PDEEC – Machine Learning 2018/19

Lecture - Unsupervised learning

Jaime S. Cardoso
jaime.cardoso@inesctec.pt

INESC TEC and Faculdade Engenharia, Universidade do Porto

Dec. 06, 2018

Empirical Learning

Supervised learning

Supervised learning is concerned with predicting the values of one or more outputs or response variables Y for a given set of input or predictor variables X.

- the predictions are based on the training sample $(\mathbf{x}_i, y_i)$ of previously solved cases
- if (X, Y) are random variables with joint probability density $P(X, Y)$, then supervised learning can be characterized as a density estimation problem where one is concerned with determining properties of the conditional density $P(Y|X)$.
- distint approaches to solving the decision problem:
  1. generative models: first solve the inference problem of determining the joint density $p(x, y)$ (or $p(x, \mathcal{C}_k)$). Then normalize to obtain the posterior probabilities $p(y|x)$. Finally the conditional mean (or class membership) for each new input x.
  2. first solve the inference problem of determining the conditional density $p(y|x)$ and then subsequently calculate the conditional mean
  3. find a regression (or a discriminant) function directly from the training data.

Empirical Learning

Unsupervised learning

Unsupervised learning is concerned with inferring properties of the probability density $p(x)$ without the help of a supervisor or teacher providing correct answers or a degree-of-error for each observation. The goal is to characterize x-values, or collections of such values, where $p(x)$ is relatively large.

- principal components, multidimensional scaling, self-organizing maps, principal curves, etc, attempt to identify low-dimensional manifolds within the x-space that represent high data density.
- **cluster analysis** attempts to find multiple (convex) regions of the x-space that contain modes of $p(x)$.
- **association rules** attempt to construct simple descriptions (conjunctive rules) that describe regions of high density in the special case of very high dimensional binary-valued data.

Empirical Learning

Unsupervised learning

Association Rules

- Association rules learning is another key unsupervised learning method, after clustering, that finds interesting associations (relationships, dependencies) in large sets of data items: $\{onions, potatoes\} \Rightarrow \{beef\}$
- the goal is to find values of the variables $X = (X_1, X_2, \ldots, X_d)$ that appear most frequently in the database
- it is most often applied to binary-valued data, being referred as "market basket" analysis:
  How does demographic information affect what the customer buys?
  Is bread usually bought together with milk?
  Does a specific milk brand make any difference?
  Where should tomatoes be placed to maximize sales?
  Is bread bought also when both milk and eggs are purchased?

Empirical Learning

Unsupervised learning

Association Rules

- Apriori algorithm: as is common in association rule mining, given a set of itemsets (for instance, sets of retail transactions, each listing individual items purchased), the algorithm attempts to find subsets which are common to at least a minimum number C (the cutoff, or confidence threshold) of the itemsets. Apriori uses a "bottom up" approach, where frequent subsets are extended one item at a time (a step known as candidate generation), and groups of candidates are tested against the data. The algorithm terminates when no further successful extensions are found.

Empirical Learning

Unsupervised learning

Cluster Analysis

- grouping or segmenting a collection of objects into subsets ot clusters, such that those within each cluster are more closely related to one another that objects assigned to different clusters
  - high intra-class similarity
  - low inter-class similarity
- an object can be described by a set of measurements, or by its relation to other objects
- central to all of the goals of cluster analysis is the notion of the degree of similarity (dissimilarity) between the individual objects being clustered.

Unsupervised learning

What is Similarity?

Hard to define!
But *we know it*
when we see it

- The real meaning of similarity is a philosophical question. We will take a more pragmatic approach
- Depends on representation and algorithm. For many rep./alg., easier to think in terms of a distance (rather than similarity) between vectors.

Unsupervised learning

(Dis)similarity measures

Fundamental to all clustering techniques is the choice of proximity (alikeness or affinity) or dissimilarity (difference or lack of affinity) between pairs of objects: $\gamma : E \times E \rightarrow R$

- Similarity indices
- Dissimilarity indices

Unsupervised learning

Dissimilarity indices

- A function $d : E \times E \to R_0^+$ is a dissimilarity index if
  1. $d(x, x) = 0$
  2. $d(x_1, x_2) = d(x_2, x_1)$
- A function $d : E \times E \to R_0^+$ is a distance index if, besides 1) and 2), it has the following property:
  3. $d(x_1, x_2) = 0 \Rightarrow x_1 = x_2$
- A function $d : E \times E \to R_0^+$ is a distance function or metric if, besides 1), 2) and 3), it has the following property:
  4. $d(x_1, x_2) \leq d(x_1, x_3) + d(x_3, x_2), \forall x_1, x_2, x_3 \in E$
- A function $d : E \times E \to R_0^+$ is a ultrametric if, besides 1), 2) and 3), it has the following property:
  5. $d(x_1, x_2) \leq \max\{d(x_1, x_3), d(x_3, x_2)\}, \forall x_1, x_2, x_3 \in E$

Unsupervised learning

Intuitions behind desirable distance measure properties

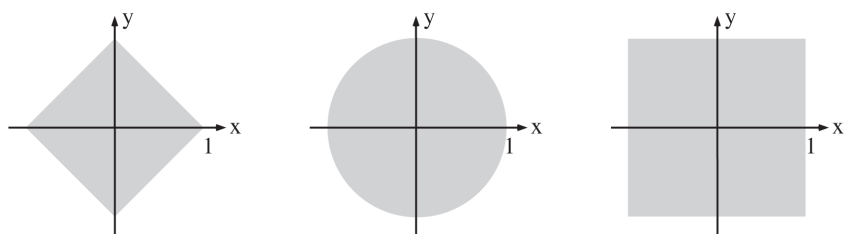
- $D(A,B) = D(B,A)$ *Symmetry*
  - *Otherwise you could claim "Alex looks like Bob, but Bob looks nothing like Alex"*
- $D(A,A) = 0$ *Constancy of Self-Similarity*
  - *Otherwise you could claim "Alex looks more like Bob, than Bob does"*
- $D(A,B) = 0$ IIf A= B *Positivity Separation*
  - *Otherwise there are objects in your world that are different, but you cannot tell apart.*
- $D(A,B) \leq D(A,C) + D(B,C)$ *Triangular Inequality*
  - *Otherwise you could claim "Alex is very like Bob, and Alex is very like Carl, but Bob is very unlike Carl"*

Unsupervised learning

Dissimilarity indices

Examples for quantitative variables

- Generalized euclidean distance: $d(x, z) = (x - z)^T A(x - z)$
  - $A = I$: euclidean distance
  - $A = \Sigma^{-1}$: Mahalanobis distance
- Minkowski Metrics:
  $L_m(\mathbf{x}, \mathbf{z}) = \sqrt[m]{|x_1 - z_1|^m + \ldots + |x_d - z_d|^m}$
  - $L_1 = |x_1 - z_1| + \ldots + |x_d - z_d|$: city block, taxicab, Manhattan distance
  - L_2: euclidean distance
  - $L_\infty = \max\{|x_1 - z_1|, \ldots, |x_d - z_d|\}$: Chebychev distance
  - $L_1 \geq L_2 \geq \ldots \geq L_\infty$



