Dimensionality reduction and data visualization

Gilles Gasso

INSA Rouen - ASI Departement
Laboratory LITIS

September 21, 2021

Introduction

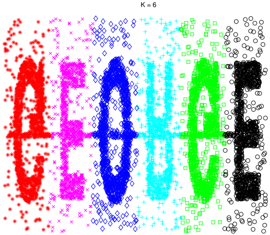
Supervised learning (predictive methods)

- Develop predictive models using labeled training data
- Ensure that the models perform well on future data (test data)

Unsupervised learning (descriptive methods)

- Data exploration
- Analyze distribution/geometry of the data
- Goal: acquire or extract knowledge / patterns from data

→ Dimension reduction, visualization, clustering



Data exploration

d variables

rcc	wcc	hc	hg	ferr	bmi	ssf	pcBfat	lbm	ht	wt	sex
4.82	7.6	43.2	14.4	58	22.37	50	11.64	53.11	163.9	60.1	f
4.32	6.8	40.6									f
5.16	7.2	44.3									f
4.66	6.4	40.9									f
4.19	9	39									f
4.53	5	40.7									f
4.42	6.4	42.8									f
4.32	4.3	41.6									m
4.73	6.7	42.8									m
4.71	7.2	43.6									m
4.93	7.3	46.2	15.1	41	21.12	34	6.39	67	184.4	71.8	m
5.21	7.5	47.5	16.5	20	21.89	46.7	9.5	70	187.3	76.8	m
5.09	8.9	46.3	15.4	44	29.97	71.1	13.97	88	185.1	102.7	m
5.11	9.6	48.2	16.7	103	27.39	65.9	11.66	83	185.5	94.2	m
4.94	6.3	45.7	15.5	50	23.11	34.3	6.43	74	184.9	79	m
4.86	3.9	44.9	15.4	73	22.83	34.5	6.56	70	181	74.8	m
4.51	4.4	41.6	12.7	44	19.44	65.1	15.07	53.42	179.9	62.9	f
4.62	7.3	43.8	14.7	26	21.2	76.8	18.08	61.85	188.7	75.5	f

Data matrix

- sample $x_i = \begin{pmatrix} x_{i,1} & \cdots & x_{i,d} \end{pmatrix}^\top$
- $X = \begin{pmatrix} x_{1,1} & \cdots & x_{1,d} \\ \vdots & & \vdots \\ x_{n,1} & \cdots & x_{n,d} \end{pmatrix} \in \mathbb{R}^{n \times d}$

n points

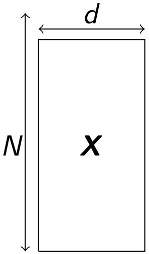
Point x_i

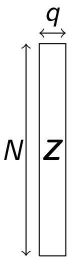
Variable j (hemoglobin)

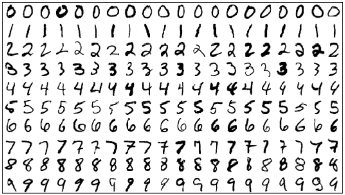
What are the relations between the variables? How close are the samples?

Dimension reduction: the goal

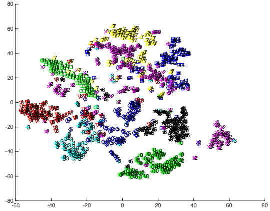
- Let $\boldsymbol{X} \in \mathbb{R}^{N \times D}$ the data (N samples of dimension d)
- Goal: find a projection of $\boldsymbol{X}$ onto $\boldsymbol{Z} \in \mathbb{R}^{N \times q}$ with $q < d$







$d = 784$



$q = 2$

What for?

- Visualization ($q = 2$ ou 3)
    - check the data
    - identify outliers
    - visualize the data according to their categories (if labelled)
- Data representation ($q < d$)
    - Noise reduction
    - pre-processing: computation issue
    - hidden structure in the data (example: manifolds)

Coding/Encoding scheme

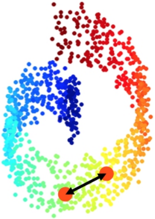
$$\begin{array}{rccccccc} cod: & \mathbb{R}^d & \longrightarrow & \mathbb{R}^q & , & \boldsymbol{x} & \longmapsto & \boldsymbol{z} = cod(\boldsymbol{x}) \\ dec: & \mathbb{R}^q & \longrightarrow & \mathbb{R}^d & , & \boldsymbol{z} & \longmapsto & \boldsymbol{x} = dec(\boldsymbol{z}) \end{array}$$

How to assess the quality of the coding?

