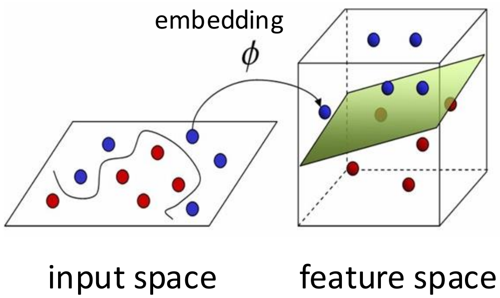
CSE 176 Introduction to Machine Learning

Lecture 8: Dimensionality Reduction

Recap: Kernel K-means



$$E(S,\mu) = \sum_{k=1}^{K} \sum_{p \in S^k} \|f_p - \mu_k\|^2$$

(Basic K-means)

$$E_k(S,\hat{\mu}) = \sum_{k=1}^{K} \sum_{p \in S^k} \|\phi(f_p) - \hat{\mu}_k\|^2$$

(Kernel K-means)

Recap: Explicit Kernel $\Leftrightarrow$ *Implicit Embedding*

$$E_k(S,\hat{\mu}) = \sum_{k=1}^{K}\sum_{p\in S^k} \|\phi(f_p) - \hat{\mu}_k\|^2$$

just plug-in

$$\hat{\mu}_k = \frac{1}{|S^k|}\sum_{q\in S^k}\phi(f_q)$$

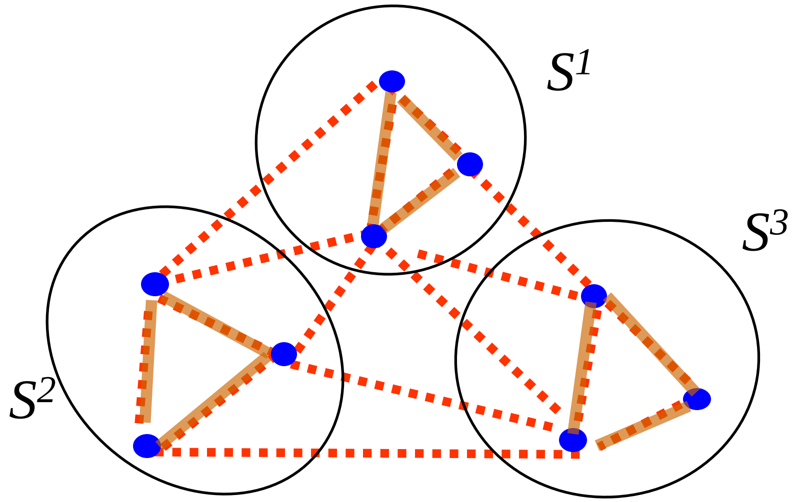
equivalent

$$E_k(S) = -\sum_{k=1}^{K}\frac{\sum_{pq\in S_k} k(f_p,f_q)}{|S^k|}$$

Recap: kernel K-means or ***average association***

"self-association" of cluster S^k

$$E(S) = -\sum_{k=1}^{K} \frac{\sum_{pq \in S_k} A_{pq}}{|S^k|}$$



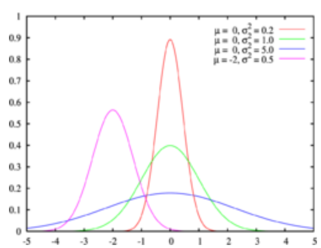
K-means

$$\boxed{\sum_{p\in\mathbf{S}}\|f_p-\mu_{\mathbf{S}}\|^2+\sum_{p\in\bar{\mathbf{S}}}\|f_p-\mu_{\bar{\mathbf{S}}}\|^2}$$

probabilistic K-means

make models more complex

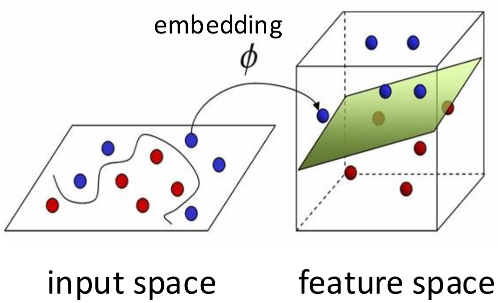
$$-\sum_{p\in\mathbf{S}}\ln\Pr(f_p|\theta_{\mathbf{S}})-\sum_{p\in\bar{\mathbf{S}}}\ln\Pr(f_p|\theta_{\bar{\mathbf{S}}})$$



kernel K-means

make data more complex

$$\sum_{p\in\mathbf{S}}\|\phi(f_p)-\hat{\mu}_{\mathbf{S}}\|^2+\sum_{p\in\bar{\mathbf{S}}}\|\phi(f_p)-\hat{\mu}_{\bar{\mathbf{S}}}\|^2$$

