COURANT INSTITUTE OF MATHEMATICAL SCIENCES

Learning representations

Optimization-Based Data Analysis

http://www.cims.nyu.edu/~cfgranda/pages/OBDA_spring16

Carlos Fernandez-Granda

4/11/2016

General problem

For a dataset of n signals

$$X := \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix}$$

learn a model such that

$$x_j \approx \sum_{i=1}^{k} \Phi_i \, A_{ij}, \quad 1 \leq j \leq n, \quad \text{for } k \ll n$$

- $\Phi_1, \ldots, \Phi_k \in \mathbb{R}^d$ are atoms
- $A_1, \ldots, A_n \in \mathbb{R}^k$ are coefficient vectors

Matrix factorization problem

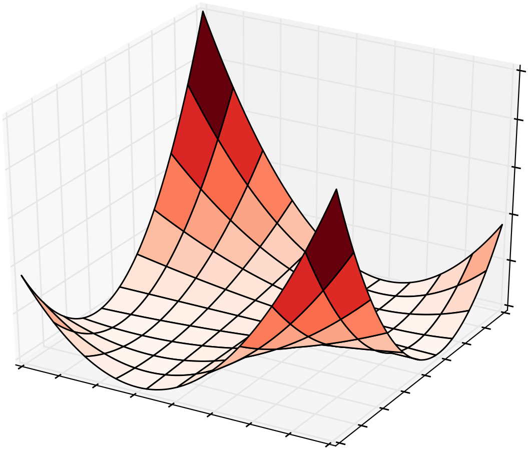
Equivalent formulation

$$X \approx \begin{bmatrix} \Phi_1 & \Phi_2 & \cdots & \Phi_k \end{bmatrix} \begin{bmatrix} A_1 & A_2 & \cdots & A_n \end{bmatrix} = \Phi\, A$$

$\Phi \in \mathbb{R}^{d \times k}$, $A \in \mathbb{R}^{k \times n}$

Nonconvex problem!

Matrix factorization problem

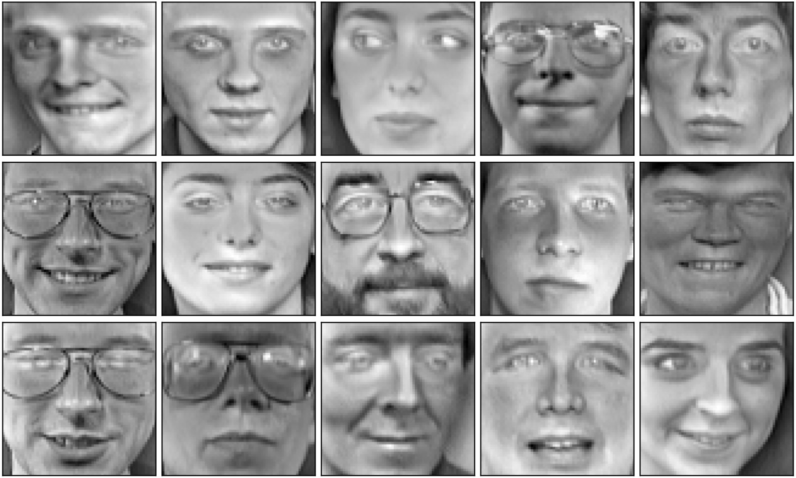


Applications

- Learned representation Φ can be used for compression, denoising, classification, etc.

- Coefficient patterns in A allow to cluster the data

Faces dataset, 10 images of 40 different people



K-means

Low-rank models

- Principal component analysis
- Nonnegative matrix factorization
- Sparse PCA

Dictionary learning

K-means

Aim: Divide $x_1, \ldots x_n$ into k classes

Learn $\Phi_1, \ldots, \Phi_k$ that minimize

$$\sum_{i=1}^{n} \left|\left|x_i - \Phi_{c(i)}\right|\right|_2^2$$

$$c(i) := \arg\min_{1 \leq j \leq k} \left|\left|x_i - \Phi_j\right|\right|_2$$

Matrix-factorization interpretation

Equivalent formulation

$$
\begin{aligned}
X &\approx \begin{bmatrix} \Phi_1 & \Phi_2 & \cdots & \Phi_k \end{bmatrix} \begin{bmatrix} e_{c(1)} & e_{c(2)} & \cdots & e_{c(n)} \end{bmatrix} \\
&= \Phi\, A
\end{aligned}
$$