Unsupervised learning

Similarity Measures: Correlation Coefficient

Pearson correlation coefficient

$$r = \frac{\sum_{i=1}^{D}(x_i - \bar{x})(z_i - \bar{z})}{\sqrt{\sum_{i=1}^{D}(x_i - \bar{x})^2 \sum_{i=1}^{D}(z_i - \bar{z})^2}}$$

Examples for binary variables (I)

In the case of binary variables, we usually do not use distance as a similarity measure. Consider two binary vectors $\mathbf{x}$ and $\mathbf{z}$, that are two strings of binary data:
$x = [x_1 x_2 \dots x_d]^T$
$z = [z_1 z_2 \dots z_d]^T$ compare them coordinate-wise and then count the number of occurrences of specific combinations of 0's and 1's:

a) when x_k and z_k are both equal to 1
b) when $x_k = 0$ and $z_k = 1$
c) when $x_k = 1$ and $z_k = 0$
d) when x_k and z_k are both equal to 0

Examples for binary variables (II)

These four combinations of numbers can be organized into a 2×2 co-occurrence matrix to show how the two strings are "close" to each other. For instance, the first row and the first column corresponds to the number of times 1s' occur in both strings (equal to a):

	1	0
1	a	b
0	c	d

- matching coefficient: $\frac{a+b}{a+b+c+d}$
- Russell and Rao index: $\frac{a}{a+b+c+d}$
- Jacard index: $\frac{a}{a+b+c}$
- etc

Unsupervised learning

Dissimilarity indices

- nominal variables: the degree of difference between pairs of values must be delineated explicitly. If the variable assumes M distinct values, these can be arranged in a symmetric $M \times M$ matrix. The most common choice is to set the dissimilarity measure equal to 1 if the values differ.
- ordinal variables: can be treated as nominal variables (ignoring the order) or, by replacing their M original values with $\frac{i-1/2}{M}, i = 1, \ldots, M$ and treated as quantitative variables on this scale.

Unsupervised learning

Edit Distance:

A generic technique for measuring similarity To measure the similarity between two objects, transform one of the objects into the other, and measure how much effort it took. The measure of effort becomes the distance measure.

The distance between Patty and Selma.

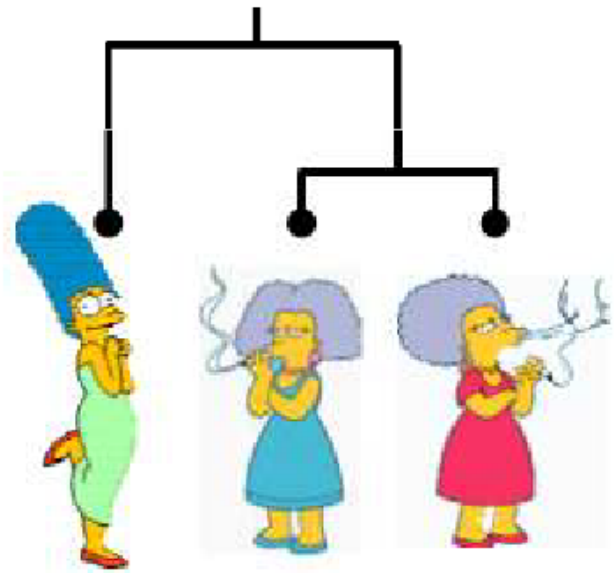
Change dress color, 1 point
Change earring shape, 1 point
Change hair part, 1 point

D(Patty,Selma) = **3**

The distance between Marge and Selma.

Change dress color, 1 point
Add earrings, 1 point
Decrease height, 1 point
Take up smoking, 1 point
Lose weight, 1 point

D(Marge,Selma) = **5**



This is called the
Edit distance
or the
Transformation distance

Unsupervised learning

Cluster Analysis

Categories of Clustering Algorithms

- Nonhierarchical clustering
  - Combinatorial or Partition-based clustering: work directly on the observed data with no direct reference to an underlying probability model
  - Model-based (a mixture of probabilities) clustering: supose that the data is an i.i.d. sample from some population described by a probability density function.
  - mode seekers, take a nonparametric perspective, attempting to directly estimate distinct modes of the probability density function.
- Hierarchical clustering

Unsupervised learning

Combinatorial or Partition-based clustering

- the data is partitioned into a prespecified number of clusters K, such that
  - every observation belongs to a cluster
  - all clusters are disjoint
- one seeks the optimal cluster assignment $C^*(n)$ that optimizes some goal function among all possible cluster assignment
- possible cost function $J(C) = \frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=k}\sum_{C(i')=k} d(x_i, x_{i'})$
- introducing the binary indicator variables $r_{nk} \in \{0, 1\}$, where $k = 1, \ldots, K$, describing which of the K clusters the data point $\mathbf{x}_n$ is assigned to, the cost function can be written as $J(C) = \frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N} r_{ik} r_{jk} d(x_i, x_j)$
- this criterion characterizes the extend to which observations assigned to the same cluster tend to be close to one another (sometimes called the 'within cluster' dispersion).
- the total dispersion is $T = \frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=k}\left(\sum_{C(i')=k} d(x_i, x_{i'}) + \sum_{C(i')\neq k} d(x_i, x_{i'})\right) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} d(x_i, x_j)$

Unsupervised learning

Combinatorial or Partition-based clustering

- Cluster analysis is a combinatorial optimization problem. Minimizing $J(C)$ by complete enumeration is feasible only for very small data sets.
- feasible strategies are based on iterative greedy descent: an initial partition is specified. At each iterative step, the cluster assignments are changed in such a way that the value of the criterion is improved from the previous value.
- these algorithms may converge to local optima

Unsupervised learning

K-means

- intended for quantitative variables and squared euclidean distance
- Noticing that $\sum_{i=1}^{N}\sum_{j=1}^{N}||x_i - x_j||^2 = 2N\sum_i ||x_i - \bar{x}||^2$, we can redefine the enlarged optimization problem

$$J(C) = \sum_{k=1}^{K}\sum_{i=1}^{N} r_{ik}||x_i - \mu_k||^2$$

which represents the sum of squares of the distances of each data point to its assigned center μ_k. The goal is to find the values for $\{r_{nk}\}$ and the $\{\mu_k\}$ so as to minimize J.