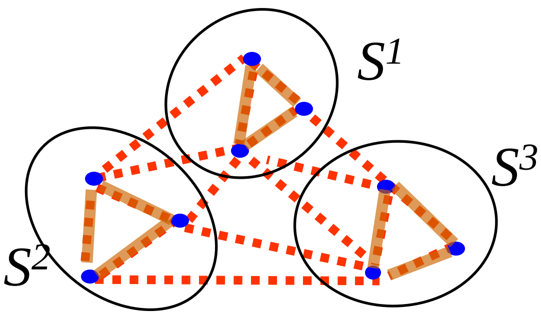


Other kernel (graph) clustering objectives

Average Association

"self-association" for S^k

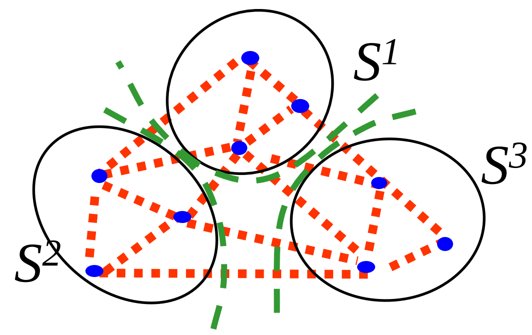
$$-\sum_{k=1}^{K}\frac{S^{k\prime}\,A\,S^{k}}{|S^{k}|}$$



Average Cut

"cut" for S^k

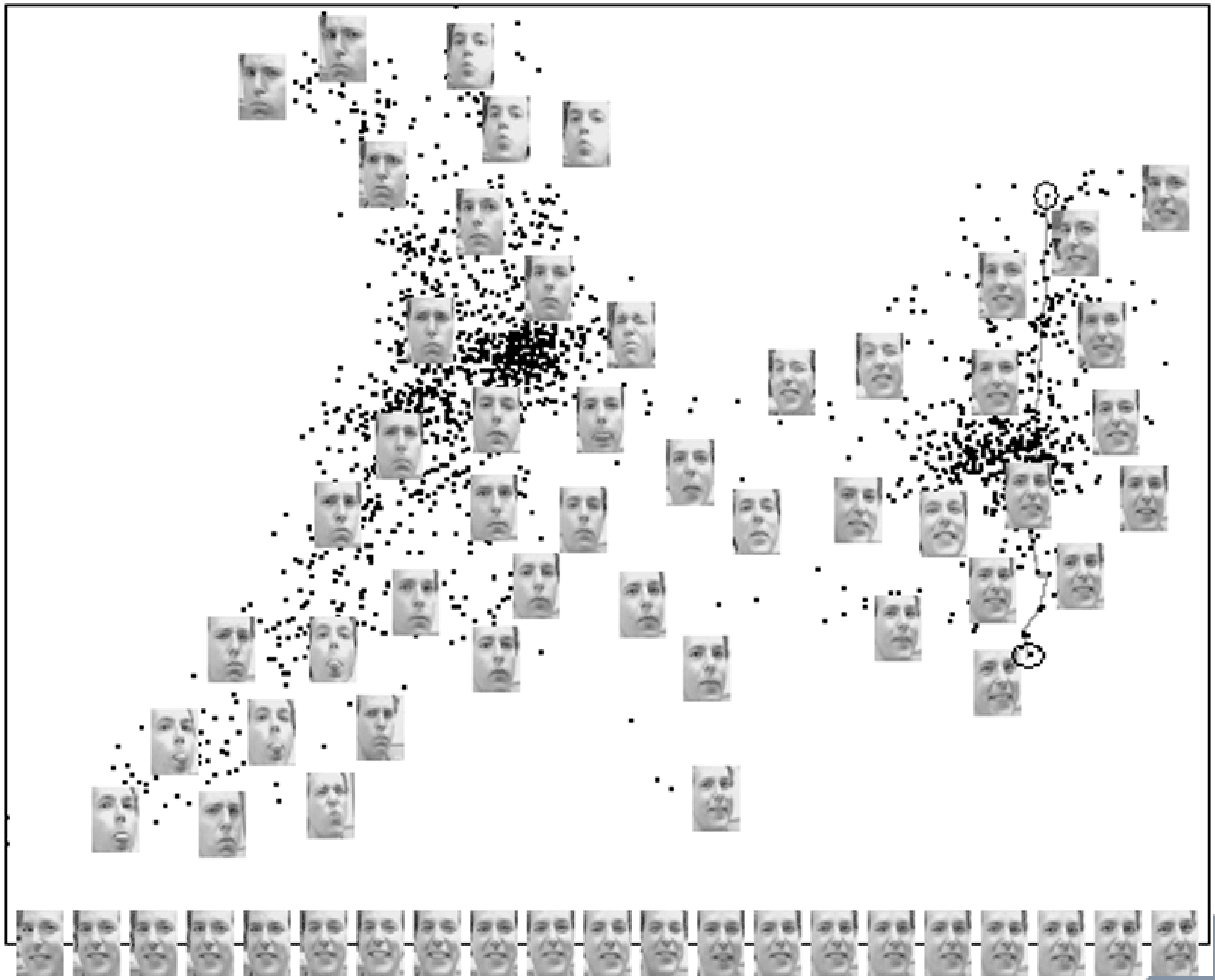
$$\sum_{k=1}^{K}\frac{S^{k\prime}\,A\,(\mathbf{1}-S^{k})}{|S^{k}|}$$



