Linear Models: PCA, FA

Shikui Tu

Shanghai Jiao Tong University

2021-04-13

Outline

- Some mathematical background
- Principal Component Analysis (PCA)
- From matrix factorization perspectives
- Hebbian learning, LMSER and PCA
- Probabilistic PCA, Factor Analysis (FA)

Outline

- **Some mathematical background**
  - **Linear algebra**

- Principal Component Analysis (PCA)

Inner product (1/3)

Inner product

The standard inner product on $\mathbf{R}^n$, the set of real n-vectors, is given by

$$\langle x, y \rangle = x^T y = \sum_{i=1}^{n} x_i y_i, \quad \text{for } x, y \in \mathbf{R}^n.$$

- The Euclidean norm, or ℓ_2-norm, of a vector $x \in \mathbf{R}^n$ is defined as

$$\|x\|_2 = (x^T x)^{1/2} = (x_1^2 + \cdots + x_n^2)^{1/2}.$$

- The (unsigned) angle $\alpha \in [0, \pi]$ between nonzero vectors $x, y \in \mathbf{R}^n$ is given by

$$\cos(\alpha) = \frac{x^T y}{\|x\|_2 \|y\|_2}$$

Inner product (2/3)

- The standard inner product on $\mathbf{R}^{m\times n}$, the set of $m \times n$ real matrices, is given by

$$\langle X, Y\rangle = \mathbf{tr}(X^T Y) = \sum_{i=1}^{m}\sum_{j=1}^{n} X_{ij}Y_{ij}, \quad \text{for } X, Y \in \mathbf{R}^{m\times n}$$

  Here **tr** denotes trace of a matrix, i.e., the sum of its diagonal elements.
- The distance between two vectors x and y: $\mathbf{dist}(x, y) = \|x-y\|$.
- The unit ball of the norm $\|\cdot\|$: the set of all vectors with norm less than or equal to one,

$$\mathcal{B} = \{x \in \mathbf{R}^n \mid \|x\| \leq 1\},$$

Inner product (3/3)

- The ℓ_1-norm: $\|x\|_1 = |x_1| + \cdots + |x_n|$.
- The Chebyshev or ℓ_∞-norm: $\|x\|_\infty = \max\{|x_1|, \ldots, |x_n|\}$.
- The ℓ_p-norm, with $p \geq 1$, is defined by

$$\|x\|_p = (|x_1|^p + \cdots + |x_n|^p)^{1/p}.$$

- The P-quadratic norms: For $P \in \mathbf{S}^n_{++}$:

$$\|x\|_P = (x^T P x)^{1/2} = \|P^{1/2} x\|_2.$$

Taylor series

The Taylor series of a real or complex-valued function $f(x)$ that is infinitely differentiable at a real or complex number a is the power series

$$\sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \cdots$$

A Taylor series expansion of a scalar-valued function of more than one variable can be written compactly as

$$f(a) + \nabla f(a)^T (x-a) + \frac{1}{2!}(x-a)^T \nabla^2 f(a)(x-a) + \cdots$$

Use in optimization:

$$f(x+\Delta x) = f(x) + \nabla f(x)^T \Delta x + \frac{1}{2!}\Delta x^T H(x) \Delta x + \cdots$$

where $H(x) = [\frac{\partial^2 f}{\partial x_i \partial x_j}]$ is the Hessian matrix.

Symmetric matrix decomposition (1/2)

Suppose $A \in \mathbf{S}^n$, i.e., A is a real symmetric $n \times n$ matrix. The spectral decomposition or (symmetric) eigenvalue decomposition of A is

$$A = Q\Lambda Q^T,$$

where $Q \in \mathbf{R}^{n\times n}$ is orthogonal, i.e., satisfies $Q^TQ = I$, and $\Lambda = \mathbf{diag}(\lambda_1, \ldots, \lambda_n)$. The (real) numbers λ_i are the eigenvalues of A, and are the roots of the characteristic polynomial $\det(sI-A)$. The columns of Q form an orthonormal set of eigenvectors of A.

Symmetric matrix decomposition (2/2)

Assume $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$. We use the notation $\lambda_i(A)$ to refer to the i-th largest eigenvalue of $A \in \mathbf{S}$. We usually write the largest or maximum eigenvalue as $\lambda_1(A) = \lambda_{\max}(A)$, and the least or minimum eigenvalue as $\lambda_n(A) = \lambda_{\min}(A)$. The determinant and trace can be expressed in terms of the eigenvalues,

$$\det(A) = \prod_{i=1}^{n} \lambda_i, \quad Tr(A) = \sum_{i=1}^{n} \lambda_i,$$

as can the spectral and Frobenius norms,

$$\|A\|_2 = \max_{i=1,\ldots,n} |\lambda_i| = \max\{\lambda_1, -\lambda_n\}, \quad \|A\|_F = \left(\sum_{i=1}^{n} \lambda_i^2\right)^{1/2}$$

In particular, for any x, we have $\lambda_{\min}(A)x^Tx \leq x^TAx \leq \lambda_{\max}(A)x^Tx$.

Definite matrix

Positive definite

A matrix $A \in \mathbf{S}^n$ is called positive definite if for all $x \neq 0, x^T Ax > 0$, denoted as $A \succ 0$. We use $\mathbf{S}^n_{++}$ to denote the set of positive definite matrices in $\mathbf{S}^n$.

- By the inequality above, we see that $A > 0$ if and only all its eigenvalues are positive, i.e., $\lambda_{\min}(A) > 0$.
- If $-A$ is positive definite, we say A is negative definite, which we write as $A \prec 0$.
- If A satisfies $x^T Ax \geq 0$ for all x, we say that A is positive semidefinite or nonnegative definite.
- For $A, B \in \mathbf{S}^n$, we use $A \prec B$ to mean $B-A \succ 0$.

Singular Value Decomposition (SVD)

Suppose $A \in \mathbf{R}^{m\times n}$ with $\mathbf{rank}\, A = r$. Then A can be factored as

$$A = U\Sigma V^T, \tag{A.12}$$

where $U \in \mathbf{R}^{m\times r}$ satisfies $U^TU = I$, $V \in \mathbf{R}^{n\times r}$ satisfies $V^TV = I$, and $\Sigma = \mathbf{diag}(\sigma_1, \ldots, \sigma_r)$, with

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0.$$

The factorization (A.12) is called the *singular value decomposition* (SVD) of A. The columns of U are called *left singular vectors* of A, the columns of V are *right singular vectors*, and the numbers σ_i are the *singular values*. The singular value decomposition can be written

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T,$$

where $u_i \in \mathbf{R}^m$ are the left singular vectors, and $v_i \in \mathbf{R}^n$ are the right singular vectors.

