CMSC 35900 (Spring 2009) Large Scale Learning **Lecture: 1**

Dimensionality Reduction

Instructors: Sham Kakade and Greg Shakhnarovich

1 Introduction

This course will cover a number of methods related to dealing with large datasets. Recently, the term 'large scale learning' refers to the supervised learning regime where the labeled sample size is 'large' (issues related to optimization when $n \to \infty$). However, we do not mean that here. Rather, this course will focus on a number of issues related to learning high dimensions.

2 Karhunen-Loeve theorem

Consider a centered stochastic process $[X]_t$, for $t \in [0, 1]$. Centered means that $\mathbb{E}[X]_t = 0$. In the discrete case we have a random vector $X \in \mathbb{R}^d$ where $[X]_t$ is the $t - thcomponent$.

The autocovariance function is:

$$K(t, s) = Cov(X_t, X_s) =< X_t | X_s >= \mathbb{E}[X_t X_s]$$

which can be viewed as a kernel.

The corresponding integral operator is:

$$T_K \Phi(t) = \int_0^1 K(t, s)\Phi(t)ds$$

which has eigenvectors and eigenvalues.

Theorem 2.1. *(KL) Consider the centered stochastic process X_t for $t \in [0, 1]$ with covariance function $K(t, s)$. Suppose this covariance function is continuous in t, s. By Mercer's theorem, the corresponding integral operator on T_K has an orthonormal basis of eigenvectors, $\{e_i(t)\}$. Define:*

$$Z_i = \int_0^1 X_t e_i(t)dt$$

Then Z_i are centered orthogonal random variables and:

$$X_t = \sum_i^\infty e_i(t)Z_i$$

(where convergence is in the mean and uniform in t). Also,

$$Var(Z_i) = \mathbb{E}(Z_i^2) = \lambda_i$$

where λ_i is the eigenvalue corresponding to e_i.

2.0.1 Wiener Processes

For things like Brownian motions, these things have well defined answers.

2.0.2 Mercer's Theorem

Theorem 2.2. *Suppose K is a continuous symmetric non-negative definite kernel. Then there is an orthonormal basis $\{e_i\}$ on $L_2[0,1]$ consisting of eigenfunctions of T_K such that the corresponding sequence of eigenvalues $\{\lambda_i\}$ is nonnegative. The eigenfunctions corresponding to non-zero eigenvalues are continuous on $[0,1]$ and K has the representation:*

$$K(s,t) = \sum_{i=1}^{\infty} \lambda_j e_j(s) e_j(t)$$

where the convergence is absolute and uniform.

In finite dimensions,

Theorem 2.3. *Suppose K is a square symmetric matrix. Then there exists a decomposition:*

$$S = UDU^\top$$

where D is diagonal and U is orthogonal. The diagonal entries of D are the eigenvalues and corresponding columns of U are the eigenvalues. If K is non-negative definite then all the eigenvalues are positive.

3 PCA

Given a finite sample $X_1, \ldots X_n$, we have the empirical covariance matrix:

$$\hat{K} = \frac{1}{n} \sum_{i=1}^{n} X_i X_i^\top$$

PCA is just the KL transform of the empirical Kernel matrix.

Alternative viewpoint:

$$w_1 = \text{argmax}_{w:\|w\|=1} \hat{\sigma}^2 (w \cdot X) = \text{argmax}_{w:\|w\|=1} \frac{1}{n} \sum_{i=1}^{n} (w \cdot X_i)^2$$

and z_1 is the value. Next,

$$X_i \leftarrow X_i - \sum_j (w_1 \cdot X_i) w_1$$

and repeat to find e_2 and z_2 and so on.

An alternative viewpoint is provided by the SVD.

4 The Best Fitting Subspace and the SVD

Now we let A be a general matrix. The maximal *singular value* is $\max_{\|w\|=1} \|Av\|_2$ and the argmax is the corresponding singular vector. We let A_i be a row of A.

