Dimensionality Reduction

Sunita Sarawagi

CS 725 Fall 2023

Dimensional Reduction

t-sne

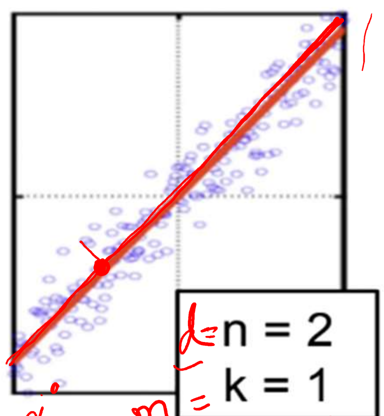
- A form of unsupervised learning where we learn a mapping from a high dimensional space $x \in R^{d}$ to a lower dimensional space $z \in R^{k}$, m where k << d

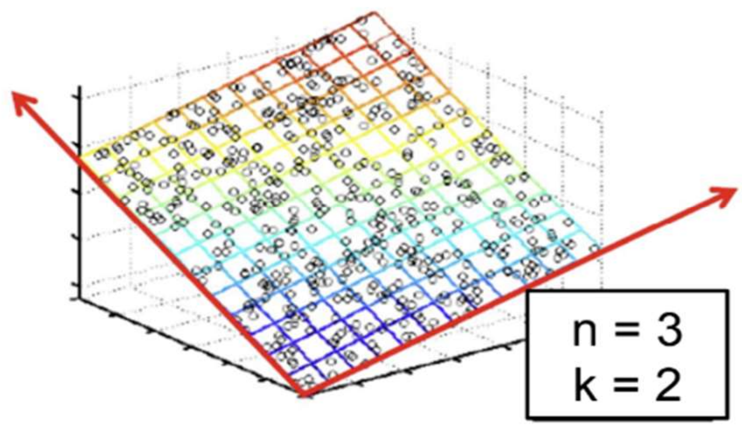
  d is large

- Often original data in raw form might have a very high dimensionality (e.g. an image), but the information in each instance can be expressed compactly in smaller dimensions.
- Applications
  - Visualization continues to be relevant. Eg: visualizing high-dimensional embedding t-sne
  - Efficient learning
  - Data understanding.

Dimension reduction

- Assumption: data (approximately) lies on a lower dimensional space
- Examples:





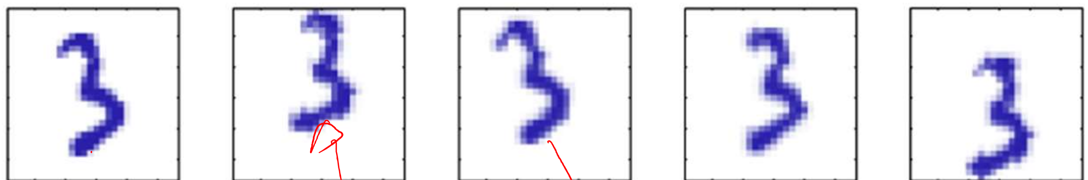
Slide from Yi Zhang

Example (from Bishop)

- Suppose we have a dataset of digits ("3") perturbed in various ways:

28

p



P = 28

d = 28 x 28

- What operations did I perform? What is the data's intrinsic dimensionality?
- Here the underlying manifold is *nonlinear*

Random displacement and rotation

Linear projections

Given: N data points $x^1, x^2 - - - - x^N$

- Each original high-dimensional $x \in R^d$ is projected to a lower dimensional space $z \in R^m$ using just a linear projection matrix.