Unsupervised learning

K-means

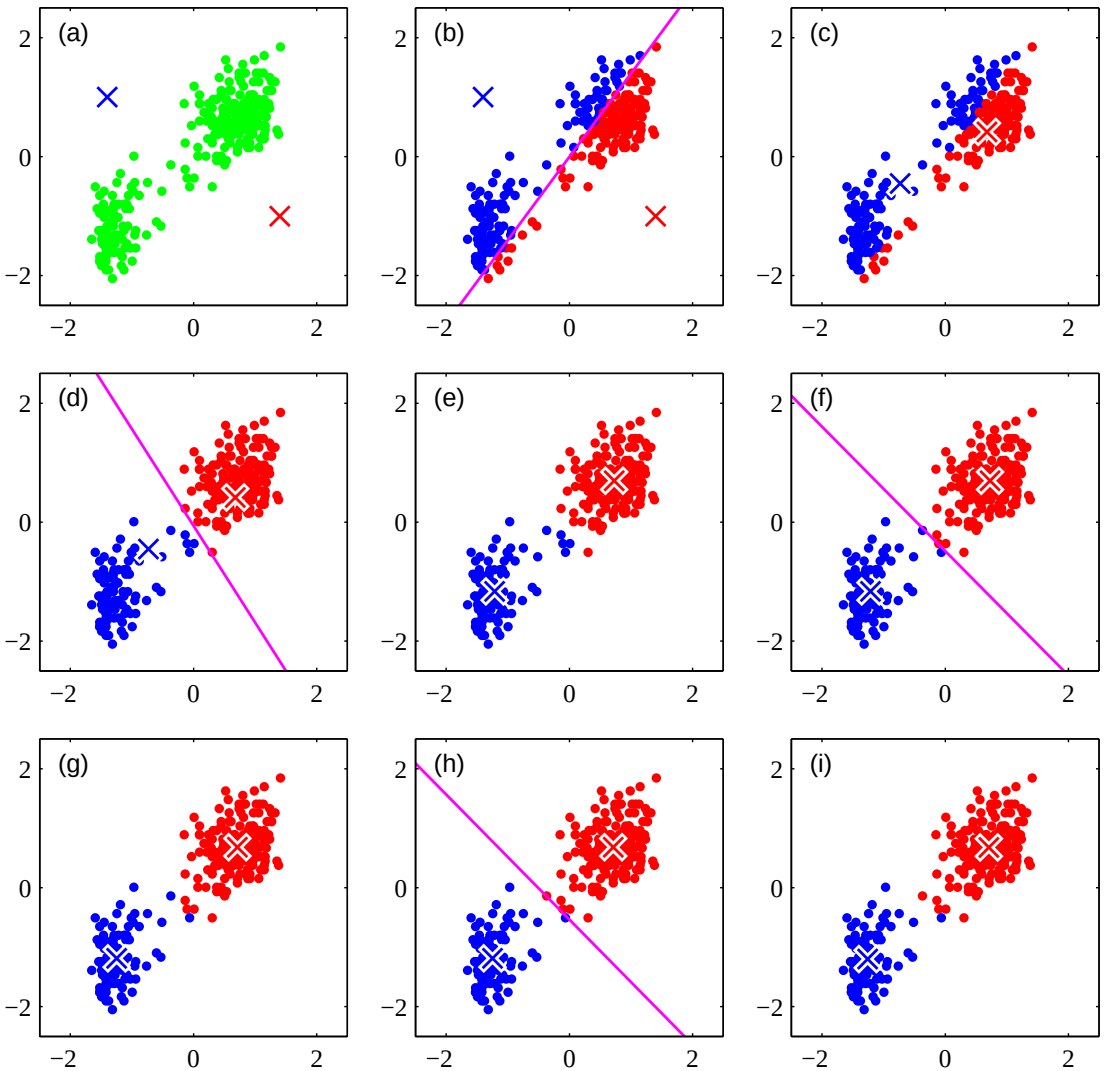
$$J(C) = \sum_{k=1}^{K} \sum_{i=1}^{N} r_{ik} ||x_i - \mu_k||^2$$

1. choose initial values for $\{\mu_k\}$
2. iterate:
   - minimize J with respect to r_{nk}, keeping μ_k fixed.
     $r_{nk} = \begin{cases} 1 \text{ if } k = \text{ argmin}_j ||x_n - \mu_j||^2 \\ 0 \text{ otherwise} \end{cases}$
     This corresponds to assigning x_n to the closest cluster center.
   - minimize J with respect to μ_k, keeping r_{nk} fixed. J is a quadratic function of μ_k and can be minimize by setting the derivative with respect to μ_k to zero: $\mu_k = \frac{\sum_n r_{nk} \mathbf{x}_n}{\sum_n r_{nk}}$
     The denominator is the number of points assigned to cluster k, resulting that μ_k is the mean of all points assigned to cluster k.

Unsupervised learning

Cluster Analysis

K-means

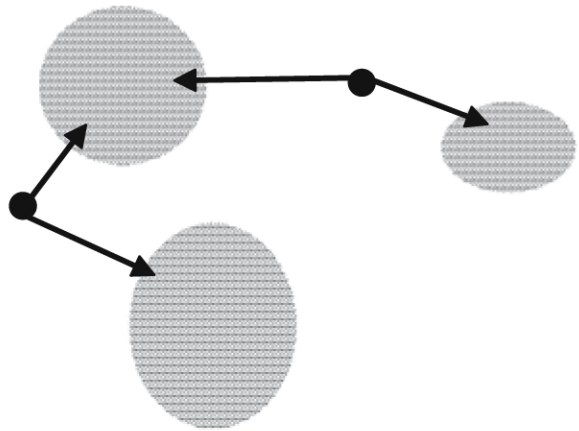


Unsupervised learning

Cluster Analysis

Soft K-means

- in K-means the responsibilities r_{nk} are restricted to $\{0, 1\}$. Points midway between two cluster centers have to be assigned to a single cluster



- soft K-means allows $r_{nk} \in [0, 1]$
- $r_{nk} = \begin{cases} 1 \text{ if } k = \text{ argmin}_j ||x_n - \mu_j||^2 \\ 0 \text{ otherwise} \end{cases}$ is replaced by a soft responsibility: $r_{nk} = \frac{G(||x_n - \mu_k||^2)}{\sum_{\ell=1}^{K} G(||x_n - \mu_\ell||^2)}$, where G(.) is a monotonically decreasing function

Unsupervised learning

Cluster Analysis

Vector Quantization

$K = 2$

$K = 3$

$K = 10$

Original image

Unsupervised learning

Cluster Analysis

K-medoids

- the restriction of squared euclidean distance with K-means requires all observations to be of qualitative type. Additionally, using squared euclidean distance places the highest influence on the largest distances: not very robust against outliers.
- In some problems we have only distances rather than raw observations
- K-means can be generalized with generic dissimilarity measures $d(.,.)$, minimizing the following distortion

$$J(C) = \sum_{k=1}^{K} \sum_{i=1}^{N} r_{ik} d(x_i, \mu_k)$$

- the E step involves assigning each data point to the cluster for which the dissimilarity is smallest ($\mathcal{O}(KN)$)
- the M step is potentially more complex than K-means, so it is common to restrict the cluster prototype (center) to be equal to one of the data vectors assigned to that cluster