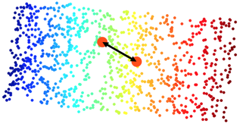
Principle of dimension reduction methods

- Project samples $\{\boldsymbol{x}_i \in \mathbb{R}^d\}_{i=1}^N$ onto $\{\boldsymbol{z}_i \in \mathbb{R}^q\}_{i=1}^N$ $(q < d)$ such that the data topology is preserved
  - preserve distance between samples
  - preserve the neighborhood ...

$\boldsymbol{x}_i \in \mathbb{R}^3$



$\boldsymbol{z}_i \in \mathbb{R}^2$: distance preservation



Methods we will study

Linear : PCA, non-linear : SNE and t-SNE variant

Principal Component Analysis (PCA)

Model: data = information + noise

$$\boldsymbol{X} = \boldsymbol{Z}\boldsymbol{P}^\top + \boldsymbol{B}$$

Linear orthogonal projection:
$$\begin{array}{lcccccc} cod: & \mathbb{R}^d & \longrightarrow & \mathbb{R}^q, & \boldsymbol{x} & \longmapsto & \boldsymbol{z} = \boldsymbol{P}^\top \boldsymbol{x} \\ dec: & \mathbb{R}^q & \longrightarrow & \mathbb{R}^d, & \boldsymbol{z} & \longmapsto & \hat{\boldsymbol{x}} = \boldsymbol{P}\boldsymbol{z} \end{array}$$

Property: columns of $\boldsymbol{P}$ are orthogonal

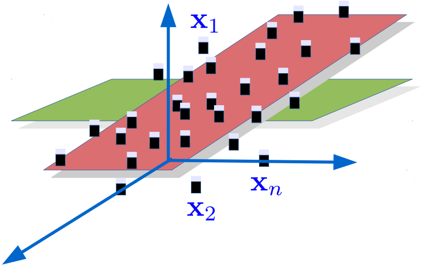
Dimensions: $\boldsymbol{X} = \begin{pmatrix} \boldsymbol{x}_1^\top \\ \vdots \\ \boldsymbol{x}_N^\top \end{pmatrix} \in \mathbb{R}^{N\times d}, \quad \boldsymbol{Z} = \begin{pmatrix} \boldsymbol{z}_1^\top \\ \vdots \\ \boldsymbol{z}_N^\top \end{pmatrix} \in \mathbb{R}^{N\times q}, \quad \boldsymbol{P} \in \mathbb{R}^{d\times q}$

Objective: minimize error between $\boldsymbol{x}_i$ and its estimation $\hat{\boldsymbol{x}}_i = dec(cod(\boldsymbol{x}_i))$

$$\min_{\boldsymbol{P}\in\mathbb{R}^{d\times q}} \sum_{i=1}^{N} \|\boldsymbol{x}_i - \boldsymbol{P}\boldsymbol{P}^\top \boldsymbol{x}_i\|^2$$

Another view of PCA

PCA linearly projects $\{\boldsymbol{x}_i \in \mathbb{R}^d\}_{i=1}^N$ onto a subspace of dimension $q\,(q < d)$ such that the variance of the projections $\{\boldsymbol{z}_i = \boldsymbol{P}^\top \boldsymbol{x}_i \in \mathbb{R}^q\}_{i=1}^N$ remains maximal



Variance maximization (case $q = 1$)

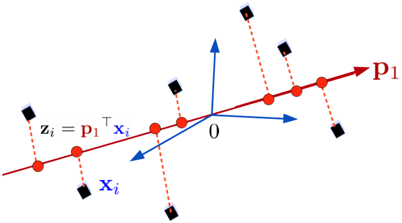
$$\max_{\boldsymbol{p}\in\mathbb{R}^q} \|\boldsymbol{X}\boldsymbol{p}\|_2^2 \quad \text{with} \quad \|\boldsymbol{p}\|_2^2 = 1 \text{ and} \quad \boldsymbol{Z} = \boldsymbol{X}\boldsymbol{p}$$

