COMS 4995: Unsupervised Learning (Summer '18) June 7, 2018

Lecture 6 – Proof for JL Lemma and Linear Dimensionality Reduction

Instructor: *Nakul Verma* Scribes: *Ziyuan Zhong, Kirsten Blancato*

This lecture gives a proof of JL-lemma and introduces linear dimensionality reduction techniques including: RP (Random Projection), PCA (Principal Component Analysis), LDA (Linear Discriminant Analysis) and MDS (Multidimensional scaling).

1 JL-Lemma

1.1 Recall

Theorem 1 (Johnson-Lidenstrauss "flattening" lemma, 1984)**.** *Pick any $0 < \epsilon < \frac{1}{2}$. Then for any integer n, let $d > \lceil \frac{4}{\epsilon^2}(2\ln n + \ln 3)\rceil$, that is, $d > \Omega(\frac{\ln n}{\epsilon^2})$. Then for any set $V \subset \mathbb{R}^D$, s.t. $|V| = n$, there exists a linear map $f : \mathbb{R}^D \to \mathbb{R}^d$ s.t. $\forall u, v \in V$,*

$$(1-\epsilon)||u-v||_2^2 \leq ||f(u)-f(v)||_2^2 \leq (1+\epsilon)||u-v||_2^2$$

Remark 2. *Since $\sqrt{1-\epsilon} \leq 1-\epsilon$ and $1+\epsilon \leq \sqrt{1+\epsilon}$, the embedding is thus a D-embedding where the distortion $D = \frac{1+\epsilon}{1-\epsilon} \leq 1+5\epsilon$ because $\frac{1}{1-\epsilon} \leq 1+2\epsilon$ $\forall 0 < \epsilon < \frac{1}{2}$. The above holds true for any random d-dim subspace (in D-dim) with high probability (minor global scaling).*

Lemma 3 (Concentration of Measure)**.** *Pick any $0 < \epsilon < \frac{1}{2}$, fix any unit vector $w \in \mathbb{R}^D$*(i.e. $||w|| = 1$)*, let $\phi : \mathbb{R}^D \to \mathbb{R}^d$, $d < D$, be a random subspace map. Then,*

$$\Pr_{\phi}\left[||\phi(w)||^2 < (1-\epsilon)\frac{d}{D} \text{ or } ||\phi(w)||^2 > (1+\epsilon)\frac{d}{D}\right] \leq 3e^{-d\epsilon^2/4}.$$

For JL, we'll want to project to a random subspace, then scale by $\sqrt{D/d}$.

1.2 Proof of JL-Lemma

Proof. Because ϕ is linear, there is a corresponding matrix $P \in \mathbb{R}^{d\times D}$ s.t. $\phi(w) = Pw$.
$f := \sqrt{\frac{D}{d}}\phi$, so $f(w) = \sqrt{\frac{D}{d}}Pw$. For any distinct $u, v \in V$,

$$
\begin{aligned}
&\Pr\left[\exists u, v \in V \text{ s.t. } ||f(u)-f(v)||^2 < (1-\epsilon)||u-v||^2 \text{ or } ||f(u)-f(v)||^2 > (1+\epsilon)||u-v||^2\right] \\
&\qquad \leq \sum_{\substack{(u,v)\in V\times V \\ \text{unordered pairs}}} \Pr_{\phi}\Big[||f(u)-f(v)||^2 < (1-\epsilon)||u-v||^2 \\
&\qquad\qquad\qquad\qquad \text{or} \quad ||f(u)-f(v)||^2 > (1+\epsilon)||u-v||^2\Big] \\
&\qquad = \sum_{\substack{(u,v)\in V\times V \\ \text{unordered pairs}}} \Pr_{\phi}\left[\left|\left|\phi\left(\frac{u-v}{||u-v||}\right)\right|\right|^2 < (1-\epsilon)\frac{d}{D}\right. \\
&\qquad\qquad\qquad\qquad \left. \text{or} \quad \left|\left|\phi\left(\frac{u-v}{||u-v||}\right)\right|\right|^2 > (1+\epsilon)\frac{d}{D}\right] \\
&\qquad \leq \binom{n}{2} 3e^{-d\epsilon^2/4} < 1
\end{aligned}
$$

