Unsupervised learning:

overview and selected topics

B. Michel

Ecole Centrale de Nantes

Statistical Learning

1 Introduction to Clustering and *k*-means

- *k*-means Algorithm
- Others clustering methods
- Quality of a clustering and number of clusters

2 Dimension reduction

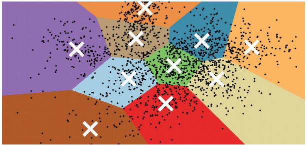
- Introduction to dimension reduction
- Principal Component Analysis
- Non linear dimension reduction

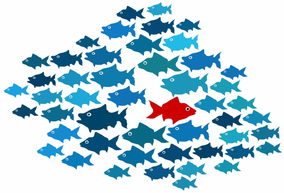
Unsupervised learning

- Unsupervised learning is learning from **unlabeled data**.

- Methods in this field study the intrinsic and hidden structure of the data in order to get meaningful insights, segment the datasets in similar groups or to simplify them.

- Main topics of unsupervised learning :
  - Clustering
  - Anomaly detection
  - Dimension reduction, auto-encoders
  - Matrix factorization
  - Manifold Learning
  - ...





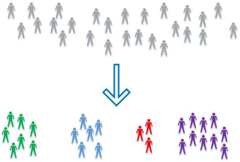
Summary

1. **Introduction to Clustering and *k*-means**
   - *k*-means Algorithm
   - Others clustering methods
   - Quality of a clustering and number of clusters

2. **Dimension reduction**
   - Introduction to dimension reduction
   - Principal Component Analysis
   - Non linear dimension reduction

Clustering

- Clustering refers to the process of partitioning a set of objects into subsets consisting of similar objects.

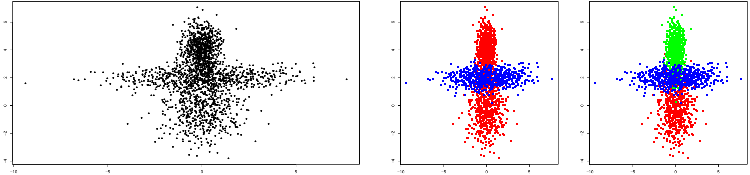


- Examples :
  - Biology : are there sub-species in a population ? For cancer studies : are there different clones ?
  - Commercial : what kinds of customers do I have ?
  - Text mining : find "similar or different" texts in a corpus.
  - Other suggestions ?

- Other types of application :
  - Image segmentation
  - Compression / quantification

An universal way to find group ?

- There is no universal way to find groups in data :



- There are many clustering methods that correspond to different ways of grouping observations into classes : geometric, probabilistic points of view, etc.
- Choosing a clustering method requires defining the notion of “class” and then a criterion to be optimized on the observations.
- Unlike supervised classification, it is more difficult to evaluate and compare the results of clustering methods (no observable truth).

Examples of distances and normalization of numeric variables

Many distances in $\mathbb{R}^p$ can be defined by a quadratic form :

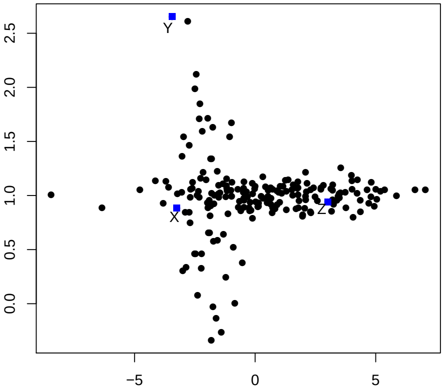
$$d^2(x_i, x_\ell) = (x_i - x_\ell)' M (x_i - x_\ell)$$

- Usual Euclidean norm : $M = I_p$

Clustering methods (and ML in general) often require normalization (or standardization) of data, so that all variables are taken into account equally in the clustering criterion used. For instance :

- $M = \text{diag}\left(\frac{1}{\sigma_1^2}, \dots, \frac{1}{\sigma_p^2}\right)$ where $\sigma_j^2 = S_{dd}$ and where S is the variance-covariance matrix of the data.
- Mahalanobis distance : $M = S^{-1}$.

Example for the Mahalanobis distance



$\|X - Y\|_2 = 1.78$ $\|X - Y\|_{2,S^{-1}} = 24.94$

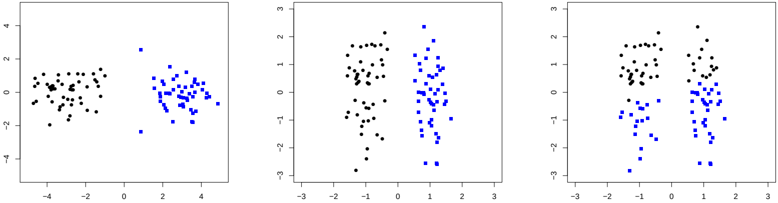
$\|X - Z\|_2 = 6.25$ $\|X - Z\|_{2,S^{-1}} = 5.92$

Normalization has a strong impact on the distances between observations.

Clustering is challenging

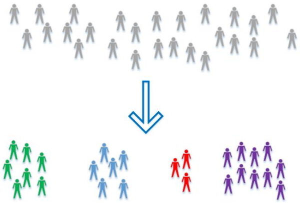
Many elements have an impact on the result of the clustering :

- The variables included in the study (and the possible repetition of certain variables),
- The metric used to compare the observations,
- The number of classes chosen,
- The selected clustering method
- The data preprocessing : standardization ?



Conclusion : data expertise is needed on the data for applying clustering methods (more than for supervised problems).

Clustering for modelling the data ?



- For some applications, the clusters really exist : tumor clones (cancer)
- But clustering is used more than it should be, an underlying domain based on discrete classes **does not always exist !** Example : clustering of costumers.
- Clustering helps understanding the structure of the data but we should keep in mind that the underlying data is **usually continuous**.
- Alternative : matrix factorization (NMF for instance).

Summary

1. **Introduction to Clustering and *k*-means**
   - *k*-means Algorithm
   - Others clustering methods
   - Quality of a clustering and number of clusters

2. **Dimension reduction**
   - Introduction to dimension reduction
   - Principal Component Analysis
   - Non linear dimension reduction

Clustering and quantification

- Let $\mathscr{X}$ be a set and d a distance on $\mathscr{X}$.
- Let $\{x_1, \ldots, x_n\} \subset \mathscr{X}$ be a sample for which we want to propose a clustering.
- Let $\{c_1, \ldots, c_K\}$ be a set of points of $\mathscr{X}$, called **codebook** which is supposed to summarize $\{x_1, \ldots, x_n\}$.
- For $r > 0$, we define the "cost" of quantizing the data $\{x_1, \ldots, x_n\}$ by $\{c_1, \ldots, c_K\}$:

$$\Phi(\{c_1, \ldots, c_K\}) = \sum_{i=1}^{n} d^r\left(x_i, \{c_1, \ldots, c_K\}\right)$$

  where $d(x, A) = \min_{a \in A} d(x, a)$.
- Objective of k-means type algorithms is finding a codebook (or a data partitioning) which minimizes Φ.
- This is not a standard convex optimization problem.

Space of partitions

- Number of partitions of a set of n individuals in K classes (Stirling number of 2nd species)

$$\frac{1}{K!}\sum_{j=0}^{K}(-1)^j (K-j)^n C_K^j$$

$\simeq 10^{47}$ partitions of $n = 100$ individuals in $K = 3$ classes
$\simeq 10^{68}$ partitions of $n = 100$ individuals in $K = 5$ classes

- Exhaustive search is not possible.

- k-means algorithms only provide approximate solutions to the optimum.