Minimization of error / maximization of variance

$$\begin{aligned}
J(\boldsymbol{P}) &= \sum_{i=1}^{N} \|\boldsymbol{x}_i - \hat{\boldsymbol{x}}_i\|^2 = \sum_{i=1}^{N} (\boldsymbol{x}_i - \boldsymbol{P}\boldsymbol{P}^\top \boldsymbol{x}_i)^\top (\boldsymbol{x}_i - \boldsymbol{P}\boldsymbol{P}^\top \boldsymbol{x}_i) \\
&= \sum_{i=1}^{N} (\boldsymbol{x}_i^\top \boldsymbol{x}_i - 2\boldsymbol{x}_i^\top \boldsymbol{P}\boldsymbol{P}^\top \boldsymbol{x}_i + \boldsymbol{x}_i^\top \boldsymbol{P}\boldsymbol{P}^\top \boldsymbol{P}\boldsymbol{P}^\top \boldsymbol{x}_i) \\
&= \sum_{i=1}^{N} \boldsymbol{x}_i^\top \boldsymbol{x}_i - \sum_{i=1}^{N} \boldsymbol{x}_i^\top \boldsymbol{P}\boldsymbol{P}^\top \boldsymbol{x}_i = \boldsymbol{x}_i^\top \boldsymbol{x}_i - \sum_{i=1}^{N} \boldsymbol{z}_i^\top \boldsymbol{z}_i \\
&= \mathit{trace}\left(\sum_{i=1}^{N} \boldsymbol{x}_i \boldsymbol{x}_i^\top - \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{z}_i \boldsymbol{z}_i^\top \right) = \mathit{trace}\left(\sum_{i=1}^{N} \boldsymbol{x}_i \boldsymbol{x}_i^\top - \sum_{i=1}^{N} \boldsymbol{P}^\top \boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{P} \right) \\
J(\boldsymbol{P}) &= \mathit{trace}\left(\boldsymbol{X}^\top \boldsymbol{X} \right) - \mathit{trace}\left(\boldsymbol{P}^\top \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{P} \right)
\end{aligned}$$

$\Rightarrow$ $\min J(\boldsymbol{P}) \Leftrightarrow$ maximizing the variance of the projections w.r.t. $\boldsymbol{P}$

First projection vector $\boldsymbol{p}_1$ of $\boldsymbol{P}$



- Data: $\{\boldsymbol{x}_i \in \mathbb{R}^{N\times d}\}_{i=1}^N$
- Assume the $\boldsymbol{x}_i$ are normalized
- Projections onto p_1: $\{\boldsymbol{z}_i = \boldsymbol{p}_1^\top \boldsymbol{x}_i \in \mathbb{R}\}_{i=1}^N$

Computing $\boldsymbol{p}_1 \in \mathbb{R}^d$

$\boldsymbol{p}_1$: a unit vector that maximizes the variance of the $\{\boldsymbol{z}_i\}_{i=1}^N$

$$\max_{\boldsymbol{p}_1 \in \mathbb{R}^d} \frac{1}{N} \sum_{i=1}^N (\boldsymbol{p}_1{}^\top \boldsymbol{x}_i)^2 \quad \text{s.t.} \quad \|\boldsymbol{p}_1\|^2 = 1$$

$\longrightarrow$ Solve a constrained optimization problem

Computing $\boldsymbol{p}_1$

$$\max_{\boldsymbol{p}_1 \in \mathbb{R}^d} \ \boldsymbol{p}_1{}^\top \boldsymbol{C} \boldsymbol{p}_1 \quad \text{s.t.} \quad \|\boldsymbol{p}_1\|^2 = 1$$

$\boldsymbol{C} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i \boldsymbol{x}_i^\top = \frac{1}{N}\boldsymbol{X}^\top \boldsymbol{X}$ is the correlation matrix

Solution derivation

- Lagrangian: $\mathcal{L}(\boldsymbol{p}_1, \lambda_1) = -\boldsymbol{p}_1^\top \boldsymbol{C} p_1 + \lambda_1(\boldsymbol{p}_1^\top \boldsymbol{p}_1 - 1)$
- Optimality conditions :

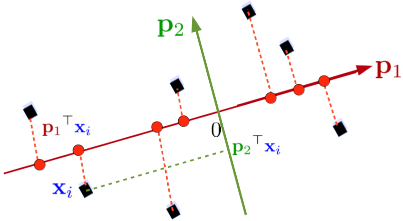
$$\nabla_{\boldsymbol{p}_1}\mathcal{L} = -2\boldsymbol{C}\boldsymbol{p}_1 + 2\lambda_1 \boldsymbol{p}_1 = 0 \quad \text{and} \quad \nabla_{\lambda_1}\mathcal{L} = \boldsymbol{p}_1^\top \boldsymbol{p}_1 - 1 = 0$$

$$\Longrightarrow \boldsymbol{C}\boldsymbol{p}_1 = \lambda_1 \boldsymbol{p}_1 \quad \text{and} \quad \boldsymbol{p}_1^\top \boldsymbol{C}\boldsymbol{p}_1 = \lambda_1$$