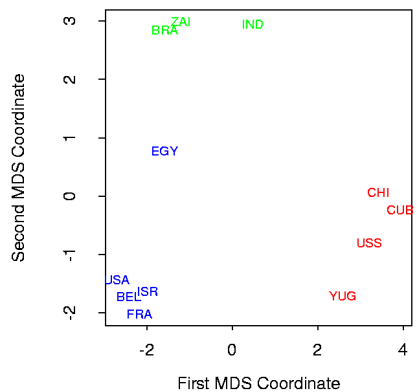
Unsupervised learning

Cluster Analysis

K-medoids

TABLE 14.3. *Data from a political science survey: values are average pairwise dissimilarities of countries from a questionnaire given to political science students.*

	BEL	BRA	CHI	CUB	EGY	FRA	IND	ISR	USA	USS	YUG
BRA	5.58										
CHI	7.00	6.50									
CUB	7.08	7.00	3.83								
EGY	4.83	5.08	8.17	5.83							
FRA	2.17	5.75	6.67	6.92	4.92						
IND	6.42	5.00	5.58	6.00	4.67	6.42					
ISR	3.42	5.50	6.42	6.42	5.00	3.92	6.17				
USA	2.50	4.92	6.25	7.33	4.50	2.25	6.33	2.75			
USS	6.08	6.67	4.25	2.67	6.00	6.17	6.17	6.92	6.17		
YUG	5.25	6.83	4.50	3.75	5.75	5.42	6.08	5.83	6.67	3.67	
ZAI	4.75	3.00	6.08	6.67	5.00	5.58	4.83	6.17	5.67	6.50	6.92



Unsupervised learning

Cluster Analysis

K-medoids

1. For a given cluster assignment C find the observation in the cluster minimizing total distance to other points in that cluster:

$$i_k^* = \underset{\{i:C(i)=k\}}{\operatorname{argmin}} \sum_{C(i')=k} D(x_i, x_{i'}).$$

   Then $m_k = x_{i_k^*}$, $k = 1, 2, \ldots, K$ are the current estimates of the cluster centers.

2. Given a current set of cluster centers $\{m_1, \ldots, m_K\}$, minimize the total error by assigning each observation to the closest (current) cluster center:

$$C(i) = \underset{1 \le k \le K}{\operatorname{argmin}} D(x_i, m_k).$$

3. Iterate steps 1 and 2 until the assignments do not change.

Clustering

Model based algorithms

Mixture of Gaussians

- we assume a certain probabilistic model of the data and then estimate its parameters: mixture density
- we assume that the data are a result of a mixture of K sources of data that might be thought of as clusters
- assuming a mixture of gaussians, the distribution can be written as $p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\mu_k, \sigma_k)$
- each Gaussian density $\mathcal{N}(\mathbf{x}|\mu_k, \sigma_k)$ is called a component
- the parameters $\pi_k \geq 0$ are called mixing coefficients and $\sum_{k=1}^{K} \pi_k = 1$

Clustering

Model based algorithms

Mixture of Gaussians

- the posterior probabilities $p(k|\mathbf{x})$ play the role of responsibilities. From Bayes' theorem these are given by $r_{nk} = p(k|\mathbf{x}_n) = \frac{p(\mathbf{x}_n|k)p(k)}{\sum_{j=1}^{K} p(\mathbf{x}_n|j)p(j)} = \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\mu_k,\Sigma_k)}{\sum_j \pi_j \mathcal{N}(\mathbf{x}_n|\mu_j,\Sigma_j)}$
- the log of the likelihood function is given by

$$\ln p(X|\pi,\mu,\Sigma) = \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_n|\mu_k,\Sigma_k) \right\}$$

Clustering

Model based algorithms

Expectation-maximization algorithm for Mixture of Gaussians

- setting the derivatives of $\ln p(X|\pi, \mu, \Sigma)$ with respect to μ_k to zero, we obtain

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} r_{nk} \mathbf{x}_n$$

  with $N_k = \sum_{n=1}^{N} r_{nk}$ the effective number of points assigned to cluster k.
- setting the derivatives of $\ln p(X|\pi, \mu, \Sigma)$ with respect to Σ_k to zero, we obtain

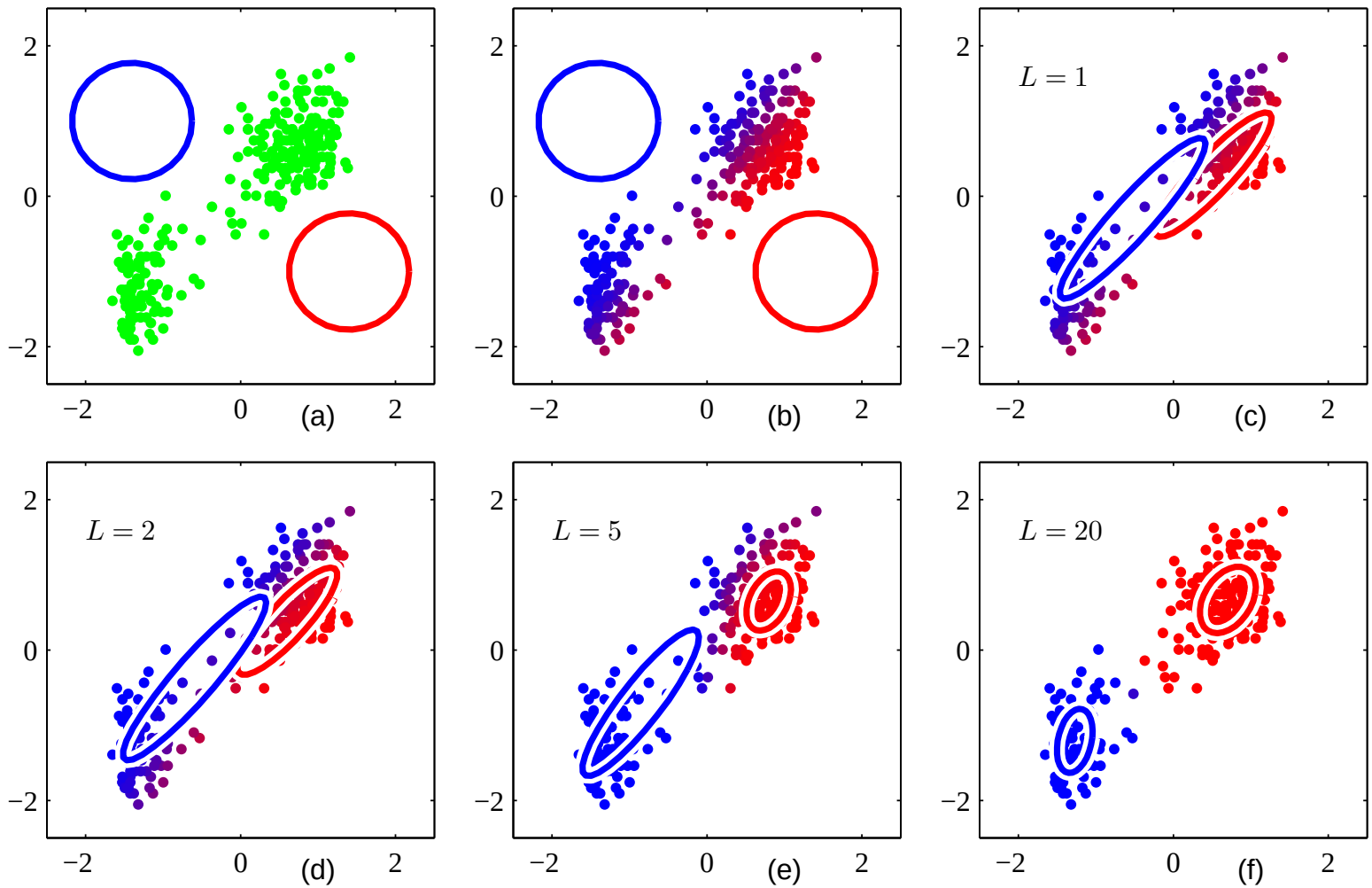
$$\Sigma_k = \frac{1}{N_k} \sum_{n=1}^{N} r_{nk} (\mathbf{x}_n - \mu_k)(\mathbf{x}_n - \mu_k)^T$$