Today’s topics

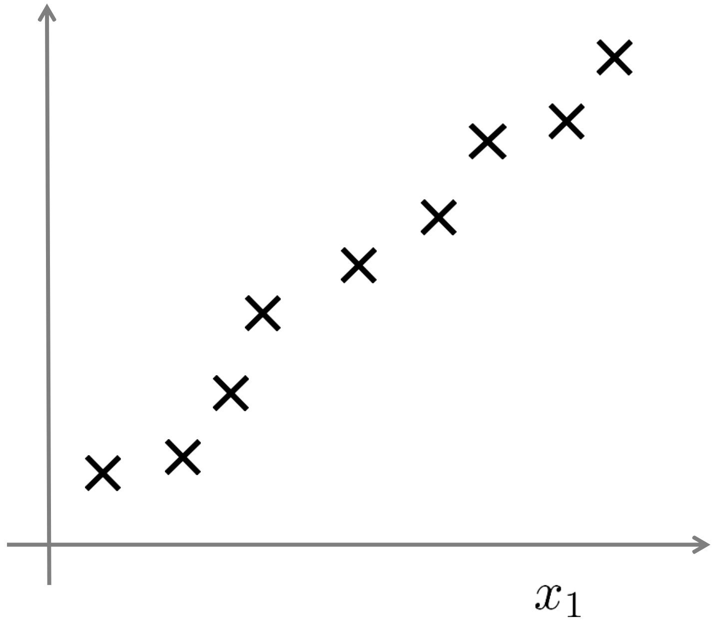
- ❑Principle Component Analysis
- ❑Multi Dimensional Scaling (MDS)

Motivation for Dimensionality Reduction



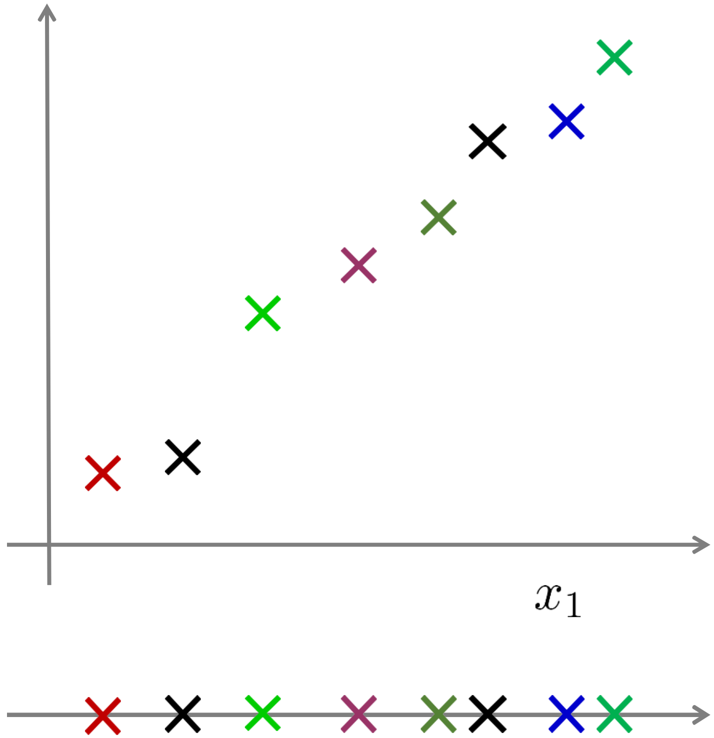
Motivation for Dimensionality Reduction

- Data Compression



Motivation for Dimensionality Reduction

❑Data Compression



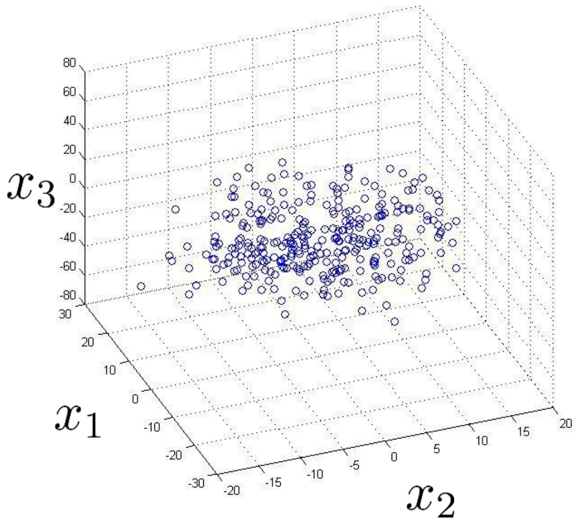
Reduce data from 2D to 1D

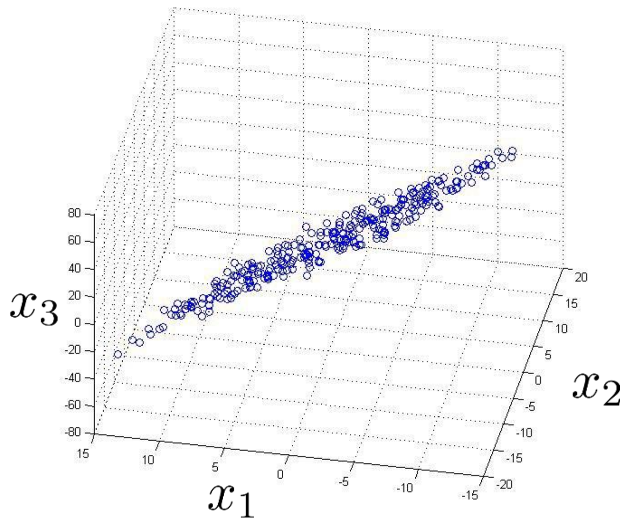
$$x^{(1)} \rightarrow z^{(1)}$$
$$x^{(2)} \rightarrow z^{(2)}$$
$$\vdots$$
$$x^{(m)} \rightarrow z^{(m)}$$