SVD and eigendecomposition

The singular value decomposition of a matrix A is closely related to the eigenvalue decomposition of the (symmetric, nonnegative definite) matrix $A^T A$. Using (A.12) we can write

$$A^T A = V \Sigma^2 V^T = \begin{bmatrix} V & \tilde{V} \end{bmatrix} \begin{bmatrix} \Sigma^2 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V & \tilde{V} \end{bmatrix}^T,$$

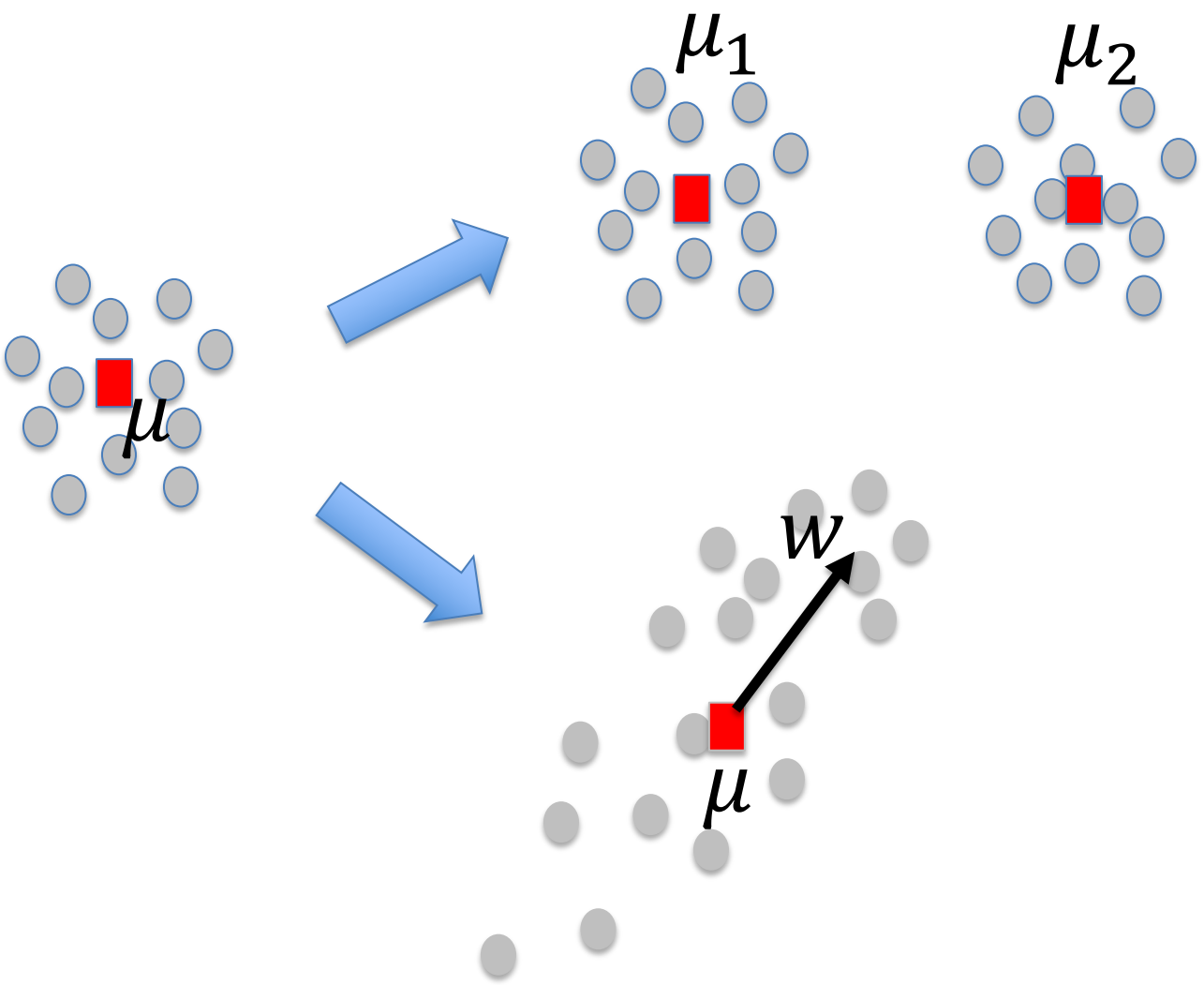
where $\tilde{V}$ is any matrix for which $[V\ \tilde{V}]$ is orthogonal. The righthand expression is the eigenvalue decomposition of $A^T A$, so we conclude that its nonzero eigenvalues are the singular values of A squared, and the associated eigenvectors of $A^T A$ are the right singular vectors of A. A similar analysis of AA^T shows that its nonzero eigenvalues are also the squares of the singular values of A, and the associated eigenvectors are the left singular vectors of A.

Outline

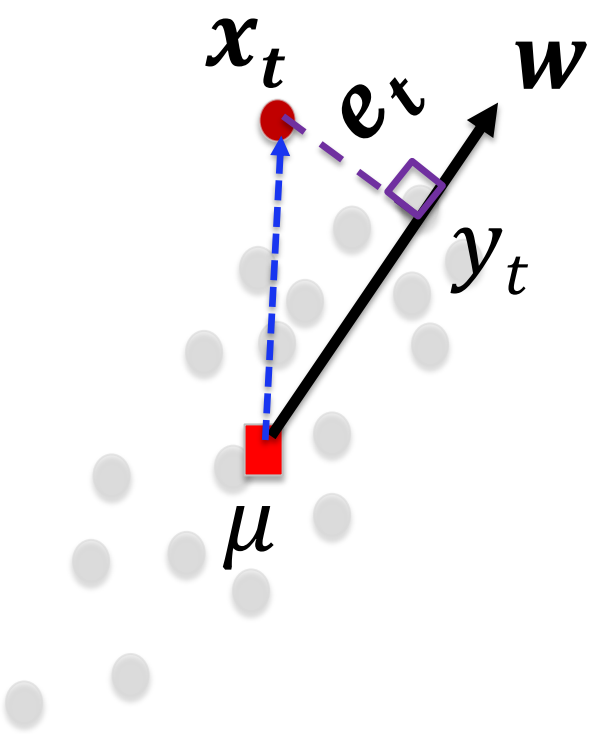
- Some mathematical background
  - Linear algebra

- **Principal Component Analysis (PCA)**

Model from “one point” to “one line”



Define the error



$$||w|| = 1$$

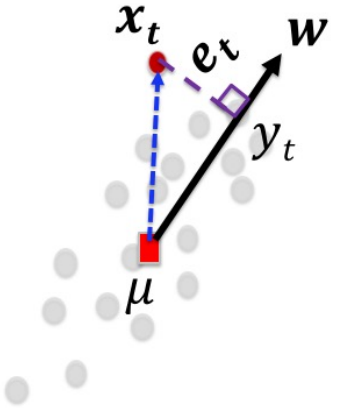
$$y_t = x_t^T w$$

$$e_t = ||x_t - y_t w||^2$$

$$J(w) = \frac{1}{N}\sum_{t=1}^{N} ||x_t - (x_t^T w)w||^2$$

Mean Square Error (MSE)

$$J(w) = \frac{1}{N}\sum_{t=1}^{N} ||\boldsymbol{x_t} - (\boldsymbol{x_t^T w})\boldsymbol{w}||^2$$



$$\boldsymbol{x_t^T x_t} - (\boldsymbol{x_t^T w})\boldsymbol{w^T x_t} - \boldsymbol{x_t^T}(\boldsymbol{x_t^T w})\boldsymbol{w} + (\boldsymbol{x_t^T w})\boldsymbol{w^T}(\boldsymbol{x_t^T w})\boldsymbol{w}$$
$$= \boldsymbol{x_t^T x_t} - \boldsymbol{w^T}(\boldsymbol{x_t x_t^T})\boldsymbol{w}$$