Lemma 4.1. *For an arbitrary matrix $A \in \mathbb{R}^{n \times d}$,*

$$\text{argmax}_{\|w\|=1} \|Aw\|^2 = \text{argmin}_{\|w\|=1} \|A - (Aw)w^\top\|_F^2 = \text{argmin}_{\|w\|=1} \sum_i \|A_i - (A_i \cdot w)w\|^2$$

where $\|\cdot\|_F^2$ is the Frobenious norm (the Frobenious norm of a matrix M is $\|\cdot\|_F^2 = \sum_{i,j} M_{i,j}^2$).

Proof. The proof essentially follows from the Pythagorus theorem. □

Theorem 4.2. *(SVD) Define the k dimensional subspace W_k as the span of the following k vectors:*

$$w_1 = \text{argmax}_{\|w\|=1}\|Aw\|^2 \quad (1)$$

$$w_2 = \text{argmax}_{\|w\|=1, w\cdot w_1=0}\|Aw\|^2 \quad (2)$$

$$\vdots \quad (3)$$

$$w_k = \text{argmax}_{\|w\|=1, \forall i\leq k,\, w\cdot w_i=0 \forall i\leq k}\|Aw\|^2 \quad (4)$$

Then W_k is optimal in the sense that:

$$W_k = \text{argmin}_{dim(W)=k}\sum_i distance(A_i, W_k)^2$$

Furthermore,

$$\sigma_1 = \|Aw_1\| \geq \sigma_2 = \|Aw_2\| \geq \ldots \sigma_{\min\{n,d\}} = \|Aw_{\min\{n,d\}}\|$$

Let $\sigma_i u_i = Av_i$, so u_i is unit length. Then the set $\{u_i\}$ is orthonormal (so is $\{v_i\}$ by construction) and the SVD decomposition of A is:

$$A = \sum_i \sigma_i u_i v_i^\top = Udiag(\sigma_1, \ldots \sigma_{\min\{n,d\}})V^\top$$

where U and V are orthogonal matrices with rows $\{u_i\}$ and $\{v_i\}$, respectively.

Proof. The interesting part of the proof is that $\{u_i\}$ is orthonormal — the rest of the proof essentially follows by construction. □

As a corollary, we have that:

Corollary 4.3. *Among all rank k matrices D, $A_k = \sigma_{i=1}^k u_i = Av_i$ is the one which minimizes $\|A - D\|_F$. Further,*

$$\|A - D\|_F^2 = \sum_{i=k+1}^{\min\{n,d\}} \sigma_i^2$$

4.1 Computation

Computing an SVD is often intensive for large matrices. There are increasingly fast algorithms for this.

4.2 Latent Semantic Analysis (LSA) or LSI (LSIndexing)

Let look at an application to information retrieval.

Say we represent a document by a vector d and a query by a vector q, then one score of a match is the cosine score:

$$\text{similarity} = \frac{d \cdot q}{\|d\|\|q\|}$$

The naive approach is to just use a bag of words to represent these vectors — so the length of the vector is the number of words (in the language or corpus) and the entry in the k-th position denote the number of times that word appears. Using just bag of word counts, two difficulties with this approach are synonymy and polysemy.

LSA is a simple way to address this, using a vector space method. Here, let X be the term/document matrix. Let:

$$X = UDV^\top$$

be the SVD of X. We can work with the k-rank approximation to X:

$$X_k = U_k D_k V_k^\top$$

So we represent each document and (new) query as a k-vector. The document j is just represented by V_j. A vector query q is now represented as:

$$x_{term}(q) = D_k^{-1} U_k^\top q x_{document}(d) = D_k^{-1} V_k^\top d$$

Now for recall we can just use the cosine score for retrieval.

5 References

Material used was Wikipedia and Santosh Vempala's lecture notes. Further reading about LSA can be found in the Information Retrieval book by Manning and Raghavan.