PCA Tutorial

Slides by Yawen Ma

Based on Sam & Marcus' tutorial notes

Image as n-Dimensional vector

- Image $X_i = [1,0,\ldots]^T x_{i1} + [0,1,\ldots]^T x_{i2} + \cdots$
- Image $X_i = B_1 y_{i1} + B_2 y_{i2} + \cdots = \sum_{j=1}^{D} B_j y_{ij}$

Principle Component Analysis

- Intuition
  - Pixel intensities are related
  - -> in new basis, only a few of the coordinates are significant
  - Less computation

Algorithm

- Find optimal set of principle components (basis)
- Compute patch coordinate in new basis
- Discard the axes with near 0 coordinates

$$(X_i = B_1 y_{i1} + B_2 y_{i2} + \cdots = \sum_{j=1}^{D} B_j y_{ij}\,, \quad X_i{}' = \sum_{j=1}^{d} B_j y_{ij})$$

Find basis

- Compute the average $\bar{X}$
- Subtract average from each patch $Z_i = X_i - \bar{X}$ (centralized)
- $Z = [Z_1, Z_2, \dots]$
- B_1, B_2 … are the eigenvectors of ZZ^T corresponding to d largest eigenvalues

Face recognition using PCA (Eigenface)

- Database creation
  - Input: N face images
  - Compute d eigenfaces (basis) :B_j
    - d < 10-15
  - Compute new coordinates for each input: $y_{ij} = {B^j}^T X_i$
  - Store B_j & y_{ij}
- Recognition
  - Input: a new image T
  - Express T by B_j (Compute T's new coordinate): t_j
  - Find closest y_i of t, and return X_i

Theory

- $Z = [Z_1, Z_2 , \dots]$
- Column: ith patch
- Row: jth coordinate

- $Y = [Y_1, Y_2 , \dots]$
- We want coordinates in the new basis:
  - Variance for each coordinate to be large
  - Covariance between every 2 coordinate to be small
- We want YY^T diagonal

Derivation

- $Z = BY$
- $Y = B^{-1}Z$
- Let A=B^{-1}, so $Y = AZ$

- Proof:
- $A^T = [e_1, e_2 ..]$, where e_j is the jth eigenvector of ZZ^T
- .. Will make YY^T diagonal

- (Define $A = B^{-1}$, so we have $Y = A * Z$.)
- Then

$$
\begin{aligned}
Y * Y^T &= (A * Z) * (A * Z)^T \\
&= A * Z * Z^T * A^T \qquad \text{[transpose rule]} \\
&= A * (Z * Z^T) * A^T \qquad \text{[regroup]}
\end{aligned}
$$

- So we want $Y * Y^T = A * (Z * Z^T) * A^T$ to be diagonal.

- Examine the eigenvector decomposition of (Z*Z^T):

$$(Z * Z^T) * e_i = e_i * \lambda_i \qquad [def]$$

$$(Z * Z^T) * [e_1 \ldots e_M] = [e_1 \ldots e_M] * diag(\lambda_1, \ldots, \lambda_M)$$

[stack the eqns]

$$(Z * Z^T) * E = E * L$$

[matrix notation]

- Left-multiply this by E^T:

$$E^T * (Z * Z^T) * E = E^T * E * L$$

$$= (E^T * E) * L \text{ [regroup]}$$

$$= I * L \quad \text{[eigenvecs are orthogonal, } E * E^T = E^T * E = I\text{]}$$

$$= L \quad \text{[eigenval matrix L is diagonal]}$$

- Observe that setting $A = E^T$ gives us

$$\begin{aligned} E^T * (Z * Z^T) * E \;&=\; A * (Z * Z^T) * A^T [A \;= E^T] \\ &= (A * Z) * (A * Z)^T \text{ [transpose rule]} \\ &= Y * Y^T \;\; [Y \;=\; A * Z \;= E^T * Z] \\ &= \; L \qquad\quad \text{[see above, note L is diagonal]} \end{aligned}$$

- and so satisfies our requirement that $Y * Y^T$ be diagonal.

Practical PCA

- Remember that Z is (M x N), with M >> N

  e.g. M = 75,000 pixels per face

  N = 200 faces in the database

- Compute eigenvectors of $Z^T Z$ first
- Then compute eigenvectors of $Z Z^T$

Proof

- Definition of eigenvectors for the *much* smaller $Z^T * Z$ (N x N):

$$(Z^T * Z) * e_i = \lambda_i * e_i \qquad \text{[def]}$$

$$(Z^T * Z) * [e_1 \ldots e_N] = [e_1 \ldots e_N] * diag(\lambda_1, \ldots, \lambda_n)$$

[stack the eqns]

$$(Z^T * Z) * E = E * L \qquad \text{[matrix notation]}$$

- Note: E and L are also (N x N)

- Now a little algebra ...

$$Z * (Z^T * Z) * E \quad = \ Z * E * L \qquad \text{[left multiply by Z]}$$

$$(Z * Z^T) * (Z * E) \ = \ (Z * E) * L \qquad \text{[regroup]}$$

- Define $e'_i = Z * e_i$, and $E' = Z * E$ (M x N), and observe how this form
- gives us N eigenvector equations for the *huge* matrix $Z * Z^T$(M x M):

$(Z * Z^T) * E' = E' * L$ [$E' = Z * E$]

$(Z * Z^T) * [e'_1 \ldots e'_N] = [e'_1 \ldots e'_N] * diag(\lambda_1, \ldots, \lambda_N)$

[matrix notation]

$(Z * Z^T) * e_i = \lambda_i * e'_i$ [unstack eqns]

- So to obtain N eigenvectors, E', for $Z * Z^T$, simply find the eigenvectors, E, for the “much” smaller $Z^T * Z$ (all N of them), and then transform them according to $E' = Z * E$.

- This smaller set of N (<< M) eigenvectors actually covers all the "interesting"/non-degenerate eigenvectors available.

- Why?
- We know that rank(Z) <= N = min(M,N), but it's also the case that rank($Z * Z^T$) = rank($Z^T * Z$) = rank(Z). So this tells us that the eigenvector decomposition for $Z * Z^T$ can have at most N non-zero λ_i's, i.e. we've covered them all.