Lloyd's algorithm

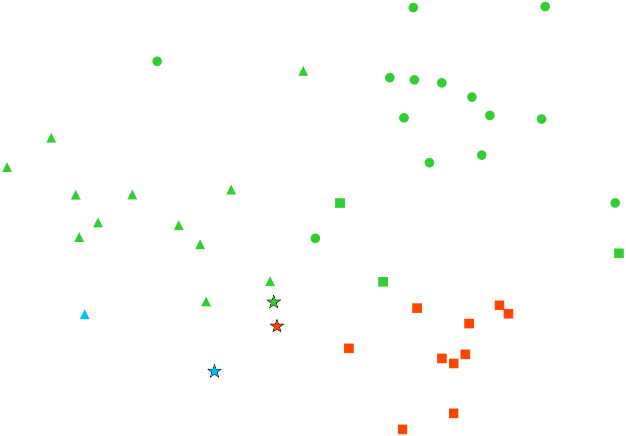
- Initialize $\Phi_1, \ldots, \Phi_k \in \mathbb{R}^d$ randomly
- Repeat
  1. Assignment step

$$c(i) := \arg \min_{1 \leq j \leq k} ||x_i - \Phi_j||_2$$

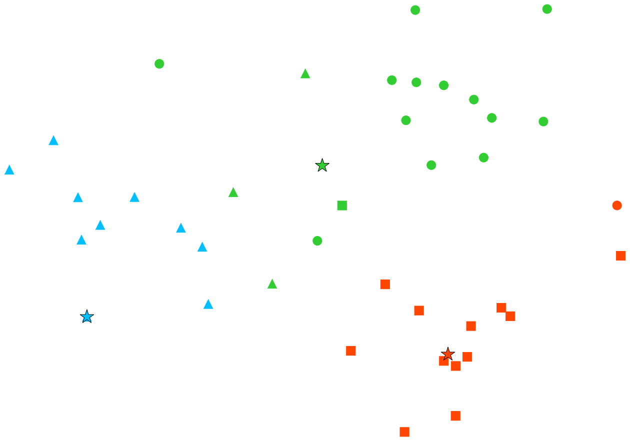
  2. Averaging step

$$\Phi_j := \frac{\sum_{i=1}^{n} \delta(c(j) = i)\ x_i}{\sum_{i=1}^{n} \delta(c(j) = i)}$$