Inertia

- Hereafter we consider the Euclidean case : observations belongs to $\mathbb{R}^p$.
- Let $\mathcal{C} = \{\mathscr{C}_1, \ldots, \mathscr{C}_K\}$ be a partition of the data $\{x_1, \ldots, x_n\}$ into K clusters.
- We consider the cost of clustering

$$\Phi(\mathcal{C}) = \Phi\left(\{m_1, \ldots, m_K\}\right) = \sum_{i=1}^{n} \|x_i - \{m_1, \ldots, m_K\}\|^2$$

where $m_k = \frac{1}{|\mathscr{C}_k|} \sum_{i \in \mathscr{C}_k} x_i$ is the barycentre of cluster $\mathscr{C}_k$ and

$$\|x_i - \{m_1, \ldots, m_K\}\| := \inf_{k=1\ldots K} \|x_i - m_k\|.$$

Inertia

$$\Phi(\mathcal{C}) = \Phi(\{m_1, \ldots, m_K\}) = \sum_{i=1}^{n} \|x_i - \{m_1, \ldots, m_K\}\|^2$$

- Total Inertia : $I_{total} = \frac{1}{n} \sum_{i=1}^{n} \|x_i - \bar{x}\|^2$

  where $\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$ is the barycentre of the point cloud.

- Between-group Inertia : $I_{inter} = \frac{1}{n} \sum_{k=1}^{K} |\mathscr{C}_k| \times \|m_k - \bar{x}\|^2$

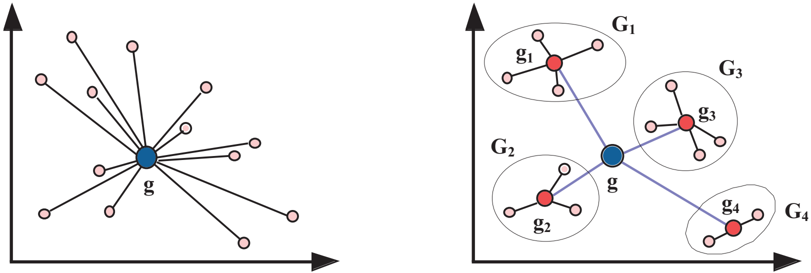
  $\Rrightarrow$ I_{inter} measures the dispersion of the K centers (barycentres)

- Within-group Inertia : $I_{intra} = \frac{1}{n} \sum_{k=1}^{K} \sum_{i \in \mathscr{C}_k} \|x_i - m_k\|^2 = \frac{1}{n}\Phi(\mathcal{C})$

  $\Rrightarrow$ I_{intra} measures the dispersion of the points inside a cluster

Inertia : Huygens Property

$$I_{total} = I_{inter} + I_{intra}$$



[Source : Bisson 2001]

Minimizing Within-group Inertia $\iff$ Maximizing Between-group Inertia

K-means Algorithm (aka Lloyd Algorithm)

```
**Data:** Number of clusters $K$, data set $\mathbf{X}$, number of iterations $T$.
**Result:** clustering $\mathcal{C} = \{\mathscr{C}_1 \dots \mathscr{C}_K\}$.
Initialization : choose $K$ initial centers $c_1, \dots, c_K$ ;
$t = 0$;
while $\textit{clustering is not stabilized or } t < T$ do
    t = t+1 ;
    for $x \in \mathbf{X}$ do
        for $k = 1 \textit{ to } K$ do
            Compute all distances to centers $\|x - c_k\|$;
        end
        Assign $x$ to the cluster for which $\|x - c_k\|$ is minimal ;
    end
    for $k = 1 \textit{ to } K$ do
        Update center in $\mathscr{C}_k$ : $c_k = m_k = \frac{1}{|\mathscr{C}_k|} \sum_{i \in \mathscr{C}_k} x_i$ ;
    end
end
```

Animation

Within-group Inertia is decreasing

Proposition :

The Within-group Inertia $I_{\text{intra}}(\mathcal{P}^{(t)})$ decreases at each step and thus convergences to a local minimum.

Sketch of Proof :

- If i moves from cluster $\mathscr{C}_k^{(t-1)}$ to cluster $\mathscr{C}_{k'}^{(t)}$ then

$$\left\| x_i - m_{k'}^{(t-1)} \right\|^2 \leq \left\| x_i - m_k^{(t-1)} \right\|^2$$

- $m_k^{(t)}$ being the center of $\mathscr{C}_k^{(t)}$, then

$$\sum_{i \in \mathscr{C}_k^{(t)}} \left\| x_i - m_k^{(t)} \right\|^2 \leq \sum_{i \in \mathscr{C}_k^{(t)}} \left| x_i - m_k^{(t-1)} \right\|^2$$

Properties of k-means algorithm

- Relatively efficient : Complexity $O(KnT)$ where T is the number of iterations.
- The Within-group inertia decreases with the iterations of the algorithm.
- But convergence to a local minimum.
- Need to specify the number of classes K.
- Discovers compact, convex and well-separated classes.
- Influence of the choice of initial kernels
- Can produce empty classes.
- Influence of outliers

Choice of initial kernels

- "Furthest Point" Strategy : NO !
- Selection based on expert knowledge.
- Preliminary study of univariate data (histograms, ...).
- Dimension reduction then initialization on a more robust structure (k-means on the first PCA components for example).
- Repetition of the method N times and selection of the classification with the lowest within-group inertia.
- k-means $++$: see further.

k-means ++

k-means ++ : a clever Initialisation of k-means.

```
Data: number of classes $K$, data set $\mathbf{X}$
Result: clustering $\mathcal{C} = \{\mathscr{C}_1 \dots \mathscr{C}_K\}$.
Choose a first center $c$ uniformly among the $x_i$ ;
Initialization of the family of centers $C = \{c\}$ ;
for $k = 2$ to $K$ do
    for $i \in \{1 \dots n\}$ do
        Calculate $d(x_i, C)$ the distance to the nearest center ;
    end
    Choose $\tilde{c}$ in the $x_i$ according to the proba $\frac{d(x_i,C)^2}{\sum_{x \in \mathbf{X}} d(x,C)^2}$ ;
    $C = C \cup \{\tilde{c}\}$ ;
end
Perform k-means with this initialization;
```

k-means ++

Theorem (Arthur, D. and Vassilvitskii, S. (2007))

Let $\mathcal{C}^{++}$ be the partition of a sample $(x_1, \ldots, x_n)$ provided by k-means++ for K classes. Then :

$$\mathbb{E}\Phi(\mathcal{C}^{++}) \leq 8(\log K + 2) \inf_{|\mathcal{C}|=K} \Phi(\mathcal{C})$$

where the infimum is taken over all partitions of size K of the sample.

Mini batch k-means

k-means for very large samples and many clusters.

