

Statistics and Learning

Lecturers (alphabetic order):

Julien Bect, Ziad Kobeissi, Gilles Faÿ, Laurent Le Brusquet,
Vincent Lescarret, Arshak Minasyan, Arthur Tenenhaus[†] & Xujia Zhu

[†] Course coordinator

1/53

Lecture 10/9

Unsupervised learning: two examples

In this lecture you will...

- ▶ Understand the main ideas of unsupervised learning through two examples of unsupervised learning tasks.
- ▶ Learn how to reduce the dimension of a dataset using **principal component analysis**.
- ▶ Learn how to partition the data into clusters of similar examples (*clustering*) using the **K-means algorithm**.

2/53

Lecture outline

- 1 – Introduction to unsupervised learning
- 2 – Principal components analysis
- 3 – Clustering
- 4 – A taste of some (more) advanced methods
- 5 – Appendices

3/53

Lecture outline

- 1 – Introduction to unsupervised learning
- 2 – Principal components analysis
- 3 – Clustering
- 4 – A taste of some (more) advanced methods
- 5 – Appendices

Recap: supervised learning

- ▶ We observe **pairs** (X_i, Y_i) :

$$(X_1, Y_1), \dots, (X_n, Y_n) \stackrel{\text{iid}}{\sim} P^{X,Y},$$

with $X_i \in \mathcal{X}$: **instance** and $Y_i \in \mathcal{Y}$: **label**.

- ▶ We want to approach the **optimal predictor**

$$h^* = \operatorname{argmin}_h \mathbb{E}(L(Y, h(X))),$$

which is a **property of the conditional distribution** $P^{Y|X}$:

$$\begin{aligned} h^*(x) &= \operatorname{argmin}_{\tilde{y} \in \mathcal{Y}} \mathbb{E}(L(Y, \tilde{y}) \mid X = x) \\ &= \operatorname{argmin}_{\tilde{y} \in \mathcal{Y}} \int L(y, \tilde{y}) P^{Y|X=x}(dy). \end{aligned}$$

4/53

Unsupervised learning

Learning without a “teacher”:

- ▶ we observe **instances only**,

$$X_1, \dots, X_n \stackrel{\text{iid}}{\sim} P^X,$$

and we are interested in the distribution P^X .

Assume that $\mathcal{X} \subset \mathbb{R}^p$ and that P^X has a pdf f^X .

Problem: curse of dimensionality

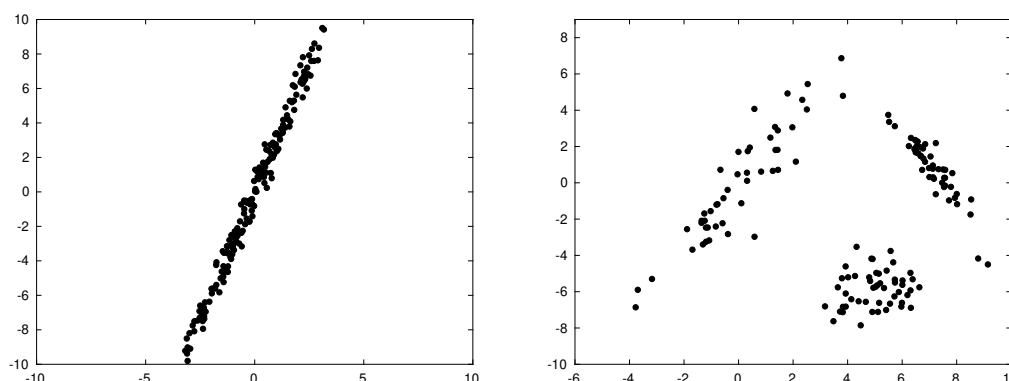
Estimating a “general” pdf f^X has a cost (sample size required to achieve a certain accuracy) that **scales exponentially with the dimension p** .[†]

[†] *Non-parametric statistics*, a branch of statistics which studies among other things density estimation under weak assumptions, provides theoretical results (not covered) that support this claim.

5/53

Goals in unsupervised learning

- 1 Ideally, **estimate the pdf** f^X of the data distribution.
 - ▮ unless p is small enough (say, $p \lesssim 5$, rare in learning problems), this problem is in general **too difficult**[†].
- 2 **Reveal underlying “structures”** in the distribution (without explicitly constructing a density estimator)



[†] In low dimension, one can use, e.g., *kernel density estimators* (not covered).

6/53

Lecture outline

1 – Introduction to unsupervised learning

2 – Principal components analysis

2.1 – Low rank approximation

2.2 – Finding the optimal subspace: SVD

2.3 – Sample variance and covariance of PCA components

3 – Clustering

4 – A taste of some (more) advanced methods

5 – Appendices

Goal: dimension reduction

We are looking for a mapping

$$\begin{aligned} T : \mathcal{X} &\rightarrow \mathcal{Z} \subset \mathbb{R}^q && \text{with } q \ll p \\ x &\mapsto z = T(x) \end{aligned}$$

together with a **reconstruction** mapping

$$\begin{aligned} \tilde{T} : \mathcal{Z} &\rightarrow \mathcal{X} \\ z &\mapsto \hat{x} = \tilde{T}(z) \end{aligned}$$

such that

$$\frac{1}{n} \sum_{i=1}^n L(x_i, \hat{x}_i) = \frac{1}{n} \sum_{i=1}^n L\left(x_i, \underbrace{\tilde{T}(T(x_i))}_{z_i}\right)$$

is as small as possible (where $L(x, \hat{x})$ denotes a loss function).

Remark: more generally, \mathcal{Z} could be a q -dimensional *manifold*, which is an abstract generalization of the concepts of curve ($q = 1$) and surface ($q = 2$); cf. differential geometry.

7/53

Lecture outline

1 – Introduction to unsupervised learning

2 – Principal components analysis

2.1 – Low rank approximation

2.2 – Finding the optimal subspace: SVD

2.3 – Sample variance and covariance of PCA components

3 – Clustering

4 – A taste of some (more) advanced methods

5 – Appendices

“Linear” dimension reduction

Let $x_1, \dots, x_n \in \mathbb{R}^p$ be an observed sample. Let $q < p$.

Definition: affine subspace

$\mathcal{A}_q \subset \mathbb{R}^p$ is an **affine subspace** of **dimension q** if there exists

- ▶ $\mu \in \mathbb{R}^p$,
- ▶ a matrix A of size $p \times q$ with **rank q** ,

such that $\mathcal{A}_q = \text{Aff}_{\mu, A} = \{y \in \mathbb{R}^p \text{ such that } y = \mu + Az, z \in \mathbb{R}^q\}$.

Definition: principal components analysis (PCA)

PCA consist in finding the **best approximation** of the data, for the **quadratic loss**, by an **affine subspace \mathcal{A}_q** .

The dimension q is either given or chosen automatically.

8/53

“Linear” dimension reduction (cont'd)

Thus, we are looking for $\mathcal{A}_q = \text{Aff}_{\mu, A}$ and (z_i) such that

$$\mu, A, (z_i) \in \operatorname{argmin} \sum_{i=1}^n \|x_i - (\mu + Az_i)\|^2. \quad (*)$$



The solution is **not unique**.

⇒ If \tilde{A} has the same range as A , then there exists \tilde{z}_i 's such that $Az_i = \tilde{A}\tilde{z}_i$ for all i .

⇒ We will assume wlog that the columns of A are **orthonormal**:

$$A^\top A = \text{Id}_q.$$

Remark: the orthonormality assumption still does not make A unique. . .

9/53

“Linear” dimension reduction (cont'd)

⇒ Fix some μ , A and (z_i) , and set $\tilde{z}_i = z_i - \bar{z}$. Then

$$\begin{aligned}\mu + Az_i &= \mu + A(\tilde{z}_i + \bar{z}) \\ &= \underbrace{\mu + A\bar{z}}_{\tilde{\mu}} + A\tilde{z}_i.\end{aligned}$$

⇒ We can constrain the z_i 's, wlog, to be such that $\bar{z} = 0$.