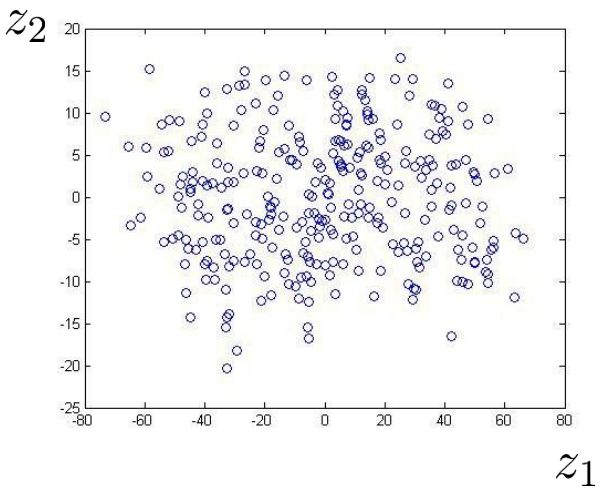
Andrew Ng

UC MERCED

Data Compression

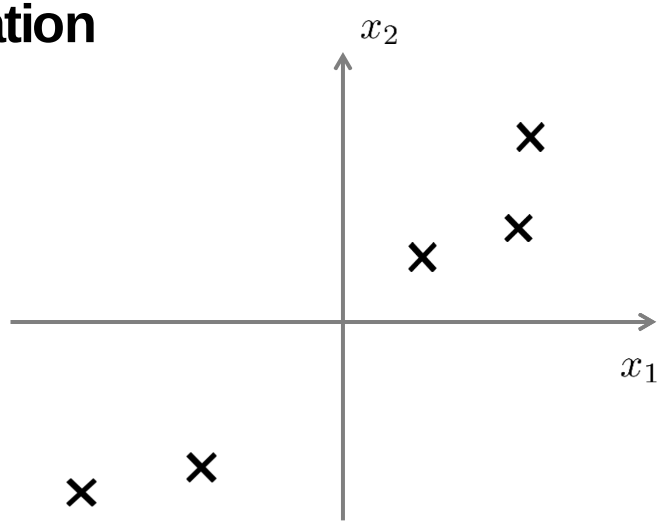




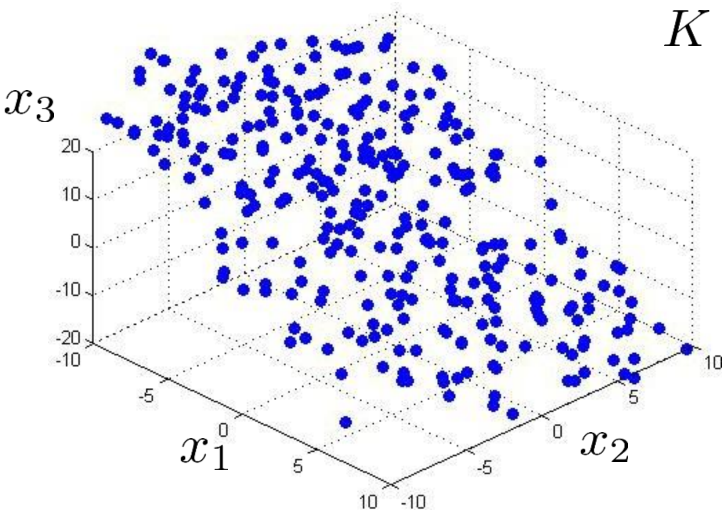


Andrew Ng

Principal Component Analysis (PCA) problem formulation



$$3D \rightarrow 2D$$
$$K = 2$$



Reduce from 2-dimension to 1-dimension: Find a direction (a vector $u^{(1)} \in \mathbb{R}^n$) onto which to project the data so as to minimize the projection error.

Reduce from n-dimension to k-dimension: Find k vectors $u^{(1)}, u^{(2)}, \ldots, u^{(k)}$ onto which to project the data, so as to minimize the projection error.