1. $(\lambda_1, \boldsymbol{p}_1)$ is the couple (eigenvalue , eigenvector) of the $\boldsymbol{C}$
2. $\boldsymbol{p}_1^\top \boldsymbol{C}\boldsymbol{p}_1 = \lambda_1$ is the objective to maximize

$\boldsymbol{p}_1$ is the eigenvector associated to the highest eigenvalue of $\boldsymbol{C}$.

Computing $\boldsymbol{p}_2$ and beyond


$\mathbf{p}_2$
$\mathbf{p}_1$
$\mathbf{p}_1^\top \mathbf{x}_i$
0
$\mathbf{p}_2^\top \mathbf{x}_i$
$\mathbf{x}_i$

- $\boldsymbol{p}_2$: unit vector orthogonal to $\boldsymbol{p}_1$ that maximizes the variance of the projections $\{\boldsymbol{p}_2^\top \boldsymbol{x}_i\}_{i=1}^N$ onto $\boldsymbol{p}_2$

Solution

- $\boldsymbol{p}_2$ is the eigenvector associated to λ_2, the 2^{nd} highest eigenvalue of $\boldsymbol{C}$

Lemma

The sub-space of size k that maximizes the variance of the projection necessarily includes the sub-space of size $k-1$.

PCA algorithm

1. Normalize the data : $\{\boldsymbol{x}_i \in \mathbb{R}^d\}_{i=1}^N \longrightarrow \{x_{ij} = \frac{x_{ij}-\bar{x}_j}{\sigma_j}, j = 1, d\}_{i=1}^N$
2. Compute the correlation matrix $\boldsymbol{C} = \frac{1}{N}\boldsymbol{X}^\top\boldsymbol{X}$
3. Find the eigenvalue decomposition $\{\boldsymbol{p}_j \in \mathbb{R}^d, \lambda_j \in \mathbb{R}\}_{j=1}^d$ of $\boldsymbol{C}$
4. Order the eigenvalues λ_j by decreasing order
5. The projection matrix is:

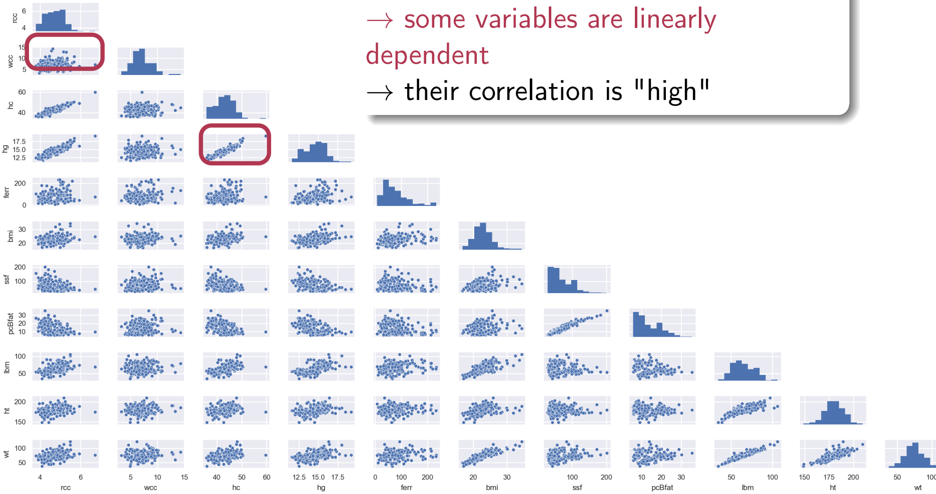
$$\boldsymbol{P} = (\boldsymbol{p}_1, \cdots, p_q) \in \mathbb{R}^{d\times q}$$

$\{\boldsymbol{p}_1, \cdots, \boldsymbol{p}_q\}$ are the q eigenvectors associated to the q highest eigenvalues.