10/53

Partial result

Proposition

Minimizing the criterion for a given matrix A leads to:

$$\begin{aligned}\mu &= \bar{x}, \\ z_i &= A^\top (x_i - \bar{x}),\end{aligned}$$

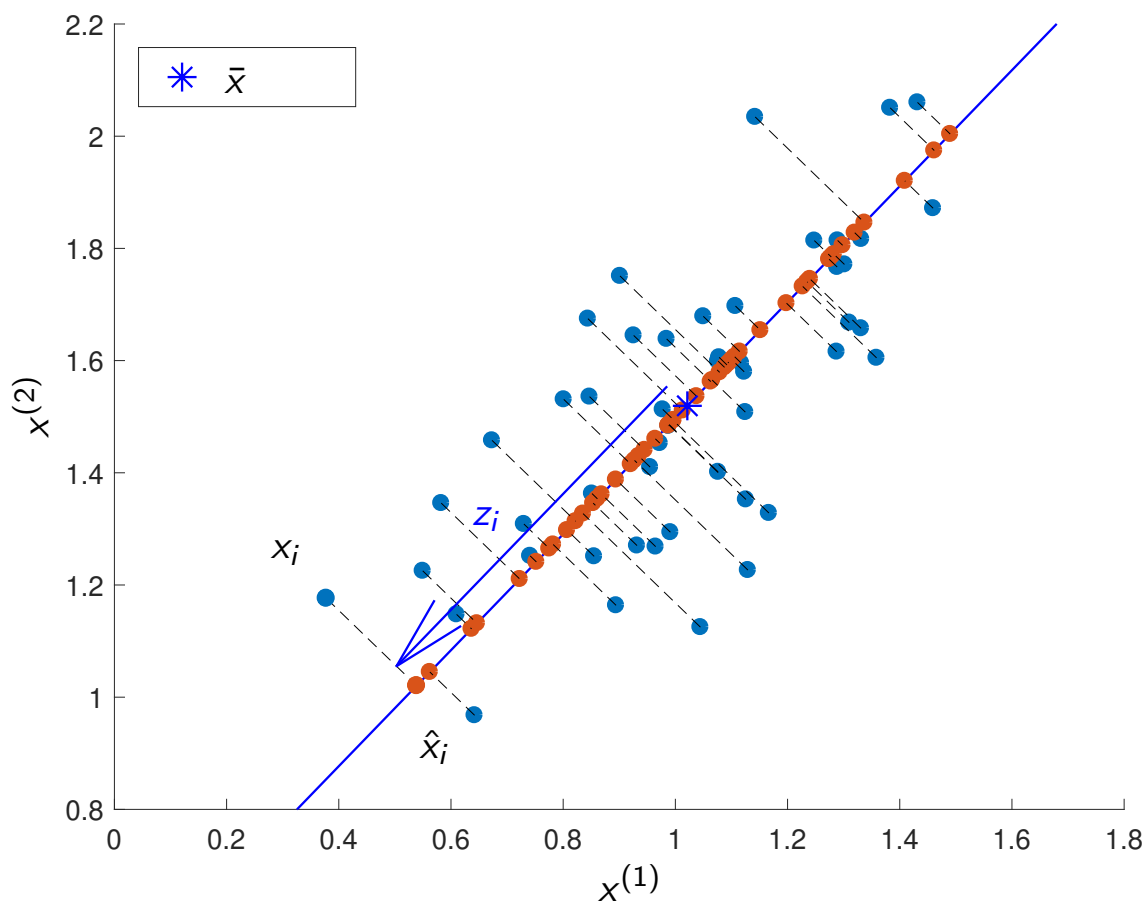
and we have the geometric interpretation:

⇒ $\hat{x}_i = \mu + Az_i$ is the **orthogonal projection** of x_i on $\text{Aff}_{\mu, A}$.

Consequence. Plugging this result into (\star) , we get

$$A = \operatorname{argmin} \sum_{i=1}^n \left\| \left(\operatorname{Id}_p - AA^\top \right) (x_i - \bar{x}) \right\|^2.$$

11/53



Partial result: proof

Fix some A and (z_i) , with $\bar{z} = 0$, and set $v_i = x_i - Az_i$. Then

$$\begin{aligned} \sum_i \|x_i - (\mu + Az_i)\|^2 &= \sum_i \|v_i - \mu\|^2 \\ &= n \left\| \mu - \frac{1}{n} \sum_i v_i \right\|^2 + c \end{aligned}$$

where c does not depend on μ . Therefore, the optimal μ is

$$\mu = \frac{1}{n} \sum_i v_i = \bar{x} - A\bar{z} = \bar{x}.$$

Thus we set $\mu = \bar{x}$, and proceed similarly to determine each of the z_i 's. For all i the minimum is attained (exercise) at

$$z_i = A^\top (x_i - \bar{x}),$$

and we check that $\bar{z} = \frac{1}{n} \sum_i z_i = A^\top (\bar{x} - \bar{x}) = 0$. □

Remark: the expressions can also be obtained quickly by setting the gradient of the criterion to zero.

Lecture outline

1 – Introduction to unsupervised learning

2 – Principal components analysis

2.1 – Low rank approximation

2.2 – Finding the optimal subspace: SVD

2.3 – Sample variance and covariance of PCA components

3 – Clustering

4 – A taste of some (more) advanced methods

5 – Appendices

Notations

Let X be the matrix of observations:

$$X = \begin{pmatrix} (x_1)^\top \\ \vdots \\ (x_n)^\top \end{pmatrix} = \begin{pmatrix} x_1^{(1)} & \dots & x_1^{(p)} \\ \vdots & & \vdots \\ x_n^{(1)} & \dots & x_n^{(p)} \end{pmatrix}$$

We will assume, wlog, that $\bar{x} = 0$.

We are looking for a matrix A such that

$$\begin{aligned} A &= \operatorname{argmin} \sum_{i=1}^n \left\| \left(\operatorname{Id}_p - AA^\top \right) x_i \right\|^2 \\ &= \operatorname{argmin} \left\| \left(\operatorname{Id}_p - AA^\top \right) X^\top \right\|_F^2 \end{aligned}$$

where $\|\cdot\|_F$ denotes the **Frobenius norm**:

$$\|M\|_F^2 = \sum_{i,j} M_{ij}^2 = \operatorname{tr}(M^\top M) = \operatorname{tr}(MM^\top).$$

Singular value decomposition (SVD)

Theorem

Let M be an $n \times p$ real matrix. There exist matrices

- ▶ U , orthogonal with size $n \times n$ ($U^\top U = \text{Id}_n$),
- ▶ V , orthogonal with size $p \times p$ ($V^\top V = \text{Id}_p$),
- ▶ $D = \text{diag}(d_1, \dots, d_r, 0, \dots, 0)$ with size $n \times p$,
with $d_1 \geq d_2 \geq \dots \geq d_r > 0$

such that :

$$M = UDV^\top,$$

and r is the rank of both D et M .

The scalars $d_1, \dots, d_r, 0, \dots, 0$ are the **singular values of M** .

- ▶ d_1^2, \dots, d_r^2 are the non-zero eigenvalues of MM^\top and $M^\top M$.

Proof. See PC 8, bonus exercise. □

15/53

Solution of the optimization problem

Let U , D and V be the matrices obtained from the SVD of X :

$$X = UDV^\top.$$

Fundamental Theorem of PCA

Let

- ▶ v_1, v_2, \dots, v_p the columns of V ,
- ▶ $V_q = (v_1 \mid \dots \mid v_q)$ the submatrix with the first q columns.

Then

$$V_q \in \operatorname{argmin}_A \left\| \left(\text{Id}_p - AA^\top \right) X^\top \right\|_F^2,$$

where A ranges over the set of all $p \times q$ matrices with rank q .

Recap: PCA

Algorithm: Principal components analysis (PCA)

Computing the PCA of a sample (x_1, \dots, x_n) consists in :

- 1 Computing the mean \bar{x} and **centering the data**: $x_i \leftarrow x_i - \bar{x}$.
- 2 Constructing the matrix X of centered data.
- 3 Computing the matrix V from the **SVD of X**
(the singular values are useful too, cf. next section)
- 4 **Reducing the dimension**: $z_i = V_q^\top x_i$.

Reconstruction. $\hat{x}_i = \bar{x} + V_q z_i$.

Vocabulary.

- ▶ v_1, \dots, v_q (columns of V_q): **principal axes**.
- ▶ $z_i^{(1)}, \dots, z_i^{(q)}$: **principal component**.

17/53

Example: handwritten digits (not MNIST, another one!)