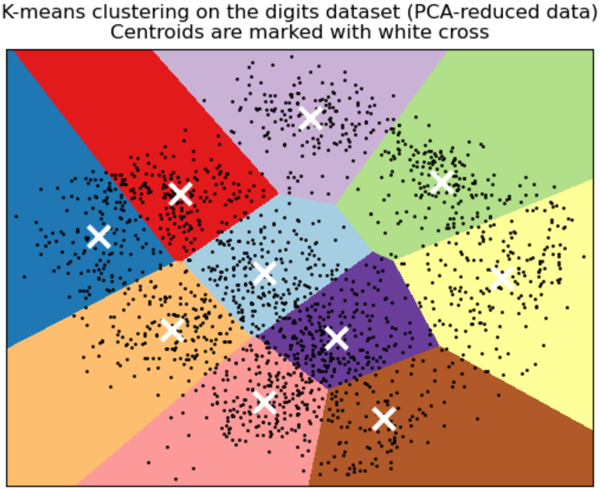
Principle : current step of k-means performed on a small subsample.

```
Data: number of classes $K$, data set $\mathbf{X}$, max iterations $T$.
Result: clustering $\mathcal{C} = \{\mathscr{C}_1 \dots \mathscr{C}_K\}$.
Initialize the centers $c_k$ by choosing them uniformly into the $x_i$;
$v = (0, \dots, 0)$ vector of zeros of length $K$;
for $t = 1$ to $T$ do
    $\tilde{\mathbf{X}}$ : subsample of size $b$ drawn uniformly in the $x_i$;
    for $x \in \tilde{\mathbf{X}}$ do
        $k_{c_x}$ : index of the nearest center $c_x$ of $x$;
    end
    for $x \in \tilde{\mathbf{X}}$ do
        Update center count $v[k_{c_x}] = v[k_{c_x}] + 1$;
        Update learning rate $\eta = 1/v[k_{c_x}]$;
        Update center $c_x = (1 - \eta)c_x + \eta x$ ;
    end
end
```

Summary

1. **Introduction to Clustering and *k*-means**
   - *k*-means Algorithm
   - Others clustering methods
   - Quality of a clustering and number of clusters
2. **Dimension reduction**
   - Introduction to dimension reduction
   - Principal Component Analysis
   - Non linear dimension reduction

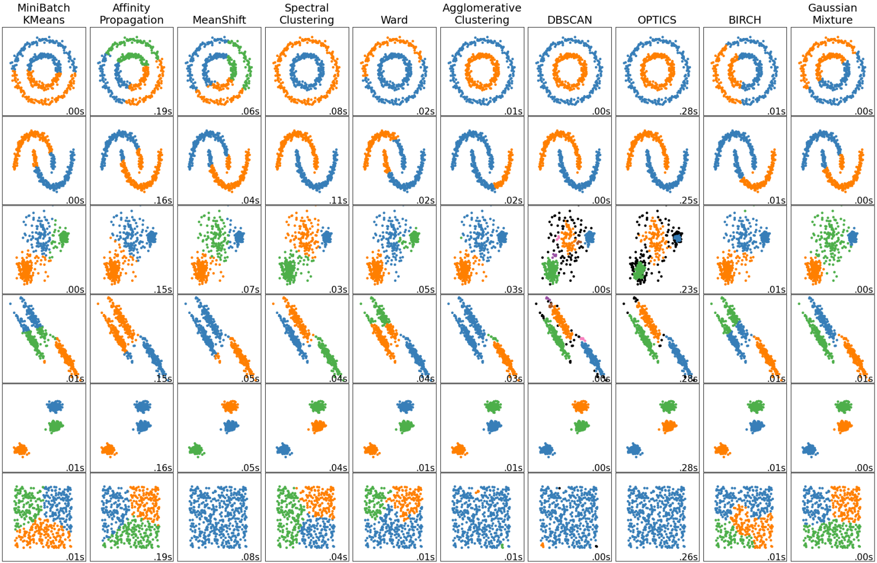
Limitation : Clusters of k-means are convex regions



[Source : scikit-learn.org]

K-means cannot separate non-convex clusters

Many alternatives methods for clustering



[Source : scikit-learn.org]

Summary

1. **Introduction to Clustering and k-means**
   - k-means Algorithm
   - Others clustering methods
   - Quality of a clustering and number of clusters

2. **Dimension reduction**
   - Introduction to dimension reduction
   - Principal Component Analysis
   - Non linear dimension reduction

Quality of a clustering

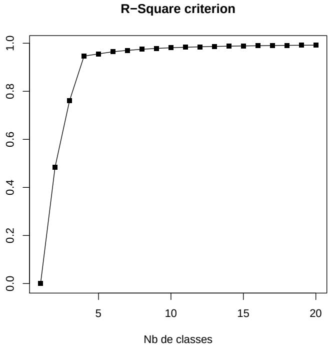
- We can evaluate the quality of a clustering by internal methods.
- These criteria can be used to choose a number of classes.
- Many possible approaches :
  - Elbow rule.
  - Silhouette criterion
  - Statistical Gap

Elbow method

- For each value of $K \in \{2, \ldots, K_{\max}\}$, we find a clustering on the data.
- R-Square :

$$K \mapsto RSQ(K) = 1 - \frac{I_{within}(\mathcal{P}_K)}{I_{total}} = \frac{I_{between}(\mathcal{P}_K)}{I_{total}}$$

- Find where the curve $K \mapsto \mathrm{RSQ}(K)$ stabilizes :

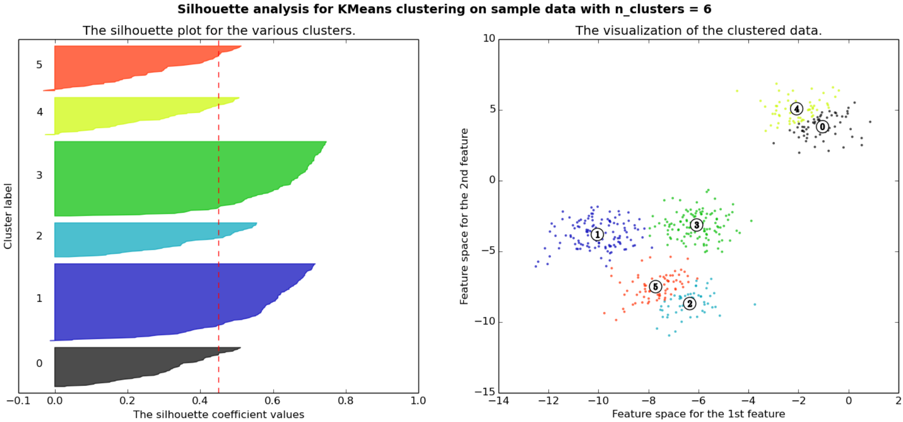


Silhouette

- $\mathcal{C}(K) = \{\mathscr{C}_1, \dots, \mathscr{C}_K\}$ a clustering of $\mathbf{X}$ into K classes.
- For $i \in \{1, \dots, n\}$ let $\mathscr{C}_k(i)$ be the class of i.
- $a(i) = \frac{1}{|\mathcal{C}_k|-1} \sum_{\substack{\ell \in \mathscr{C}_k(i) \\ \ell \neq i}} d(x_i, x_\ell)$: mean distance between i and the other observations of the cluster.
- $b(i) = \min_{k' \neq k} \frac{1}{|\mathcal{C}_{k'}(i)|} \sum_{\ell \in \mathcal{C}_{k'}} d(x_i, x_\ell)$: mean distance between i and the observations of the closest cluster to i.
- $s(i) = \frac{b(i)-a(i)}{\max(b(i), a(i))} \in [-1, 1]$
- $S(\mathcal{C}(K)) = \frac{1}{n} \sum_{i=1 \dots n} s(i)$
- For a collection of clusterings $\mathcal{C}(2), \dots, \mathcal{C}(K_{\max})$, the selected number of clusters is

$$\hat{K} = \underset{1 \leq K \leq K_{\max}}{\operatorname{argmax}} S(\mathcal{C}(K))$$