Application

rcc	wcc	hc	hg	ferr	bmi	ssf	pcBfat	lbm	ht	wt	sex
4.82	7.6	43.2	14.4	58	22.37	50	11.64	53.11	163.9	60.1	f
4.32	6.8	40.6	13.7	46	17.54	54.6	12.16	46.12	173	52.5	f
5.16	7.2	44.3	14.5	88	18.29	61.9	12.92	48.76	175	56	f
4.66	6.4	40.9	13.9	109	18.37	38.2	8.45	41.93	157.9	45.8	f
4.19	9	39	13.4	69	18.93	43.5	10.16	42.95	158.9	47.8	f
4.53	5	40.7	14	41	17.79	56.8	12.55	38.3	156.9	43.8	f
4.42	6.4	42.8	14.5	63	20.31	58.9	13.46	39.03	149	45.1	f
4.32	4.3	41.6	14	177	26.73	35.2	6.46	91	190.4	96.9	m
4.73	6.7	42.8	14.9	8	19.81	41.8	7.19	70	195.2	75.5	m
4.71	7.2	43.6	14	32	20.39	30.5	5.63	67	186.6	71	m
4.93	7.3	46.2	15.1	41	21.12	34	6.59	67	184.4	71.8	m
5.21	7.5	47.5	16.5	20	21.89	46.7	9.5	70	187.3	76.8	m
5.09	8.9	46.3	15.4	44	29.97	71.1	13.97	88	185.1	102.7	m
5.11	9.6	48.2	16.7	103	27.39	65.9	11.66	83	185.5	94.2	m
4.94	6.3	45.7	15.5	50	23.11	34.3	6.43	74	184.9	79	m
4.86	3.9	44.9	15.4	73	22.83	34.5	6.56	70	181	74.8	m
4.51	4.4	41.6	12.7	44	19.44	65.1	15.07	53.42	179.9	62.9	f
4.62	7.3	43.8	14.7	26	21.2	76.8	18.08	61.85	188.7	75.5	f

Pair plots of the variables

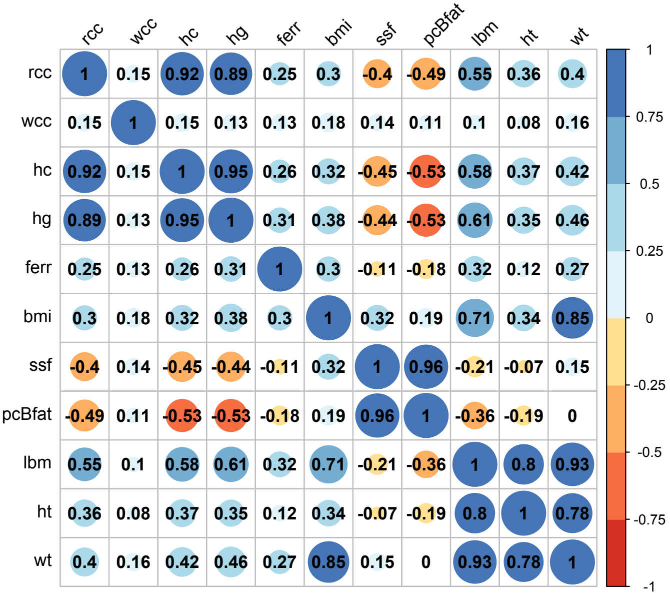


Bivariate representation

→ some variables are linearly dependent

→ their correlation is "high"