where the first inequality is a union bound, and the last inequality holds if we choose d such that:

$$
d > \left\lceil \frac{4}{\epsilon^2}(2\ln n + \ln 3) \right\rceil.
$$

The inner inequality follows from the linearity of $f = \sqrt{\frac{D}{d}}\phi$, followed by an application of Lemma 3:

$$
\begin{aligned}
||f(u)-f(v)||^2 < (1-\epsilon)||u-v||^2 &\Leftrightarrow \left|\left|\sqrt{\frac{D}{d}}\phi\left(\frac{u-v}{||u-v||}\right)\right|\right|^2 < (1-\epsilon) \\
&\Leftrightarrow \left|\left|\phi\left(\frac{u-v}{||u-v||}\right)\right|\right|^2 < (1-\epsilon)\frac{d}{D}.
\end{aligned}
$$

□

Recap:

- JL is a linear dimensionality-reduction technique. The goal is to preserve ℓ_2 distances up to distortions of $1 \pm \epsilon$.
- This is also a concentration result, $||\phi(w)||^2 < (1-\epsilon)d/D$, where $||\phi(w)||^2$ is the actual length of a particular draw and d/D is the expected length. The actual draw will be concentrated towards a specific value, typically the expected value.

1.3 Aside: A list of concentration inequalities

Markov
Chebychev
Chernoff
Hoeffding
Bernstein

Effron-Stein
Azuma
Mcdiarmid
Talagrand

2 Linear Dimensionality Reduction

The goal of JL is to presere ℓ_2 distances. However, depending on what property you care about, there will be different linear dimensionality reduction techniques appropriate for your task.

Common linear dimensionality reduction techniques:

- RP (Random Projection)
- PCA (Principal Component Analysis)
- LDA (Linear Discriminate Analysis)
- MDS (Multi-dimensional Scaling)
- ICA/BSS (Independent Component Analysis/Blind Source Separation)
- CCA (Canonical Correlation Analysis)
- DML (Distance Metric Learning)
- Factor (Factor Analysis)
- NMF/MF ((Non-negative) Matrix Factorization)

2.1 RP (Random Projection)

Method

For random projection, $P \in \mathbb{R}^{d \times D}$ with $P_{ij} = \mathcal{N}(0,1)$. i.e.

$$P = \begin{bmatrix} \mathcal{N}(0,1) & \mathcal{N}(0,1) & \cdots \\ \vdots & \ddots & \end{bmatrix}$$

If a projection matrix is wanted, apply Gram-Schmidt to P.

2.1.1 Practical application

PCA has time complexity $O(n^3)$, if we assume both the number of data points and the number of features are equal to n. In practice, if we are working with large amounts of data our first instinct to speed up PCA might be to subsample the data, i.e. if our dataset has 10k samples in $\mathbb{R}^{10k}$, randomly subsample 1k points and perform PCA on this reduced dataset. However, the quality of the k^{th} eigenvector of the subsampled data decays with respect to the k^{th} eigenvector of the full dataset. While the first eigenvector of the subsampled and full dataset will be similar, all subsequent eigenvectors of the subsampled data will be of worsening quality.

The better approach to speeding up PCA is to first do a random projection, and then perform PCA. While the distances between points will be distorted within $1 \pm \epsilon$, the quality of the eigenvectors will be better.

2.2 PCA (Principal Component Analysis)

2.2.1 Outline

Data: $x_1, x_2, ..., x_n \in \mathbb{R}^d$

Goal: Find the best linear transformation $\phi : \mathbb{R}^d \to \mathbb{R}^k$ that best maintains reconstruction accuracy. Equivalently, minimize aggregate residual error.