Data: $n = 658$ images 16×16 of the digit "3" $\rightarrow p = 256$

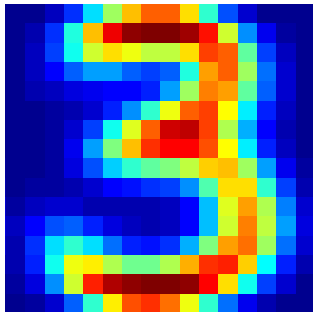


Source : The Elements of Statistical Learning, Springer

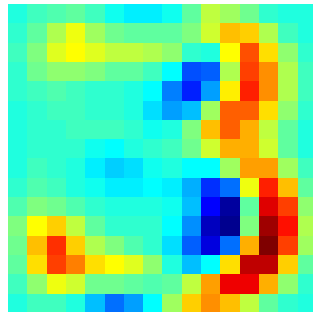
18/53

Example: handwritten digits (cont'd)

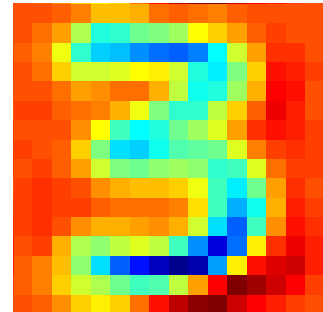
Visualization of the first two principal axes



mean \bar{x}



principal axis v_1

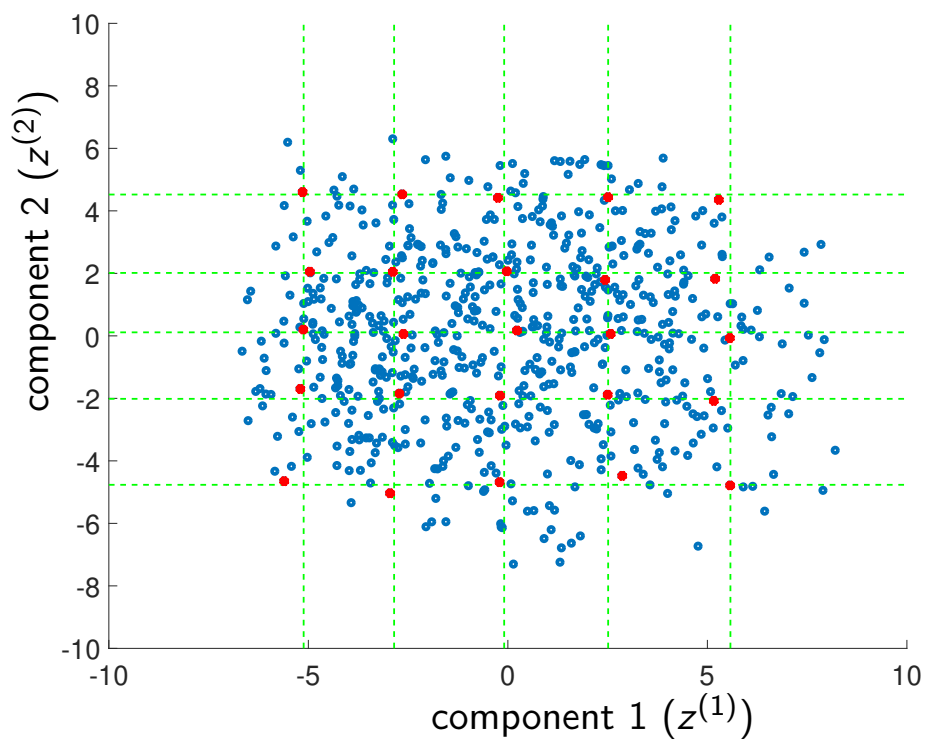


principal axis v_2

$$\forall i, \hat{x}_i = \bar{x} + z_i^{(1)} v_1 + z_i^{(2)} v_2$$

19/53

Principal plane $(z^{(1)}, z^{(2)})$

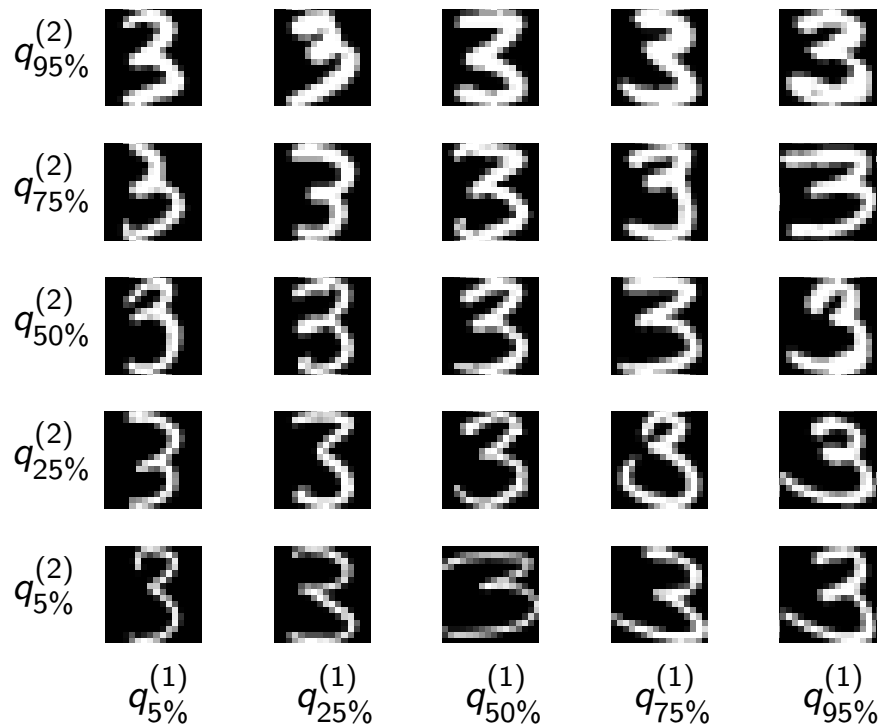


Dashed lines: 5%, 25%, 50%, 75%, 95% quantiles.

Red dots: examples shown on the next slide.

20/53

Interpretation of the components ($z^{(1)}, z^{(2)}$) based on the 25 examples selected on the previous slide.



21/53

Lecture outline

1 – Introduction to unsupervised learning

2 – Principal components analysis

2.1 – Low rank approximation

2.2 – Finding the optimal subspace: SVD

2.3 – Sample variance and covariance of PCA components

3 – Clustering

4 – A taste of some (more) advanced methods

5 – Appendices

Sample covariance matrix of the components

Let $\hat{\Sigma}_Z$ denote the sample covariance matrix of the q components

$$\begin{aligned}\hat{\Sigma}_Z &= \frac{1}{n} \sum_{i=1}^n (z_i - \bar{z})(z_i - \bar{z})^\top \\ &= \frac{1}{n} \sum_{i=1}^n z_i z_i^\top \quad (\text{car } \bar{z} = \frac{1}{n} \sum_{i=1}^n z_i = 0) \\ &= \frac{1}{n} \mathbf{Z}^\top \mathbf{Z}\end{aligned}$$

with $Z = \begin{pmatrix} z_1^\top \\ \vdots \\ z_n^\top \end{pmatrix}$. Recall that $z_i = V_q^\top x_i$, and thus $\mathbf{Z} = \mathbf{X} V_q$.

Using $\mathbf{X} = \mathbf{U} \mathbf{D} \mathbf{V}^\top$, we get

$$\begin{aligned}\hat{\Sigma}_Z &= \frac{1}{n} V_q^\top V D^\top D V^\top V_q \\ &= \frac{1}{n} \text{diag}(d_1^2, \dots, d_q^2).\end{aligned}$$

22/53

Sample covariance matrix of the components (cont'd)

Conclusions.

- ▶ The (sample) variance of component $z^{(j)}$ is $\frac{d_j^2}{n}$.
 - ▮ Components sorted by decreasing variance.
- ▶ The (sample) covariances are equal to zero.
 - ▮ The components are uncorrelated.

23/53

Total variance of a sample

Definition / Proposition

The **total variance** of the p -variate sample (x_1, \dots, x_n) is

$$VT(x_1, \dots, x_n) = \sum_{j=1}^p \text{var} \left(x_1^{(j)}, \dots, x_n^{(j)} \right).$$

With centered x_i 's, we have

$$VT(x_1, \dots, x_n) = \frac{1}{n} \text{tr}(X^\top X) = \frac{1}{n} \sum_{j=1}^r d_j^2.$$

Proof. Using that the x_i 's are centered, we have

$$VT(x_1, \dots, x_n) = \sum_{j=1}^p \left(\frac{1}{n} \sum_{i=1}^n \left(x_i^{(j)} \right)^2 \right) = \frac{1}{n} \|X\|_F^2 = \frac{1}{n} \text{tr}(X^\top X).$$

Then, using $X = UDV^\top$, with $U^\top U = \text{Id}_n$ and $V^\top V = \text{Id}_p$, we obtain

$$VT(x_1, \dots, x_n) = \frac{1}{n} \text{tr}(D^\top D) = \frac{1}{n} \sum_{j=1}^r d_j^2.$$

24/53

Proportion of explained variance

Total variance of the **reconstructed sample** $(\hat{x}_1, \dots, \hat{x}_n)$:

$$VT(\hat{x}_1, \dots, \hat{x}_n) = \frac{1}{n} \text{tr}(\hat{X}^\top \hat{X}) = ?.$$

Using $\hat{X} = ZV_q^\top$, we get:

$$VT(\hat{x}_1, \dots, \hat{x}_n) = \text{tr}(V_q \hat{\Sigma}_Z V_q^\top) = \frac{1}{n} \sum_{j=1}^q d_j^2.$$