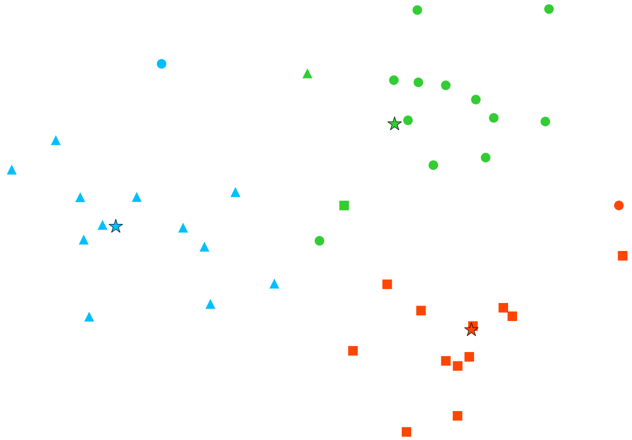
Lloyd's algorithm



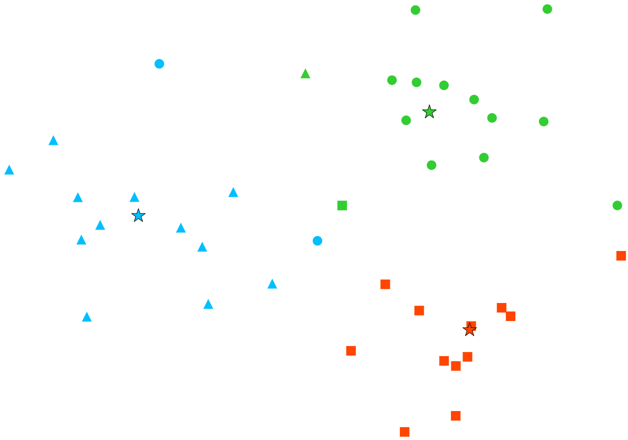
Lloyd's algorithm



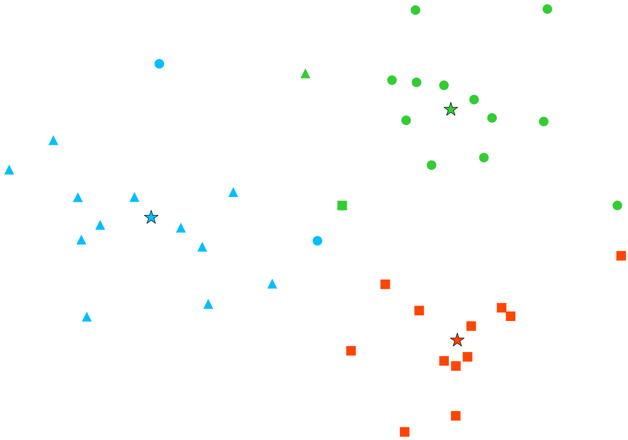
Lloyd's algorithm



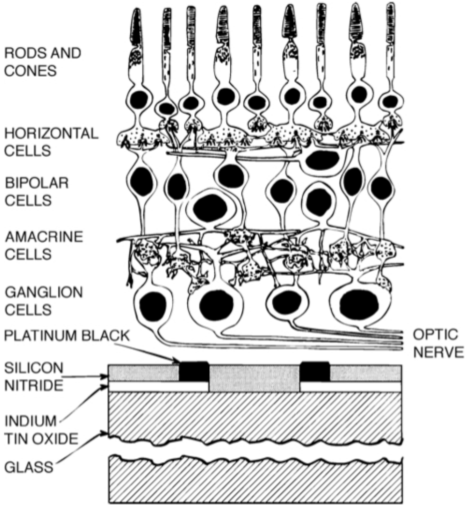
Lloyd's algorithm



Lloyd's algorithm



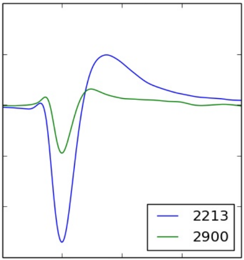
Spike sorting in neuroscience



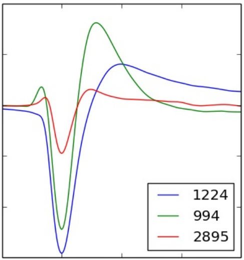
Electrode data from the retina [Litke *et al* 2004]