- setting the derivatives of $\ln p(X|\pi, \mu, \Sigma)$ with respect to π_k to zero, we obtain $\pi_k = \frac{N_k}{N}$

Clustering

Model based algorithms

Expectation-maximization algorithm for Mixture of Gaussians



Clustering

Model based algorithms

Mixture of Gaussians

1. initialize the means μ_k, covariances Σ_k and the mixing coefficients π_k
2. E step. Evaluate the responsibilities using the current parameters values $r_{nk} = \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\mu_k,\Sigma_k)}{\sum_j \pi_j \mathcal{N}(\mathbf{x}_n|\mu_j,\Sigma_j)}$
3. M step. Re-estimate the parameters using the current responsibilities:

$$\mu_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} r_{nk} \mathbf{x}_n$$

$$\Sigma_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} r_{nk} (\mathbf{x}_n - \mu_k^{new})(\mathbf{x}_n - \mu_k^{new})^T$$

$$\pi_k^{new} = \frac{N_k}{N}$$

The EM Algorithm

A more general view

- Consider a general scenario in which we have observed data $\mathbf{x}$, and a set of unknown parameters $\boldsymbol{\theta}$.
- Let us also assume some prior $p(\boldsymbol{\theta})$ for the parameters, which could well be a flat prior.
- The a posteriori probability function $p(\boldsymbol{\theta}|\mathbf{x})$ is proportional to $p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})$.
- Now, suppose that finding the MAP estimate of $\boldsymbol{\theta}$ would be easier if we had access to some other data $\boldsymbol{z}$, that is, it would be easy to maximize $p(\mathbf{x}; \boldsymbol{z}|\boldsymbol{\theta})p(\boldsymbol{\theta})$, where $p(\mathbf{x}; \boldsymbol{z}|\boldsymbol{\theta})$ is related to $p(\mathbf{x}|\boldsymbol{\theta})$ via marginalization

$$p(\mathbf{x}|\boldsymbol{\theta}) = \int p(\mathbf{x}; \boldsymbol{z}|\boldsymbol{\theta}) d\boldsymbol{z}$$

- The expectation-maximization (EM) algorithm is an iterative procedure which can be shown to converge to a (local) maximum of the marginal a posteriori probability function $p(\boldsymbol{\theta}|\mathbf{x}) = p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})$, without the need to explicitly manipulate the marginal likelihood $p(\mathbf{x}|\boldsymbol{\theta})$.

The EM Algorithm

A more general view

- Usually, $\boldsymbol{u} = (\mathbf{x}; \boldsymbol{z})$ is called the complete data, while $p(\boldsymbol{u}|\boldsymbol{\theta})$ is termed the complete likelihood function.
- The complete likelihood is supposed to be relatively easy to maximize with respect to $\boldsymbol{\theta}$.
- In many cases, the missing data is artificially inserted as a means of allowing the use of the EM algorithm to find a difficult ML estimate.
- Specifically, when $p(\mathbf{x}|\boldsymbol{\theta})$ is difficult to maximize with respect to $\boldsymbol{\theta}$, but there is an alternative model $p(\mathbf{x}; \boldsymbol{z}|\boldsymbol{\theta})$ which is easy to maximize with respect to $\boldsymbol{\theta}$, and such that $p(\mathbf{x}|\boldsymbol{\theta})$ is related to $p(\mathbf{x}; \boldsymbol{z}|\boldsymbol{\theta})$ via marginalization.

The EM Algorithm

A more general view

The EM algorithm works iteratively by alternatingly applying two steps: the E-Step (expectation) and the M-Step (maximization). Formally, let $\hat{\boldsymbol{\theta}}^{(t)}$ for $t = 0, 1, 2, \cdots$, denote the successive parameter estimates; the E and M steps are defined as:

E-Step : Compute the conditional expectation (with respect to the missing $\boldsymbol{z}$) of the logarithm of the complete a posteriori probability function, $\log p(\boldsymbol{z}; \boldsymbol{\theta}|\mathbf{x})$, given the observed data $\mathbf{x}$ and the current parameter estimate $\hat{\boldsymbol{\theta}}^{(t)}$ (usually called the Q-function):

$$Q(\boldsymbol{\theta}|\hat{\boldsymbol{\theta}}^{(t)}) \equiv E[\log p(\boldsymbol{z}, \boldsymbol{\theta}|\mathbf{x})|\mathbf{x}, \hat{\boldsymbol{\theta}}^{(t)}]$$

$$\propto \log p(\boldsymbol{\theta}) + E[\log p(\boldsymbol{z}, \mathbf{x}|\boldsymbol{\theta})|\mathbf{x}, \hat{\boldsymbol{\theta}}^{(t)}]$$

M-Step : Update the parameter estimate according to

$$\hat{\boldsymbol{\theta}}^{(t+1)} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}|\hat{\boldsymbol{\theta}}^{(t)})$$

The process continues until some stopping criterion is met.

