Clustering

Exemplifying the application of hierarchical agglomerative clustering (single-, complete- and average-linkage)

CMU, 2012 fall, Tom Mitchell, Ziv Bar-Joseph, HW4, pr. 2.a

extended by Liviu Ciortuz

The table below is a distance matrix for 6 objects.

	A	B	C	D	E	F
A	0					
B	0.12	0				
C	0.51	0.25	0			
D	0.84	0.16	0.14	0		
E	0.28	0.77	0.70	0.45	0	
F	0.34	0.61	0.93	0.20	0.67	0

Show the final result of hierarchical clustering with single-, complete- and average-linkage by drawing the corresponding dendrograms.

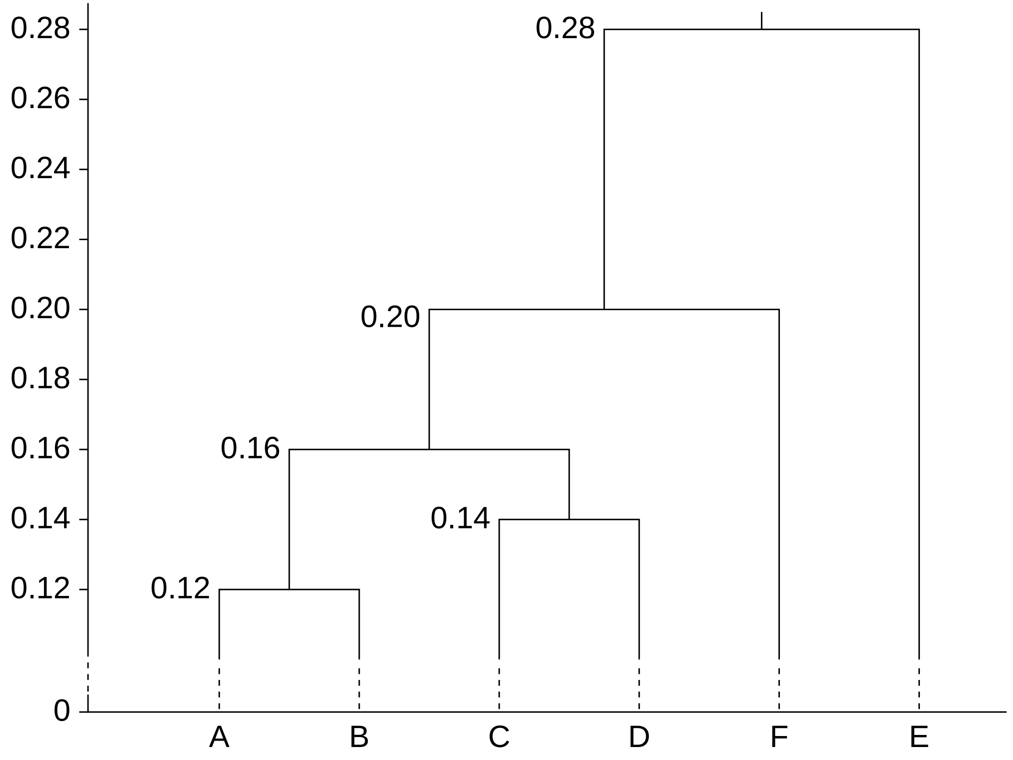
Solution:
Single-linkage:

	AB	C	D	E	F
AB	0				
C	0.25	0			
D	0.16	**0.14**	0		
E	0.28	0.70	0.45	0	
F	0.34	0.93	0.20	0.67	0

	AB	CD	E	F
AB	0			
CD	**0.16**	0		
E	0.28	0.45	0	
F	0.34	0.20	0.67	0

	$ABCD$	E	F
$ABCD$	0		
E	0.28	0	
F	**0.20**	0.67	0

	$ABCDF$	E
$ABCDF$	0	
E	**0.28**	0



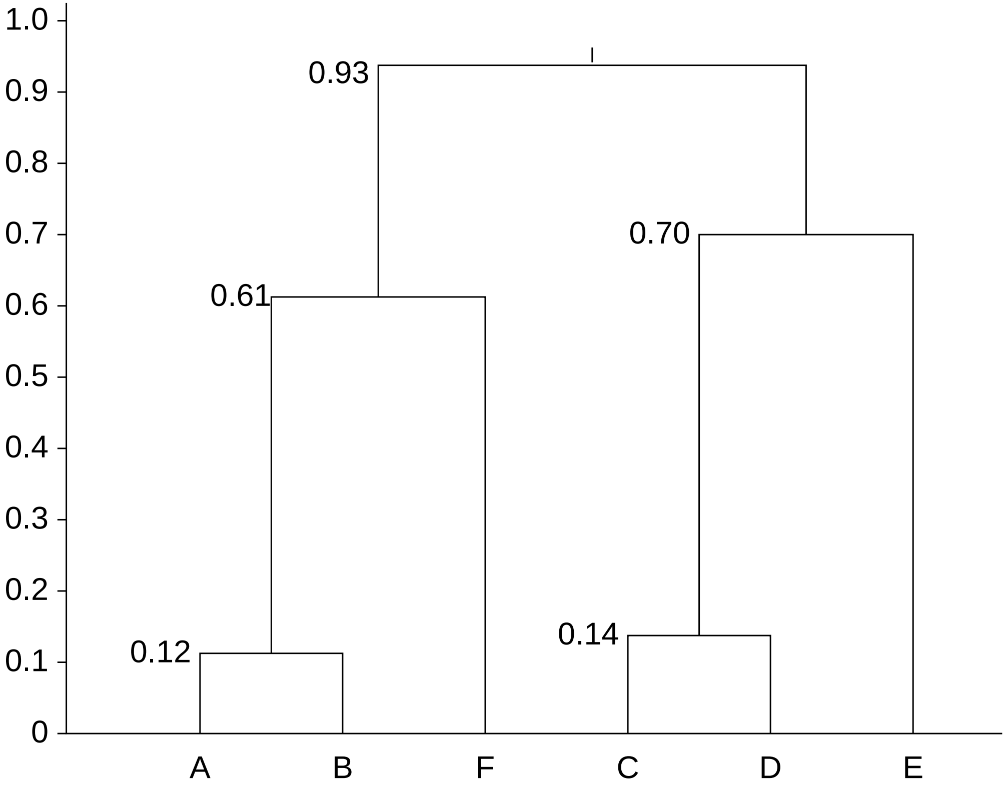
Complete-linkage:

	AB	C	D	E	F
AB	0				
C	0.51	0			
D	0.84	**0.14**	0		
E	0.77	0.70	0.45	0	
F	0.61	0.93	0.20	0.67	0

	AB	CD	E	F
AB	0			
CD	0.84	0		
E	0.77	0.70	0	
F	**0.61**	0.93	0.67	0

	ABF	CD	E
ABF	0		
CD	0.93	0	
E	0.77	**0.70**	0

	ABF	CDE
ABF	0	
CDE	**0.93**	0



Average-linkage:

	AB	C	D	E	F
AB	0				
C	0.38	0			
D	0.50	**0.14**	0		
E	0.525	0.70	0.45	0	
F	0.475	0.93	0.20	0.67	0

$$d(AB,C) = \frac{d(A,C)+d(B,C)}{2} = \frac{0.51+0.25}{2} = 0.38$$

$$d(AB,D) = \frac{d(A,D)+d(B,D)}{2} = \frac{0.84+0.16}{2} = 0.5$$

$$d(AB,E) = \frac{d(A,E)+d(B,E)}{2} = \frac{0.28+0.77}{2} = 0.525$$

$$d(AB,F) = \frac{d(A,F)+d(B,F)}{2} = \frac{0.34+0.61}{2} = 0.475$$

	AB	CD	E	F
AB	0			
CD	**0.44**	0		
E	0.525	0.575	0	
F	0.475	0.565	0.67	0

$$d(AB,CD) \overset{(*)}{=} \frac{2d(AB,C)+2d(AB,D)}{4} = \frac{0.38+0.50}{2} = 0.44$$