Define: $\Pi^k : \mathbb{R}^d \to \mathbb{R}^d$ minimize $\frac{1}{n} \sum_{i=1}^{n} ||x_i - \Pi^k(x_i)||^2$

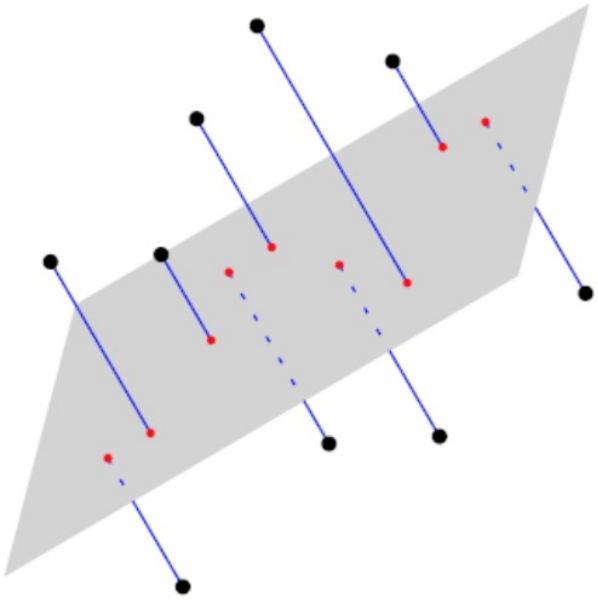

Figure 1: An illustration of PCA

2.2.2 Method

A k dimensional subspace can be represented by $q_1, ..., q_k \in \mathbb{R}^d$ orthonormal vectors.
The projection of any $x \in \mathbb{R}^d$ in the $span(q_1, ..., q_k)$ is given by

$$\underbrace{(\sum_{i=1}^{k} q_i q_i^T)}_{\Pi^k} x = \sum_{i=1}^{k} (q_i \cdot x) q_i$$

To represent it in $\mathbb{R}^k$ (using basis $q_1, ..., q_k$) the coefficients simply are: $(q_1, x), ..., (q_k, x)$.

2.2.3 The $k = 1$ case

In $k = 1$ case, the objective is the following:

$$\text{minimize}_{||q||=1} \frac{1}{n} \sum_{i=1}^{n} ||x_i - (qq^T)x_i||^2$$

Equivalently,

$$\text{maximize}_{||q||=1} q^T (\frac{1}{n} XX^T) q,$$

where $\frac{1}{n} XX^T$ is the covariance of data, if the data is mean-centered. The solution is the top eigenvector $(1/n)XX^T$.

Remark: For any q, the quadratic form $q^T(\frac{1}{n}XX^T)q$ is the empirical variance of data in the direction q.

2.2.4 General k case

The general k case is similar and the solution is basically the top k eigenvectors of the matrix XX^T.

2.3 LDA (Linear Discriminate Analysis)

2.3.1 Motivation

The goal of PCA is to minimize the reconstruction error. However, this is not necessarily going to help for classification purposes.

2.3.2 Method

Define $\mu_1 = \frac{1}{|C_1|} \sum_{x \in C_1} x$ and $\mu_2 = \frac{1}{|C_2|} \sum_{x \in C_2} x$. Define $\bar{\mu}_1 = w^T \mu_1$ and $\bar{\mu}_2 = w^T \mu_2$
One intuitive formulation would be:

$$\max_{w, s.t. ||w||=1} L(w) = |\bar{\mu}_1 - \bar{\mu}_2| = |w^T \mu_1 - w^T \mu_2|$$

It is easy to see that $w^* = \frac{\mu_1 - \mu_2}{||\mu_1 - \mu_2||}$.

Problem: Maintaining the distance between the means is not sufficient. What we want are for the:

- class means to be as far as possible after projection, and
- class variances to be as small as possible.