Unsupervised learning

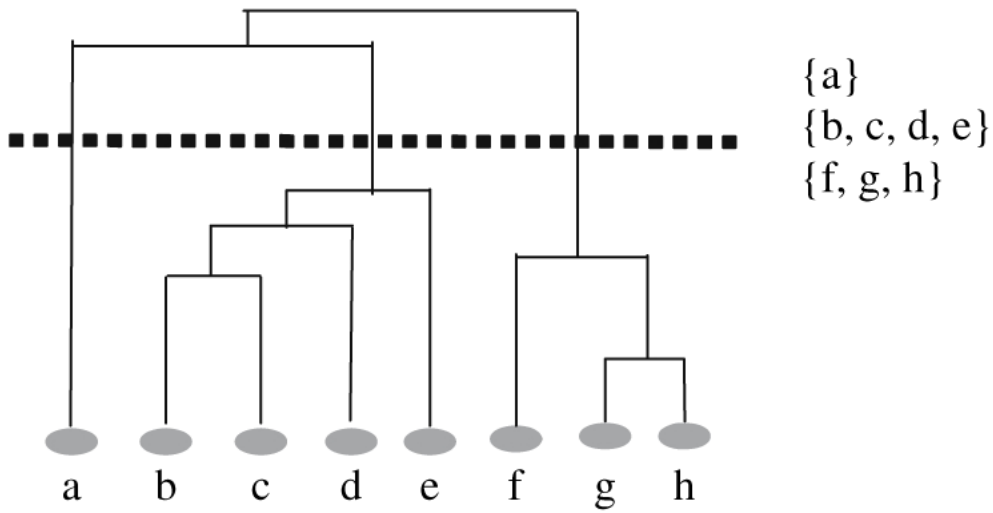
Hierarchical Clustering

- hierarchical clustering does not require the specification of the number of clusters
- hierarchical clustering methods require a measure of dissimilarity between groups of observations, based on the pairwise dissimilarities among observations in the groups
- the output is a hierarchical representation: the clusters at each level are created by merging clusters at the next lower level
- at the highest level there is only one cluster
- at the lowest level each cluster contains a single observation

Unsupervised learning

Hierarchical Clustering

- two main strategies: bottom-up (agglomerative) and top-down (divisive)
- the result can be represented in the form of a dendrogram
- dissimilarity between merged clusters is monotone increasing with the level of the merger. The height of each node is proportional to the value of the intergroup dissimilarity between its two daughters.



Unsupervised learning

Hierarchical Clustering

Intergroup dissimilarity

- Single linkage (SL):

$$d(A, B) = \min_{x \in A, y \in B} d(x, y)$$

  Also called nearest-neighbour

- Complete linkage (CL):

$$d(A, B) = \max_{x \in A, y \in B} d(x, y)$$

  Also called furthest-neighbour

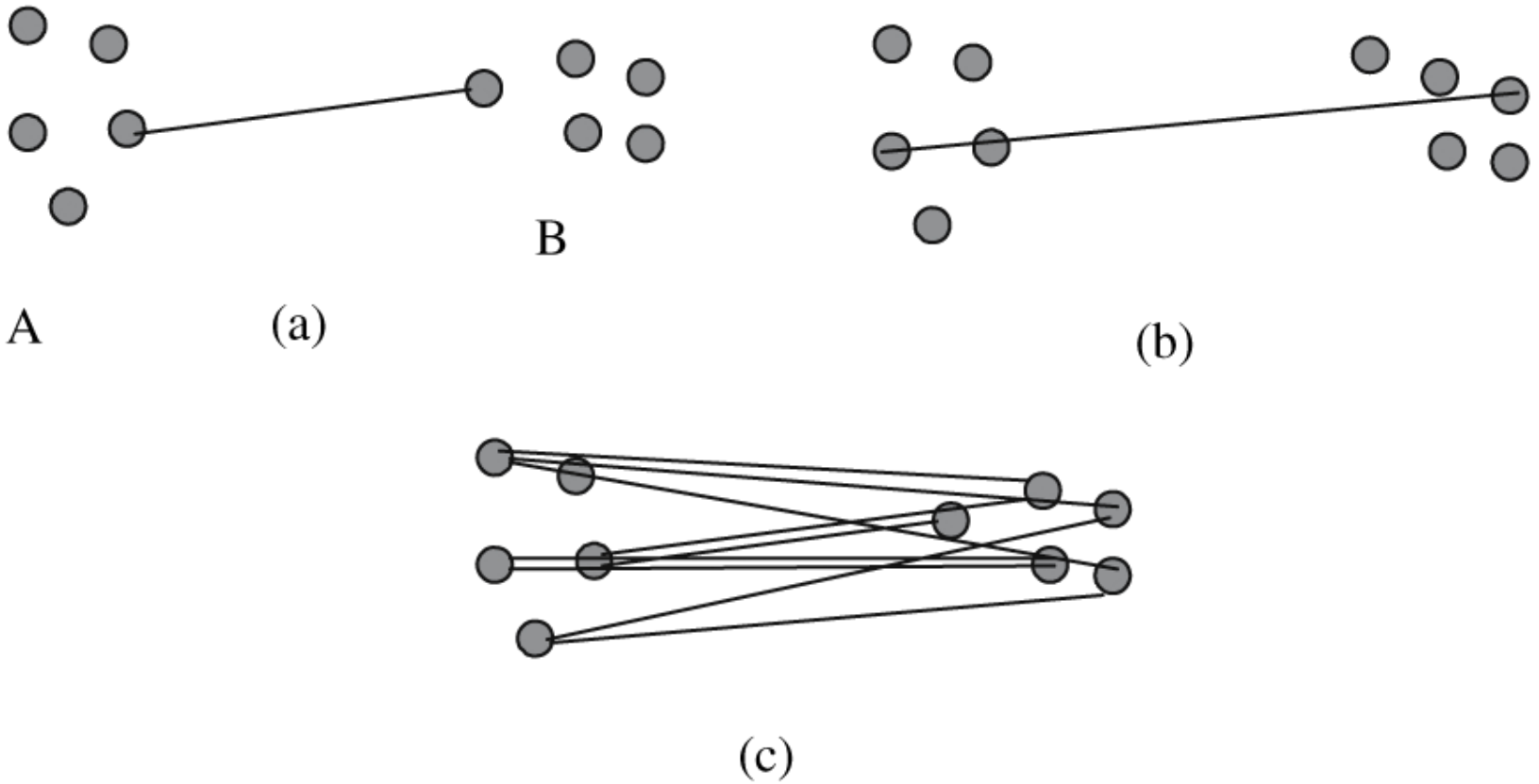
- Group average (GA):