Andrew Ng

Principal Component Analysis

Goal: Find r-dim projection that best preserves variance

1. Compute mean vector μ and covariance matrix Σ of original points
2. Compute eigenvectors and eigenvalues of Σ
3. Select top r eigenvectors
4. Project points onto subspace spanned by them:

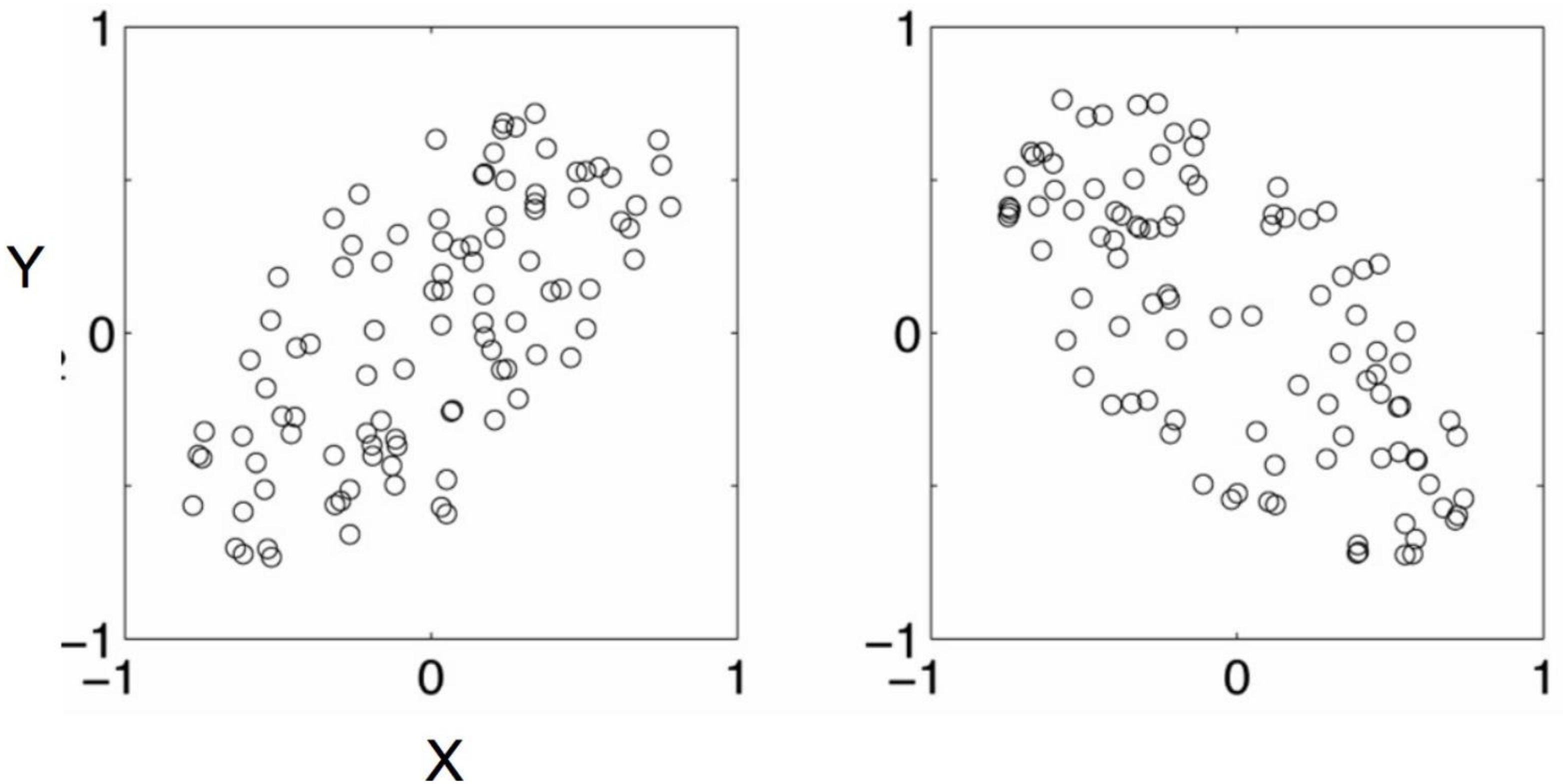
$$y = A(x - \mu)$$

   where y is the new point, x is the old one, and the rows of A are the eigenvectors

Covariance

- Variance and Covariance:
  - Measure of the “spread” of a set of points around their center of mass(mean)
- Variance:
  - Measure of the deviation from the mean for points in one dimension
- Covariance:
  - Measure of how much each of the dimensions vary from the mean with **respect to each other**

- **Covariance is measured between two dimensions**
- **Covariance sees if there is a relation between two dimensions**
- **Covariance between one dimension is the variance**



Positive: Both dimensions increase or decrease together **Negative: While one increase the other decrease**

Covariance

- Used to find relationships between dimensions in high dimensional data sets

$$q_{jk} = \frac{1}{N}\sum_{i=1}^{N}(X_{ij} - E(X_j))(X_{ik} - E(X_k))$$

The Sample mean

Eigenvector and Eigenvalue

$$Ax = \lambda x$$

A: Square Matirx

λ: Eigenvector or characteristic vector

X: Eigenvalue or characteristic value

- ***The zero vector can not be an eigenvector***
- ***The value zero can be eigenvalue***

Eigenvector and Eigenvalue

Ax = λx

A: Square Matirx

λ: Eigenvector or characteristic vector

X: Eigenvalue or characteristic value

Example

$$Show\ x = \begin{bmatrix} 2 \\ 1 \end{bmatrix} is\, an\, eigenvector\ for\ A = \begin{bmatrix} 2 & -4 \\ 3 & -6 \end{bmatrix}$$

$$Solution : Ax = \begin{bmatrix} 2 & -4 \\ 3 & -6 \end{bmatrix}\begin{bmatrix} 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$But\quad for\ \ \lambda = 0,\ \ \lambda x = 0\begin{bmatrix} 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$Thus, x\, is\, an\, eigenvector\, of\ A, and\, \lambda = 0\ is\, an\, eigenvalue.$$