$$d(CD,E) = \frac{d(C,E)+d(D,E)}{2} = \frac{0.70+0.45}{2} = 0.575$$

$$d(CD,F) = \frac{d(C,F)+d(D,F)}{2} = \frac{0.93+0.20}{2} = 0.565$$

	$ABCD$	E	F
$ABCD$	0		
E	0.55	0	
F	**0.52**	0.67	0

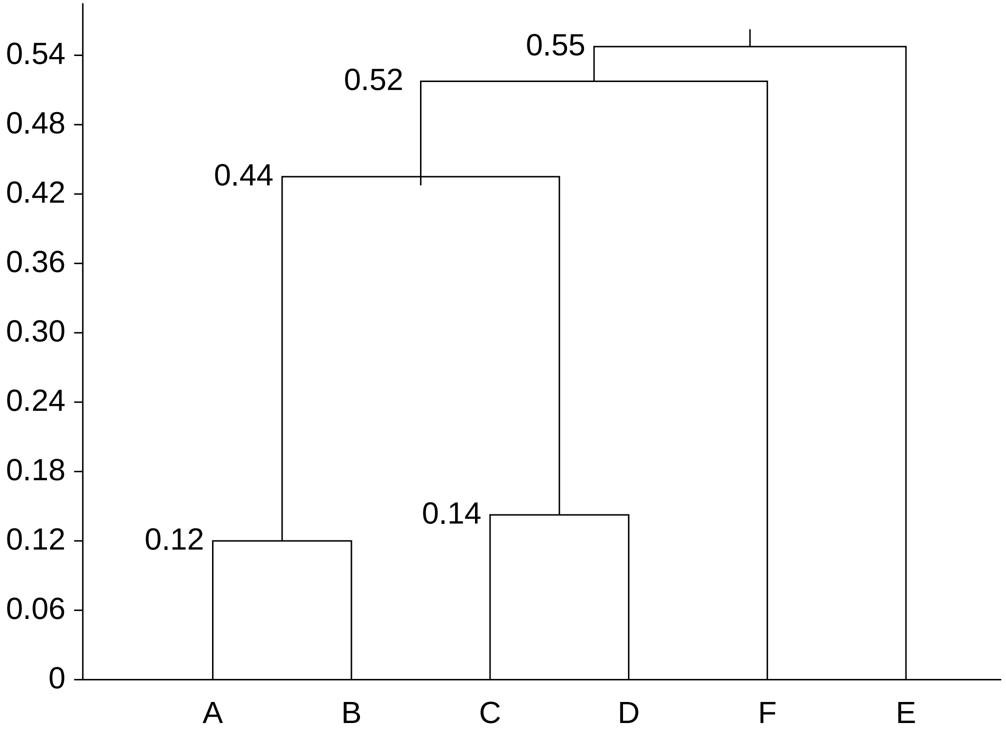
$$d(ABCD,E) \overset{(*)}{=} \frac{2d(AB,E)+2d(CD,E)}{4} = \frac{0.525+0.575}{2} = 0.55$$

$$d(ABCD,F) \overset{(*)}{=} \frac{2d(AB,F)+2d(CD,F)}{4} = \frac{0.475+0.565}{2} = 0.52$$

Average-linkage (cont'd):

	$ABCDF$	E
$ABCDF$	0	
E	**0.55**	0

$$d(ABCDF, E) \overset{(*)}{=} \frac{4d(ABCD, E) + d(F, E)}{5} = \frac{2.87}{5} = 0.574$$



Note: **For the proof of the (*) relation, see problem CMU, 2010 fall, Aarti Singh, HW3, pr. 4.2:**

$$d(X \cup Y, Z) \quad = \quad \frac{|X|\, d(X, Z) + |Y|\, d(Y, Z)}{|X| + |Y|}$$

Exemplifying

the application of hierarchical devisive clustering and the relationship between slingle-linkage hierarchies and Minimum Spanning Trees (MSTs)

CMU, 2009 spring, Ziv Bar-Joseph, final exam, pr. 9.3

Hierarchical clustering may be bottom-up or top-down. In this problem we will see whether a top-down clustering algorithm can be exactly analogous to a bottom-up clustering algorithm.

Consider the following *top-down clustering algorithm*:

1. Calculate the pairwise distance $d(P_i, P_j)$ between every two objects P_i and P_j in the set of objects to be clustered, and build a complete graph on the set of objects with edge weights being the corresponding distances.

2. Generate the Minimum Spanning Tree of the graph, i.e. choose the subset of edges E' with minimum sum of weights such that $G' = (P, E')$ is a single connected tree.

3. Throw out the edge with the heviest weight to generate two disconnected trees corresponding to top level clusters.

4. Repeat the previous step recursively on the lower level clusters to generate a top-down clustering on the set of n objects.

a. Apply this algoritm on the dataset given in the nearby table, using the Euclidian distance.

b. Does this top-down algorithm perform analogously to any bottom-up algorithm that you have encountered in class? Why?

Point	x	y
P_1	1	2
P_2	2	2
P_3	3	6
P_4	6	4
P_5	6	6
P_6	12	12

Solution:

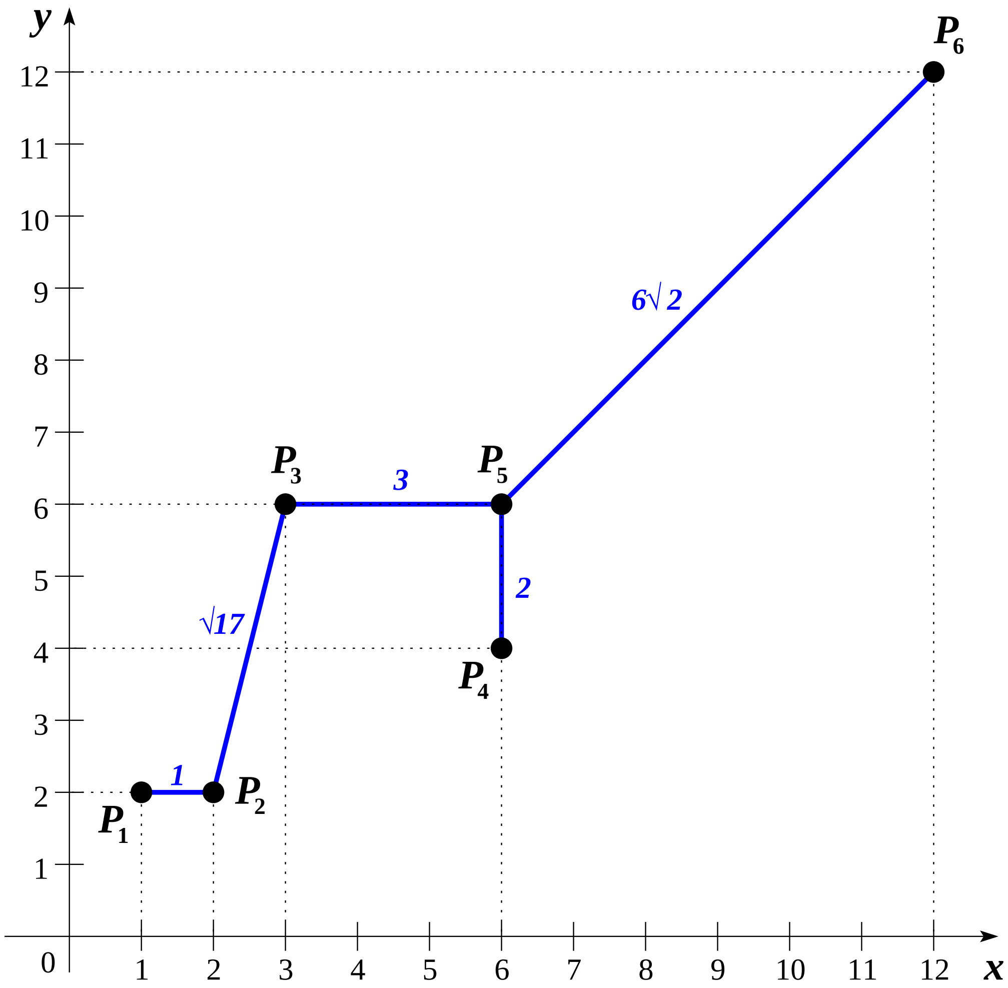
a.

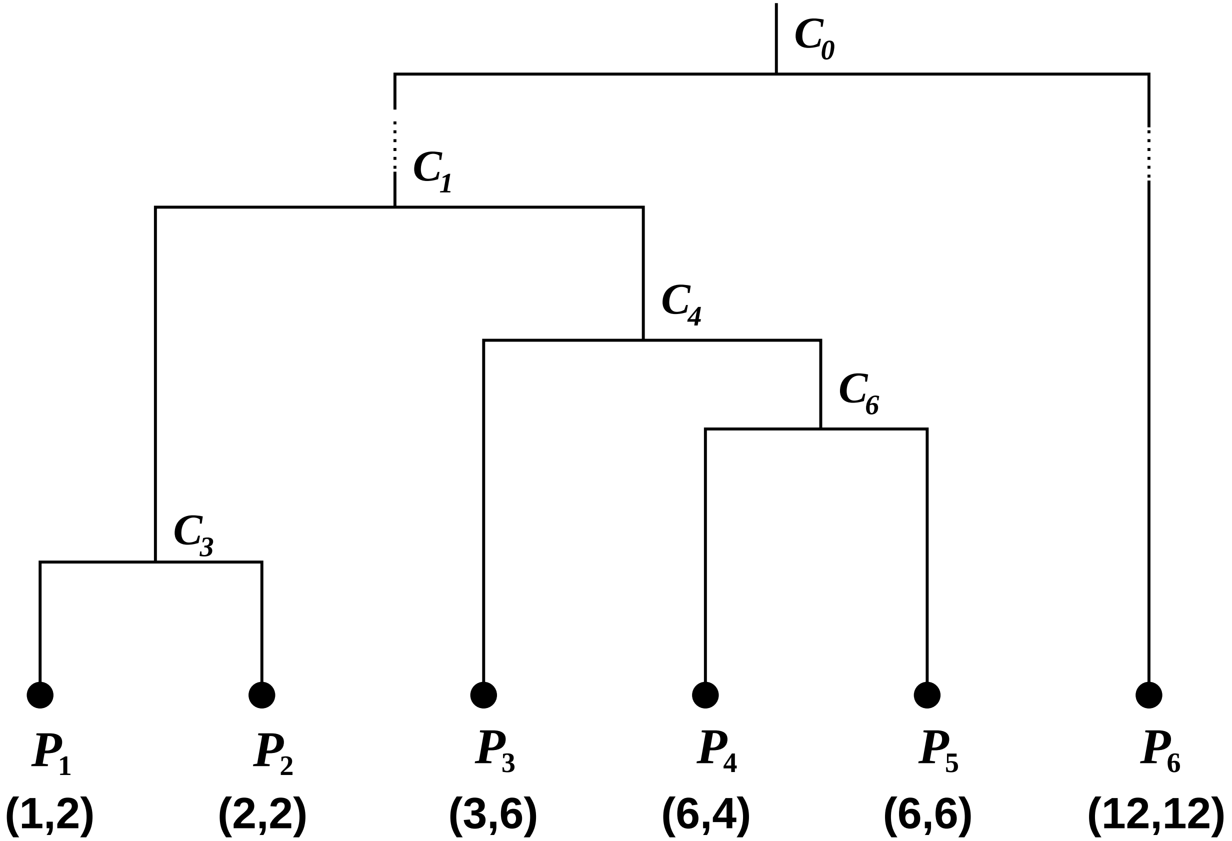
Kruskal algorithm:

1. (P_1, P_2), **cost** 1,
2. (P_4, P_5), **cost** 2,
3. (P_3, P_5), **cost** 3,
4. (P_2, P_3), **cost** $\sqrt{17}$
5. (P_5, P_6), **cost** $6\sqrt{2}$.

Prim algorithm:

1. (P_1, P_2), **cost** 1,
2. (P_2, P_3), **cost** $\sqrt{17}$
3. (P_3, P_5), **cost** 3,
4. (P_5, P_4), **cost** 2,
5. (P_5, P_6), **cost** $6\sqrt{2}$.




C_0
C_1
C_4
C_6
C_3
P_1
(1,2)
P_2
(2,2)
P_3
(3,6)
P_4
(6,4)
P_5
(6,6)
P_6
(12,12)

Note: **If there is only one MST for the given dataset, then both Kruskal's and Prim's algorithm will find it. Otherwise, the two algorithms can produce differents results.**

One can see (both on this dataset and also in general) that Kruskal's algorithm is exactly analogous to the single-linkage bottom-up clustering algorithm.

Therefore, there is indeed a bottom-up equivalent to the top-down clustering algorithm presented in this exercise.

Hierachical [bottom-up] clustering

Ward's metric

CMU, 2010 fall, Aarti Singh, HW3, pr. 4.1

In this problem you will anlayze an alternative approach to quantify the distance between two disjoint clusters, proposed by Joe H. Ward in 1963. We will call it Ward's metric.

Ward's metric simply says that the distance between two disjoint clusters, X and Y, is how much the sum of squares will increase when we merge them. More formally,

$$\Delta(X, Y) = \sum_{i \in X \cup Y} \|x_i - \mu_{X \cup Y}\|^2 - \sum_{i \in X} \|x_i - \mu_X\|^2 - \sum_{i \in Y} \|x_i - \mu_Y\|^2 \tag{1}$$

where μ_i is the centroid of cluster i and x_i is a data point in the X cluster. By definition, here we will consider $\mu_X = \frac{1}{n_X} \sum_{x_i \in X} x_i$, where n_X is the number of elements in X.

$\Delta(X, Y)$ can be thought as the *merging cost* of combining clusters X and Y into one cluster. That is, in agglomerative clustering those two clusters with the lowest *merging cost* are merged, when Ward's metric is used as a *closeness* measure.

a. Can you reduce the formula in equation (1) for $\Delta(X, Y)$ to a simpler form? Give the simplified formula.

***Hint*: Your formula should be in terms of the cluster sizes (let's denote them as n_X and n_Y) and the distance $\|\mu_X - \mu_Y\|^2$ between cluster centroids μ_X and μ_Y *only*.**

