princeton univ. F'18 cos 521: Advanced Algorithm Design

Lecture 10: Dimensionality Reduction and the Johnson-Lindenstrauss Lemma

Lecturer: *Christopher Musco*

1 Preliminaries

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. We saw an example in Lecture 4 where the "bag-of-words" model was used to represent documents. In this case, we constructed a vector that had length N, where N is the *total number of words in the language the documents are written in.* At position $i \in 1, \dots N$ we recorded the number of occurrences of word i. The bag-of-words model is often extended to include data on all n-grams in a document – i.e. counts of every possible phrase of n words. When n-grams are include, our documents are represented as vectors of length $O(N^n)$.

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.

2 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.

3 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 2 (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.

We're going to choose f randomly. In particular, let G be a $d \times m$ *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). More on this next lecture.

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

Theorem 3 ((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 $(1-\epsilon)\|x\|_2 \leq \|\Pi x\|_2 \leq (1+\epsilon)\|x\|_2$. Then, to prove Theorem 2 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 2. So, we can focus our attention on proving Theorem 3.

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 $\mathrm{Var}[w_i] = \mathbb{E}[w_i^2]$. It follows that:

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

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 4 (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"[1].

So $w_i \sim \mathcal{N}(0, \|x\|^2) = \|x\|_2 \cdot \mathcal{N}(0,1)$. It follows that w_i^2 is a χ^2 (chi-squared) random variable and $\frac{1}{m}\sum_{i=1}^{m} w_i^2$ is a chi-squared random variable with m degrees of freedom. You can look up the CDF on Wikipedia for a χ^2 tail bound, but it essentially concentrates around its mean as well as a Gaussian. In particular, if $v = \frac{1}{m}\sum_{i=1}^{m} w_i^2$, then[2]:

$$\Pr\left[|\mathbb{E}v - v| \geq \epsilon \mathbb{E}v\right] \leq 2e^{-m\epsilon^2/8}.$$

[1] 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.

[2] See e.g. https://www.stat.berkeley.edu/~mjwain/stat210b/Chap2_TailBounds_Jan22_2015.pdf

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

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

with probability $1 - \delta$. □ It's worth noting that Theorem 2 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 [2]. The fully tight result was only obtained last year [3]. The result was proven first for *linear embeddings* and then extended to a lower-bound for all possible functions f.

4 Applications

There are many, many applications of the JL lemma. Here are a few that we will see on the problem set or in later classes:

- Approximate all-pairs distances in $O(n^2 \log n + nd)$ time vs. the naive $O(n^2 d)$ time.
- Approximate distance based clustering.
- Approximate support vector machine (SVM) classification and more.
- Sparse recovery.compressed sensing.
- Approximate linear regression.

4.1 Linear regression

In addition to its use in proving the original lemma about distances, the (ϵ, δ)-JL property for norm preservation is often directly useful in applications. Furthermore, many applications crucially use the *linearity* of the Johnson-Lindenstrauss embedding, not just its approximation properties. Here we consider a classic example: least squares regression.

Given n data vectors $a_1, \ldots, a_n \in \mathbb{R}^d$ and n response values $y_1, \ldots, y_n \in \mathbb{R}$. Usually we think of $a_1, \ldots, a_n$ as the rows in an $n \times d$ matrix A and $y_1, \ldots, y_n$ as entries in n length vector y. Goal:

$$\min_{x \in \mathbb{R}^d} \sum_{i=1}^{n} (a_i \cdot x - y_i)^2 = \min_{x \in \mathbb{R}^d} \|Ax - y\|_2^2 \tag{3}$$

Typically this probably requires $O(nd^2)$ time to solve (you will show this on the homework). We will speed this up by reducing n using the Johnson-Lindenstrauss Lemma. In particular, let $\Pi \in \mathbb{R}^{m \times n}$ be chosen from a random family of matrices satisfying Theorem 3. To obtain an approximate solution we will solve the "sketched" problem:

$$\min_{x \in \mathbb{R}^d} \|\Pi A - \Pi y\|_2^2, \tag{4}$$

which can be solved in $O(md^2)$ time (once πA and Πy are computed – more on this next class). We want to prove that a solution to this smaller problem is a good approximate solution to the original. Before doing so, we claim a simpler result:

Lemma 5
As long as $m = O(\log(1/\delta)/\epsilon^2)$ then, for any particular x,

$$(1-\epsilon)\|Ax - y\|_2^2 \leq \|\Pi Ax - \Pi y\|_2^2 \leq (1+\epsilon)(1-\epsilon)\|Ax - y\|_2^2$$

with probability $1-\delta$.

This is a direct consequence of Theorem 3.

If we could show the same result *for all* x then we would be in good shape. Specifically, let x^* be the optimal solution for the original regression problem (3) and let $\tilde{x}^*$ be the optimal solution for the sketched problem (4). We have:

$$\|A\tilde{x}^* - y\|_2^2 \leq \frac{1}{1-\epsilon}\|\Pi A\tilde{x}^* - \Pi y\|_2^2 \leq \frac{1}{1-\epsilon}\|A\Pi x^* - \Pi y\|_2^2 \leq \frac{1+\epsilon}{1-\epsilon}\|Ax^* - y\|_2^2$$

For $\epsilon \leq .25$, $\frac{1+\epsilon}{1-\epsilon} \leq 1 + 3\epsilon$. So we would get a relative error approximation to the regression problem. Question: For the argument above, why did we need a bound *for all* x?

But how would we extend Lemma 5 to all x? We certainly can't use a union bound argument – there are an infinite number of possible vectors x.

Notice that to extend Lemma 5 to all x, it suffices to show that Π preserves the length of all vectors in the $d+1$ dimensional linear subspace spanned by A's d columns and y. Next class we will prove the following:

Theorem 6 (Subspace Embedding)
Let U be a d dimensional linear subspace. If Π is chosen as described, with scaled random Gaussian entries (any transformation satisfying Lemma 3 works), and $m = O\left(\frac{d+\log(1/\delta)}{\epsilon^2}\right)$ then, for all $v \in U$,

$$(1-\epsilon)\|v\|_2^2 \leq \|\Pi v\|_2^2 \leq (1+\epsilon)\|v\|_2^2.$$

This results lets us obtain an approximate solution with a regression problem with just $O(d)$ examples, which takes $O(d^3)$ time to solve.

We will prove Theorem 6 using what's known as an ϵ-net argument. In particular, we will show that there are $O(1/\epsilon)^d$ fixed points in U such that, if we can establish the claim for all of those points, we establish it for all $x \in U$. Doing so will require taking a union bound over all of these points, which leads to a dependence on $\log(O(1/\epsilon)^d) = O(d\log(1/\epsilon))$.

Subspace embeddings have proven valuable far beyond regression: they're important in applying dimensionality reduction to other linear algebra problems, sparse recovery problems, and graph problems. Moreover, the *proof technique* is a very powerful and commonly used approach: it's useful to have in our toolkit.

References

[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 combinatoricsI. Discrete Mathematics, 273(1-3):31 53, 2003.

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