms. J. ACM 61(1): 4:1-4:23, 2014.

[5] Nir Ailon, Bernard Chazelle. The Fast Johnson-Lindenstrauss Transform and Approximate Nearest Neighbors. SIAM J. Comput. 39(1): 302-322, 2009.