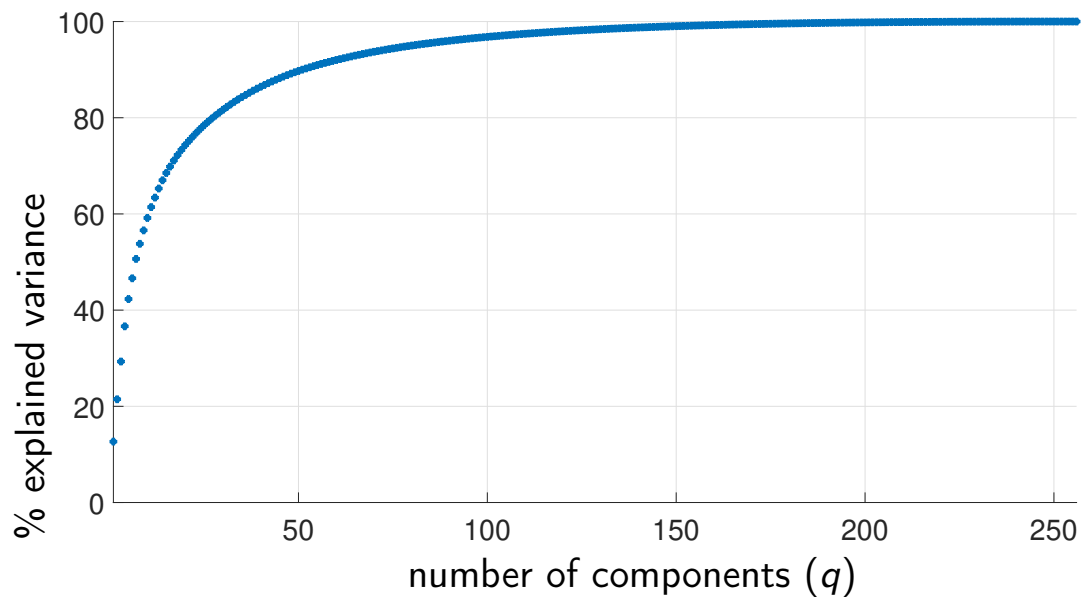
Proportion of explained variance

The **proportion of explained variance** is defined as

$$\frac{VT(\hat{x}_1, \dots, \hat{x}_n)}{VT(x_1, \dots, x_n)} = \frac{\sum_{j=1}^q d_j^2}{\sum_{j=1}^r d_j^2}.$$

25/53

Example: handwritten digits (MNIST, $p = 28^2 = 784$)



Remark: similarity with the coefficient of determination (R^2) in regression.

26/53

Lecture outline

1 – Introduction to unsupervised learning

2 – Principal components analysis

3 – Clustering

3.1 – Dissimilarity

3.2 – K -means algorithm

3.3 – Choice of the number of clusters

4 – A taste of some (more) advanced methods

5 – Appendices

Definition : clustering, clusters

Let $E = \{x_1, \dots, x_n\}$ be a sample of n observations $x_i \in \mathcal{X}$.

- We assume that $\mathcal{X} \subset \mathbb{R}^p$, thus $E \subset \mathbb{R}^p$.

Definitions

Clustering[†] consists in **partitioning** the set E in K **non-empty parts** $E_k \subset E$, $1 \leq k \leq K$, that contain “similar” observations.

The number K is either given or chosen automatically.

The sets E_k are called **groups** or **clusters**.

Notations.

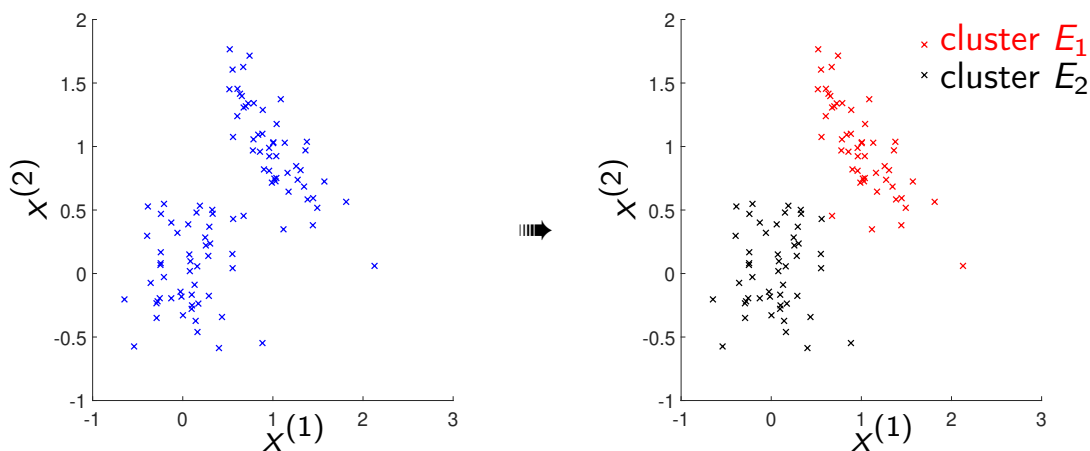
- Denote by $\pi^{(k)} = \{i \leq n \mid x_i \in E^{(k)}\}$ the indices in E_k .
- $\Pi = \{\pi_1, \dots, \pi_K\}$ is a partition of $\{1, \dots, n\}$.

[†] also called *data partitioning*.

27/53

Example of clustering result

Example with $p = 2$ and $K = 2$



28/53

Lecture outline

1 – Introduction to unsupervised learning

2 – Principal components analysis

3 – Clustering

3.1 – Dissimilarity

3.2 – K -means algorithm

3.3 – Choice of the number of clusters

4 – A taste of some (more) advanced methods

5 – Appendices

Dissimilarity: definition

We are looking for a partition such that, for all k ,

- ▶ the instances[†] in cluster E_k are “similar” to each other,
- ▶ and as dissimilar as possible to those in other clusters.

Definition

In clustering algorithms, we call **dissimilarity** the function $D : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ that is used to measure the “distance” between examples.

Remark: **not always a distance** but satisfies in general

- ▶ the **symmetry** property: $D(x, y) = D(y, x)$,
- ▶ the **positivity** property: $D(x, y) \geq 0$.

[†] a.k.a. “examples”, “observations”, “data”, “individuals”...

Dissimilarity: examples

- ▶ General form: $D(x_i, x_{i'}) = \sum_{j=1}^p d_j(x_i^{(j)}, x_{i'}^{(j)})$
- ▶ Quantitative variable: $d_j(x_i^{(j)}, x_{i'}^{(j)}) = f(|x_i^{(j)} - x_{i'}^{(j)}|)$.
Example: $d_j(x_i^{(j)}, x_{i'}^{(j)}) = (x_i^{(j)} - x_{i'}^{(j)})^2$.
Remark: it is often beneficial to normalize the variables:
 $x_i^{(j)} \rightarrow \frac{x_i^{(j)}}{s_j}$, (usual choice for s_j : sample standard deviation)
- ▶ Qualitative variable: $d_j(x_i^{(j)}, x_{i'}^{(j)}) = \text{cste}$ if $x_i^{(j)} \neq x_{i'}^{(j)}$ (0 otherwise)

30/53

Within-cluster and between-cluster inertia

Let us write $d_{ii'} = D(x_i, x_{i'})$.

Within-cluster inertia

Within-cluster inertia is defined as:

$$W(\Pi) = \frac{1}{2} \sum_{k=1}^K \sum_{i, i' \in \pi_k} d_{ii'}.$$

(W=Within)

Between-cluster inertia

Between-cluster inertia is defined as:

$$B(\Pi) = \frac{1}{2} \sum_{k, k' \neq k} \sum_{i \in \pi_k} \sum_{i' \in \pi_{k'}} d_{ii'}.$$

(B=Between)

31/53

Within-cluster and between-cluster inertia (cont'd)

Property

$$W(\Pi) + B(\Pi) = \frac{1}{2} \sum_{i,j'} d_{ij'}$$

Definition

$T = \frac{1}{2} \sum_{i,j'} d_{ij'}$ is the **total inertia**.

- Does not depend on the partition.

Proof of the property:

$$\begin{aligned} T &= \frac{1}{2} \sum_{i,j'} d_{ij'} = \frac{1}{2} \sum_{k,k'} \sum_{i \in \pi_k} \sum_{i' \in \pi_{k'}} d_{ii'} \\ &= \underbrace{\frac{1}{2} \sum_k \sum_{i,i' \in \pi_k} d_{ii'}}_{W(\Pi)} + \underbrace{\frac{1}{2} \sum_{k,k' \neq k} \sum_{i \in \pi_k} \sum_{i' \in \pi_{k'}} d_{ii'}}_{B(\Pi)} \end{aligned}$$

32/53

Optimal partition

We would like to find the **optimal partition**:

$$\Pi_{\star} = \arg \min_{\Pi} W(\Pi)$$

Remark: since $W(\Pi) + B(\Pi) = T$, $\Pi_{\star} = \arg \max_{\Pi} B(\Pi)$.