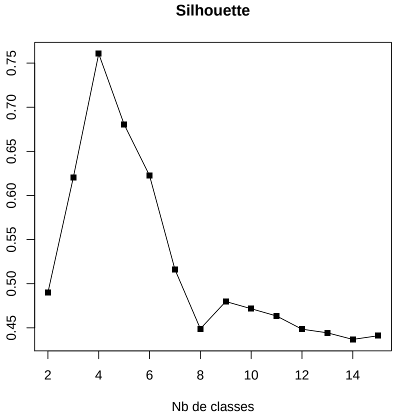
Example

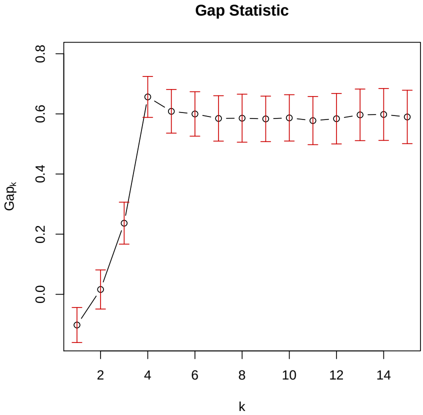

Silhouette analysis for KMeans clustering on sample data with n_clusters = 6
The silhouette plot for the various clusters.
Cluster label
5
4
3
2
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values
The visualization of the clustered data.
Feature space for the 2nd feature
10
5
0
−5
−10
−15
−14
−12
−10
−8
−6
−4
−2
0
2
Feature space for the 1st feature
4
0
1
3
5
2

Statistical Gap

- For every K in $\{1, \dots, K_{\max}\}$:
  - Compute the within-group covariance matrix $W_K = \frac{1}{n} \sum_{k=1}^{K} \sum_{i \in \mathscr{C}_k} (x_i - m_k)'(x_i - m_k)$ and its determinant $\det(W_K)$,
  - Build curve $K \mapsto \log(det(W_K))$,
  - Compare this curve to the one obtained from the uniformly distributed data.
- Choose the number of clusters K corresponding to the greatest difference between the two curves (”gap”)

Silhouette et Gap Statistique





Summary

1. **Introduction to Clustering and *k*-means**
   - *k*-means Algorithm
   - Others clustering methods
   - Quality of a clustering and number of clusters

2. **Dimension reduction**
   - Introduction to dimension reduction
   - Principal Component Analysis
   - Non linear dimension reduction

Summary

1. **Introduction to Clustering and *k*-means**
    - *k*-means Algorithm
    - Others clustering methods
    - Quality of a clustering and number of clusters

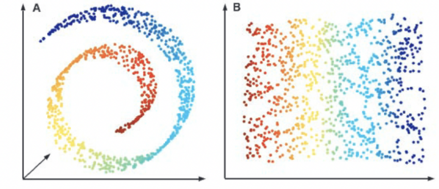
2. **Dimension reduction**
    - Introduction to dimension reduction
    - Principal Component Analysis
    - Non linear dimension reduction

Why dimension reduction ?

- **Visualization.** Beyond three axes, it is difficult to represent the structure of a cloud of points.
- **Extraction or creation of "features"** that best summarize the information.
- **Statistical efficiency.** Many methods in statistics are inefficient in high dimension.
  Ex : estimation of a density by histograms.
- **Computational burden.** The computational complexity of learning algorithms depends on the dimension of the data.
- The last two points both result from **the curse of dimensionality** : when the dimensionality increases, the volume of the space increases so fast that the available data become sparse.

Manifold Assumption

- High-dimensional data is usually concentrated on or near a lower-dimensional structure $M \subset \mathbb{R}^D$ (submanifold).
- We try to "embed" the data in a lower dimensional space while preserving the geometry of the data as much as possible :



$\Phi : \mathbb{R}^D \to \mathbb{R}^d$

[Source : Isomap]

- The dimension reduction map Φ is generally learned from the data.
- **Linear dimension reduction** means that Φ is linear.

Multivariate statistics

Let $\mathbf{y}$ and $\mathbf{z}$ in $\mathbb{R}^n$.

Vector $\mathbf{e}_n = (1, \ldots, 1)' \in \mathbb{R}^n$.

- Means of $\mathbf{y}$: $\bar{y} = \frac{1}{n}\sum_{i=1\ldots n} y_i = \frac{1}{n}\mathbf{y}'\mathbf{e}_n$.
- Covariance of $\mathbf{y}$ and $\mathbf{z}$:

$$\begin{aligned}\text{cov}(\mathbf{y},\mathbf{z}) &= \frac{1}{n}\sum_{i=1\ldots n}(y_i - \bar{y})(z_i - \bar{z}) \\ &= \frac{1}{n}\left(\mathbf{y} - \bar{y}\mathbf{e}_n\right)'\left(\mathbf{z} - \bar{z}\mathbf{e}_n\right).\end{aligned}$$

- Variance of $\mathbf{y}$:

$$\begin{aligned}\text{var}(\mathbf{y}) &= \frac{1}{n}\sum_{i=1\ldots n}(y_i - \bar{y})^2 \\ &= \frac{1}{n}\left(\mathbf{y} - \bar{y}\mathbf{e}_n\right)'\left(\mathbf{y} - \bar{y}\mathbf{e}_n\right).\end{aligned}$$

Multivariate statistics

$$\mathbf{X} = \begin{bmatrix} x_1^1 & \dots & x_1^p \\ \vdots & \vdots & \vdots \\ x_n^1 & \dots & x_n^p \end{bmatrix}$$

- Variables $\mathbf{x}^1, \dots, \mathbf{x}^j, \dots, \mathbf{x}^p$ (colomns)
- Observations $\mathbf{x}_1, \dots, \mathbf{x}_i, \dots, \mathbf{x}_n$ (raws)

- Barycenter of the point cloud : $\bar{\mathbf{x}} = (\bar{x}^1, \dots, \bar{x}^p)' \in \mathbb{R}^p$.
- Covariance matrix of the variables :

$$\begin{aligned} \mathbf{S} := \left[\mathrm{cov}(\mathbf{x}^j, \mathbf{x}^k)\right]_{1 \le j,k \le p} &= \frac{1}{n}\left(\mathbf{X} - \mathbf{e}_n\bar{\mathbf{x}}'\right)'\left(\mathbf{X} - \mathbf{e}_n\bar{\mathbf{x}}'\right) \\ &= \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})'. \end{aligned}$$

- Inertia of the point cloud in $\mathbb{R}^p$:

$$\mathcal{I}(\mathbf{X}) := \frac{1}{n}\sum_{i=1\dots n}\|\mathbf{x}_i - \bar{\mathbf{x}}\|^2 = \frac{1}{n}\,\mathrm{tr}\left[\left(\mathbf{X} - \mathbf{e}_n\bar{\mathbf{x}}'\right)\left(\mathbf{X} - \mathbf{e}_n\bar{\mathbf{x}}'\right)'\right] = \mathrm{tr}(S)$$

For centered data :

$$\bar{\mathbf{x}} = (0, \dots, 0)' \quad , \quad \mathbf{S} = \frac{1}{n}\mathbf{X}'\mathbf{X} \quad \text{et } \mathcal{I}(\mathbf{X}) = \frac{1}{n}\mathrm{tr}(\mathbf{X}'\mathbf{X}).$$

Summary

1. **Introduction to Clustering and *k*-means**
   - *k*-means Algorithm
   - Others clustering methods
   - Quality of a clustering and number of clusters

2. **Dimension reduction**
   - Introduction to dimension reduction
   - Principal Component Analysis
   - Non linear dimension reduction

Introduction

Context : a dataset $\mathbf{X}$ of size $n \times p$ corresponding to n observations described by p variables :

$$\begin{bmatrix} x_1^1 & \dots & x_1^p \\ \vdots & \vdots & \vdots \\ x_n^1 & \dots & x_n^p \end{bmatrix}$$

- Observations $\mathbf{x}_1,\ldots,\mathbf{x}_i,\ldots,\mathbf{x}_n$ in rows,
- **Quantitative continuous** variables $\mathbf{x}^1,\ldots,\mathbf{x}^j,\ldots,\mathbf{x}^p$ in columns.
- The data matrix $\mathbf{X}$ is assumed to be centered : $\bar{\mathbf{x}} = 0$.
- **PCA :** provides :
  - An "optimal" representation of observations in a subspace of dimension q of $\mathbb{R}^p$
  - New variables - called principal components- summarizing in an "optimal way" the information contained in the initial set of variables.