$$||w|| = 1$$

Introduce a Lagrange multiplier λ

$$L(\{\boldsymbol{x_t}\}, \boldsymbol{w}) = J(\{\boldsymbol{x_t}\}, \boldsymbol{w}) - \lambda \cdot (\boldsymbol{w^T w} - 1)$$

$$\frac{\partial J(\boldsymbol{w})}{\partial \boldsymbol{w}} - \lambda \cdot \frac{\partial(\boldsymbol{w^T w} - 1)}{\partial \boldsymbol{w}} = -2(\Sigma_x \boldsymbol{w}) - \lambda \cdot 2\boldsymbol{w} = \boldsymbol{0}$$

$$\Sigma_x = \frac{1}{N}\sum_{t=1}^{N} \boldsymbol{x_t x_t^T}$$

$$\boxed{\Sigma_x \boldsymbol{w} = (-\lambda) \cdot \boldsymbol{w}}$$

Eigenvalues and Eigenvectors

Lagrange multiplier

From wiki

maximize $f(x, y)$
subject to $g(x, y) = 0$

Lagrange multiplier λ

$$\mathcal{L}(x, y, \lambda) = f(x, y) - \lambda \cdot g(x, y)$$

$$\nabla_{x,y} f = \lambda \nabla_{x,y} g$$

$$\nabla_{x,y} f = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right)$$

$$\nabla_{x,y} g = \left(\frac{\partial g}{\partial x}, \frac{\partial g}{\partial y} \right)$$

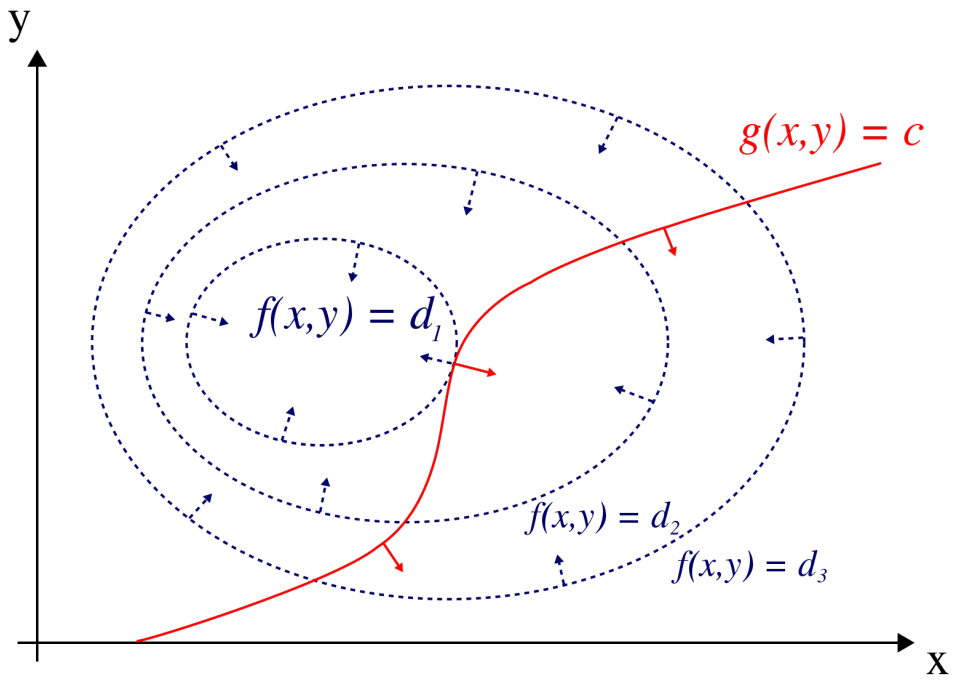
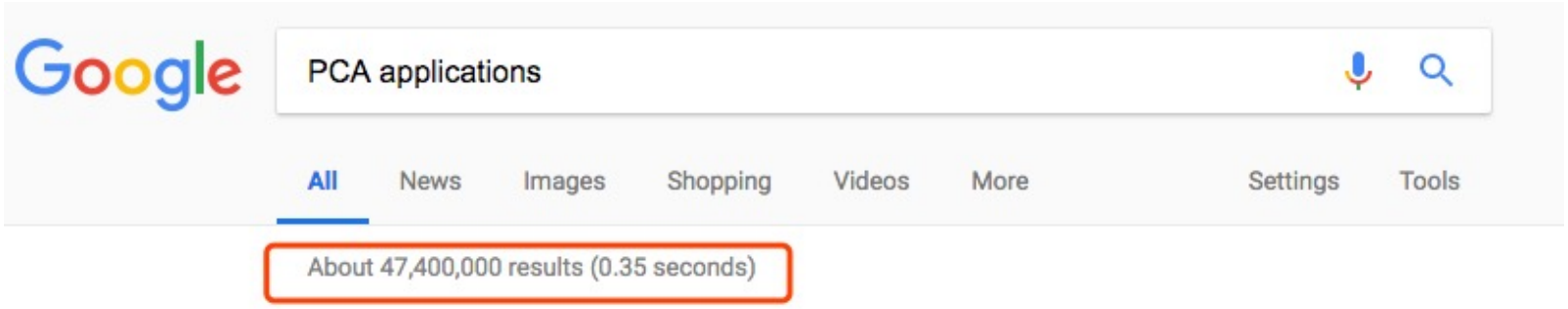


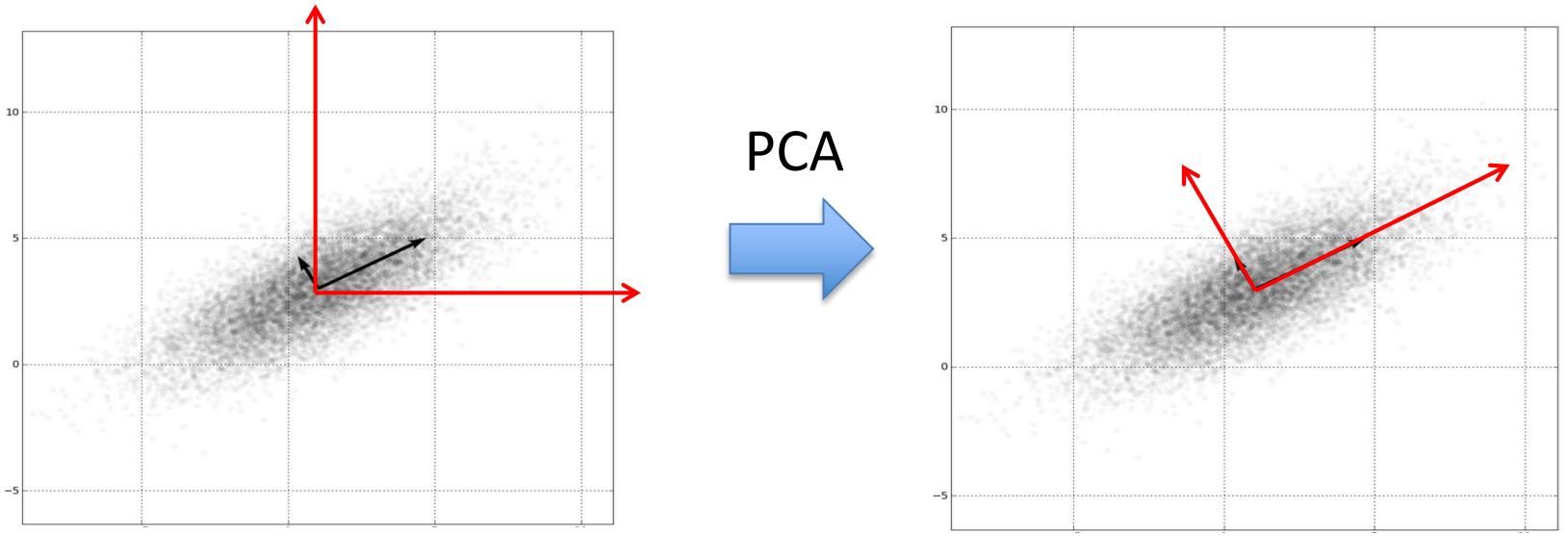
Figure 1: The red line shows the constraint $g(x, y) = c$. The blue lines are contours of $f(x, y)$. The point where the red line tangentially touches a blue contour is the maximum of $f(x, y)$, since $d_1 > d_2$.