$$d(A, B) = \frac{1}{card(A)card(B)} \sum_{x \in A, y \in B} d(x, y)$$

Unsupervised learning

Hierarchical Clustering

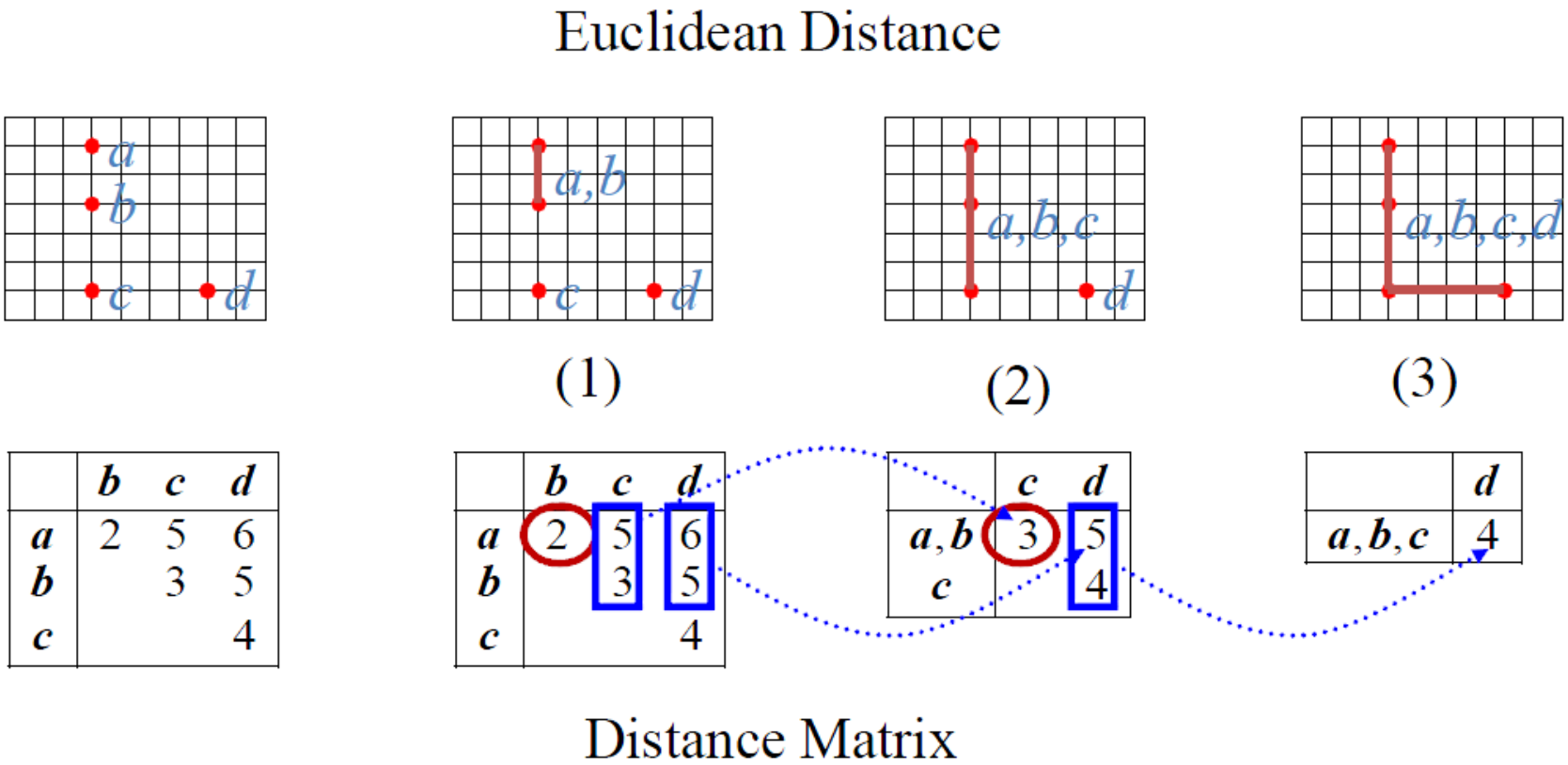
Intergroup dissimilarity



Unsupervised learning

Hierarchical Clustering

Single-Link Method



	b	c	d
a	2	5	6
b		3	5
c			4

	b	c	d
a	2	5	6
b		3	5
c			4

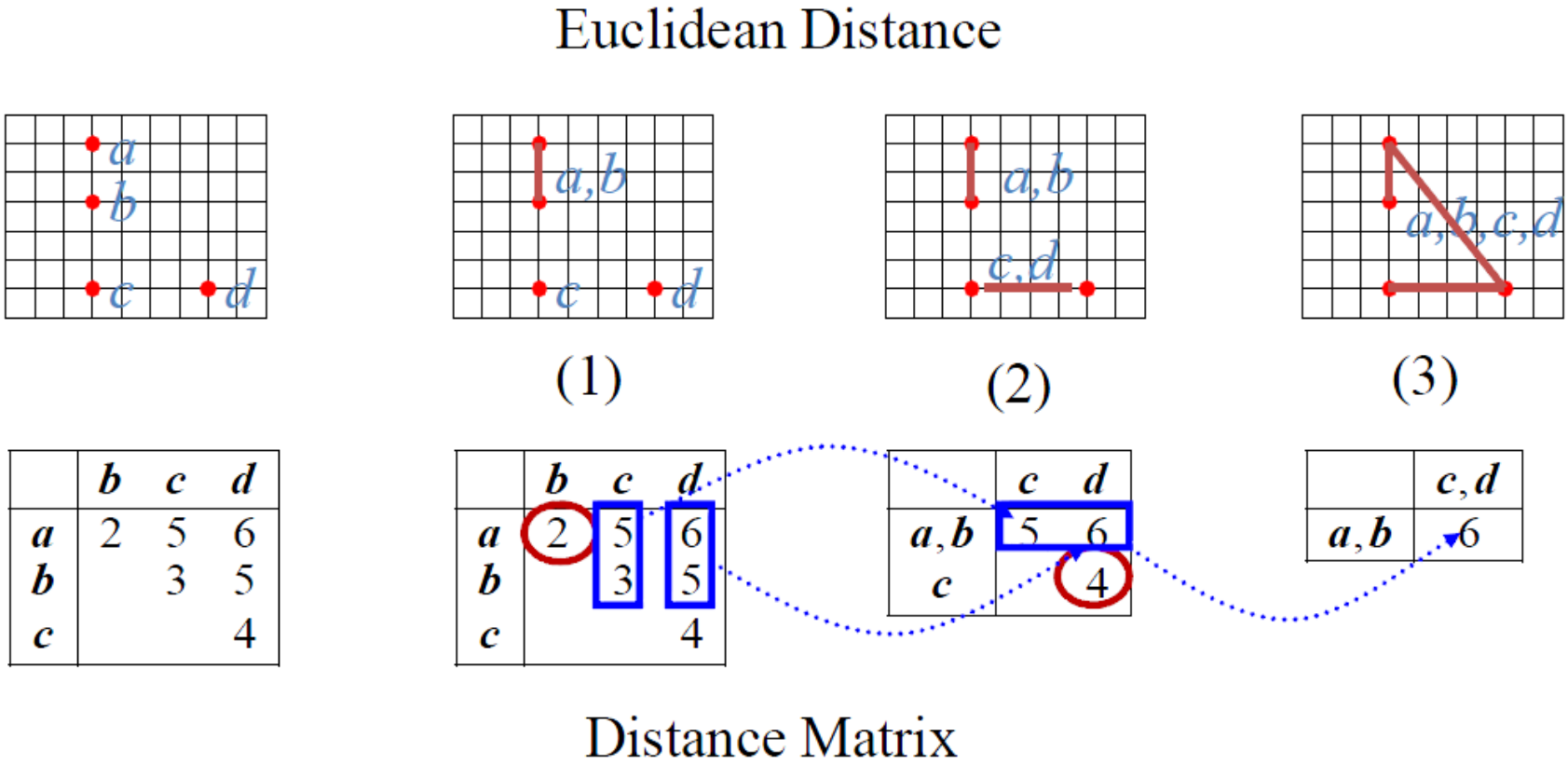
	c	d
a,b	3	5
c		4

	d
a,b,c	4

Unsupervised learning

Hierarchical Clustering

Complete-Link Method



Unsupervised learning

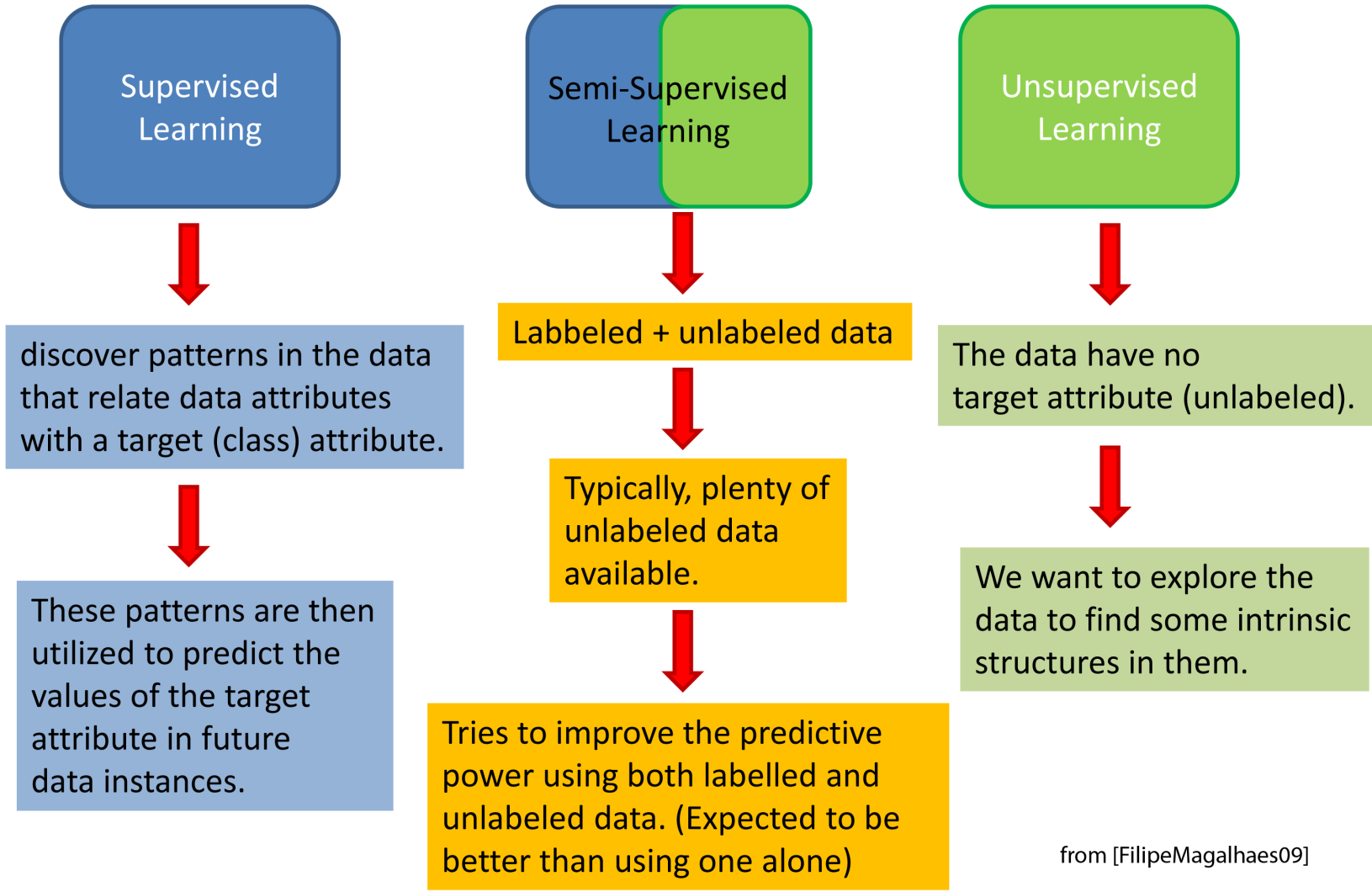
Hierarchical Clustering

- single linkage can produce the chaining phenomenon, leading to clusters with very large diameters
- Hierarchical Clustering impose a hierarchical structure whether or not such structure actually exists in the data

empirical learning

- mixing unsupervised with supervised learning
- Unsupervised learning as supervised learning

Semi-Supervised Learning

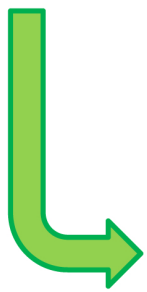


from [FilipeMagalhaes09]

Semi-Supervised Learning

Unlabeled data is easy to obtain

Labelled data can be difficult to obtain
- human annotation is boring
- may require experts
- may require special equipment
- very time-consuming