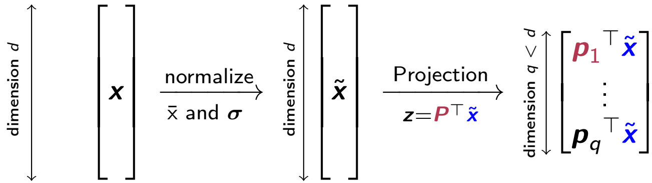
Correlation matrix



	rcc	wcc	hc	hg	ferr	bmi	ssf	pcBfat	lbm	ht	wt
rcc	1	0.15	0.92	0.89	0.25	0.3	-0.4	-0.49	0.55	0.36	0.4
wcc	0.15	1	0.15	0.13	0.13	0.18	0.14	0.11	0.1	0.08	0.16
hc	0.92	0.15	1	0.95	0.26	0.32	-0.45	-0.53	0.58	0.37	0.42
hg	0.89	0.13	0.95	1	0.31	0.38	-0.44	-0.53	0.61	0.35	0.46
ferr	0.25	0.13	0.26	0.31	1	0.3	-0.11	-0.18	0.32	0.12	0.27
bmi	0.3	0.18	0.32	0.38	0.3	1	0.32	0.19	0.71	0.34	0.85
ssf	-0.4	0.14	-0.45	-0.44	-0.11	0.32	1	0.96	-0.21	-0.07	0.15
pcBfat	-0.49	0.11	-0.53	-0.53	-0.18	0.19	0.96	1	-0.36	-0.19	0
lbm	0.55	0.1	0.58	0.61	0.32	0.71	-0.21	-0.36	1	0.8	0.93
ht	0.36	0.08	0.37	0.35	0.12	0.34	-0.07	-0.19	0.8	1	0.78
wt	0.4	0.16	0.42	0.46	0.27	0.85	0.15	0	0.93	0.78	1

Dimension reduction

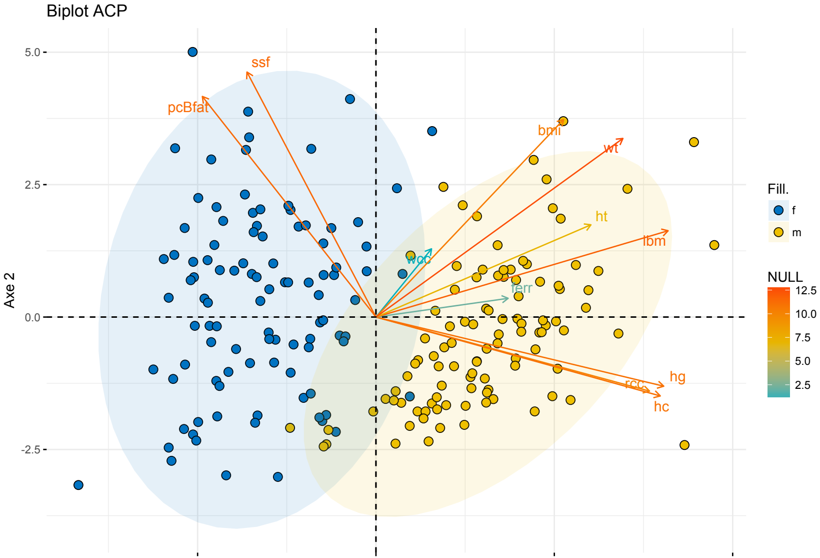
Knowing $\boldsymbol{P} \in \mathbb{R}^{d\times q}$, the mean $\bar{x}$ and std $\boldsymbol{\sigma}$



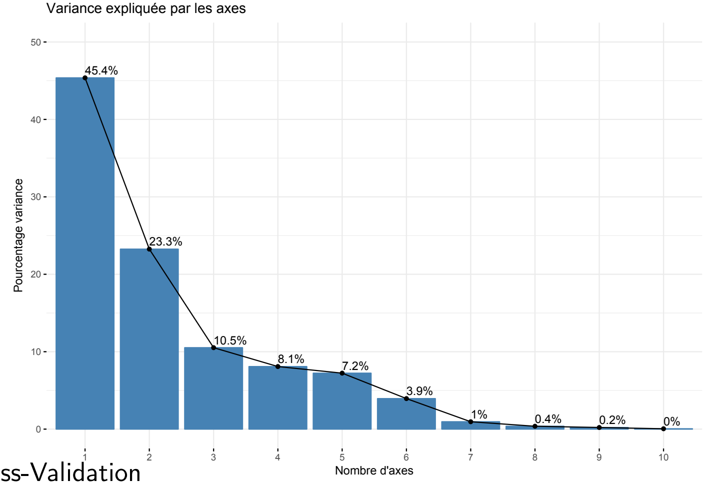
$$\boldsymbol{x} \xrightarrow[\bar{x}\text{ and }\boldsymbol{\sigma}]{\text{normalize}} \tilde{\boldsymbol{x}} \xrightarrow[\boldsymbol{z}=\boldsymbol{P}^\top\tilde{\boldsymbol{x}}]{\text{Projection}} \begin{bmatrix} \boldsymbol{p}_1{}^\top\tilde{\boldsymbol{x}} \\ \vdots \\ \boldsymbol{p}_q{}^\top\tilde{\boldsymbol{x}} \end{bmatrix}$$

⟶ we achieve dimensionality reduction

Data visualization ($q = 2$)



How to choose q?