Applications of PCA

- Popular in multivariate statistics, signal processing
- Reduce data noise, redundancy, correlation, …
- Dimensionality reduction
- Feature selection and feature extraction
- Data visualization (to 2D or 3D)

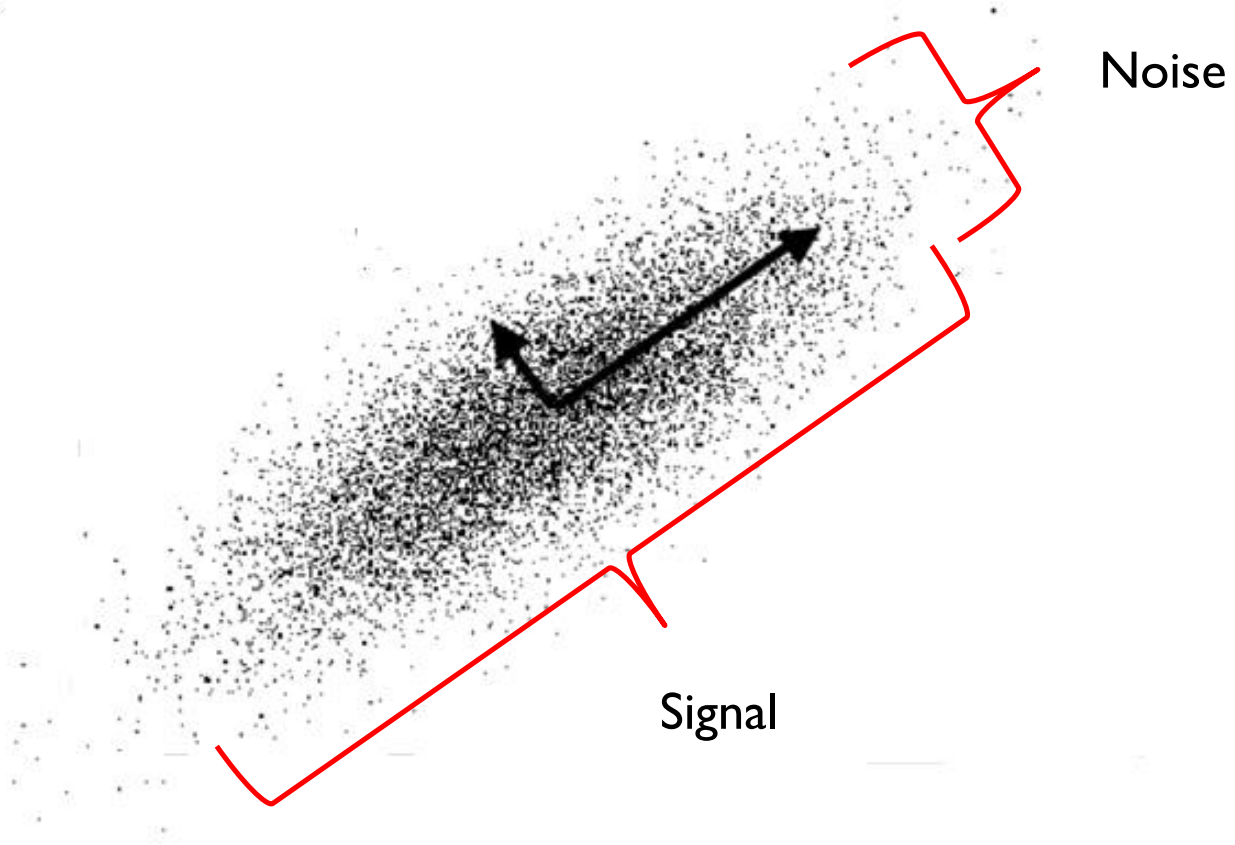


Data correlation



Correlation can be removed by rotating the coordinates to principal components.

Signal-noise ratio maximization



Keep one signal dimension, discard one noisy dimension.