$z = Wx$ (W: $m \times d$)

$$W = \begin{bmatrix} w_1^T \\ w_2^T \\ \vdots \\ w_m^T \end{bmatrix} \qquad w_j = \begin{bmatrix} w_{j1} \\ \vdots \\ w_{jd} \end{bmatrix} \qquad x = \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \qquad z = \begin{bmatrix} z_1 \\ \vdots \\ z_m \end{bmatrix}$$

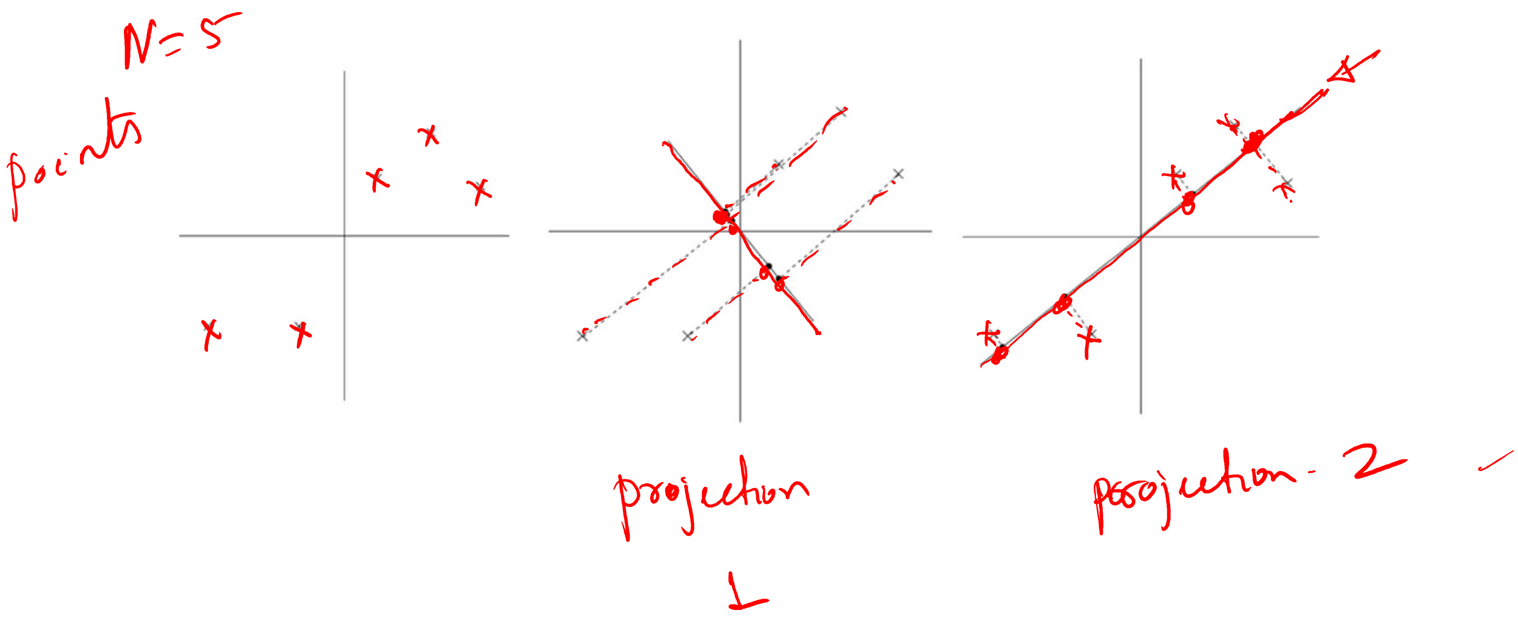
$z_j = w_j^T x$

$= \langle w_j, x \rangle$

$$W = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1d} \\ \vdots & & & \\ w_{j1} & w_{j2} & \cdots & w_{jd} \\ \vdots & & & \\ w_{m1} & w_{m2} & \cdots & w_{md} \end{bmatrix} \begin{array}{l} \to w_1^T \\ \\ \to w_j^T \\ \\ \to w_m^T \end{array}$$

No labelled supervision on the desired z. How do we choose the best z?

Which projection is better?