Spike sorting in neuroscience

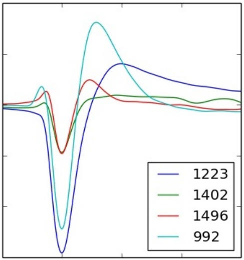
$k = 2$



$k = 3$

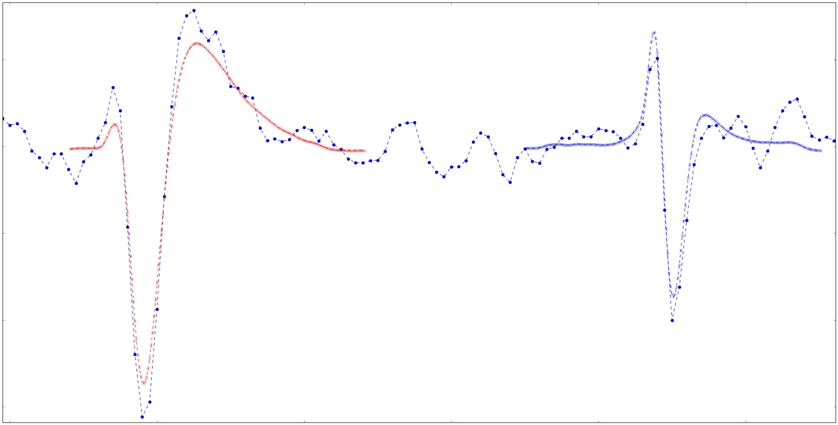


$k = 4$

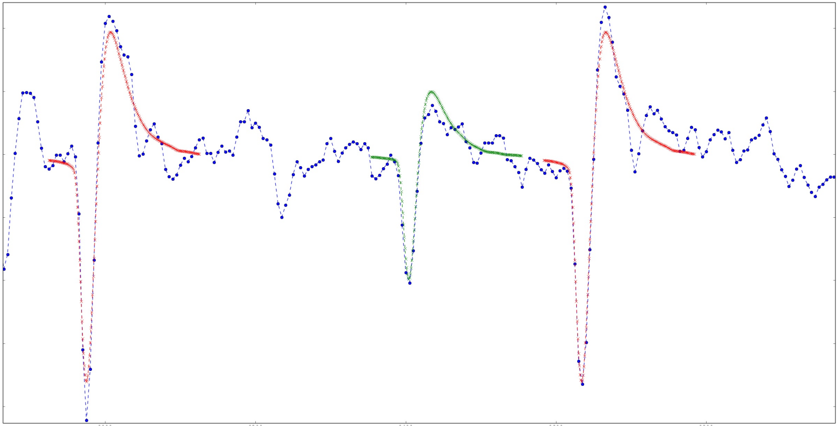


Data from Chichilnisky lab at Stanford

Original data

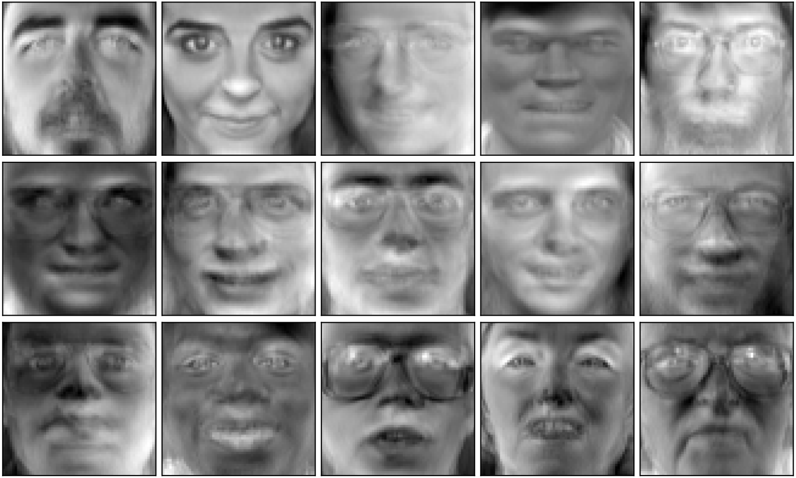


Original data

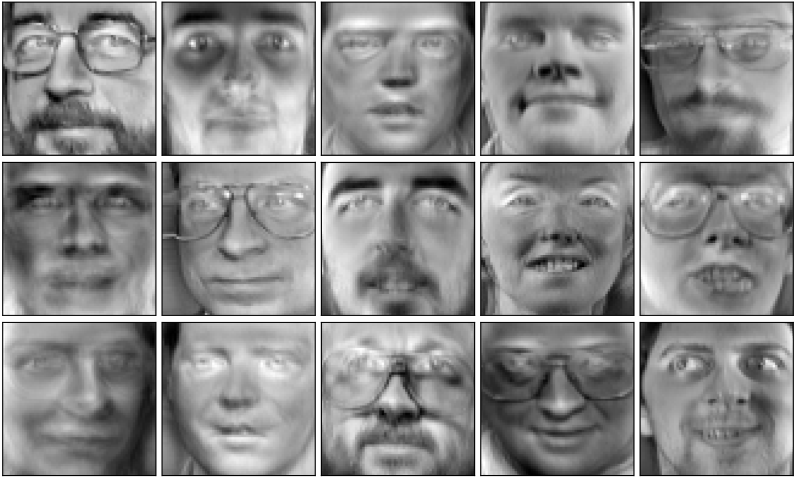


Faces dataset, $k = 5$

Faces dataset, $k = 15$



Faces dataset, $k = 40$



K-means

Low-rank models

- Principal component analysis
- Nonnegative matrix factorization
- Sparse PCA

Dictionary learning

K-means

Low-rank models

Principal component analysis

Nonnegative matrix factorization

Sparse PCA

Dictionary learning

Singular-value decomposition

Every real matrix $A \in \mathbb{R}^{d \times n}$, $d \leq n$ has a unique singular-value decomposition (SVD)

$$
\begin{aligned}
A &= \begin{bmatrix} u_1 & u_2 & \cdots & u_d \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ & & \cdots & \\ 0 & 0 & \cdots & \sigma_d \end{bmatrix} \begin{bmatrix} v_1^T \\ v_2^T \\ \cdots \\ v_d^T \end{bmatrix} \\
&= U \Sigma V^T
\end{aligned}
$$

The singular values are $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_d \geq 0$

The left singular vectors $u_1, \ldots, u_d \in \mathbb{R}^d$ are a basis of the column space

The right singular vectors $v_1, \ldots, v_d \in \mathbb{R}^n$ are a basis of the row space

Variation of the data in a certain direction

Variation of dataset in the direction of unit vector u

$$\begin{aligned}\sum_{i=1}^{n} \left|\left|\mathcal{P}_{\text{span}(u)} x_i\right|\right|_2^2 &= \sum_{i=1}^{n} u^T x_i x_i^T u \\ &= u^T X X^T u \\ &= \left|\left|X^T u\right|\right|_2^2\end{aligned}$$

Directions of maximum variation

$$\sigma_1 = \max_{||u||_2=1} \left|\left|X^T u\right|\right|_2 \quad \text{Nonconvex problem!}$$

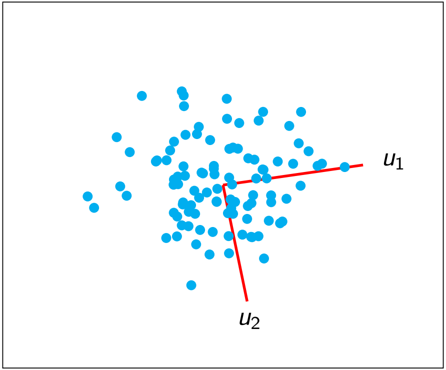
$$U_1 = \arg \max_{||u||_2=1} \left|\left|X^T u\right|\right|_2$$

$$\sigma_j = \max_{\substack{||u||_2=1 \\ u \perp U_1,\ldots,U_{j-1}}} \left|\left|X^T u\right|\right|_2, \quad 2 \leq j \leq d$$

$$U_j = \arg \max_{\substack{||u||_2=1 \\ u \perp U_1,\ldots,U_{j-1}}} \left|\left|X^T u\right|\right|_2, \quad 2 \leq j \leq d$$