Information redundancy

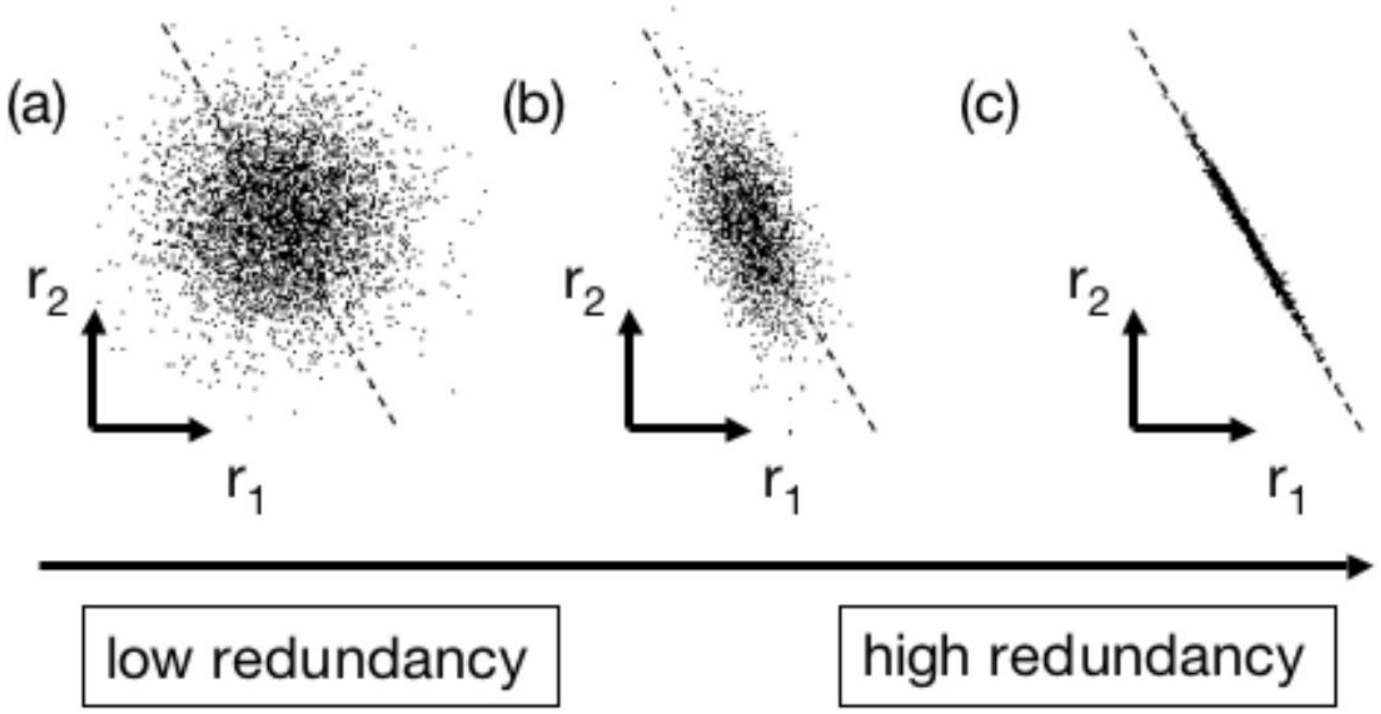


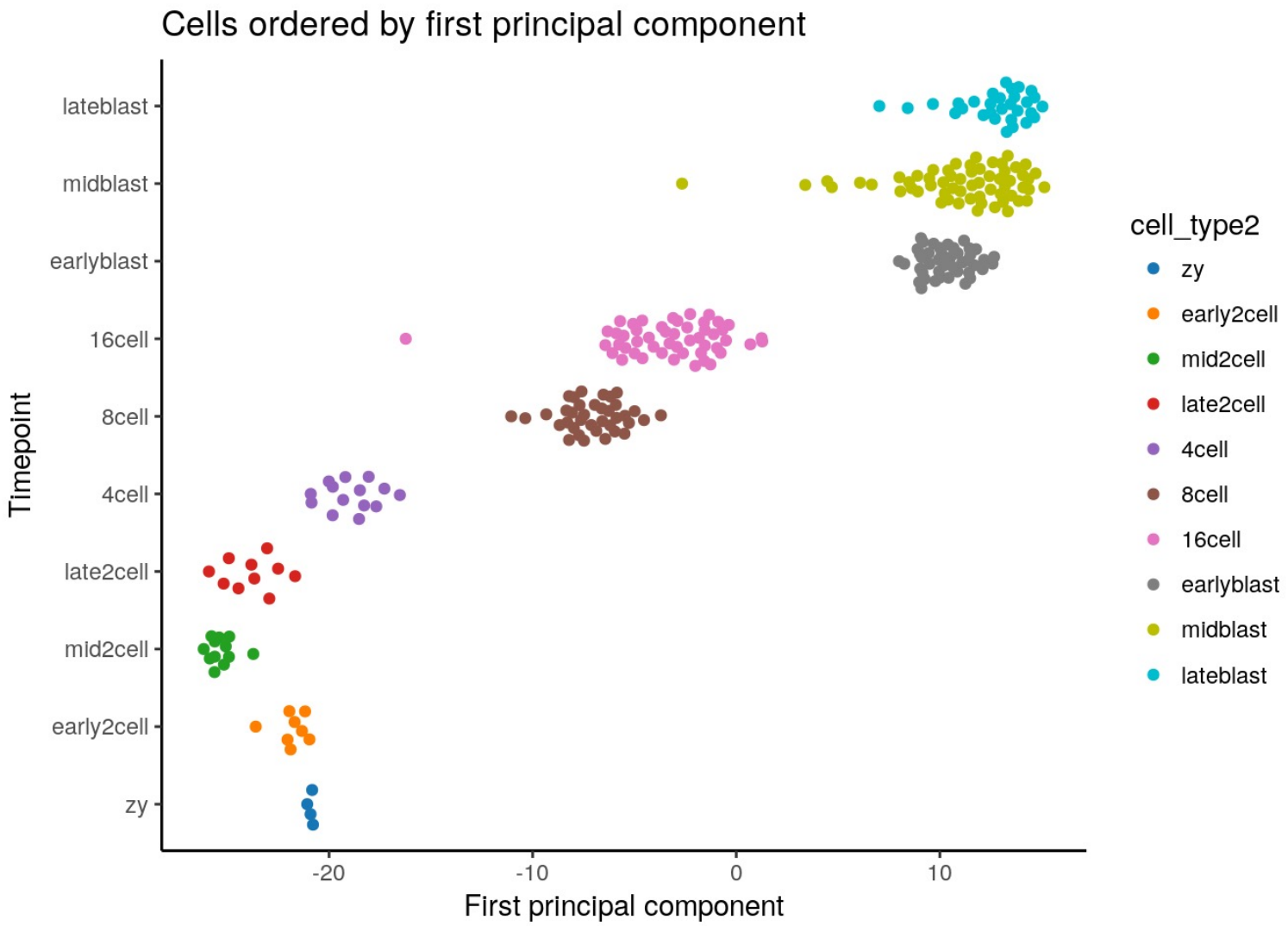
Image denoising by PCA

(a) Noisy image (b) PGPCA (PSNR=33.6) (c) PLPCA (PSNR=34.8)