From notes by Andrew Ng

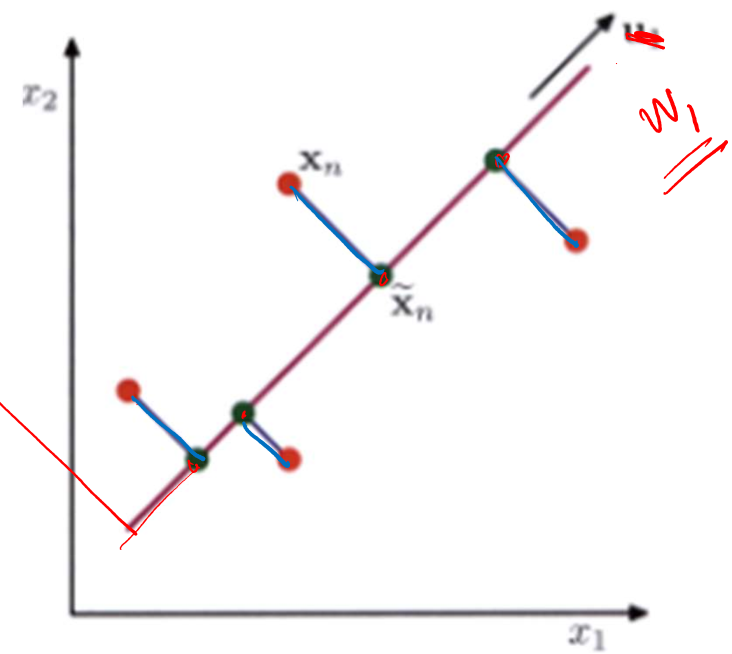
Principal Component Analysis (PCA)

$m = 1$

- Reading material: Chapter 12 of Bishop book
- Simple and widely used linear projection
- Projection basis are orthogonal
  - $w_j \perp w_r \quad \text{j, r} \leq m$
- Objectives
  - Maximize the variance (spread) of the projected points --- green points
  - Equivalently, minimizing the sum of squares of the projection error --- blue lines.

w_2

$w_1 \perp w_2$



Consider single dimensional projection m=1, to ~~minimize~~ maximize variance of projected points

Assume $\|W_1\| = 1$

Given $x^1, x^2 \ldots x^N$ $\quad x^i \in R^d$

find W_1 s.t $\quad z^i = W_1^T x^i$

variance of $\{z^1, z^2 \ldots z^N\}$ is ~~minimized~~ maximized

where $z^i = W_1^T x$

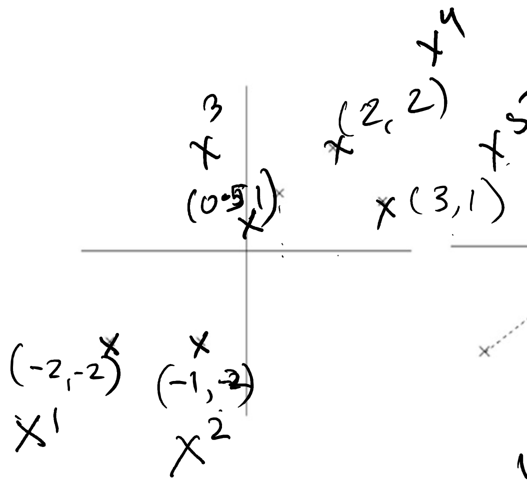
$$W_1 = \underset{W}{\arg\max} \sum_{i=1}^{N} (z^i - \bar{z})^2 \quad \text{where}$$