Example : Basketball Dataset

The dataset provides the results of four basketball teams during the 2012-2013 regular season. For each of the 69 players :

- Height
- Width
- Age
- Salary
- Team
- Position
- nb of Games played
- Minutes played
- Fields goals made
- Fields goal attempted
- ...

Fitting a linear subspace on $\mathcal{N}_{\text{ind}}$

- we are looking for the linear subspace E_q of dim q in $\mathbb{R}^p$ that best fits on $\mathbf{X}$.
- By Pythagoras : $\|\mathbf{x}_i\|^2 = \|\mathbf{x}_i - \mathbf{P}_{E_q}(\mathbf{x}_i)\|^2 + \|\mathbf{P}_{E_q}(\mathbf{x}_i)\|^2$
- $\mathcal{I}(\mathbf{X}) := \frac{1}{n}\sum_{i=1}^{n}\|\mathbf{x}_i\|^2$. Thus

$$\text{Minimize } \frac{1}{n}\sum_{i=1\dots n}\|\mathbf{x}_i - \mathbf{P}_{E_q}(\mathbf{x}_i)\|^2 \Longleftrightarrow \text{Maximize } \mathcal{I}\left(\mathbf{P}_{E_q}(\mathbf{X})\right)$$

- Singular value decomposition of $\mathbf{X} := \sum_{s=1}\mu_s \mathbf{v}^s \mathbf{u}'_s$.
- According to SVD, it can be checked that $\mathcal{I}\left(\mathbf{P}_{E_q}(\mathbf{X})\right)$ is maximum for

$$\hat{E}_q := \text{Vect}(\mathbf{u}_1, \dots, \mathbf{u}_q).$$

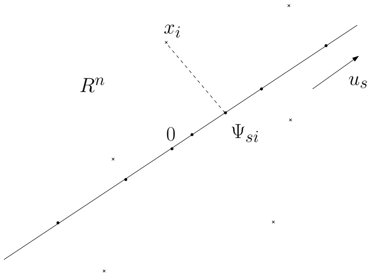
- Equivalently : fitting a linear subspace $\hat{E}_q$ on $\mathbf{X}$ is optimal for $\hat{E}_q$.

Principal Components

- For $s = 1 \dots p$, the **principal component** $\boldsymbol{\Psi}^s = (\Psi_1^s, \dots, \Psi_n^s)$ is the vector :

$$\boldsymbol{\Psi}^s = \begin{bmatrix} < \mathbf{x}_1, \mathbf{u}_s > \\ \vdots \\ < \mathbf{x}_n, \mathbf{u}_s > \end{bmatrix} = \mathbf{X}\mathbf{u}_s$$

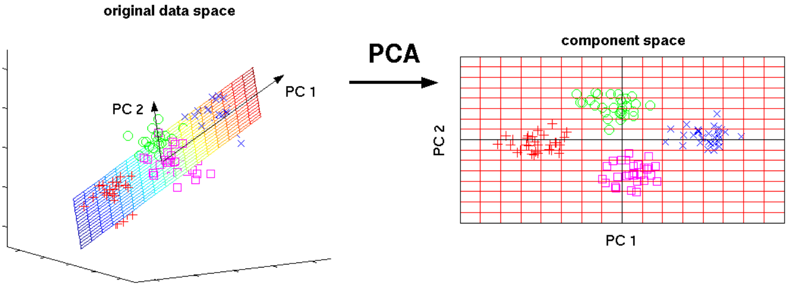
- The vector $\boldsymbol{\Psi}_s \in \mathbb{R}^n$ can be considered as a new variable. It "measures" each observation in the direction $\mathbf{u}_s$.



- The principal components are orthogonal (as vectors of $\mathbb{R}^n$).

PCA : find directions of highest variance

- SVD : $\mathbf{X} := \sum_{s=1} \mu_s \mathbf{v}^s \mathbf{u}_s'$.
- We have seen that PCA corresponds to projecting data on $\hat{E}_q := \mathrm{Vect}(\mathbf{u}_1, \ldots, \mathbf{u}_q)$..
- A little of algebra gives that the $\mathbf{u}_s$ are the eigenvectors of the covariance matrix S.
- The solution of PCA makes sense : we project the data into the directions of largest variance.

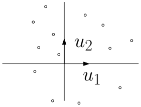


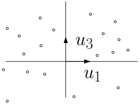
PCA in a nutshell

- Starting with a raw dataset $\mathbf{R} = (r_{ij})$ of size $n \times p$. There are two types of **principal component analysis** (PCA) :
  - **non-normalized PCA** : centered array analysis $\mathbf{X} = \mathbf{R} - \bar{\mathbf{r}}$;
  - **the normalized PCA** : analysis of the reduced centered tableau $\mathbf{X} = \left(\frac{r_i^j - \bar{r}^j}{\sqrt{\text{var}(\mathbf{r}^j)}} \right)_{i,j}$.
- In both cases, PCA consists in diagonalizing the variance-covariance matrix $\mathbf{S} = \frac{1}{n}\mathbf{X}'\mathbf{X}$.
- PCA makes it possible to explore the data, it is also a preliminary step to many statistical methods such as regression or data classification.

Representation of individuals

We analyze the cloud of individuals projected on the first factorial directions :





We can define indicators to measure :

- the overall quality of the representation of the projected cloud,
- the quality of the representation of an individual,
- the contribution of an individual to the total inertia of the cloud,
- the contribution of an individual to the variance of a principal component.

All these indicators allow in particular

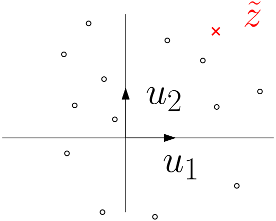
- to discuss the quality of PCA,
- to analyze and explain by individuals the geometry of the cloud,
- to detect the most influential observations on the principal components,
- to identify possible outliers (which we can then remove data to proceed to a new PCA).

Representation of additional observations

We call *additional observation* any observation that has not been taken into account in the calculation of the PCA (i.e. for the diagonalization of $\mathbf{S}$).