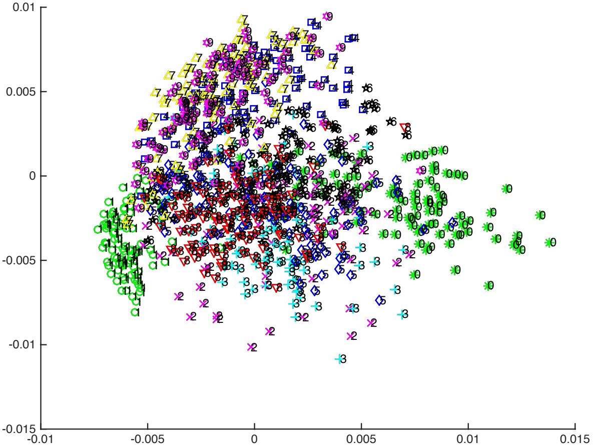


- Cross-Validation
- "Elbow trick" on the graph of eigenvalues
- Set a proportion (for instance 95%) of the recovered variance

Visualizing Mnist dataset

$d = 784$ $q = 2$

$\boldsymbol{Z} =$



Drawbacks of PCA

- Only linear projection
- PCA solely relies on order 2 statistics (mean and variance)

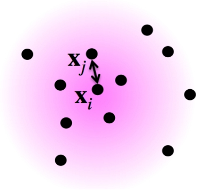
Beyond PCA

- Non-linear PCA
- ISOMAP, LLE, MVU, SNE, t-SNE . . .
- Neural networks
  - auto-encoders
  - embeddings for custom data: word2vec, doc2vec (text), signal2vec (time series)

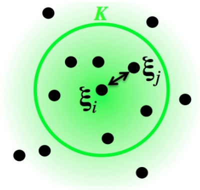
SNE (Stochastic Neighbor Embedding) and t-SNE

Intuition

- Transform pairwise distances $dist(\boldsymbol{x}_i, \boldsymbol{x}_j)$ into probability $\mathbb{P}_X(\boldsymbol{x}_i|\boldsymbol{x}_j)$ (that $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are close)
  - low distance → high probability to be close
- Same for the projections: $dist(\boldsymbol{z}_i, \boldsymbol{z}_j) \rightarrow \mathbb{P}_Z(\boldsymbol{z}_i|\boldsymbol{z}_j)$
- Find $\{\boldsymbol{z}_i\}$ that minimize the distance between the distributions $\mathbb{P}_X$ and $\mathbb{P}_Z$



$dist(\boldsymbol{x}_i, \boldsymbol{x}_j) = \|\mathbf{x}_i - \mathbf{x}_j\|_2$



$\delta_{ij} = \|\boldsymbol{\xi}_i - \boldsymbol{\xi}_j\|_2$

SNE: the maths

for $\{\boldsymbol{x}_i\}_{i=1}^n$

- define the probability that $\boldsymbol{x}_j$ and $\boldsymbol{x}_i$ are close

$$\mathbb{P}_X(\boldsymbol{x}_i|\boldsymbol{x}_j) = \frac{\exp^{-d_{ij}^2}}{\sum_{k=1}^N \sum_{l\neq k} \exp^{-d_{kl}^2}}$$

$$\text{with} \qquad d_{ij} = \frac{dist(\boldsymbol{x}_i, \boldsymbol{x}_j)^2}{2\sigma_i^2}$$

σ_i defines the number of neighbors of sample $\boldsymbol{x}_i$. It is selected such that

$$\log(K) = -\sum_{j=1}^N \mathbb{P}_X(\boldsymbol{x}_i|\boldsymbol{x}_j) \log \mathbb{P}_X(\boldsymbol{x}_i|\boldsymbol{x}_j)$$

for $\{\boldsymbol{y}_i\}_{i=1}^n$ (the unknowns)

- Prob. that $\boldsymbol{z}_j$ is neighbor of $\boldsymbol{z}_i$

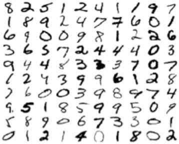
$$\mathbb{P}_Z(\boldsymbol{z}_i|\boldsymbol{z}_j) = \frac{\exp^{-\delta_{ij}^2}}{\sum_{k=1}^N \sum_{l\neq k} \exp^{-\delta_{kl}^2}}$$

$$\text{with} \qquad \delta_{ij} = \|\boldsymbol{z}_i - \boldsymbol{z}_j\|$$

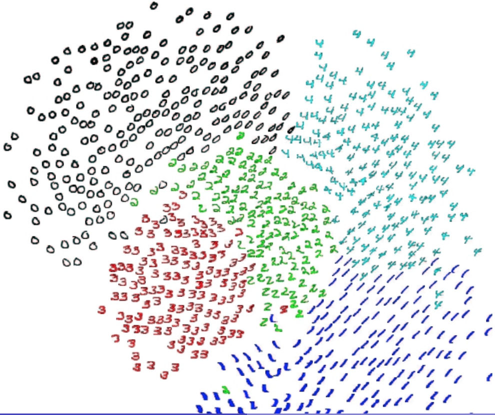
Computing the $\{\boldsymbol{z}_i\}_{i=1}^n$