Problem : this is a **combinatorial** optimization problem

- 34105 partitions for $n = 10$ and $K = 4$,
- $\approx 7.5 \cdot 10^{11}$ partitions for $n = 20$ and $K = 5$.

Solution : look for a sub-optimal solution

- ➡ K -means algorithm

33/53

Lecture outline

1 – Introduction to unsupervised learning

2 – Principal components analysis

3 – Clustering

3.1 – Dissimilarity


3.2 – K -means algorithm

3.3 – Choice of the number of clusters

4 – A taste of some (more) advanced methods

5 – Appendices

Dissimilarity considered here : $d_{ii'} = \|x_i - x_{i'}\|^2$.

With this choice of dissimilarity ( proof):

$$W(\Pi) = \sum_{k=1}^K n_k \sum_{i \in \pi_k} \|x_i - \bar{x}_k\|^2$$

where $\bar{x}_k = \frac{1}{|\pi_k|} \sum_{i \in \pi_k} x_i$ is the barycenter of the cluster, $n_k = |\pi_k|$.

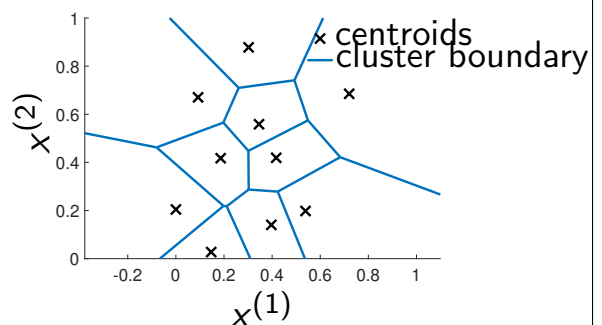
⇒ \bar{x}_k is called the **centroid** of cluster k .

Principle of the K -means algorithm

Iteratively,

- ▶ Given a partition Π , compute the centroids \bar{x}_k .
- ▶ Modify Π in such a way that each x_i is associated to the cluster π_k whose (current) centroid \bar{x}_k is the closest.

⇒ Voronoï diagram



K-means algorithm

Require: $K > 0$

{number of clusters}
{centroids initialization}

Require: $(\bar{x}_{1,0}, \dots, \bar{x}_{K,0})$

$t \leftarrow 0$

repeat

Step 1

{construction of Π_t from the centroids}

for all k **do**

$$\pi_{k,t} = \{i \text{ s.t. } k = \arg \min_{k'} \|x_i - \bar{x}_{k',t}\|\}$$

end for

Step 2

{centroids update}

for all k **do**

$$\bar{x}_{k,t} = \frac{1}{|\pi_{k,t}|} \sum_{i \in \pi_{k,t}} x_i$$

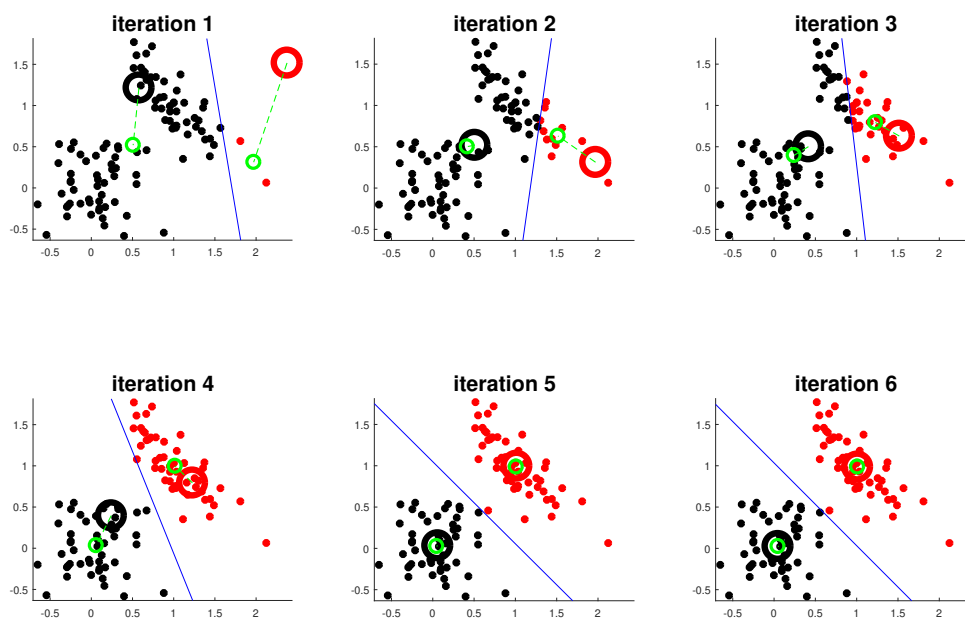
end for

$t \leftarrow t + 1$

until $W(\Pi_{t-1}) = W(\Pi_{t-2})$

return Π_{t-1}

35/53



36/53

Properties of the K –means algorithm

Proposition

Let $(\Pi_t)_{t \geq 0}$ denote the sequence of partitions constructed by the algorithm.

Then, there exists T such that :

- ① $\forall t \leq T, W(\Pi_t) < W(\Pi_{t-1}),$
- ② $W(\Pi_{T+1}) = W(\Pi_T).$



The algorithm terminates in a finite number of iterations, but

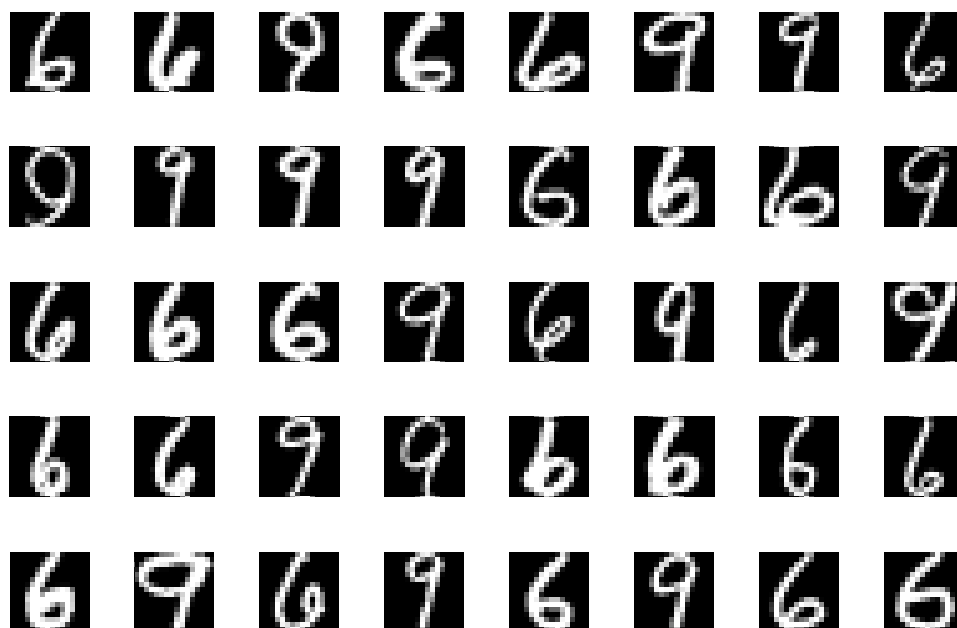
- ▶ the partition Π_T is **not, in general, the optimal partition**;
- ▶ it depends on the starting point $(\bar{x}_{1,0}, \dots, \bar{x}_{K,0})$.

⇒ Recommended: several trials with random starting points.

37/53

Example: handwritten digits

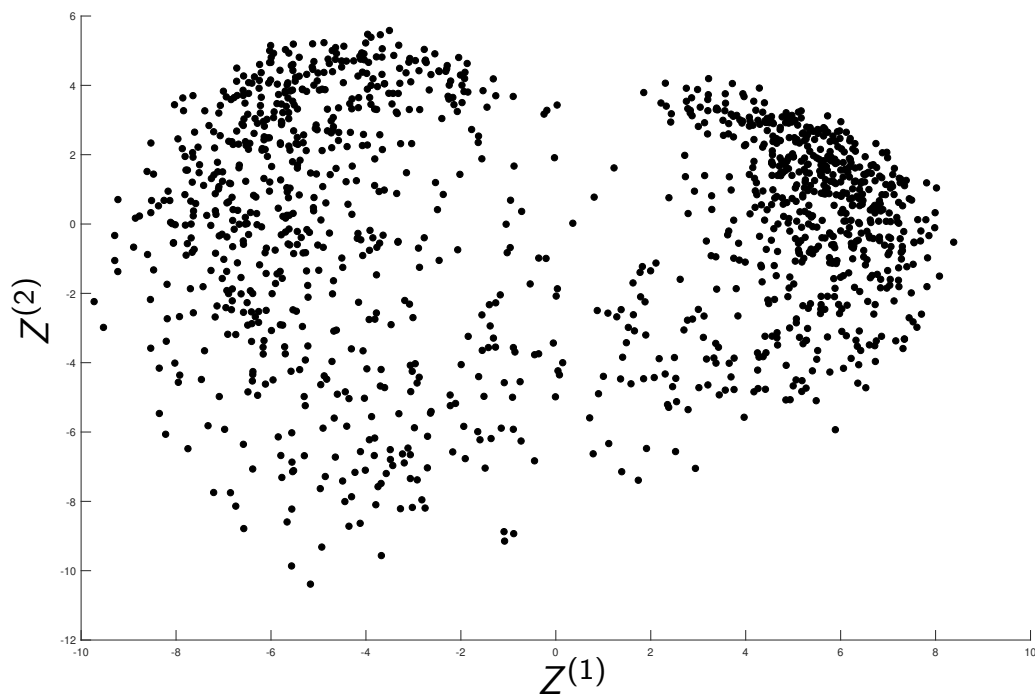
Consider the digits “6” and “9” (644 images each).