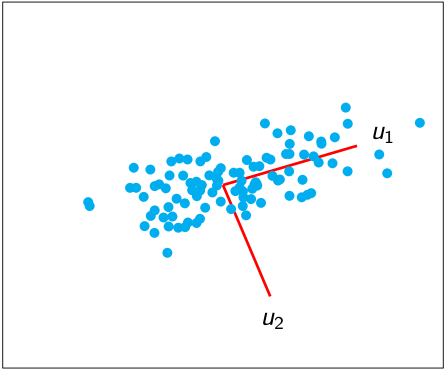
Example: 2D data

$$\frac{\sigma_1}{\sqrt{n}} = 0.705 \qquad \frac{\sigma_2}{\sqrt{n}} = 0.690$$



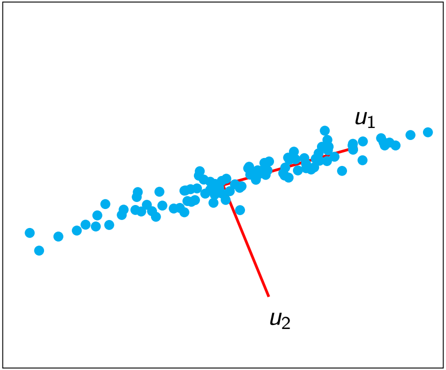
Example: 2D data

$$\frac{\sigma_1}{\sqrt{n}} = 0.9832 \qquad \frac{\sigma_2}{\sqrt{n}} = 0.3559$$



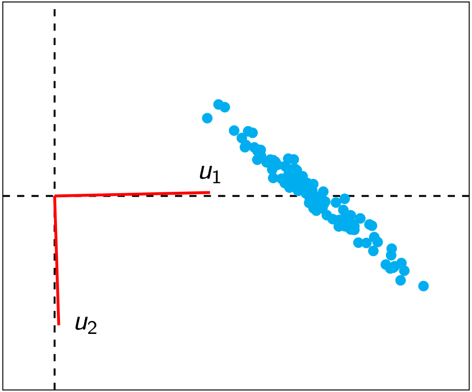
Example: 2D data

$$\frac{\sigma_1}{\sqrt{n}} = 1.3490 \qquad \frac{\sigma_2}{\sqrt{n}} = 0.1438$$



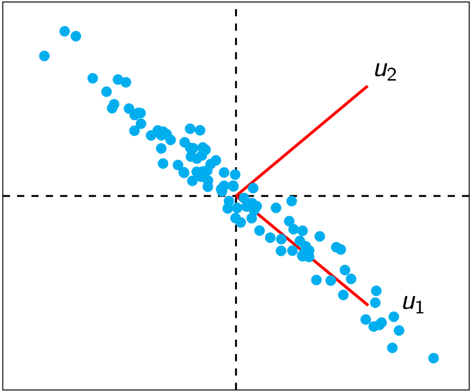
Centering is important!

$$\frac{\sigma_1}{\sqrt{n}} = 5.077 \qquad \frac{\sigma_2}{\sqrt{n}} = 0.889$$



Centering is important!

$$\frac{\sigma_1}{\sqrt{n}} = 1.261 \qquad \frac{\sigma_2}{\sqrt{n}} = 0.139$$



Covariance of a random vector

Assume data $x_1, \ldots, x_n$ are samples from a random d-dimensional vector $\check{x}$ with covariance matrix $\Sigma_{\check{x}}$

The variance of the projection of $\check{x}$ onto $\text{span}(u)$ is

$$\text{Var}\left(\check{x}^T u\right) = u^T \Sigma_{\check{x}} u$$

Eigenvectors of $\Sigma_{\check{x}}$ are directions of maximum variance

Empirical covariance

Given samples $x_1, \ldots, x_n$, the empirical mean is

$$\overline{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$$

and the empirical covariance is

$$\begin{aligned} \overline{\Sigma}_n &:= \frac{1}{n} \sum_{i=1}^{n} \left(x_i - \overline{x} \right) \left(x_i - \overline{x} \right)^T \\ &= \frac{1}{n} X X^T \end{aligned}$$

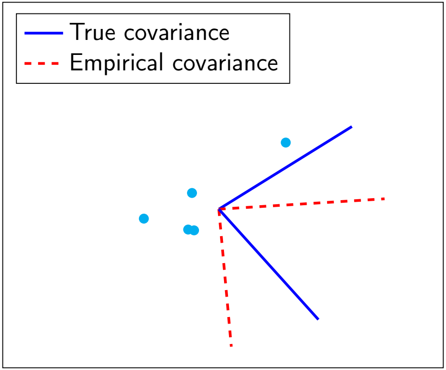
If the mean is known, it's an unbiased estimate of the true covariance

Probabilistic interpretation

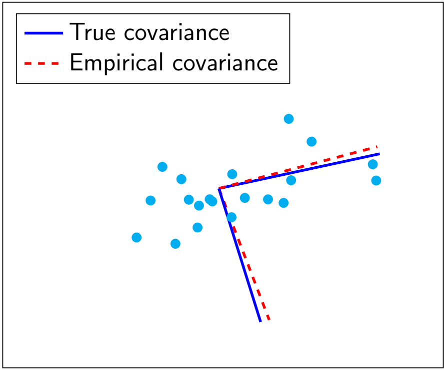
When the estimate converges, PCA finds directions of maximum variance

$$\begin{aligned}\mathrm{Var}\left(\check{x}^T u\right) &= u^T \Sigma_{\check{x}} u \\ &\approx \frac{1}{n} u^T X X^T u \\ &= \frac{1}{n} \left|\left| X^T u \right|\right|_2^2\end{aligned}$$