Solution

$$
\begin{aligned}
\Delta(X,Y) = &\sum_{i\in X\cup Y} ||x_i - \mu_{X\cup Y}||^2 - \sum_{i\in X} ||x_i - \mu_X||^2 - \sum_{i\in Y} ||x_i - \mu_Y||^2 \\
\overset{(3)}{=} &\sum_{i\in X\cup Y} \left(x_i - \frac{n_X\mu_X + n_Y\mu_Y}{n_X + n_Y}\right)^2 - \sum_{i\in X}(x_i - \mu_X)^2 - \sum_{i\in Y}(x_i - \mu_Y)^2 \\
\overset{(4)}{=} &\sum_{i\in X\cup Y} x_i^2 - \left(\frac{2n_X\mu_X + 2n_Y\mu_Y}{n_X + n_Y}\right) \sum_{i\in X\cup Y} x_i + \sum_{i\in X\cup Y} \left(\frac{n_X\mu_X + n_Y\mu_Y}{n_X + n_Y}\right)^2 \\
&- \sum_{i\in X} x_i^2 + 2\mu_X \sum_{i\in X} x_i - \sum_{i\in X} \mu_X^2 - \sum_{i\in Y} x_i^2 + 2\mu_Y \sum_{i\in Y} x_i - \sum_{i\in Y} \mu_Y^2 \\
= &-\frac{2(n_X\mu_X + n_Y\mu_Y)^2}{n_X + n_Y} + \frac{(n_X\mu_X + n_Y\mu_Y)^2}{n_X + n_Y} + 2\mu_X n_X \mu_X + 2\mu_Y n_Y \mu_Y - n_X\mu_X^2 - n_Y\mu_Y^2 \\
= &-\frac{(n_X\mu_X + n_Y\mu_Y)^2}{n_X + n_Y} + n_X\mu_X^2 + n_Y\mu_Y^2 = \frac{n_X n_Y}{n_X + n_Y}(\mu_X^2 - 2\mu_X\mu_Y + \mu_Y^2) \\
= &\frac{n_X n_Y}{n_X + n_Y} ||\mu_X - \mu_Y||^2.
\end{aligned}
$$

3 explicative notes

$$\mu_X \overset{def.}{=} \frac{1}{|X|} \sum_{x_i \in X} x_i \Rightarrow \sum_{x_i \in X} x_i = |X|\,\mu_X \tag{2}$$

$$\begin{aligned}
\mu_{X \cup Y} \quad & \overset{def.}{=} \quad \frac{1}{|X \cup Y|} \sum_{x_i \in X \cup Y} x_i \overset{X \cap Y = \emptyset}{=} \frac{1}{|X| + |Y|} \left(\sum_{x_i \in X} x_i + \sum_{y_i \in Y} y_i \right) \\
& \overset{(2)}{=} \quad \frac{1}{|X| + |Y|} (|X|\,\mu_X + |Y|\,\mu_Y) = \frac{1}{n_X + n_Y} (n_X \mu_X + n_Y \mu_Y)
\end{aligned} \tag{3}$$

$$\begin{aligned}
\frac{2n_X \mu_X + 2n_Y \mu_Y}{n_X + n_Y} \sum_{x_i \in X \cup Y} x_i \quad & \overset{(2),(3)}{=} \quad \frac{2(n_X \mu_X + n_Y \mu_Y)}{n_X + n_Y} \, (n_X + n_Y) \, \frac{n_X \mu_X + n_Y \mu_Y}{n_X + n_Y} \\
& = \quad \frac{2(n_X \mu_X + n_Y \mu_Y)^2}{n_X + n_Y}.
\end{aligned} \tag{4}$$

b. Give an interpretation for Ward's metric. What do you think it is trying to achieve?

***Hint*: The simplified formula from above will be helpful to answer this part.**

Solution:

With agglomerative hierarchical clustering, the sum of squares starts out at zero (because every point is in its own cluster) and then grows as we merge clusters. Ward's metric aims to keep this growth in sum of squares as small as possible. This is nice if we believe that the sum of squares (as a measure of cluster coherence) should be small.

Notice that both n_X and n_Y (in fact, [half of] their *harmonic mean*) show up in $\Delta(X, Y)$ as computed at part a. The *intuition* is that given two pairs of clusters whose centers are equally far apart, Ward's method will prefer to merge the smaller ones.

c. Assume that you are given two *pairs* of clusters P_1 and P_2. The centers of the two clusters in P_1 are farther apart than the centers of the two clusters in P_2. Using Ward's metric, does agglomerative clustering always choose to merge the two clusters in P_2 (those with less 'distance' between their centers)? Why (not)? Justify your answer with a simple example.

Solution:

No, not always. Which pair will be merged also depends on the size of the clusters. One simple counter example is where the size of the clusters in P_1 are 1 and 99, respectively; and similarly 50 and 50 for P_2.

Then the harmonic mean $\frac{2n_X n_Y}{n_X + n_Y}$ of cluster sizes is $2 \cdot 0.99$ for P_1 and $2 \cdot 25$ for P_2.