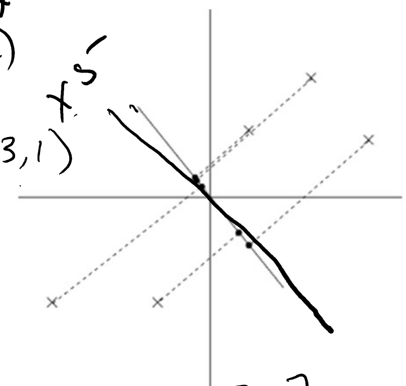
$$z^i = W_1^T x^i$$

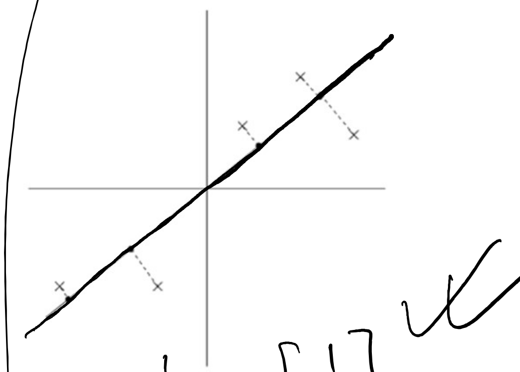
$$\bar{z} = \frac{1}{N} \sum_{i=1}^{N} z^i \quad (\text{Average } z)$$

Two

$N = 5$







$\underline{w^1} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$

$z^1 \quad 0 = w_1^T x^1$

$z^2 \quad 1$

$z^3 \quad -0.5$

$z^4 \quad 0$

$z^5 \quad 2$

Variance $\{0, 1, -0.5, 0, 2\}$

$= \quad \bar{z} = 0.5$

$w_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$z^1 = -4$

$z^2 = -3$

$z^3 = 1.5$

$z^4 = 4$

$z^5 = 4$

$\bar{z} = 0.5$

Solving for the optimal projection

$$\underline{w}^1 = \operatorname{argmax} \sum_{i=1}^{N} (\underline{w_1^T x^i} - w_1^T \bar{x})^2$$

It can be shown that

$$\sum_{i=1}^{N} (w_1^T x^i - w_1^T \bar{x})^2 \equiv w_1^T S w_1 \quad \text{where} \quad S = \sum_{i=1}^{N} (x_i - \bar{x})(x_i - \bar{x})^T$$

S: covariance ~~correlation~~ between the dimensions of x

$$\max_{w^1} \boxed{w_1^T S w_1} \equiv \text{Eigen value of } w_1$$

$w_1 \equiv$ Eigen vector

where $\|w_1\| = 1$

or $\boxed{w_1^T w_1 = 1}$

$$S \quad d \begin{bmatrix} S_{jr} & \\ & \end{bmatrix}_{d}$$

S_{jr} → covariance between dimension j & r of N points

$$\max_{w_1} \quad \underline{w_1^T S w_1} - \lambda \underline{(w_1^T w_1 - 1)}$$ Lagrangian multiplier.

Equating gradient w.r.t w_1 to 0 $\Rightarrow$ λ is a scalar $\in \mathbb{R}$. S is a square matrix

$$\boxed{S w_1 = \lambda w_1}$$

$\Rightarrow$ w_1 = is an Eigen vector of S

Left multiply by w_1

$$w_1^T S w_1 = w_1^T \lambda w_1 = \lambda \boxed{w_1^T w_1} = \lambda. \quad [\because w_1^T w = 1].$$

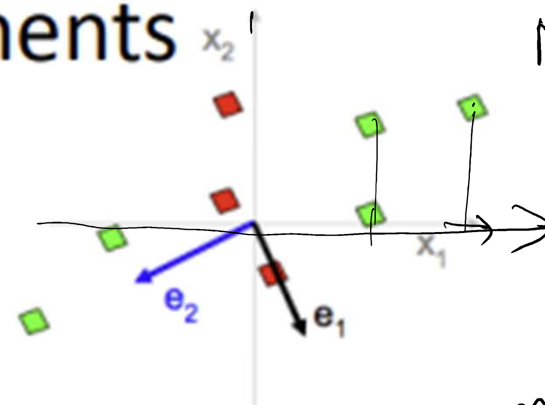
objective:

w_1 = Eigen-vector corresponding to which Eigen value is maximum.

In general..

- The best m dimensional linear projection of a d-dimensional dataset for maximizing variance of the projected points are
  - The first m eigen vectors of the covariance matrix S of the given data points.