from [FilipeMagalhaes09]

Examples:
- Web page classification (billions of pages)
- Email classification (SPAM or No-SPAM)
- Speech annotation (400h for each hour of conversation)
- ...

Semi-Supervised Learning

Sometimes, it may not be so hard to label data...

www.espgame.org

Tries to guess the user's gender based on his/her choices.

After that, we tell if it was right or wrong

Takes advantage of player's intervention in order to enrich the training of automatic learning algorithms

from [FilipeMagalhaes09]

Semi-Supervised Learning

Self-Training

$L = (X_i , Y_i)$ ⟸ Set of labelled data

$U = (X_i , ?)$ ⟸ Set of unlabeled data

Algorithm

Repeat

- Train a classifier C with training data L
- Classify data in U with C
- Find a subset U′ of U with the most confident scores
- L + U′ ➔ L
- U – U′ ➔ U

from [FilipeMagalhaes09]

References

Bishop

Pattern recognition and machine learning,

Book.

Trevor Hastie and Robert Tibshirani and Jerome Friedman

The elements of statistical learning,

Springer.

Mário A. T. Figueiredo

Lecture Notes on the EM Algorithm

http://www.stat.duke.edu/courses/Spring06/sta376/Support/EM/EM.Mixtures.Figueiredo.2004.pdf