Math 123
Lecture Notes

Jan 19th

Linear algebra review:

Let $A \in \mathbb{R}^{m \times n}$, with entries $a_{i,j}$, $0 \leq i \leq m, 0 \leq j \leq n$. Of course, if $m = n$, A is square.

Recall that if $x \in \mathbb{R}^n$, we say that we can act on x via $Ax \in \mathbb{R}^m$, where $y = Ax$ has entries $y_i = \Sigma_{j=1}^n a_{i,j} x_j$

Somewhat obvious remark: Vector inner products can be realized as matrix multiplication, where we realize the inner product for $x, y \in \mathbb{R}^d$ as $\Sigma_{i=1}^d x_i y_i$. Alternatively, we may consider it as matrix multiplication of xy^T, where we realize x, y as $1 \times d$ matrices.

On the other hand, if we still realize these as $1 \times d$ matrices, we can call the outer product something like $x^T y \in \mathbb{R}^{d \times d}$.

Remark: if we have a rank 1 matrix, then we have a theorem that states that we can always decompose it as the outer product of two vectors.

Recall we may define $\|x\| = \sqrt{< x, x >}$

Recall that we may call the angle of two vectors as $\arccos\left(< x, y > / \|x\|^2 \|y\|^2\right)$

Recall that, for $A \in \mathbb{R}^{m \times n}$, we call $A^T \in \mathbb{R}^{n \times m}$, where $a_{i,j} = a_{j,i}^T$.

If $A^T = A$, then we call A symmetric. Note that of course, if A is symmetric, then A must be square.

Remark: If A is symmetric, then we may decompose A as:

$$A = U^T \Lambda U$$

where

(1) U is orthogonal/orthonormal (that is, $U^T U = U U^T = I$)

(2) Λ is diagonal

In particular, the values of Λ are the eigenvalue of the corresponding eigenvector in U.

Why would we care?

1) U orthogonal $\implies \{u_i\}_{i=1}^n \subset \mathbb{R}^n$ are linearly independent, that is, they span $\mathbb{R}^n$.

2) Thus, when we want to compute Ax, we may rewrite $x = \Sigma_{i=1}^n c_i u_i$

3) Thus, when we take $y = Ax$ rewritten in the eigenbasis, this is very easy: we can compute this easily as $y = \Sigma_{j=1}^n c_j \lambda_j u_j$

Jan 24th

Principal Component Analysis:

Suppose we have data points $\{x_i\}_{i=1}^n \subset \mathbb{R}^d$. We wish to find the directions of maximum variance in the data, and reduce the analysis into components $< D$, in order to capture as much of the variance in a lower dimensional subspace. In other words, we want to find low-dimensional structure in the data, generally when D is large.

We do this by taking a spectral value decomposition of the empirical covariance matrix.

We organize the data into a $n \times D$ matrix, where a row is a data point, and each column is a measurement of a variable across data points.

Given an orthogonal matrix $U \in \mathbb{R}^{n \times n}$, for any $x \in \mathbb{R}^n$, we may express $x = \sum_{i=1}^n c_i u_i$ where u_i is the i-th row/column of U. That is, the u_i span $\mathbb{R}^n$. However, this is not great for a generic matrix M, since this grows as the cube of the dimension. (Interesting point: solving a generic linear system grows as n^3 where n is the dimension). However, for an orthogonal matrix, we abuse the fact that $u_i \cdot u_j = \delta_{ij}$. We notice then that $Ux = (c_1, ..., c_n)$. (note that we can of course, compute each coefficient one by one by taking a dot product against each of the orthogonal vectors in turn) In particular, this has complexity n^2.

In any case, in this context, PCA is an effort to learn the "best" orthogonal matrix for my data. When we say best, we mean that if we were to take cutoffs, each choice of component preserves the maximum variance in the data, under the projection onto the component. Formally:

The projection of the data onto u_1 should be variance maximizing over all possible choices of u_1. More generally, we wish this to be variance maximizing over n components, for all n dimensional subspaces.

First, recall what variance means: for $x_1, ..., x_n \in \mathbb{R}$, we define the variance as:

$$\sigma = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2, \mu = \frac{1}{n}\sum_{i=1}^{n} x_i$$

where we call μ the mean.

Then, we need only compute the projection onto a direction $u \in \mathbb{R}^{D \times 1}$ by taking $u^T x$ for a data point $x \in \mathbb{R}^{D \times 1}$, and take the variance of these pure numbers. So, we wish to find:

$$\arg \max_{u \in \mathbb{R}^n, \|u\|=1} \frac{1}{n}\sum_{i=1}^{n}[u^T x_i - \mu]^2, \mu = \frac{1}{n}\sum_{i=1}^{n} u^T x_i$$

This kinda sucks. However, we claim that WLOG, that $\mu = 0$, because we can recenter the data at the origin by taking $\overline{x} = x - \mu$, where this μ is the mean of the original data points.

Well, let's now do some tricks:

$$(u^T x_i)^2 = (u^T x_i)(u^T x_i)^T = u^T x_i x_i^T u = u^T (x_i x_i^T) u$$

where we use the fact that since $u^T x_i$ is 1×1, so $(u^T x_i)(u^T x_i)^T$

Therefore, using the linearity:

$$\frac{1}{n}\sum_{i=1}^{n}[u^T x_i]^2 = \frac{1}{n}\sum_{i=1}^{n} u^T(x_i x_i^T)u = \frac{1}{n}u^T\left[\sum_{i=1}^{n}(x_i x_i^T)\right]u$$

We call

$$\Sigma = \frac{1}{n}\sum_{i=1}^{n}(x_i x_i^T)$$

the (empirical) covariance matrix of the data, where we understand $x_i \in \mathbb{R}^{D\times 1}$, i.e. a column vector.