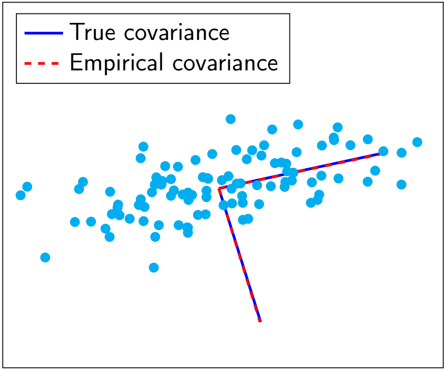
Example: $n = 5$



Example: $n = 20$



Example: $n = 100$



Best k-rank approximation

For any subspace $\mathcal{S}$ of dimension $k \leq \min\{m, n\}$

$$\left|\left|U_{1:k}U_{1:k}^T X\right|\right|_F^2 = \sum_{i=1}^{n} \left|\left|\mathcal{P}_{\text{span}(U_1,U_2,\ldots,U_k)} x_i\right|\right|_2^2$$
$$\geq \sum_{i=1}^{n} \left|\left|\mathcal{P}_{\mathcal{S}} x_i\right|\right|_2^2$$

This implies that for any matrix M of rank k

$$\left|\left|X - U_{1:k}\Sigma_{1:k}V_{1:k}^T\right|\right|_F \leq \left|\left|X - M\right|\right|_F$$

The truncated SVD is the best low-rank approximation

Principal component analysis

Data: $\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_n$

1. Center the data. Compute

$$x_i = \tilde{x}_i - \frac{1}{n}\sum_{i=1}^{n} \tilde{x}_i,$$

2. Group the centered data in a data matrix X

$$X = \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix}$$

   Compute the SVD of X and extract the first k left singular vectors

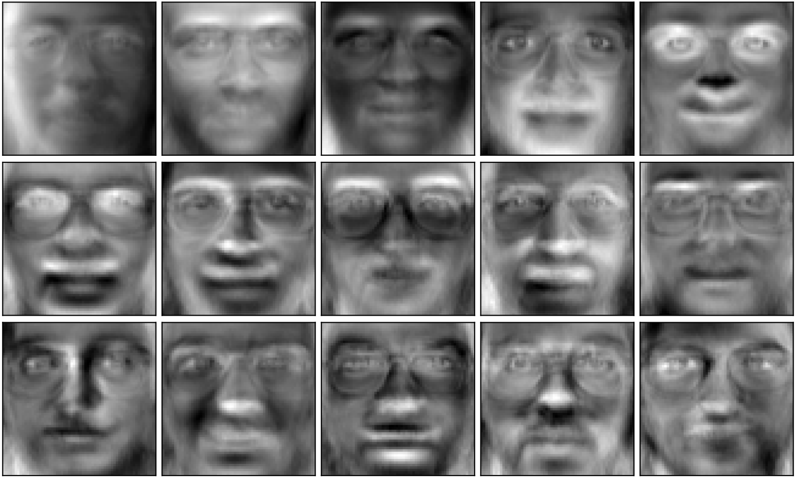
Principal component analysis (PCA)

First k left singular vectors can be interpreted as *atoms*

$$X \approx \begin{bmatrix} U_1 & U_2 & \cdots & U_k \end{bmatrix} A := \Phi A$$

$$A := \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ & & \cdots & \\ 0 & 0 & \cdots & \sigma_k \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \\ \cdots \\ V_k^T \end{bmatrix}$$

Faces dataset



Collaborative filtering

$$A := \begin{pmatrix} 1 & 1 & 5 & 4 \\ 2 & 1 & 4 & 5 \\ 4 & 5 & 2 & 1 \\ 5 & 4 & 2 & 1 \\ 4 & 5 & 1 & 2 \\ 1 & 2 & 5 & 5 \end{pmatrix}$$

	Bob	Molly	Mary	Larry
The Dark Knight	1	1	5	4
Spiderman 3	2	1	4	5
Love Actually	4	5	2	1
Bridget Jones's Diary	5	4	2	1
Pretty Woman	4	5	1	2
Superman 2	1	2	5	5

Centering

$$\mu := \frac{1}{n} \sum_{i=1}^{m} \sum_{j=1}^{n} A_{ij},$$

$$\bar{A} := \begin{bmatrix} \mu & \mu & \cdots & \mu \\ \mu & \mu & \cdots & \mu \\ \cdots & \cdots & \cdots & \cdots \\ \mu & \mu & \cdots & \mu \end{bmatrix}$$

SVD

$$A - \bar{A} = U\Sigma V^T = U \begin{bmatrix} 7.79 & 0 & 0 & 0 \\ 0 & 1.62 & 0 & 0 \\ 0 & 0 & 1.55 & 0 \\ 0 & 0 & 0 & 0.62 \end{bmatrix} V^T$$

First left singular vector

	D. Knight	Sp. 3	Love Act.	B.J.'s Diary	P. Woman	Sup. 2
$U_1 =$	$(-0.45$	-0.39	0.39	0.39	0.39	$-0.45)$

Interpretations:

- Score atom: Centered scores for each person are proportional to U_1
- Coefficients: They cluster movies into action (+) and romantic (-)