Principal components



$N = 8$

- Compute covariance matrix $\Sigma = S$
  - covariance of dimensions x_1 and x_2:
    - do x_1 and x_2 tend to increase together?
    - or does x_2 decrease as x_1 increases?
  - covariance: measure of variability

$$\begin{array}{c|cc} & x_1 & x_2 \\ \hline x_1 & \boxed{2.0} & 0.8 \\ x_2 & 0.8 & 0.6 \end{array}$$

covariance between x_1 & x_2

$$\mathrm{var}(x_2) = \frac{1}{N}\sum_{i=1}^{n}(x_{2,i} - \mu_2)^2$$

$$\mathrm{cov}(x_1, x_2) = \frac{1}{N}\sum_{i=1}^{n}(x_{1,i} - \mu_1)(x_{2,i} - \mu_2)$$

- Find the basis of $\Sigma = S$
  - find vectors e_i which aren't turned by Σ
    - $\Sigma e_i = \lambda_i e_i$: eigenvalue / eigenvector
  - 1^{st} PC: "longest" e_i (has largest λ_i), 2^{nd} PC: next longest, ...

$$\begin{array}{cc} \lambda_1 & 0.26 \\ \lambda_2 & \boxed{2.42} \end{array} \qquad \begin{array}{c|cc} & e_1 & e_2 \\ \hline x & 0.4 & -0.9 \\ y & -0.9 & -0.4 \end{array}$$