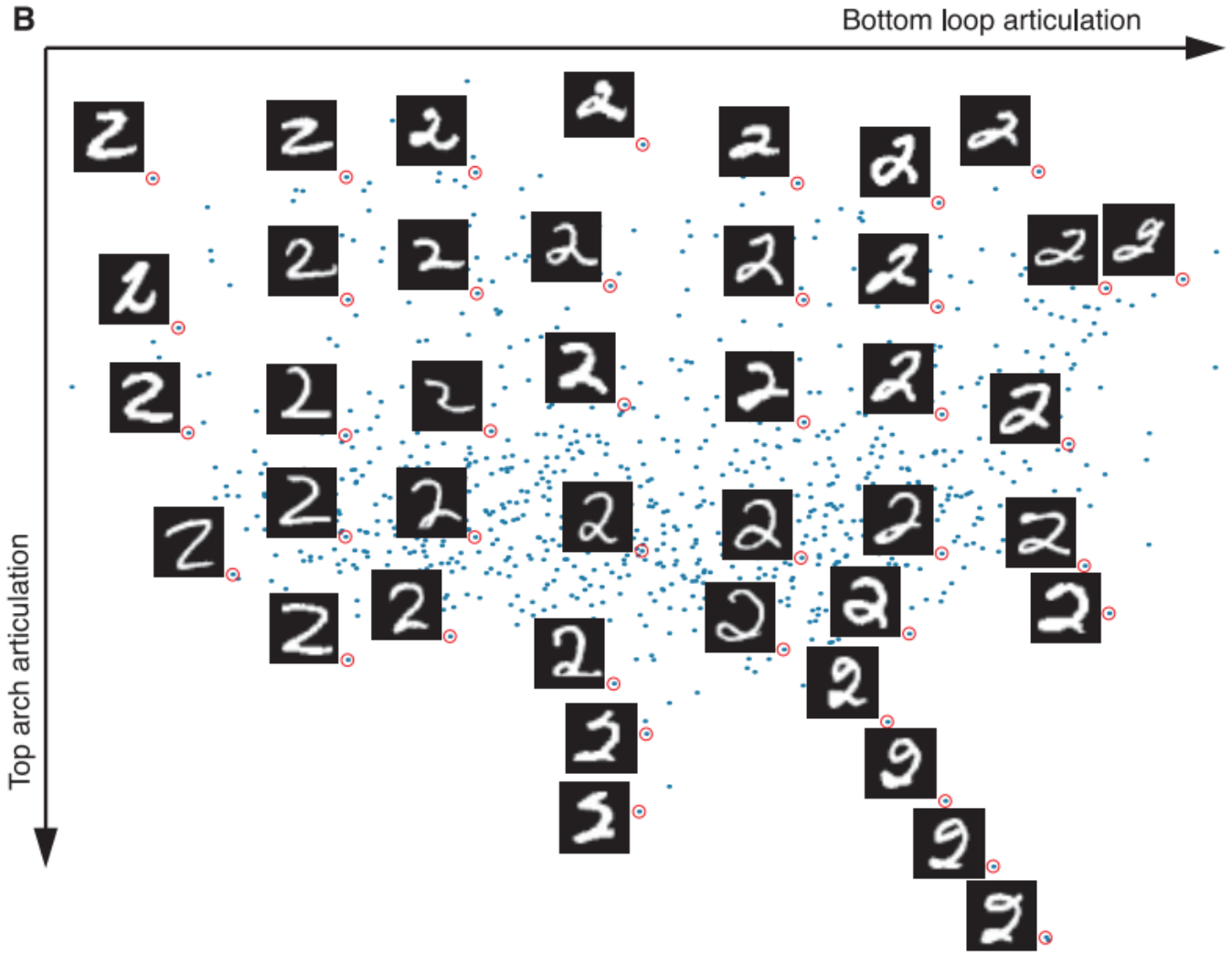
Figure 6: Visual evaluation of the denoising performance of PGPCA and PLPCA on an image (Barbara) damaged by an AWGN with noise level $\sigma = 10$, with their PSNR.

Charles-Alban Deledalle, Joseph Salmon, Arnak Dalalyan; BMVC 2011

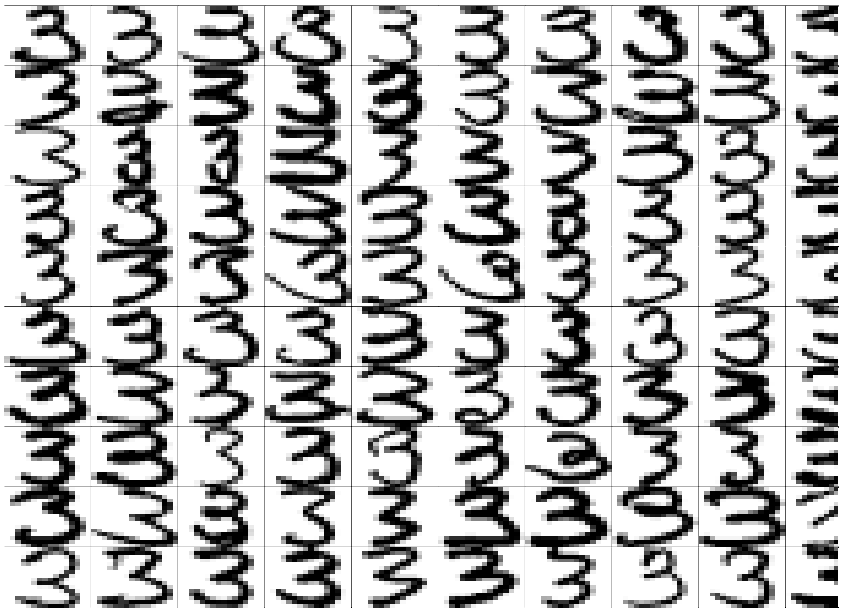
Order the cells by PCA



Visualize the hand-written digits



PCA of hand-written digits



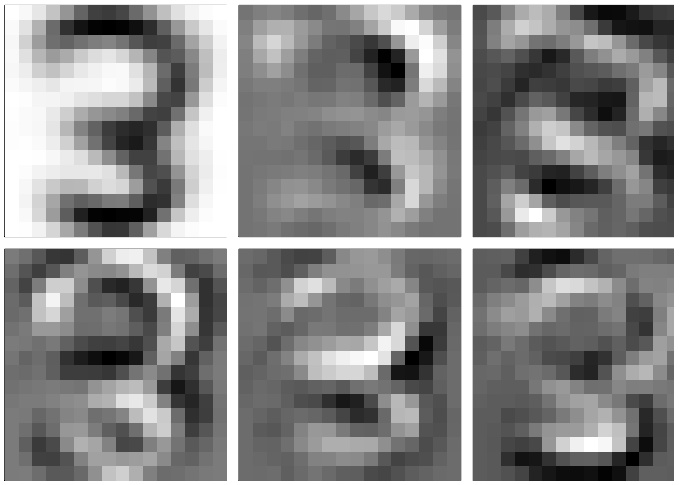


Figure 4: Digits: mean and eigenvectors

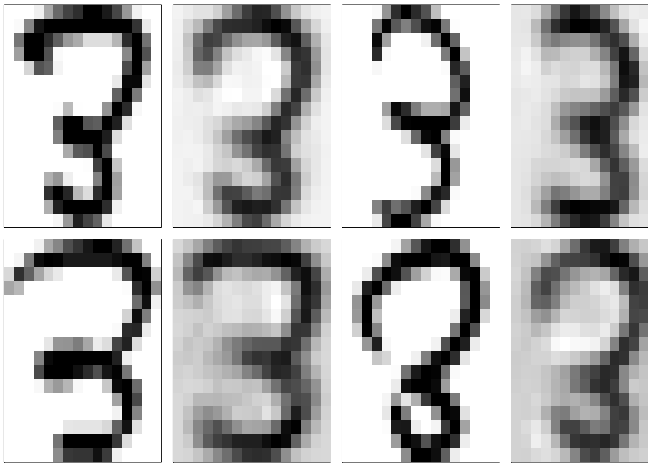


Figure 5: Digits data: Top: digits. Bottom: their reconstructions.

Eigen-face method

- Sirovich and Kirby (1987) showed that PCA could be used on a collection of face images to form a set of basis features.

- Given input image vector $U \in \Re^n$, the mean image vector from the database M, calculate the weight of the kth eigenface as:

  $$w_k = V_k^T (U - M)$$

  Then form a weight vector $W = [w_1, w_2, \ldots, w_k, \ldots, w_n]$

- Compare W with weight vectors W_m of images in the database. Find the Euclidean distance.