First right singular vector

$$V_1 = \begin{pmatrix} \text{Bob} & \text{Molly} & \text{Mary} & \text{Larry} \\ 0.48 & 0.52 & -0.48 & -0.52 \end{pmatrix}$$

Interpretations:

- Score atom: Centered scores for each movie are proportional to V_1
- Coefficients: They cluster people into action (-) and romantic (+)

Rank 1 model

$$\bar{A} + \sigma_1 U_1 V_1^T = \begin{pmatrix} 1.34\,(1) & 1.19\,(1) & 4.66\,(5) & 4.81\,(4) \\ 1.55\,(2) & 1.42\,(1) & 4.45\,(4) & 4.58\,(5) \\ 4.45\,(4) & 4.58\,(5) & 1.55\,(2) & 1.42\,(1) \\ 4.43\,(5) & 4.56\,(4) & 1.57\,(2) & 1.44\,(1) \\ 4.43\,(4) & 4.56\,(5) & 1.57\,(1) & 1.44\,(2) \\ 1.34\,(1) & 1.19\,(2) & 4.66\,(5) & 4.81\,(5) \end{pmatrix}$$

	Bob	Molly	Mary	Larry
The Dark Knight	1.34 (1)	1.19 (1)	4.66 (5)	4.81 (4)
Spiderman 3	1.55 (2)	1.42 (1)	4.45 (4)	4.58 (5)
Love Actually	4.45 (4)	4.58 (5)	1.55 (2)	1.42 (1)
B.J.'s Diary	4.43 (5)	4.56 (4)	1.57 (2)	1.44 (1)
Pretty Woman	4.43 (4)	4.56 (5)	1.57 (1)	1.44 (2)
Superman 2	1.34 (1)	1.19 (2)	4.66 (5)	4.81 (5)

K-means

Low-rank models

- Principal component analysis
- Nonnegative matrix factorization
- Sparse PCA

Dictionary learning

Nonnegative matrix factorization

Nonnegative atoms/coefficients can make results easier to interpret

$$X \approx \Phi\, A, \quad \Phi_{i,j} \geq 0, \; A_{i,j} \geq 0, \; \text{for all } i, j$$

Nonconvex optimization problem:

$$\begin{aligned} \text{minimize} \quad & \left|\left| X - \tilde{\Phi}\, \tilde{A} \right|\right|_2^2 \\ \text{subject to} \quad & \tilde{\Phi}_{i,j} \geq 0, \\ & \tilde{A}_{i,j} \geq 0, \qquad \text{for all } i, j \end{aligned}$$

$\tilde{\Phi} \in \mathbb{R}^{d \times k}$ and $\tilde{A} \in \mathbb{R}^{k \times n}$ for a fixed k

Faces dataset

Topic modeling

$$A := \begin{pmatrix} 6 & 1 & 1 & 0 & 0 & 1 & 9 & 0 & 8 \\ 1 & 0 & 9 & 5 & 8 & 1 & 0 & 1 & 0 \\ 8 & 1 & 0 & 1 & 0 & 0 & 9 & 1 & 7 \\ 0 & 7 & 1 & 0 & 0 & 9 & 1 & 7 & 0 \\ 0 & 5 & 6 & 7 & 5 & 6 & 0 & 7 & 2 \\ 1 & 0 & 8 & 5 & 9 & 2 & 0 & 0 & 1 \end{pmatrix}$$

singer	GDP	senate	election	vote	stock	bass	market	band	Articles
6	1	1	0	0	1	9	0	8	a
1	0	9	5	8	1	0	1	0	b
8	1	0	1	0	0	9	1	7	c
0	7	1	0	0	9	1	7	0	d
0	5	6	7	5	6	0	7	2	e
1	0	8	5	9	2	0	0	1	f

SVD

$$A - \bar{A} = U\Sigma V^T = U \begin{bmatrix} 19.32 & 0 & 0 & 0 & & \\ 0 & 14.46 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4.99 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2.77 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.67 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.93 \end{bmatrix} V^T$$

Left singular vectors

	a	b	c	d	e	f
U_1 =	(−0.51	−0.40	−0.54	−0.11	−0.38	−0.38)
U_2 =	(0.19	−0.45	−0.19	−0.69	−0.2	−0.46)
U_3 =	(0.14	−0.27	−0.09	−0.58	−0.69	−0.29)

Right singular vectors

	singer	GDP	senate	election	vote	stock	bass	market	band
V_1 =	−0.38	0.05	0.40	0.27	0.40	0.17	−0.52	0.14	−0.38
V_2 =	0.16	−0.46	0.33	0.15	0.38	−0.49	0.10	−0.47	0.12
V_3 =	−0.18	−0.18	−0.04	−0.74	−0.05	0.11	−0.10	−0.43	−0.43

Nonnegative matrix factorization

$$X \approx W\ H$$

$$W_{i,j} \geq 0,\ H_{i,j} \geq 0,\ \text{for all } i, j$$

Right nonnegative factors

	singer	GDP	senate	election	vote	stock	bass	market	band
H_1 =	(0.34	0	3.73	2.54	3.67	0.52	0	0.35	0.35)
H_2 =	(0	2.21	0.21	0.45	0	2.64	0.21	2.43	0.22)
H_3 =	(3.22	0.37	0.19	0.2	0	0.12	4.13	0.13	3.43)