Eigenvector and Eigenvalue

$$Ax = \lambda x \longrightarrow \begin{array}{l} Ax - \lambda x = 0 \\ (A - \lambda I)x = 0 \end{array}$$

If we define a new matrix B: $\longrightarrow$ $\begin{array}{l} B = A - \lambda I \\ Bx = 0 \end{array}$

If B has an inverse: $\longrightarrow$ $x = B^{-1}0 = 0$ ✘ BUT! an eigenvector cannot be zero!!

x will be an eigenvector of A if and only if B does not have an inverse, or equivalently det(B)=0 :

$$\boxed{\det(A - \lambda I) = 0}$$

Eigenvector and Eigenvalue

Example 1: Find the eigenvalues of

$$A = \begin{bmatrix} 2 & -12 \\ 1 & -5 \end{bmatrix}$$

Eigenvector and Eigenvalue

Example 1: Find the eigenvalues of

$$A = \begin{bmatrix} 2 & -12 \\ 1 & -5 \end{bmatrix}$$

$$|\lambda I - A| = \begin{vmatrix} \lambda - 2 & 12 \\ -1 & \lambda + 5 \end{vmatrix} = (\lambda - 2)(\lambda + 5) + 12$$

$$= \lambda^2 + 3\lambda + 2 = (\lambda + 1)(\lambda + 2)$$

two eigenvalues: $-1, -2$

Note: The roots of the characteristic equation can be repeated. That is, $\lambda_1 = \lambda_2 = \ldots = \lambda_k$. If that happens, the eigenvalue is said to be of multiplicity k.

Principal Component Analysis

Input: $\mathbf{x} \in \mathbb{R}^D\colon\ \mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$

Set of basis vectors: $\mathbf{u}_1, \ldots, \mathbf{u}_K$

Summarize a *D* dimensional vector *X* with *K* dimensional feature vector *h(x)*

$$h(\mathbf{x}) = \begin{bmatrix} \mathbf{u}_1 \cdot \mathbf{x} \\ \mathbf{u}_2 \cdot \mathbf{x} \\ \cdots \\ \mathbf{u}_K \cdot \mathbf{x} \end{bmatrix}$$

Principal Component Analysis

$$\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_K]$$

Basis vectors are orthonormal

$$\mathbf{u}_i^T \mathbf{u}_j = 0$$
$$||\mathbf{u}_j|| = 1$$

New data representation *h(x)*

$$z_j = \mathbf{u}_j \cdot \mathbf{x}$$
$$h(\mathbf{x}) = [z_1, \ldots, z_K]^T$$

The space of all face images

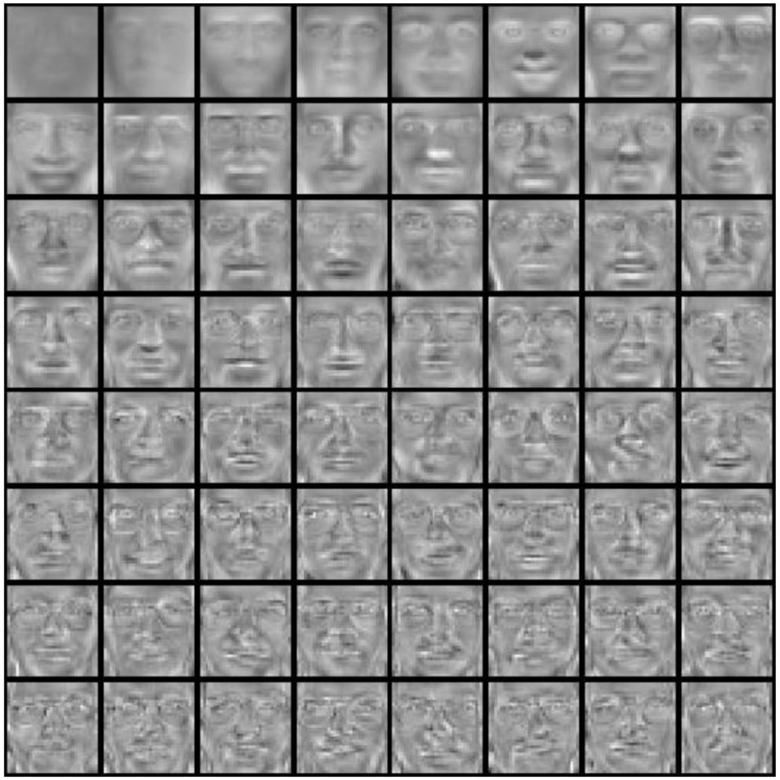
- When viewed as vectors of pixel values, face images are extremely high-dimensional
  - 100x100 image = 10,000 dimensions
  - Slow and lots of storage
- But very few 10,000-dimensional vectors are valid face images
- We want to effectively model the subspace of face images

slide by Derek Hoiem

Eigenfaces example

Mean: μ

Top eigenvectors: $u_1,\ldots u_k$



slide by Derek Hoiem

Representation and reconstruction

- Face **x** in "face space" coordinates:

$$\mathbf{x} \rightarrow [\mathbf{u}_1^{\mathrm{T}}(\mathbf{x} - \mu), \ldots, \mathbf{u}_k^{\mathrm{T}}(\mathbf{x} - \mu)]$$
$$= \quad w_1, \ldots, w_k$$