w_1

w_2

$\|w_1\| = 1^2$

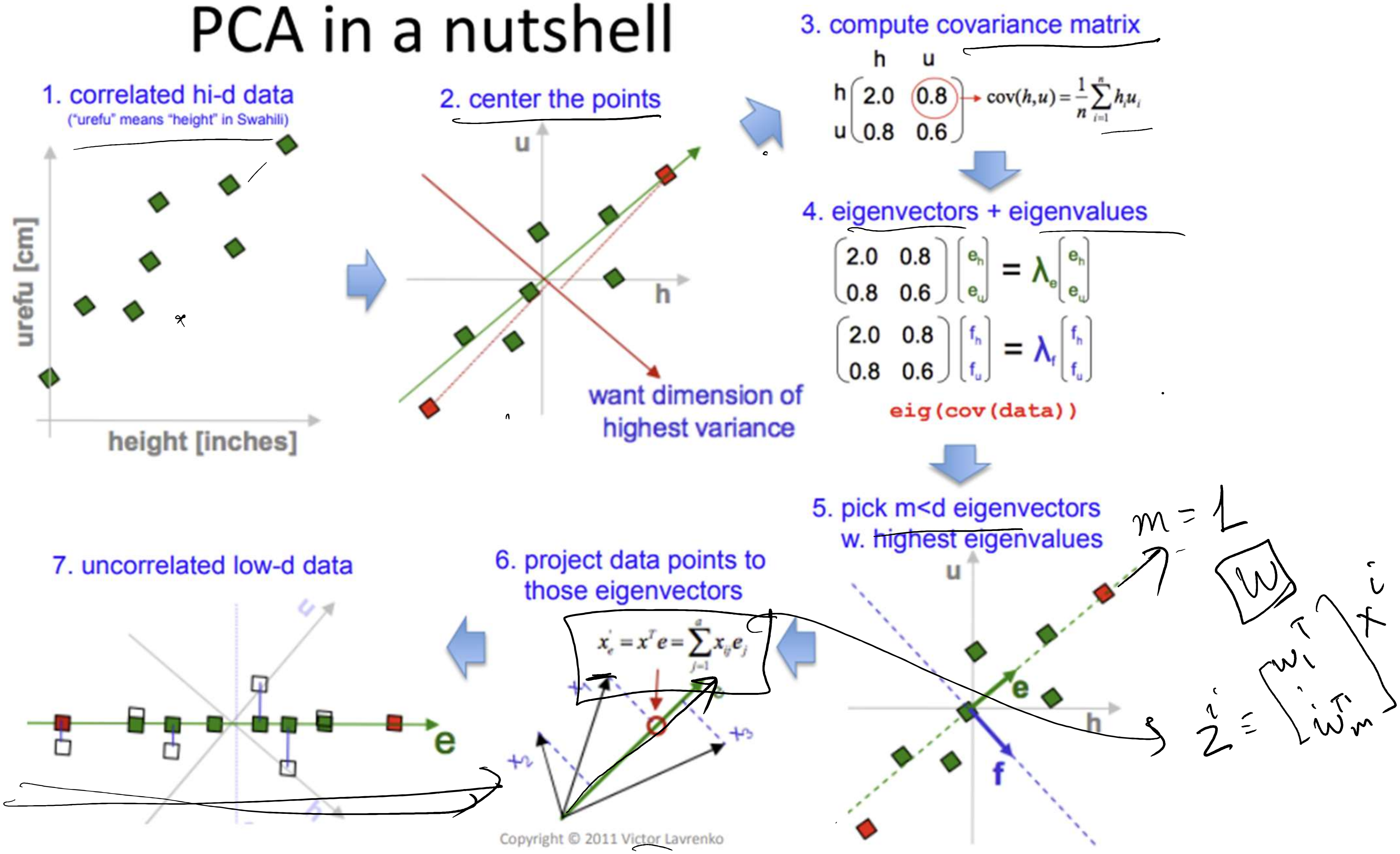
$= (0.9)^2 + (0.4)^2$

$= 0.81 + 0.16$

≈ 1

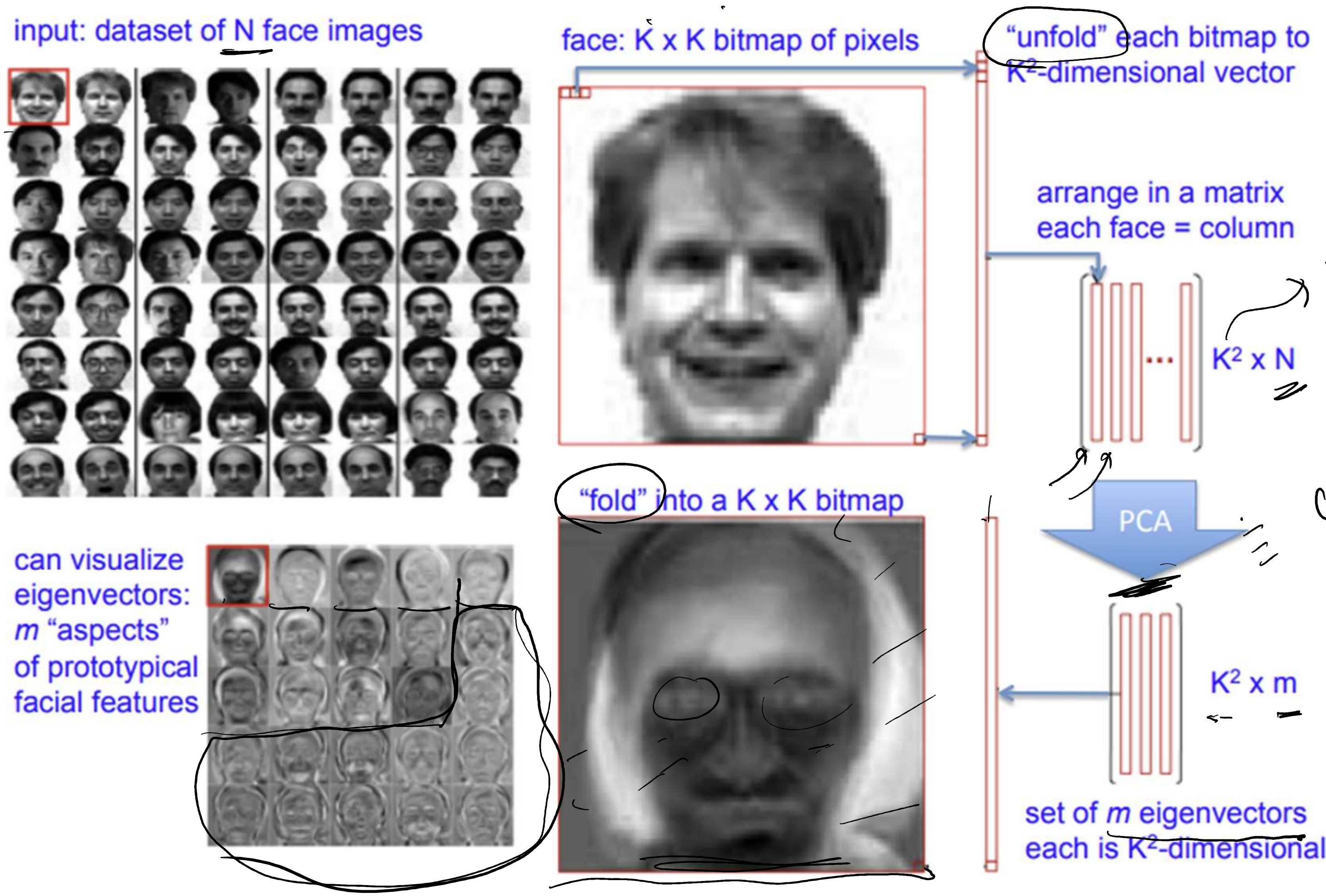
pca.pdf (ed.ac.uk)