Interpretations:

- Count atom: Counts for each doc are weighted sum of H_1, H_2, H_3
- Coefficients: They cluster words into politics, music and economics

Left nonnegative factors

$$
\begin{array}{ccccccccc}
 & & a & b & c & d & e & f \\
W_1 & = & (0.03 & 2.23 & 0 & 0 & 1.59 & 2.24) \\
W_2 & = & (\ 0.1 & 0 & 0.08 & 3.13 & 2.32 & 0\ ) \\
W_3 & = & (2.13 & 0 & 2.22 & 0 & 0 & 0.03)
\end{array}
$$

Interpretations:

- Count atom: Counts for each word are weighted sum of W_1, W_2, W_3
- Coefficients: They cluster docs into politics, music and economics

K-means

Low-rank models

Principal component analysis

Nonnegative matrix factorization

Sparse PCA

Dictionary learning

Sparse PCA

Sparse atoms can make results easier to interpret

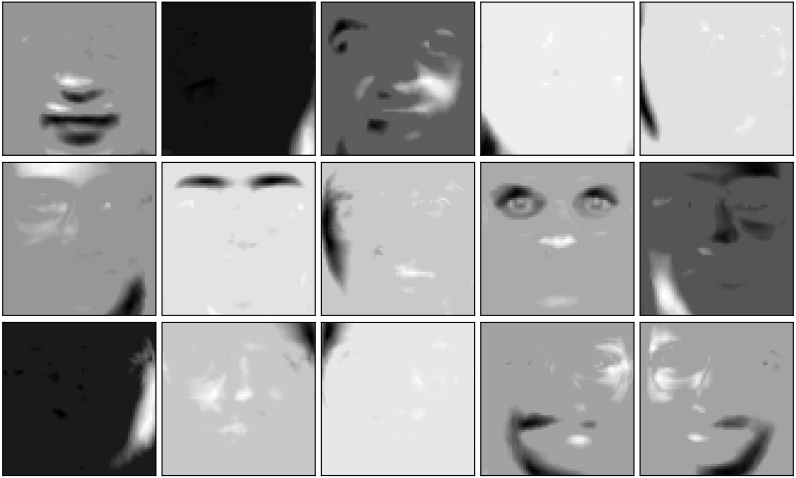
$$X \approx \Phi\, A, \quad \Phi \text{ sparse}$$

Nonconvex optimization problem:

$$\begin{aligned} \text{minimize} \quad & \left|\left|X - \tilde{\Phi}\,\tilde{A}\right|\right|_2^2 + \lambda \sum_{i=1}^{k} \left|\left|\tilde{\Phi}_i\right|\right|_1 \\ \text{subject to} \quad & \left|\left|\tilde{\Phi}_i\right|\right|_2 = 1, \qquad 1 \le i \le k \end{aligned}$$

$\tilde{\Phi} \in \mathbb{R}^{d \times k}$ and $\tilde{A} \in \mathbb{R}^{k \times n}$ for a fixed k

Faces dataset



K-means

Low-rank models

- Principal component analysis
- Nonnegative matrix factorization
- Sparse PCA

Dictionary learning

Dictionary learning

Learn sparsifying dictionary

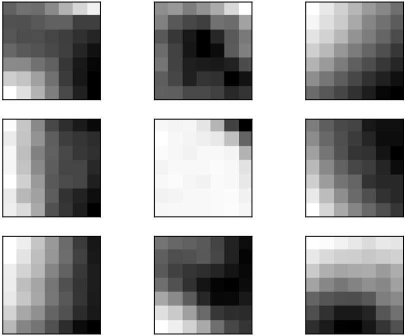
$$X \approx \Phi \, A, \quad A \text{ sparse}$$

Nonconvex optimization problem:

$$\begin{aligned} \text{minimize} \quad & \left|\left| X - \tilde{\Phi} \, \tilde{A} \right|\right|_2^2 + \lambda \sum_{i=1}^{k} \left|\left| \tilde{A}_i \right|\right|_1 \\ \text{subject to} \quad & \left|\left| \tilde{\Phi}_i \right|\right|_2 = 1, \qquad 1 \leq i \leq k \end{aligned}$$

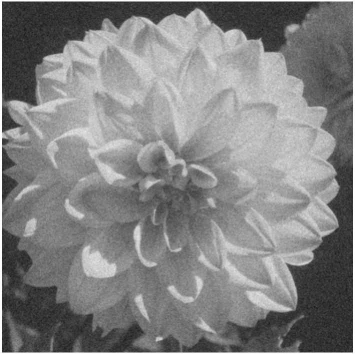
$\tilde{\Phi} \in \mathbb{R}^{d \times k}$ and $\tilde{A} \in \mathbb{R}^{k \times n}$ for a fixed k

Dictionary learning



Denoising via dictionary learning

Denoising via dictionary learning



Denoising via dictionary learning