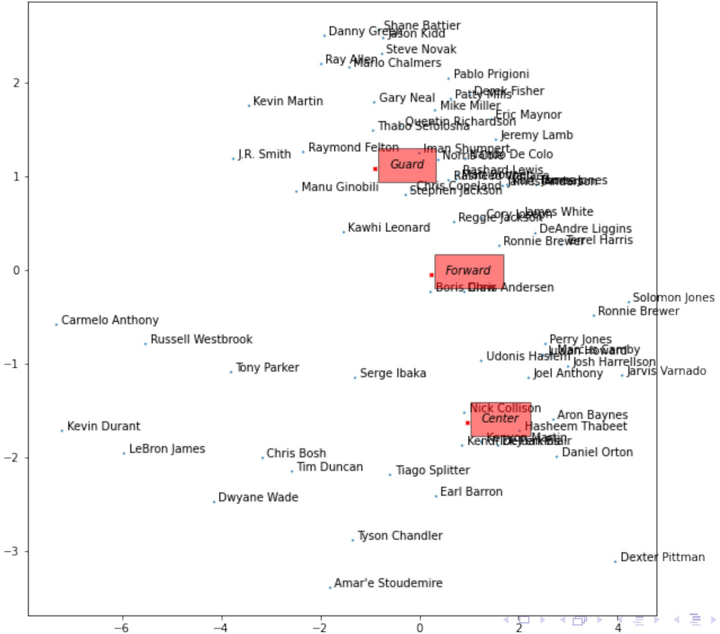
We can represent additional observations by projecting them into the factorial planes. Let $\mathbf{z}$ be such an individual, we recenter (and reduce if normalized PCA) $\mathbf{z}$: $\tilde{\mathbf{z}} = \mathbf{z} - \bar{\mathbf{x}}$. The coordinates of the projection of $\tilde{\mathbf{z}}$ on each of the factorial axes are given by :

$$\Psi_z^s := \mathbf{u}_s' \tilde{\mathbf{z}}.$$

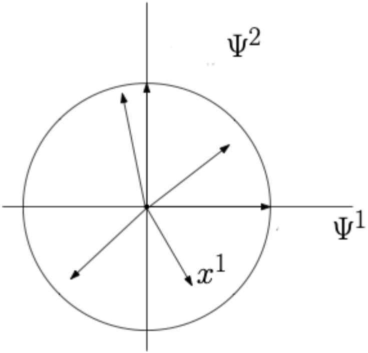


You can also give the quality of the representation of $\tilde{\mathbf{z}}$.

Basketball Dataset : PC1 and PC2



Correlation circle

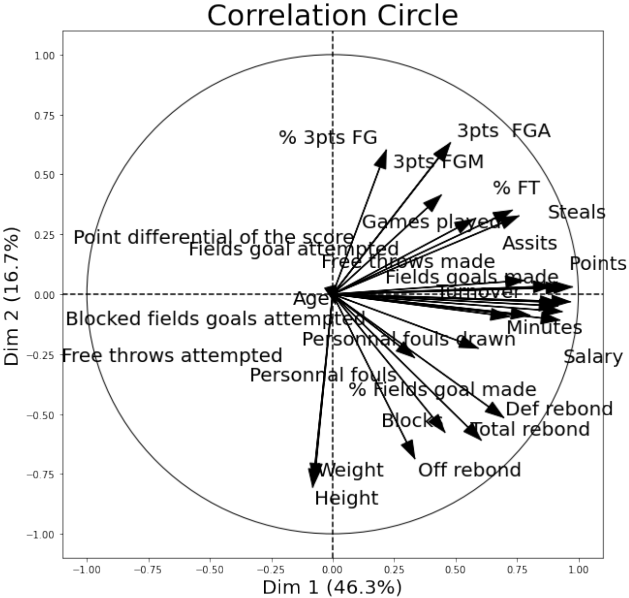


- The **correlation circle** visualizes the correlation between
  - two PC (first and second for instance)
  - the variables

- The variable $\mathbf{x}^j$ is represented by the vector pointing

$$\left(\mathrm{cor}(\mathbf{x}^j, \Psi^1), \mathrm{cor}(\mathbf{x}^j, \Psi^2)\right)$$

Basketball Dataset



How to choose the number of principal components

Several methods and heuristics :

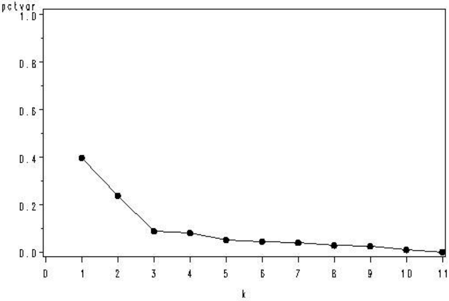
- Proportion of imposed inertia :

$$r_q := \frac{\mathcal{I}\left(\mathbf{P}_{\hat{E}_q}(\mathbf{X})\right)}{\mathcal{I}(\mathbf{X})} = \frac{\lambda_1 + \cdots + \lambda_q}{\lambda_1 + \cdots + \lambda_p} > \text{threshold ?}$$

- Kaiser's rule (normalized PCA) : keep only the eigenvalues larger than the average :

$$q = \max\{\ell \mid \ \lambda_\ell \geq \frac{1}{p} \sum_{s=1} \lambda_s\}$$

- Elbow criterion

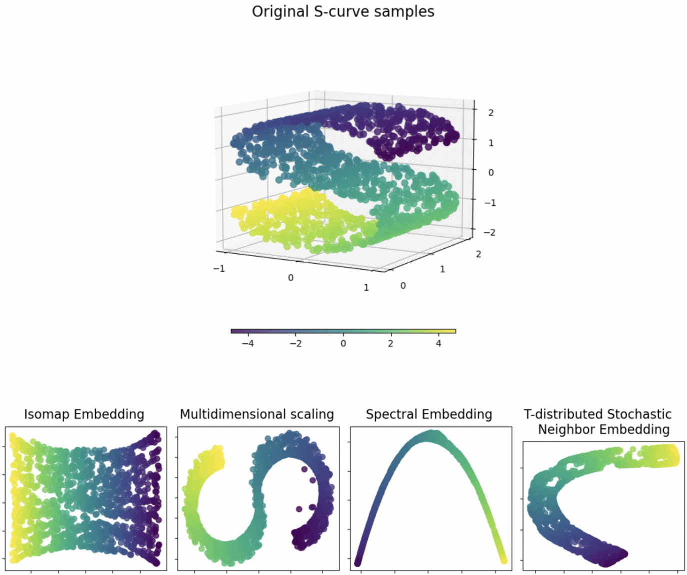


Summary

1. **Introduction to Clustering and *k*-means**
   - *k*-means Algorithm
   - Others clustering methods
   - Quality of a clustering and number of clusters

2. **Dimension reduction**
   - Introduction to dimension reduction
   - Principal Component Analysis
   - Non linear dimension reduction