PCA in a nutshell



[Eigen faces demo]

PCA example: Eigen Faces

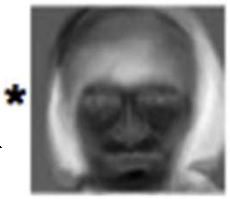


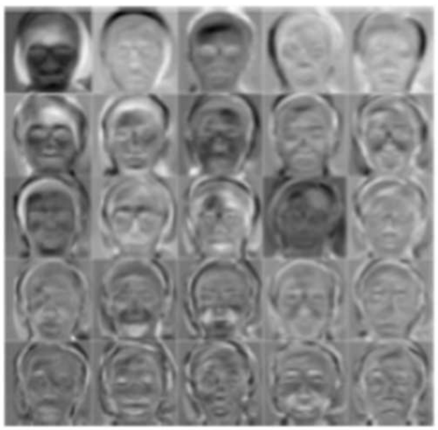
= d

Create the Σ or S matrix.

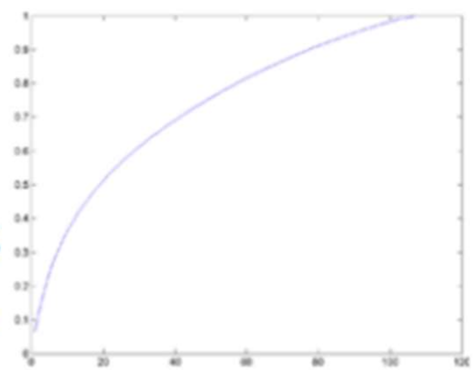
(can find PCA without explicitly creating Σ)

Eigen Faces: Projection

 = 0.9 *  - 0.2 *  + 0.4 *  + …



- Project new face to space of eigen-faces
- Represent vector as a linear combination of principal components
- How many do we need?



m=1 m=2 m=6

(Eigen) Face Recognition

- Face similarity
  - in the reduced space
  - insensitive to lighting expression, orientation
- Projecting new "faces"
  - everything is a face

new face (not in training)

projected to eigenfaces

Non-linear dimensionality reduction

- Also called manifold learning
- Manifold is a topological space that is locally Euclidean
  - Example: surface of the earth is a curved 2d surface embedded in a high-dimensional space by at each point on the surface, the earth seems flat
- Manifold hypothesis:
  - Most “naturally occurring” high-dimensional dataset lie on a low-dimensional manifold, also called the intrinsic dimensonality of the data

- Many methods exist for learning manifolds: main idea is to preserve local neighborhood of each point in the given dataset.

Examples

Stochastic Neighborhood Embedding (SNE)

- Convert high-dimensional Euclidean distances into conditional probabilities that represent similarities