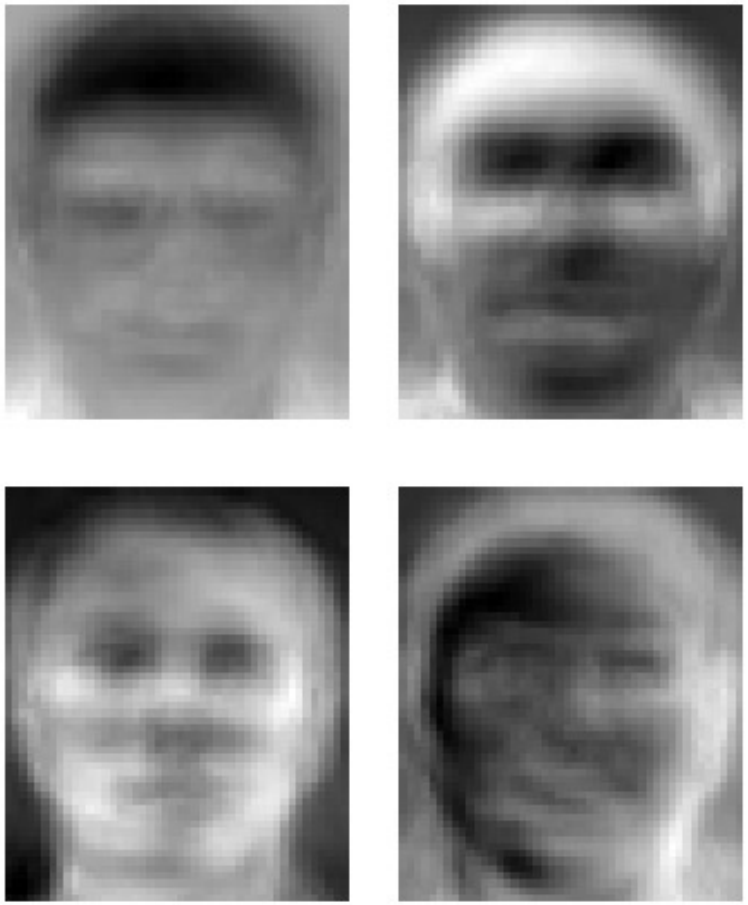
  $$d = ||W - W_m||^2$$

- If $d < \epsilon_1$, then the mth entry in the database is a candidate of recognition.
- If $\epsilon_1 < d < \epsilon_2$, then U may be an unknown face and can be added to the database.
- If $d > \epsilon_2$, U is not a face image.

https://en.wikipedia.org/wiki/Eigenface

Eigen-face examples

Eigen-face



Reconstruction by top-k eigenvectors

k=2 k=15 k=40

Some eigenfaces from AT&T Laboratories Cambridge

Outline

- Recall
  - Principal Component Analysis (PCA)

- **From matrix factorization perspectives**

- Hebbian learning, LMSER and PCA

From matrix factorization perspectives

- Eigen-decomposition

$$\Sigma_x \approx U \Lambda U^T$$

$$U = [\boldsymbol{u}_1, \dots, \boldsymbol{u}_{\mathrm{k}}]^{\mathrm{T}}_{d \times N}$$

$$\Lambda = \mathrm{diag}[\lambda_1, \dots, \lambda_k]$$

Covariance matrix (assume $\boldsymbol{x}_t$ zero mean):

$$\Sigma_x = \frac{1}{N} \sum_{t=1}^{N} \boldsymbol{x}_t \boldsymbol{x}_t^T = \frac{1}{N} X X^T$$

$$X = [\boldsymbol{x}_1, \dots, \boldsymbol{x}_N]_{d \times N}$$

- Singular Value Decomposition (SVD)

$$X = U D V^T$$

$$X X^T = U D V^T \cdot V D U^T = U D^2 U^T$$

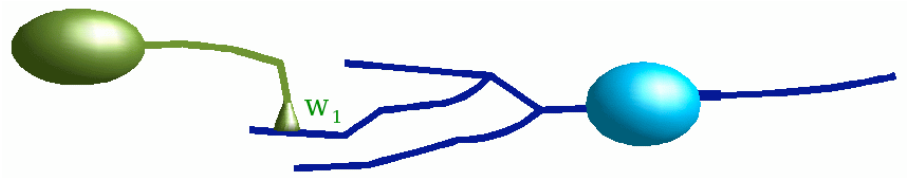
Outline

- Recall
  - Principal Component Analysis (PCA)
- From matrix factorization perspectives
- **Hebbian learning, LMSER and PCA**

Hebbian learning

- Donald Hebb wrote in 1949:
  - When an axon in cell A is near enough to excite cell B and repeatedly and persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency in firing B is increased.

- The Hebbian synapse



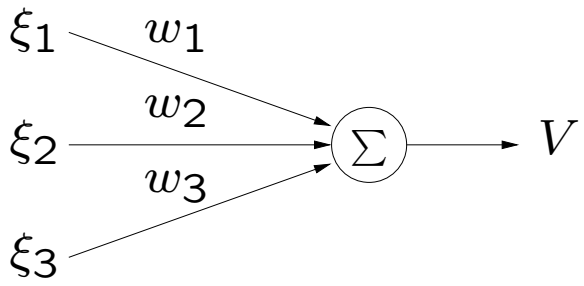
$$\Delta w_1(t) \propto x(t)\, y(t)$$

The Hebbian Neuron

A computational system which implements Hebbian learning.

Let's assume a linear unit; experiment shows this is largely sufficient:

$$V = \sum_j w_j \xi_j = \bar{w}^T \bar{\xi} \tag{2}$$



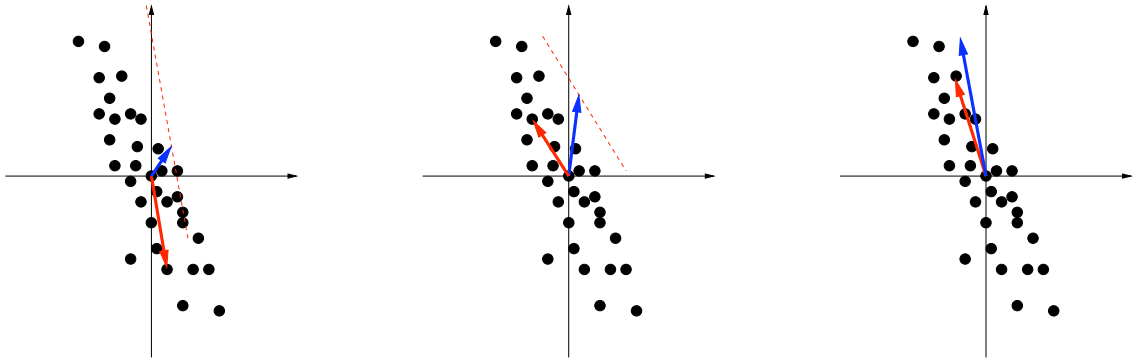
Plain Hebbian learning:

$$\Delta \bar{w} = \eta \, V \, \bar{\xi} \tag{3}$$

A adaptive learning rule

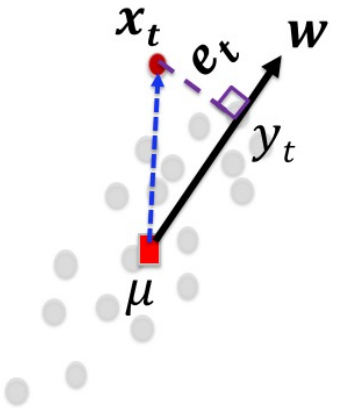
Hebbian learning implements PCA

Recall that $\Delta \bar{w} = \eta V \bar{\xi}$ and that $V = \bar{w}^T \bar{\xi}$.