If the distance between the centers of the clusters in P_1 is less than 25/0.99 = 25.(25) times the distance between the centers of the clusters in P_2, then $\Delta(P_1)$ will be still smaller and so the clusters in P_1 are merged.

d. In clustering it is usually not trivial to decide what is the right number of clusters the data falls into. Using Ward's metric for agglomerative clustering, can you come up with a simple heuristic to pick the number of clusters k?

Solution:

Ward's algorithm can give us a hint to pick a reasonable k through the merging cost. If the cost of merging increases a lot, it is probably going too far, and losing a lot of structure.

So one possible *heuristic* is to keep reducing k until the cost jumps, and then use the k right before the jump. In other words, pick the k just before the merging cost takes off.

Exemplifying non-hierarchical clustering using the K-means algorithm

T.U. Dresden, 2006 summer, Steffen Höldobler, Axel Grossmann, HW3

Folosiţi algoritmul K-means şi distanţa euclidiană pentru a grupa următoarele 8 instanţe din $\mathbb{R}^2$ în 3 clustere:

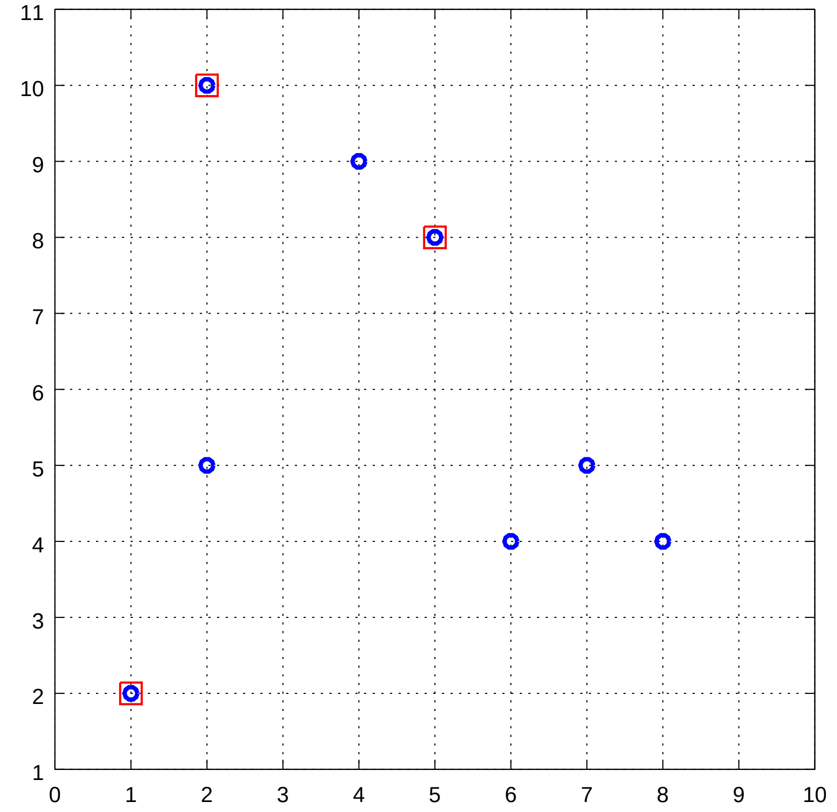
$$A(2,10),\ B(2,5),\ C(8,4),\ D(5,8),\ E(7,5),\ F(6,4),\ G(1,2),\ H(4,9).$$

Se vor lua drept centroizi iniţiali punctele A, D şi G.

a. Rulaţi prima iteraţie a algoritmului K-means. Pe un grid de valori 10×10 veţi marca instanţele date, poziţiile centroizilor la începutul primei iteraţii şi componenţa fiecărui cluster la finalul acestei iteraţii. (Trasaţi mediatoarele segmentelor determinate de centroizi, ca separatori ai clusterelor.)

b. Câte iteraţii sunt necesare pentru ca algoritmul K-means să conveargă? Desenaţi pe câte un grid rezultatul rulării fiecărei iteraţii.

ALGORITMUL de clusterizare K-MEANS - Datele initiale



Solution:

***Iteration 0*:**

$$\left.\begin{array}{l}\mu_1^0 = (2,\ 10)\\ \mu_2^0 = (5,\ 8)\\ \mu_3^0 = (1,\ 2)\end{array}\right\} \Rightarrow$$