- Minimize the Kullback-Leibler divergence between $\mathbb{P}_X$ et $\mathbb{P}_Z$
- $\min_{\boldsymbol{z}_1,\ldots,\boldsymbol{z}_N} \sum_{i,j=1}^N \mathbb{P}_X(\boldsymbol{x}_i|\boldsymbol{x}_j) \log \frac{\mathbb{P}_X(\boldsymbol{x}_i|\boldsymbol{x}_j)}{\mathbb{P}_Z(\boldsymbol{z}_i|\boldsymbol{z}_j)}$
- Solution via numerical methods

Illustration of SNE

$\boldsymbol{X} =$ 

$\boldsymbol{Z} =$



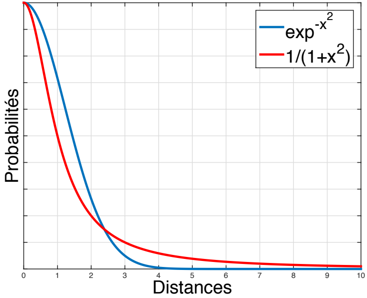
t-SNE variant

SNE

Prob. that $\boldsymbol{z}_j$ is neighbor of $\boldsymbol{z}_i$

$$\mathbb{P}_Z(\boldsymbol{z}_i|\boldsymbol{z}_j) = \frac{\exp^{-\delta_{ij}^2}}{\sum_{k=1}^N \sum_{l \neq k} \exp^{-\delta_{lk}^2}}$$

with $\delta_{ij} = \|\boldsymbol{z}_i - \boldsymbol{z}_j\|$



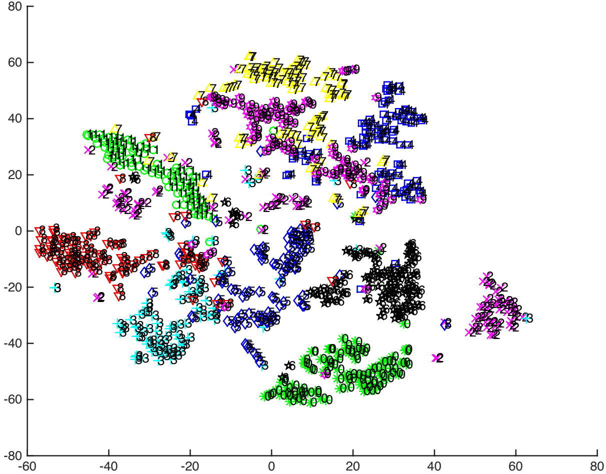
t-SNE

Prob. that $\boldsymbol{z}_j$ is neighbor of $\boldsymbol{z}_i$

$$\mathbb{P}_Z(\boldsymbol{z}_i|\boldsymbol{z}_j) = \frac{\left(1+\delta_{ij}^2\right)^{-1}}{\sum_{k=1}^N \sum_{l \neq k} \left(1+\delta_{lk}^2\right)^{-1}}$$

- $\mathbb{P}_x = \mathbb{P}_Z$ large $\Rightarrow \delta_Z < d_X$ (attraction)
- $\mathbb{P}_x = \mathbb{P}_Z$ low $\Rightarrow \delta_Z > d_X$ (repulsion)

Illustration of t-SNE



https://lvdmaaten.github.io/tsne/

Conclusions

- PCA: linear dimensionality reduction method
- Several non-linear methods (t-SNE, UMAP, auto-encoder . . .)
- They involve advanced optimization methods
- Useful for data visualization and dimension reduction
- Some toolboxes
  - Matlab : `https://lvdmaaten.github.io/drtoolbox/`
  - Python : `http://scikit-learn.org/stable/modules/manifold.html#manifold`
  - Graphical tool : `http://divvy.ucsd.edu/`