38/53

Example: handwritten digits

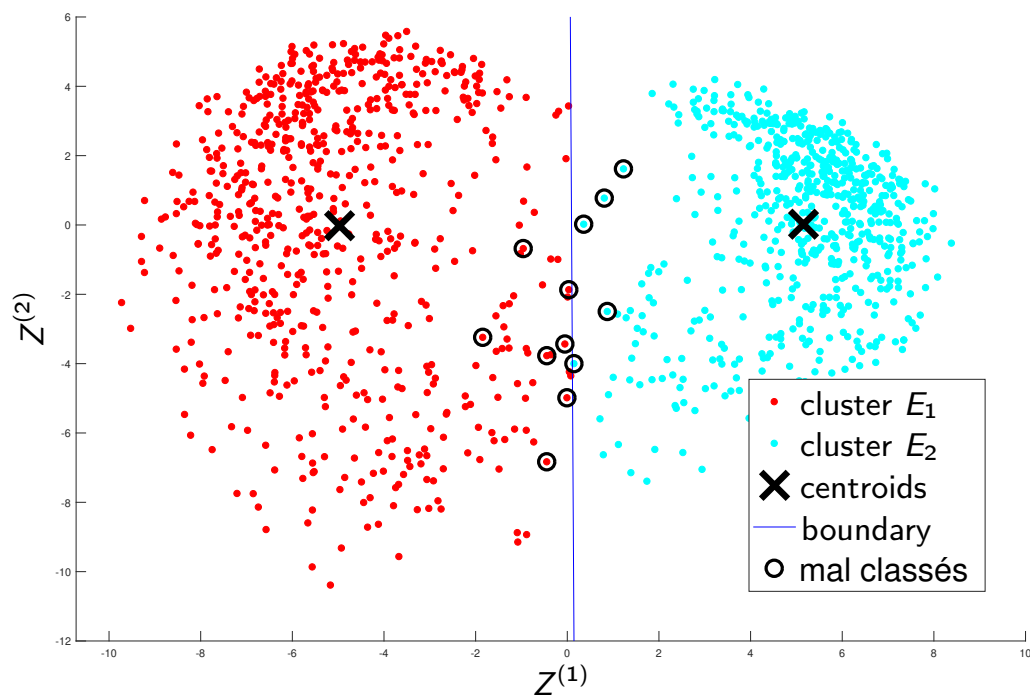
Represent each image by its first two principal components.



39/53

Example: handwritten digits

missclassification rate: 0.92%



Note: here we use the labels, which are assumed unavailable in the non-supervised setting, to the sole purpose of evaluating the quality of the partition that we have obtained.

40/53

Lecture outline

1 – Introduction to unsupervised learning

2 – Principal components analysis

3 – Clustering

3.1 – Dissimilarity

3.2 – K -means algorithm

3.3 – Choice of the number of clusters

4 – A taste of some (more) advanced methods

5 – Appendices

Homogeneity / dispersion

Reminder. We are looking for a partition such that, for all k ,

- ▶ the instances[†] in cluster E_k are “similar” to each other,
- ▶ and as dissimilar as possible to those in other clusters.

Definition: dispersion measure

The dispersion of cluster E_k is (often) measured by

$$S_k = \left(\frac{1}{|\pi_k|} \sum_{i \in \pi_k} \|x_i - \bar{x}_k\|^q \right)^{\frac{1}{q}},$$

with q a positive real number, to be chosen[†].

Interpretation. The smaller S_k , the more homogeneous the cluster.

[†] P.-H. Cournède's lecture notes and scikit-learn use $q = 1$.

Davies-Bouldin index

Definition: similarity of clusters E_k and $E_{k'}$

$$R_{k,k'} = \frac{S_k + S_{k'}}{\|\bar{x}_k - \bar{x}_{k'}\|}, \quad 1 \leq k, k' \leq K, \quad k \neq k'.$$

Interpretation. The clusters are more dissimilar when their dispersions are small with respect to the distance between their centroids.

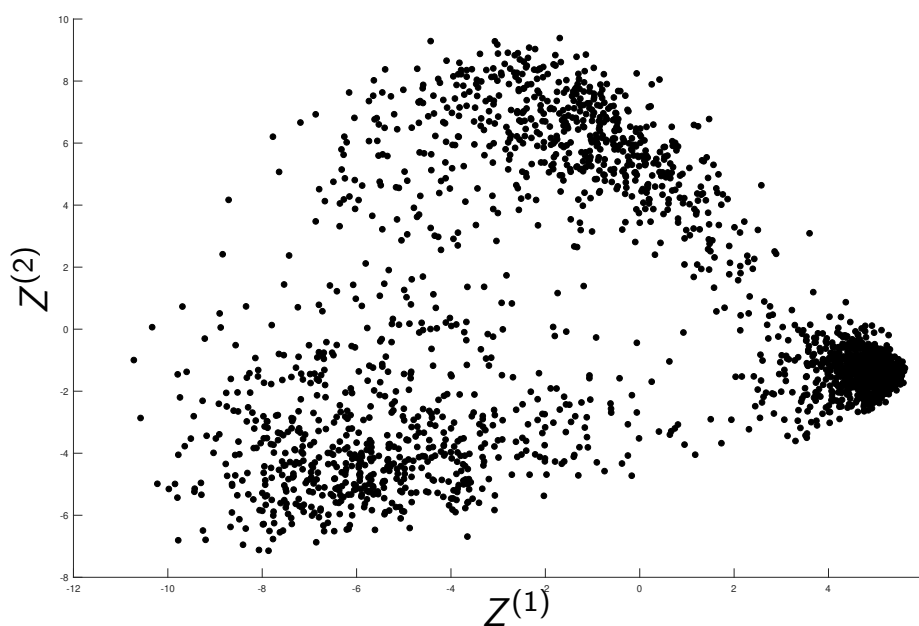
Definition: Davies-Bouldin index of a partition

$$DB = \frac{1}{K} \sum_{k=1}^K \max_{k' \neq k} R_{k,k'}$$

► Use: choose K in order to **minimize DB**.

42/53

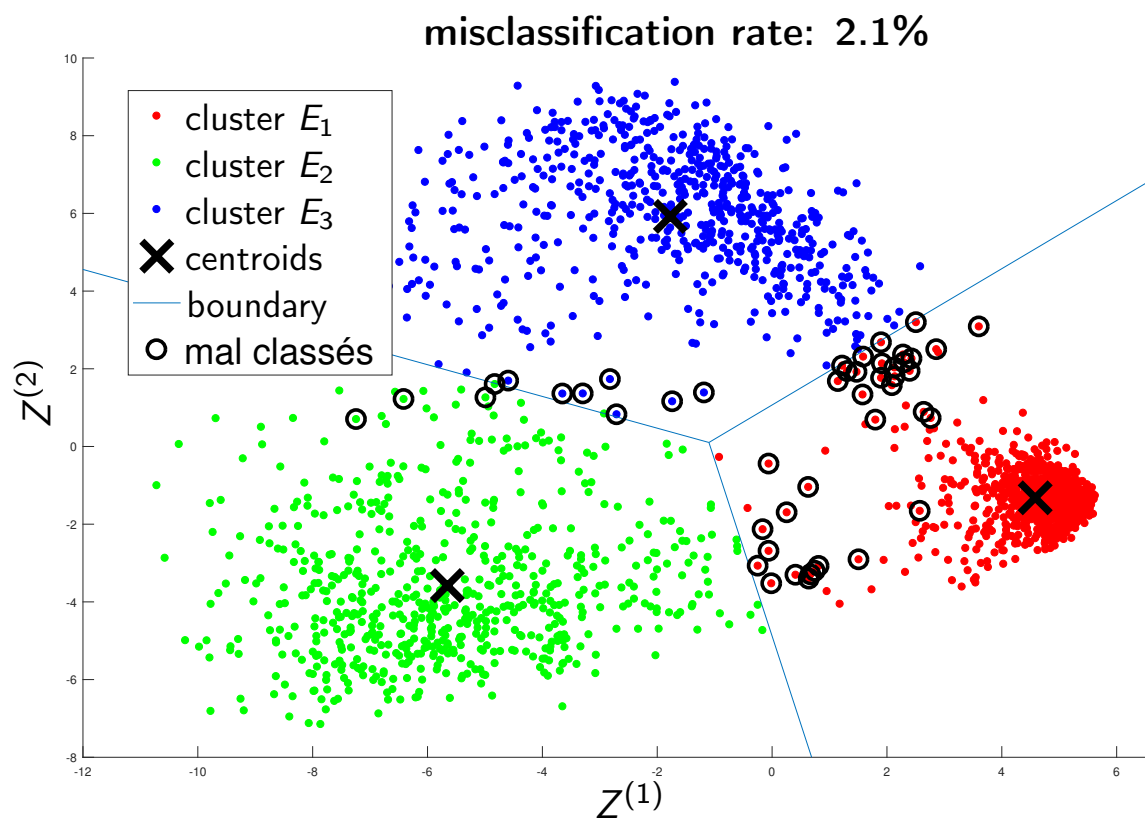
Example: handwritten digits with digits 1, 6 and 9