After projection, denote $\bar{s}_1^2 = \sum_{\bar{x} \in C_1} (\bar{x} - \bar{\mu}_1)^2$, $\bar{s}_2^2 = \sum_{\bar{x} \in C_2} (\bar{x} - \bar{\mu}_2)^2$. The formulation according to our criteria would be the following:

$$\max_w L(w) = \frac{(\bar{\mu}_1 - \bar{\mu}_2)^2}{\bar{s}_1^2 + \bar{s}_2^2}$$

We will expand each term in this equation in the following:

$$\begin{aligned}
\bar{s}_1^2 &= \sum_{\bar{x} \in C_1} (\bar{x} - \bar{\mu}_1)^2 \\
&= \sum_{\bar{x} \in C_1} (w^T x - w^T \mu_1)^2 \\
&= w^T \Big(\sum_{\bar{x} \in C_1} (x - \mu_1)(x - \mu_1)^T \Big) w \\
&= w^T S_1 w \text{ where } S_1 \text{ is the scatter matrix for 1}
\end{aligned}$$

Similarly, $\bar{s}_2^2 = w^T S_2 w$

$$\begin{aligned}
(\bar{\mu}_1 - \bar{\mu}_2)^2 &= (w^T(\mu_1 - \mu_2))^2 \\
&= w^T(\mu_1 - \mu_2)(\mu_1 - \mu_2)^T w \\
&= w^T S_B w \text{ where } S_B \text{ is the between class scatter matrix}
\end{aligned}$$

Note: rank of $S_B = 1$.
Thus,

$$L(w) = \frac{w^T S_B w}{w^T S_w w}$$

It follows that it has derivative (if we ignore its denominator):

$$\frac{d}{dw} L(w) = w^T S_w w 2 S_B w - w^T S_B w 2 S_w w$$

When we divide it by $2 w^T S_w w$, we get:

$$S_B w - \frac{w^T S_B w}{w^T S_w w} (S_w w) = S_B w - L(w)(S_w w)$$

If we let it equal to 0, we get:

$$S_B w = S_w L(w) w \Leftrightarrow S_w^{-1} S_B w = L(w) w$$

Maximizing $L(w) \Leftrightarrow$ finding eigenvector corresponding to the largest eigenvalues of $S_w^{-1} S_B$.

Remark: When you have c-classes, LDA will find a $c - 1$ subspace.

2.4 MDS (Multi-dimensional Scaling)

MDS is useful when you don't have a Euclidean representation of the data, and wish to come up with one based on distances between data points. There are three types of MDS:

- classical MDS
- metric MDS
- non-metric MDS

Here, we are mostly going to discuss metric MDS.

2.4.1 Metric MDS method

Given a distance matrix $P \in \mathbb{R}^{n \times n}$, define the "stress function", for $x_i \in \mathbb{R}^k \ \forall i = 1, ..., n$, to be the following:

$$S(x_1, ..., x_n) = \sum_{i<j} (||x_i - x_j|| - p_{ij})^2$$

We want to minimize the stress function over the data points $x_1, ..., x_n$.
The problem can be formulated as the following:

$$\min S(x_1, ..., x_n)$$
$$s.t. \sum x_i = 0$$

There is no easy way to write this as an eigenvalue problem, so to solve this we can simply do any gradient based optimization.

2.4.2 Non-metric MDS

The goal on non-metric MDS is to maintain the order of distances between data points. For example, if x_i is closer to x_j than x_k, then maintain this ordering. Can make $||x_i - x_j||$ into any monotonic function.

References

[1] An Elementary Proof of a Theorem of Johnson and Lindenstrauss. Sanjoy Dasgupta, Anupam Gupta.

[2] Verma 4771 Machine Learning Lecture Slide: `http://www.cs.columbia.edu/~verma/classes/ml/lec/lec8_unsupervised_dim_redux.pdf`

[3] Linear Dimensionality Reduction- Survey, Insights, and Generalizations. John P. Cunningham, Zoubin Ghahramani.