Principal Component Analysis (PCA)

Salvador Dalí, “Galatea of the Spheres”

CSC411: Machine Learning and Data Mining, Winter 2017

Slides from Derek Hoiem and Alysha Efros
The space of all face images

- When viewed as vectors of pixel values, face images are extremely high-dimensional
  - 100x100 image = 10,000 dimensions
  - Slow and lots of storage
- But very few 10,000-dimensional vectors are valid face images
- We want to effectively model the subspace of face images
The space of faces

- Each image is a point in space
The space of all face images

- Eigenface idea: construct a low-dimensional linear subspace that contains most of the face images possible (possibly with small errors)

- Here: a 1D subspace arguably suffices
Rotating a Cloud to Be Axis-Aligned

• Consider the covariance matrix of all the points in a cloud

\[ \Sigma = \sum_i (x^{(i)} - \mu)(x^{(i)} - \mu)^T \]

• Using the Spectral Theorem, we know that we can diagonalize \( \Sigma \):

\[ R^T \Sigma R = D = \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_k \end{bmatrix}, \]

R is the matrix of the Eigenvectors of R
Now:

\[
\sum_i R(x^{(i)} - \mu)(R(x^{(i)} - \mu)^T) = \\
R\left(\sum_i (x^{(i)} - \mu)(x^{(i)} - \mu)^T\right) R^T
\]

\[
= R\Sigma R^T = D
\]

So if we rotate the \((x^{(i)} - \mu)\) using \(R\), the covariance matrix will be diagonal!
Change of Basis

• (On the board)
Reconstruction

• For a subspace with the orthonormal basis of size $k$ $V_k = \{v_0, v_1, v_2, \ldots v_k\}$, the best reconstruction of $x$ in that subspace is:
  \[ \hat{x}_k = (x \cdot v_0)v_0 + (x \cdot v_1)v_1 + \cdots + (x \cdot v_k)v_k \]
  – If $x$ is in the span of $V_k$, this is an exact reconstruction
  – If not, this is the projection of $x$ on $V$

• Squared reconstruction error: $(\hat{x}_k - x)^2$
Reconstruction cont’d

• \( \hat{x}_k = (x \cdot v_0)v_0 + (x \cdot v_1)v_1 + \cdots + (x \cdot v_k)v_k \)

• Note: in \( (x \cdot v_0)v_0 \),
  – \( (x \cdot v_0) \) is a measure of how similar \( x \) is to \( v_0 \)
  – The more similar \( x \) is to \( v_0 \), the larger the contribution from \( v_0 \) is to the sum
Representation and reconstruction

- Face $\mathbf{x}$ in “face space” coordinates:

$$\mathbf{x} \rightarrow [\mathbf{u}_1^T (\mathbf{x} - \mu), \ldots, \mathbf{u}_k^T (\mathbf{x} - \mu)]$$

$$= w_1, \ldots, w_k$$

- Reconstruction:

$$\hat{x} = \mu + w_1 u_1 + w_2 u_2 + w_3 u_3 + w_4 u_4 + \ldots$$
Reconstruction

After computing eigenfaces using 400 face images from ORL face database
Principal Component Analysis

• Suppose the columns of a matrix $X_{N \times K}$ are the datapoints ($N$ is the size of each image, $K$ is the size of the dataset), and we would like to obtain an orthonormal basis of size $k$ that produces the smallest sum of squared reconstruction errors for all the columns of $X - \bar{X}$.

  $- \bar{X}$ is the average column of $X$

• Answer: the basis we are looking for is the $k$ eigenvectors of $(X - \bar{X})(X - \bar{X})^T$ that correspond to the $k$ largest eigenvalues.
PCA – cont’d

• If $x$ is the datapoint (obtained after subtracting the mean), and $V$ an orthonormal basis, $V^T x$ is a column of the dot products of $x$ and the elements of $x$

• So the reconstruction for the centered $x$ is
  $$\hat{x} = V(V^T x)$$

• PCA is the procedure of obtaining the $k$ eigenvectors $V_k$
NOTE: centering

• If the image $x$ is not centred (i.e., $\overline{X}$ was not subtracted), the reconstruction is:

$$\hat{x} = \overline{X} + V(V^T(x - \overline{X}))$$
Proof that PCA produces the best reconstruction

- (Fairly easy calculus – look it up, or we can talk in office hours, or possibly we’ll do it next week)
Obtaining the Principal Components

- $XX^T$ can be huge
- There are tricks to still compute the EVs
PCA as dimensionality reduction

The set of faces is a “subspace” of the set of images

- Suppose it is K dimensional
- We can find the best subspace using PCA
- This is like fitting a “hyper-plane” to the set of faces
  - spanned by vectors \( \mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_K \)
  - any face \( \mathbf{x} \approx \bar{x} + a_1 \mathbf{v}_1 + a_2 \mathbf{v}_2 + \ldots + a_k \mathbf{v}_k \)
Eigenfaces example

Mean: $\mu$

Top eigenvectors: $u_1, \ldots u_k$
Another Eigenface set
Linear subspaces

\[ x \rightarrow ((x - \bar{x}) \cdot v_1, (x - \bar{x}) \cdot v_2) \]

What does the \( v_2 \) coordinate measure?
- distance to line
- use it for classification—near 0 for orange pts

What does the \( v_1 \) coordinate measure?
- position along line
- use it to specify which orange point it is
Dimensionality reduction

How to find $v_1$ and $v_2$?

- We can represent the orange points with *only* their $v_1$ coordinates
  - since $v_2$ coordinates are all essentially 0
- This makes it much cheaper to store and compare points
- A bigger deal for higher dimensional problems