Non linear dimension reduction



Original S-curve samples

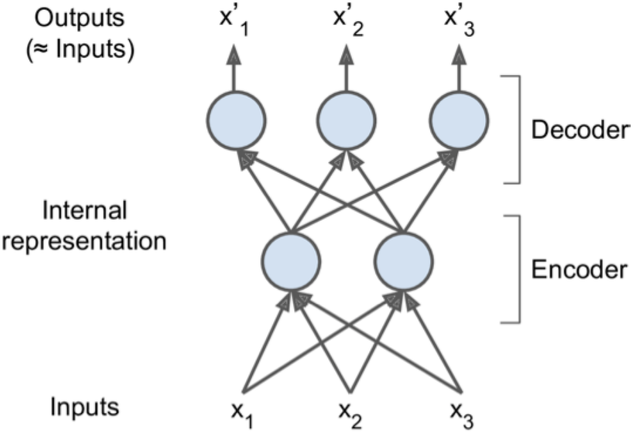
Isomap Embedding

Multidimensional scaling

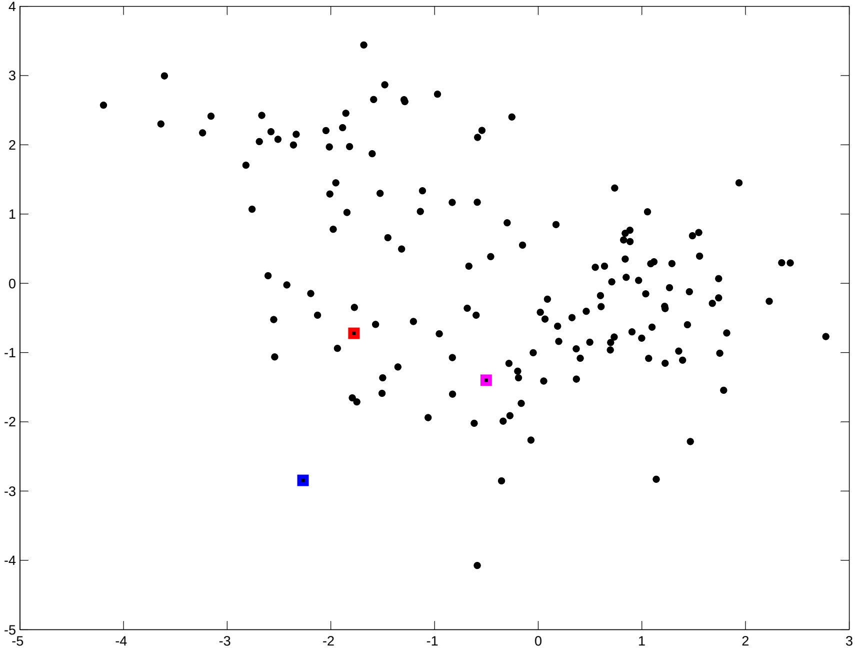
Spectral Embedding

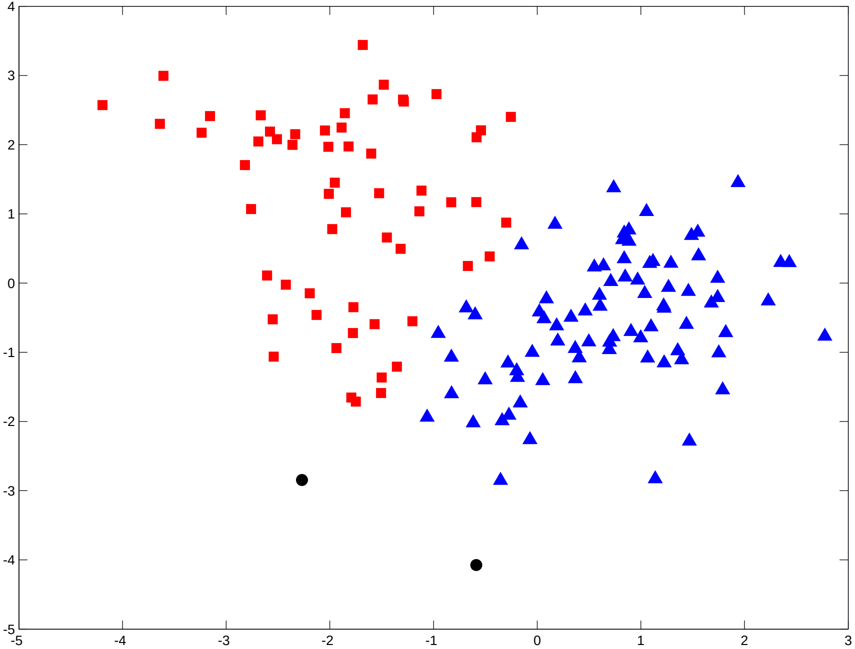
T-distributed Stochastic Neighbor Embedding

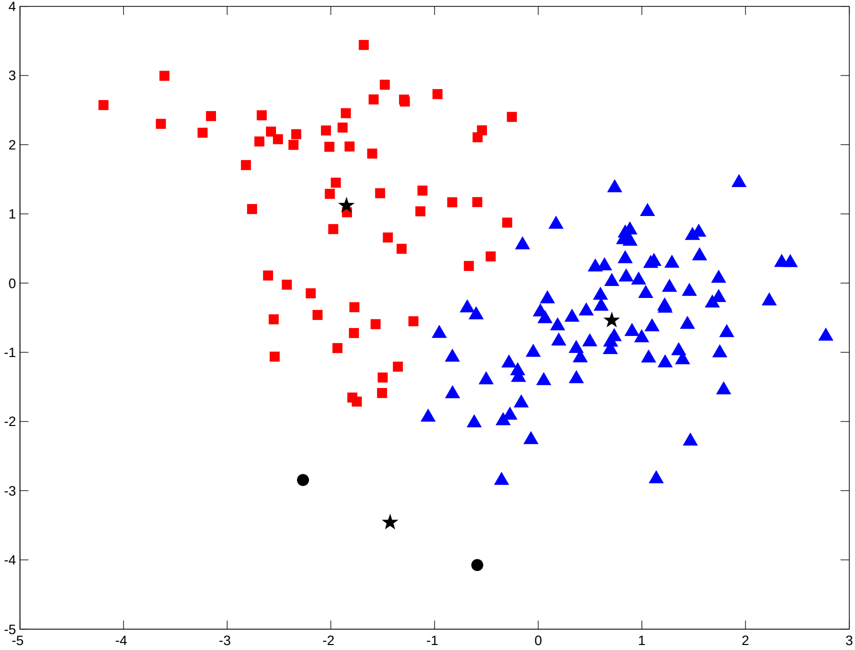
Non linear dimension reduction : autoencoder