In this case, the weight vector will ultimately align itself with the *direction of greatest variance* in the data.

$$J(w) = \frac{1}{N} \sum_{t=1}^{N} ||\boldsymbol{x_t} - (\boldsymbol{x_t^T w})\boldsymbol{w}||^2$$



$$\boldsymbol{x_t^T x_t} - (\boldsymbol{x_t^T w})\boldsymbol{w^T x_t} - \boldsymbol{x_t^T}(\boldsymbol{x_t^T w})\boldsymbol{w} + (\boldsymbol{x_t^T w})\boldsymbol{w^T}(\boldsymbol{x_t^T w})\boldsymbol{w}$$
$$= \boldsymbol{x_t^T x_t} - \boldsymbol{w^T}(\boldsymbol{x_t x_t^T})\boldsymbol{w}$$

J. Math. Biology (1982) 15: 267–273

Journal of
Mathematical
Biology

$$\mu_i(t+1) = \mu_i(t) + \gamma\eta(t)[\xi_i(t) - \eta(t)\mu_i(t)] + O(\gamma^2).$$

A Simplified Neuron Model as a Principal Component Analyzer

Erkki Oja

University of Kuopio, Institute of Mathematics, 70100 Kuopio 10, Finland

$$\mu_i(t+1) = \frac{\mu_i(t) + \gamma\eta(t)\xi_i(t)}{\{\sum_{i=1}^{n}[\mu_i(t) + \gamma\eta(t)\xi_i(t)]^2\}^{1/2}}, \qquad \eta = \sum_{i=1}^{n} \mu_i\xi_i.$$

Theorem. *In* (8), *let* C *be positive semidefinite with the largest eigenvalue of multiplicity one, and let* c *be the corresponding normalized eigenvector* (*either of the two possible choices*). *Then if* $z(0)^T c > 0$ (< 0), $z(t)$ *tends to* c $(-c)$ *as* $t \to \infty$. *The points* c *and* $-c$ *are uniformly asymptotically* (*exponentially*) *stable.*

$$\frac{d}{dt}z(t) = Cz(t) - (z(t)^T Cz(t))z(t) \tag{8}$$

$$C = E\{x(t)x(t)^T$$

LMSER for PCA

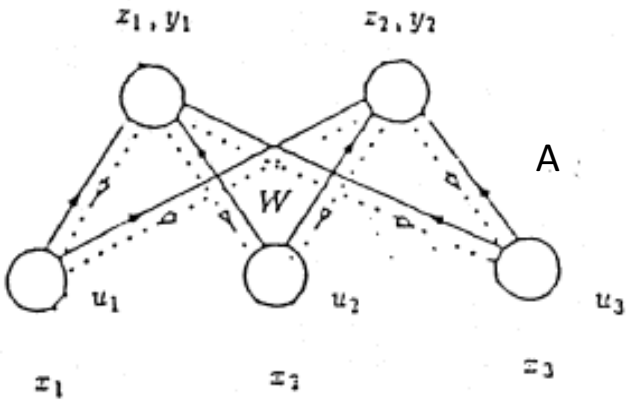
Least Mean Square Error Reconstruction (LMSER)

Xu, 1991, 93 *Proc. IJCNN 91,* Singapore, pp. 2362-2373
Neural Networks, vol. 6, pp. 627-648, 1993

$$s(\bar{y}) = [s(\bar{y}^{(1)}), \ldots, s(\bar{y}^{(m)})]^T$$
$$s(r) = \frac{1}{1+e^{-r}}, \quad \bar{y} = W(x-\mu)$$

$$\boldsymbol{J(W) = \frac{1}{N}\sum_{t=1}^{N} ||x_t - W^T W x_t||^2}$$

W = A^T



A

$$\vec{y} = W\vec{x}, \vec{u} = W^t\vec{y}, \vec{y}^r = W\vec{u}$$

$$\bar{z} = S(\bar{y}) = A_m\bar{y}, A_m = diag\left[a_1, \cdots, a_{n_1}\right]$$

$a_1 > \cdots > a_{n_1}$ are all positive.

$$\tau^W \frac{dW}{dt} = \bar{z}\bar{x}^t - \bar{y}\bar{u}^t$$

$$\tau^W \frac{dW}{dt} = \bar{z}\bar{x}^t - \bar{y}\bar{u}^t + \bar{z}\bar{x}^t - \bar{y}^t\bar{x}^t$$

LMSER implements PCA/PSA

THEOREM 3. *Assume* $\lambda_1 \geq \lambda_2 \ldots \lambda_{n_1} > \lambda_{n_1+1} \geq \lambda_{n_0} > 0$ *and* $\Phi = [\vec{\phi}_1, \ldots, \vec{\phi}_{n_0}]$. *Then all the critical points of eqn* (10*b*) *given by Theorem* 2 *are saddle points of the energy landscape of* J, *except those with* $W = R'\Phi''^t$,

$R''^t R' = I$, $\Phi'^t \Phi' = I$, *where* $\Phi' = D\Phi'$, *and* $D = [D_1 | \vec{0}]$ *with* D_1 *being diagonal matrix and its diagonal elements being either* $+1$ *or* -1. *Furthermore, these* $W = R'\Phi''^t$ *let* $J = \frac{1}{2}E(\|\vec{x} - \vec{u}\|^2) = \frac{1}{2}E(\|\vec{x} - \mathbf{W}^t W\vec{x}\|^2)$ *reach its only local (also global) minimum* $J_{min} = \frac{1}{2} \sum_{i=n_1+1}^{n_0} \lambda_i$.

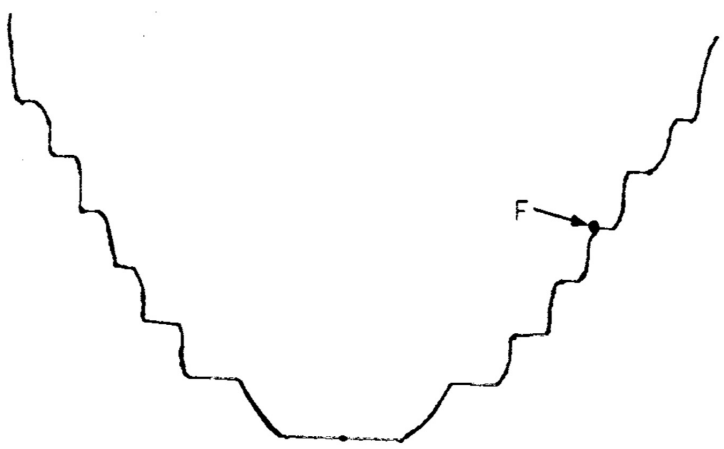


FIGURE 3. The landscape of J. There is only one unique minimum which is the flat bottom, and there are in total $\sum_{l=1}^{n_1} C_{n_0}^l - 1$ plateaux which are the saddle points.

Thank you!