- Reconstruction:

 =  + 

$\hat{x} = \mu + w_1u_1+w_2u_2+w_3u_3+w_4u_4+ \ldots$

slide by Derek Hoiem

Reconstruction

P = 4

P = 200

P = 400

After computing eigenfaces using 400 face images from ORL face database

slide by Derek Hoiem

Application: Image compression

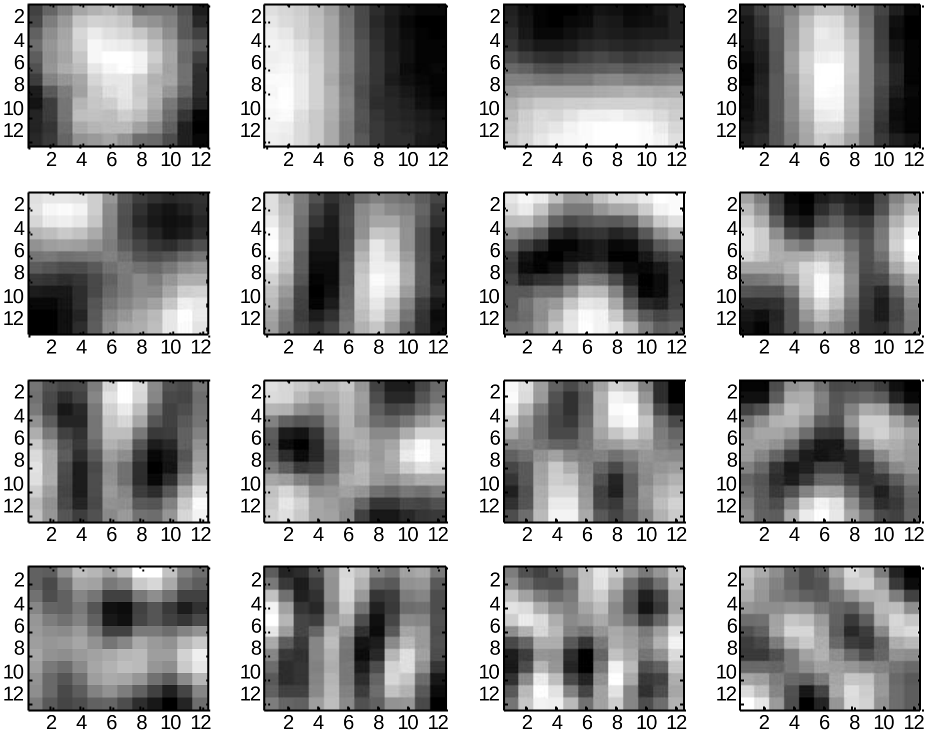
Original Image

- Divide the original 372x492 image into patches:
  - Each patch is an instance that contains 12x12 pixels on a grid
- View each as a 144-D vector

PCA compression:

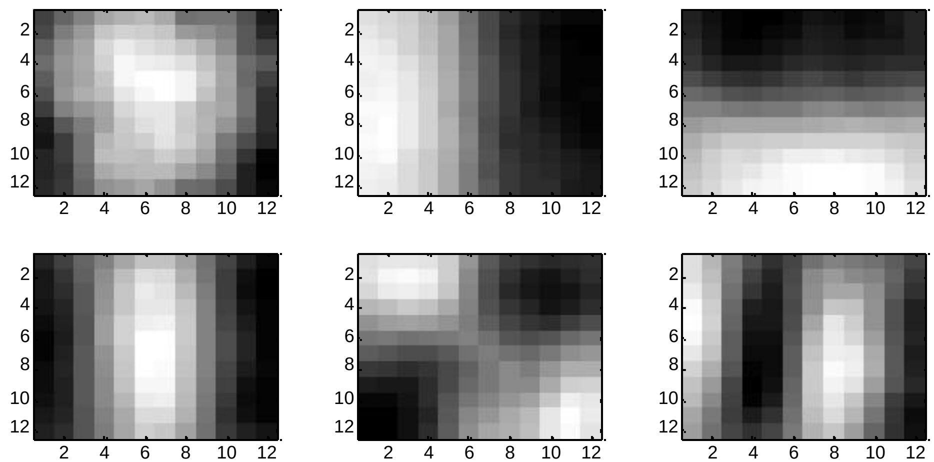
PCA compression:

16 most important eigenvectors



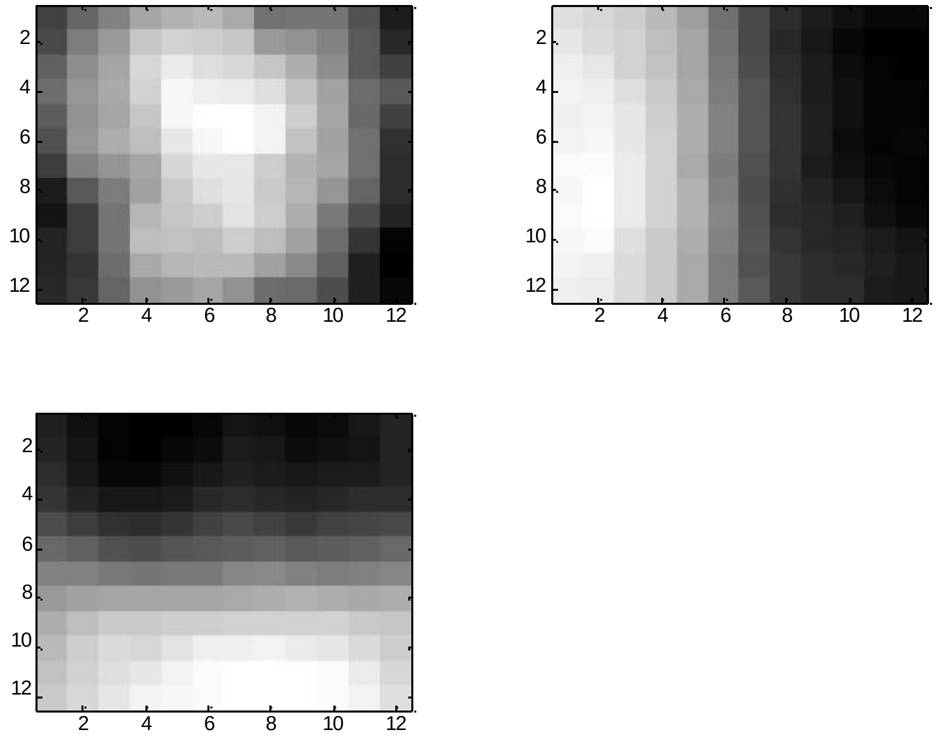
PCA compression: 144D -> 6D

6 most important eigenvectors

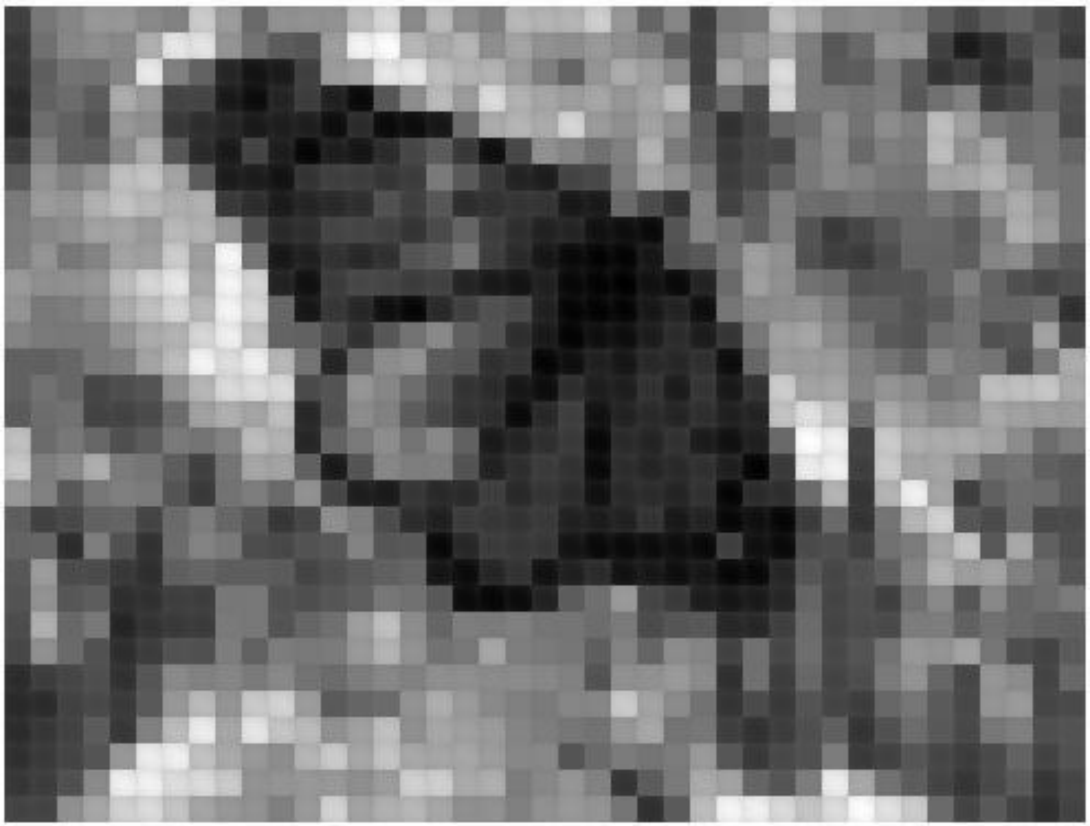


PCA compression: 144D → 3D

3 most important eigenvectors



PCA compression: 144D → 1D



Dimensionality reduction

- PCA (Principal Component Analysis):
  - Find projection that maximize the variance
- LDA(Linear Discriminant Analysis):
  - Maximizing the component axes for class-separation
- Multidimensional Scaling:
  - Find projection that best preserves inter-point distances
- …