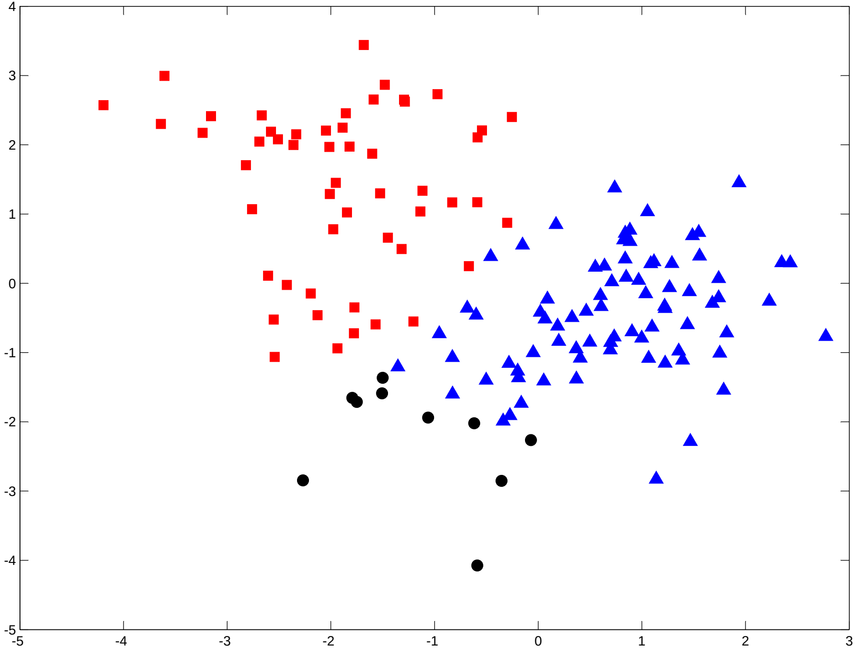


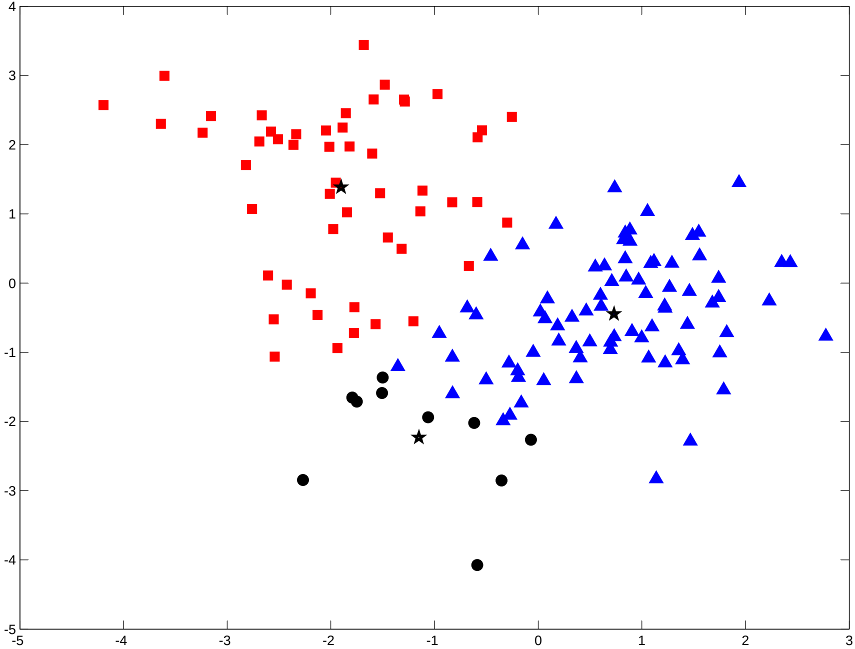
Appendices

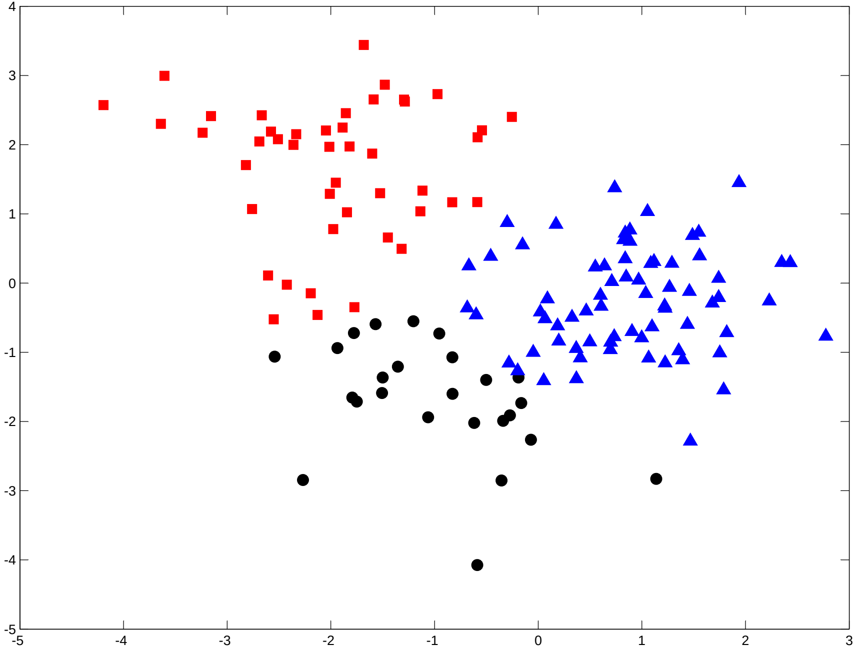

4
3
2
1
0
-1
-2
-3
-4
-5
-5
-4
-3
-2
-1
0
1
2
3

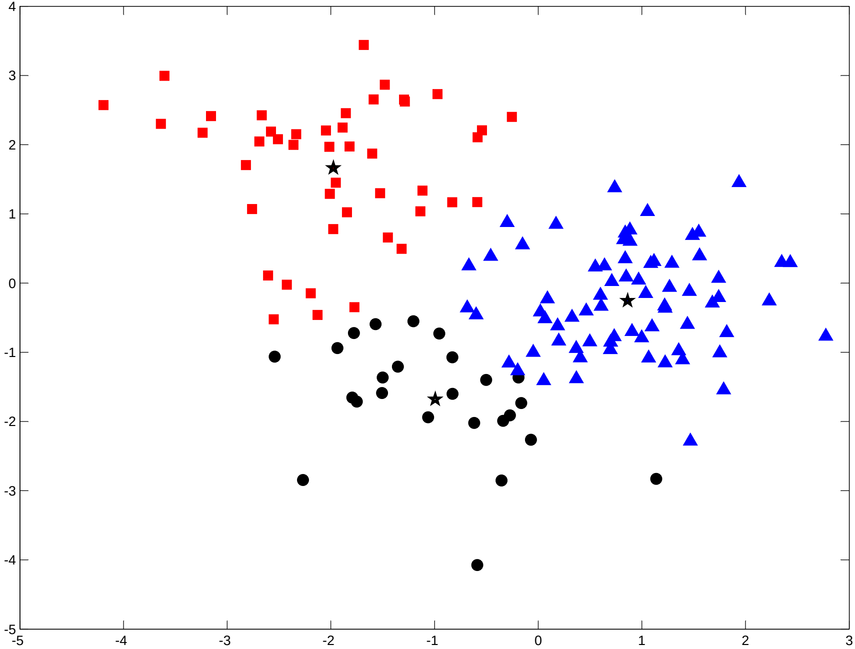

4
3
2
1
0
-1
-2
-3
-4
-5
-5
-4
-3
-2
-1
0
1
2
3

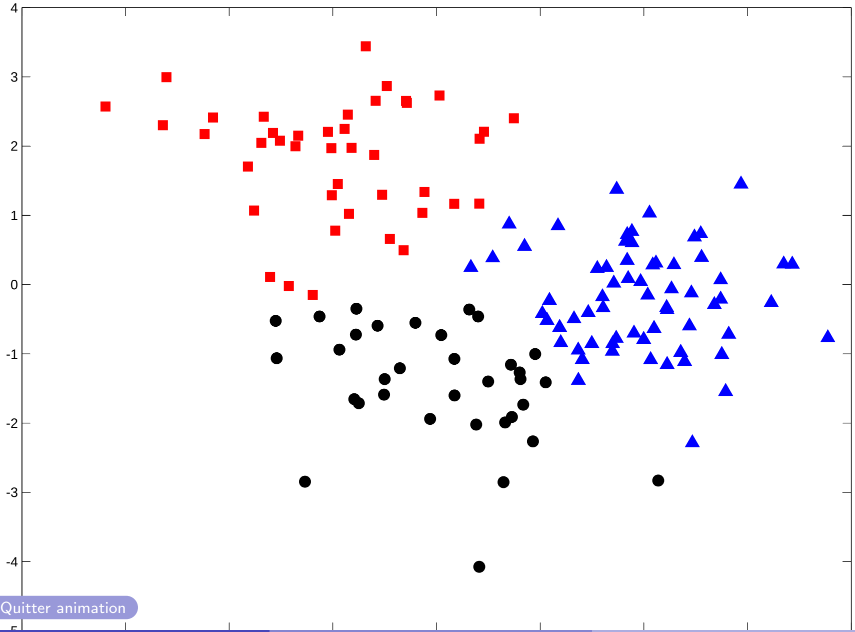

4
3
2
1
0
-1
-2
-3
-4
-5
-5
-4
-3
-2
-1
0
1
2
3


4
3
2
1
0
-1
-2
-3
-4
-5
-5
-4
-3
-2
-1
0
1
2
3


4
3
2
1
0
-1
-2
-3
-4
-5
-5
-4
-3
-2
-1
0
1
2
3


4
3
2
1
0
-1
-2
-3
-4
-5
-5
-4
-3
-2
-1
0
1
2
3


4
3
2
1
0
-1
-2
-3
-4
-5
-5
-4
-3
-2
-1
0
1
2
3



◂ Quitter animation