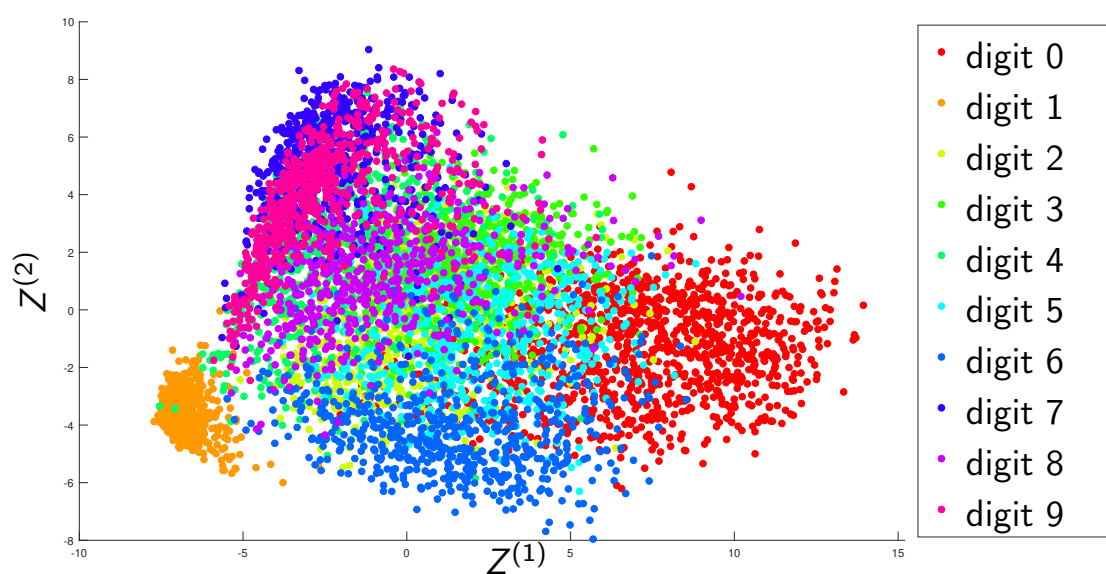
K	2	3	4	5	6	7	8
$DB(K)$	0.76	0.42	0.77	0.89	0.76	0.77	0.79

43/53

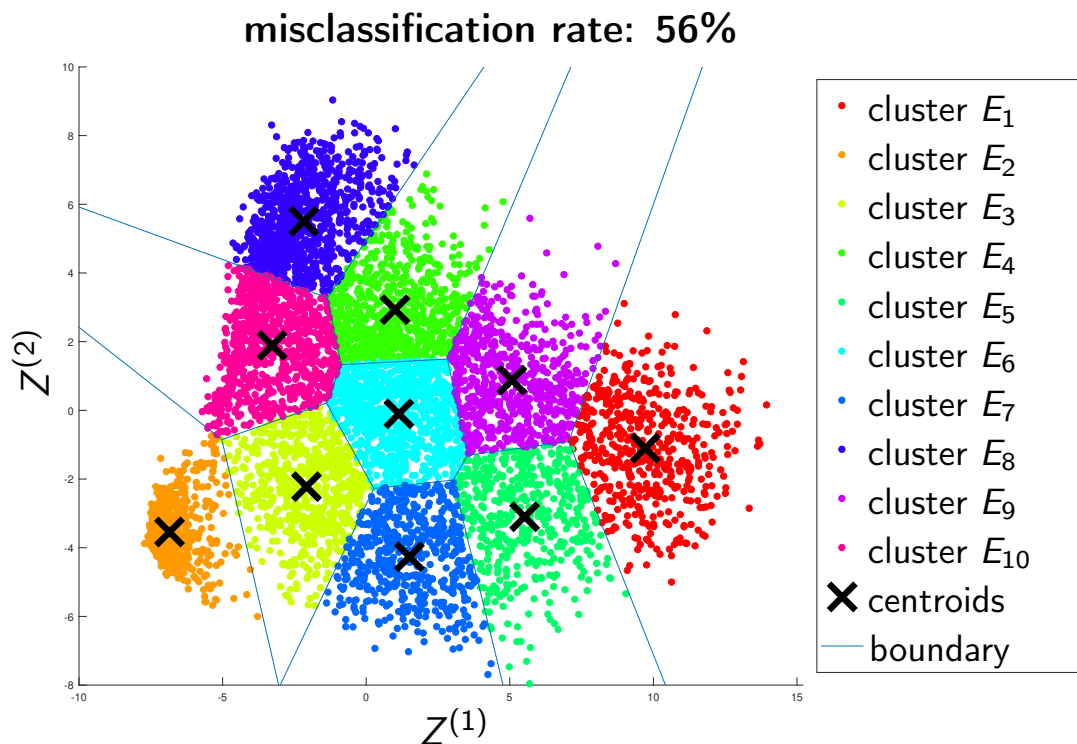
Example: handwritten digits with digits 1, 6 and 9



Example: handwritten digits with all digits



Example: handwritten digits with all digits



46/53

Example: handwritten digits with all digits

	E_1	E_2	E_3	E_4	E_5	E_6	E_7	E_8	E_9	E_{10}	total
"0"	498	0	22	6	260	82	64	0	262	0	1194
"1"	0	1000	4	0	0	0	0	0	0	1	1005
"2"	3	1	234	122	12	202	54	3	60	40	731
"3"	1	0	29	230	4	211	5	5	131	42	658
"4"	0	21	70	112	2	42	3	144	19	239	652
"5"	2	0	61	37	66	171	88	1	119	11	556
"6"	3	6	135	0	128	43	335	0	10	4	664
"7"	0	2	2	49	0	6	0	458	1	127	645
"8"	2	7	82	138	1	93	1	17	41	160	542
"9"	0	10	0	64	0	3	0	303	7	257	644
total	509	1047	639	758	473	853	550	931	650	881	7291

Poor result ➡ need for a better dissimilarity measure !

(and, in particular, for a *better representation*)

47/53

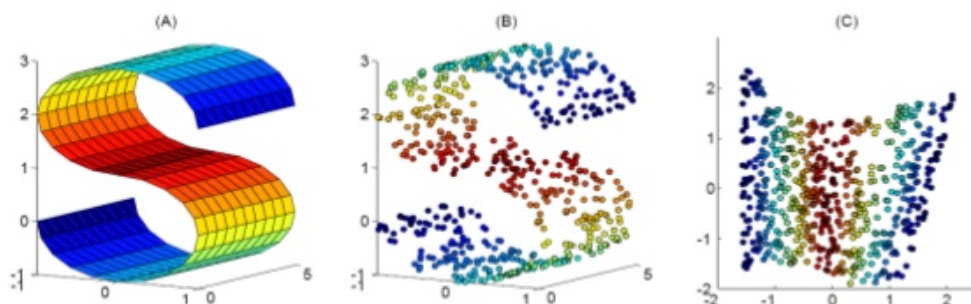
Lecture outline

- 1 – Introduction to unsupervised learning
- 2 – Principal components analysis
- 3 – Clustering
- 4 – A taste of some (more) advanced methods
- 5 – Appendices