In this case, we can do this via Lagrange multipliers.

$$\arg\max_{u\in\mathbb{R}^n, \|u\|=1} u^T\Sigma u - \lambda\left[uu^T - 1\right]$$

Differentiating with respect to u, we end up changing this into the case where

$$\frac{\partial}{\partial u}\left[u^T\Sigma u - \lambda(u^T u - 1)\right]$$

, where we claim that this is

$$2\Sigma u - 2\lambda u = 0 \implies \Sigma u = \lambda u$$

that is, u is an eigenvector of Σ.

Jan 26th

We recall that, given data $\{x_i\}_{i=1}^n \subset \mathbb{R}^d$, we wish to find the directions of maximum variance in the data such that $u_1 \perp u_2 \perp ... \perp u_d$, such that u_1 represents the direction of maximum variance, $\{u_1, u_2\}$ to be a plane, etc.

Recall this is an optimization problem as such:

Define $F(u) = \frac{1}{n}\sum_{i=1}^{n}(u^T x_i)^2$. Where we noticed that we could rewrite this as:

$$F(u) = \frac{1}{n}u^T\Sigma u$$

via a transpose trick.

After taking the derivatives, we recall that this becomes a eigenvalue equation. In particular, we notice that because this is symmetric, by the spectral theorem, that we have a D dimensional orthonormal basis from the eigenvectors.

Now, let's prove that the greedy algorithm works.

Suppose it works up to m vectors. Then, let's rewrite the condition for the $m+1$ vector as a Lagrange multiplier:

$$\mathcal{L}(u) = u^T\Sigma u - \sum_{i=1}^{m}\alpha_i(u^T v_i) - \alpha_{m+1}(uu^T - 1)$$

Taking a derivative again, we see that:

$$\frac{\partial}{\partial u}\mathcal{L}(u) = 2\Sigma u - \sum_{i=1}^{m} \alpha_i v_i - 2\alpha_{m+1} u = 0$$

Here, we look at this with respect to each v_i in turn.

$$v_j^T \left[2\Sigma u - \sum_{i=1}^{m} \alpha_i v_i - 2\alpha_{m+1} u \right] = 0 \implies 2v_j^T \Sigma u - \alpha_j - 2\alpha m + 1 v_j^T u = 0$$

Uh. I think James doesn't know what he's doing here.

Jan 31st

Redoing last time's proof.

Let Σ be a covariance matrix. Let $\{v_i\}_{i=1}^{D}$ be a set of orthonormal eigenvectors such that the corresponding eigenvalues $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_D$.

Suppose we know that the k dimensional hyperplane of greatest variance is spanned by the span of $\{v_i\}_{i=1}^{k}$. We wish to show that if we impose that u is perpendicular to v_i for $0 \leq i \leq k$, $u^T u = 1$, then $u = v_{k+1}$.

Writing this as a Lagrange multiple question:

$$\mathcal{L}(u) = u^T \Sigma u - \sum_{i=1}^{m} \alpha_i (u^T v_i) - \alpha_{m+1}(uu^T - 1)$$

Differentiating again, and setting to 0 such that this is a critical point:

$$\frac{\partial}{\partial u}\mathcal{L}(u) = 2\Sigma u - \sum_{i=1}^{m} \alpha_i v_i - 2\alpha_{m+1} u = 0$$

We claim that we may show that $\alpha_i = 0$ for $i \leq i \leq m$. If we multiply both sides by v_j^T for $1 \leq j \leq m$:

Then, since Σ is normal, and u is perpendicular to every v_j, we have that:

$$\alpha_j v_j^T v_j = \alpha_j = 0$$

Thus, we have that:

$$2\Sigma u = 2\alpha_{m+1} u \implies \Sigma u = \alpha_{m+1} u$$

Thus, u is an eigenvector, and to maximize the variance, we take an eigenvector with largest eigenvalue remaining.

What is this computational complexity of PCA?

Note that we usually just count scalar multiplications and additions and count them as equal in this toy model.

Well, suppose we have $\{x_n\}_{i=1}^{n} \subset \mathbb{R}^D$.

(1) Centering data: compute $\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$. This has complexity nD

(2) Building $\Sigma = \frac{1}{n}\sum_{i=1}^{n} x_i^T x_i$. You can do some tricks, but it's of form nD^2

(3) Eigenvectors/values of Σ: of form D^3

So, we can say that overall, this has complexity of form $D^2(n + D)$, or $D^2 \max\{n, D\}$

Note that the eigendecomposition of Σ is available here because we have a symmetric, and thus normal matrix.

However, we can also do this for a matrix $A \in \mathbb{R}^{n\times d}$, for even $n \neq d$. We will lose potentially orthogonality though between left and right eigenvectors.

The singular value decomposition of a matrix $A \in \mathbb{R}^{n\times d}$ is the factorization:

$$A = U\Lambda V^T$$

such that Λ is "diagonal", $U \in \mathbb{R}^{n\times n}$, orthogonal, and $V^T \in \mathbb{R}^{d\times d}$, orthogonal.

More precisely, Λ has only non-0 entries when, if λ_{ij} represents the i-th row, j-th column, then λ_{ij} is non-0 only if $i = j$. Hence, we denote these as λ_i.

We call the columns of U the left singular vectors of A, and the rows of V^T the right singular vectors of A, and λ_i the singular values.