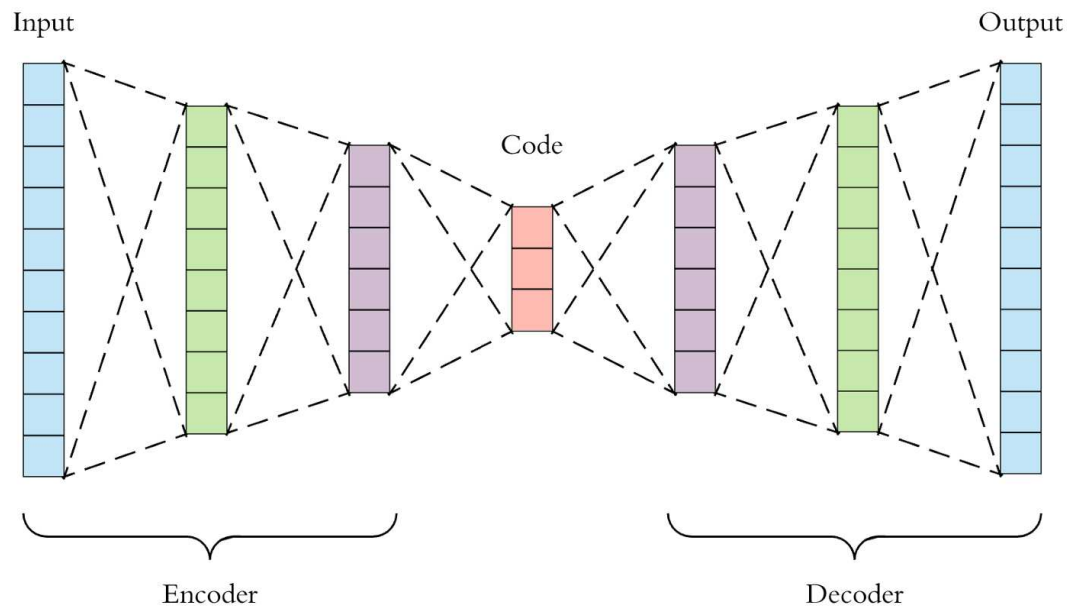
Non-linear dimension reduction

Nonlinear Dimensionality Reduction

- Many data sets contain essential nonlinear structures that invisible to PCA.



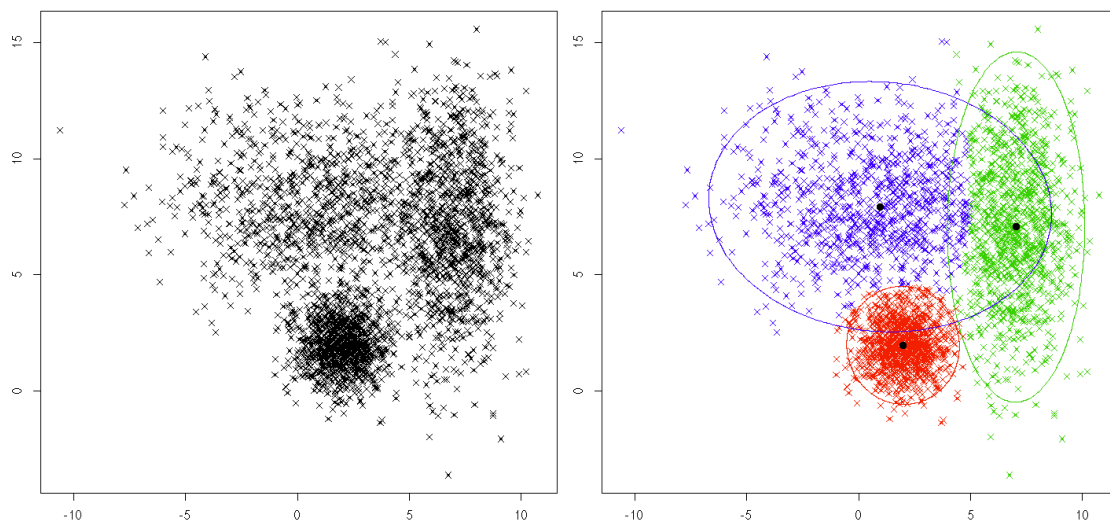
Example: auto-encoder



source: <https://towardsdatascience.com>, Applied Data Deep Learning Part 3

49/53

Clustering based on mixture models



source: bioinfo-fr.net

50/53

Lecture outline

1 – Introduction to unsupervised learning

2 – Principal components analysis

3 – Clustering

4 – A taste of some (more) advanced methods

5 – Appendices

5.1 – Proof of the fundamental theorem of PCA

5.2 – Expressions of T and $W(\Pi)$ for $d_{ii'} = \|x_i - x_{i'}\|^2$

5.3 – Silhouette of a partition

Lecture outline

1 – Introduction to unsupervised learning

2 – Principal components analysis

3 – Clustering

4 – A taste of some (more) advanced methods

5 – Appendices

5.1 – Proof of the fundamental theorem of PCA

5.2 – Expressions of T and $W(\Pi)$ for $d_{ii'} = \|x_i - x_{i'}\|^2$

5.3 – Silhouette of a partition

Proof of the fundamental theorem of PCA

$$\|(\text{Id}_p - AA^\top) X^\top\|_F^2 = \|VD^\top U^\top - AA^\top VD^\top U^\top\|_F^2$$

Properties of the Frobenius norm: if U and V are orthogonal,

$$\|VMU^\top\|_F^2 = \|M\|_F^2.$$

$$\text{Hence : } \|(\text{Id}_p - AA^\top) X^\top\|_F^2 = \|D^\top - V^\top AA^\top VD^\top\|_F^2.$$

Let $\mathcal{M}_{n,p,q}$ denote the set of all rank q matrices of size $n \times p$. Then

$$D_q = \text{diag}(d_1, \dots, d_q, 0, \dots, 0) \in \text{argmin}_{M \in \mathcal{M}_{n,p,q}} \|D^\top - M^\top\|_F^2$$

(diagonal matrix with the q largest singular values).

We obtain the result by checking that $V^\top V_q V_q^\top VD^\top = D_q^\top$. □

[back to slide 16](#)

51/53

Lecture outline

1 – Introduction to unsupervised learning

2 – Principal components analysis

3 – Clustering

4 – A taste of some (more) advanced methods

5 – Appendices

5.1 – Proof of the fundamental theorem of PCA

5.2 – Expressions of T and $W(\Pi)$ for $d_{ii'} = \|x_i - x_{i'}\|^2$

5.3 – Silhouette of a partition

Expressions of T and $W(\Pi)$ for $d_{ii'} = \|x_i - x_{i'}\|^2$

$$\begin{aligned} T &= \frac{1}{2} \sum_{i,i'} \|x_i - x_{i'}\|^2 \\ &= \frac{1}{2} \sum_{i,i'} \|(x_i - \bar{x}) - (x_{i'} - \bar{x})\|^2 \\ &= n \sum_i \|x_i - \bar{x}\|^2 - \sum_{i,i'} (x_i - \bar{x})^\top (x_{i'} - \bar{x}) \\ &= n \sum_i \|x_i - \bar{x}\|^2 \end{aligned}$$

$$\begin{aligned} W(\Pi) &= \frac{1}{2} \sum_k \sum_{i,i' \in \pi_k} \|x_i - x_{i'}\|^2 \\ &= \frac{1}{2} \sum_k \sum_{i,i' \in \pi_k} \|(x_i - \bar{x}_k) - (x_{i'} - \bar{x}_k)\|^2 \\ &= \sum_k n_k \sum_{i \in \pi_k} \|x_i - \bar{x}_k\|^2 \end{aligned}$$

[back to slide 34](#)

52/53

Lecture outline

1 – Introduction to unsupervised learning

2 – Principal components analysis

3 – Clustering

4 – A taste of some (more) advanced methods

5 – Appendices

5.1 – Proof of the fundamental theorem of PCA

5.2 – Expressions of T and $W(\Pi)$ for $d_{ii'} = \|x_i - x_{i'}\|^2$

5.3 – Silhouette of a partition

Silhouette of a partition

Another indicator of the quality of a partition Π .

Let $i \in \pi_k$. For each x_i , define

- ▶ $a(x_i)$: average distance to other points in the same cluster
- ▶ $b(x_i)$: minimum average distance to points in another cluster

$$a(x_i) = \frac{1}{|\pi_k|} \sum_{i' \in \pi_k} \|x_{i'} - x_i\|$$
$$b(x_i) = \min_{k' \neq k} \left(\frac{1}{|\pi_{k'}|} \sum_{i' \in \pi_{k'}} \|x_{i'} - x_i\| \right)$$

Interpretation : $a(x_i) \ll b(x_i)$ if the clusters are homogeneous and well separated.

Silhouette of partition Π

$$s(\Pi) = \frac{1}{n} \sum_{i=1}^n \frac{b(x_i) - a(x_i)}{\max(a(x_i), b(x_i))}$$

Choice of the number K of clusters:

$\forall \Pi, s(\Pi) \leq 1$ and we choose the partition such that $s(\Pi)$ is maximal.