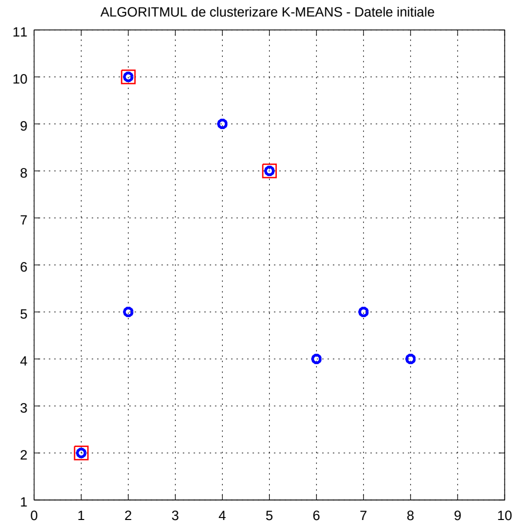
P_i	$d(\mu_1^0, P_i)$	$d(\mu_2^0, P_i)$	$d(\mu_3^0, P_i)$
$A(2,10)$	0	$\ldots$	$\ldots$
$B(2,5)$	5	$3\sqrt{2}$	$\sqrt{10}$
$C(8,4)$			
$D(5,8)$	$\ldots$	0	$\ldots$
$E(7,5)$			
$F(6,4)$			
$G(1,2)$	$\ldots$	$\ldots$	0
$H(4,9)$			

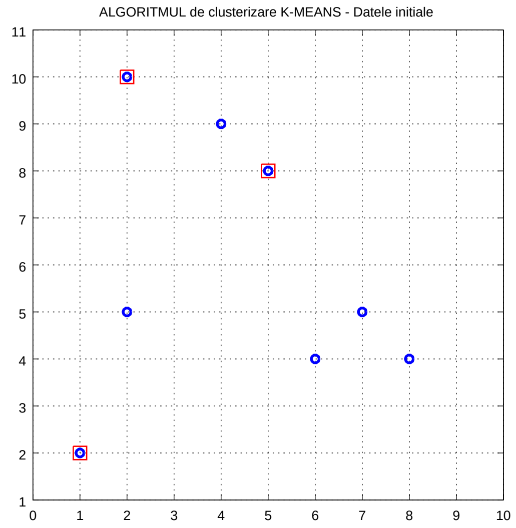
$$\Rightarrow \left\{\begin{array}{l} C_1^0 = \{A\}\\ C_2^0 = \{C, D, E, F, H\}\\ C_3^0 = \{B, G\}\end{array}\right.$$

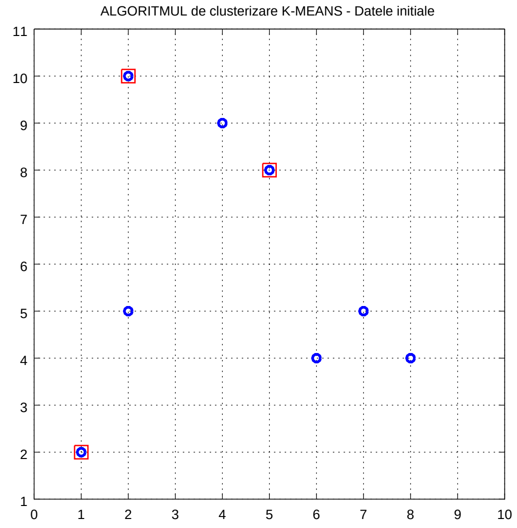
***Iteration 1*:**

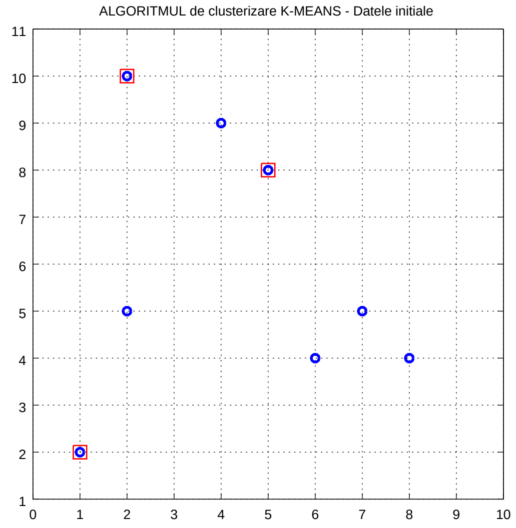
$$\left.\begin{array}{l}\mu_1^1 = \mu_1^0 = (2, 10)\\ \mu_2^1 = \left(\dfrac{4+5+6+7+8}{5}, \dfrac{4+4+5+8+9}{5}\right) = (6,\ 6)\\ \mu_3^1 = \left(\dfrac{2+1}{2}, \dfrac{5+2}{2}\right) = (1.5,\ 3.5)\end{array}\right\} \Rightarrow \ldots \Rightarrow \left\{\begin{array}{l} C_1^1 = \{A, H\}\\ C_2^1 = \{C, D, E, F\}\\ C_3^1 = \{B, G\}\end{array}\right.$$

To be used in class







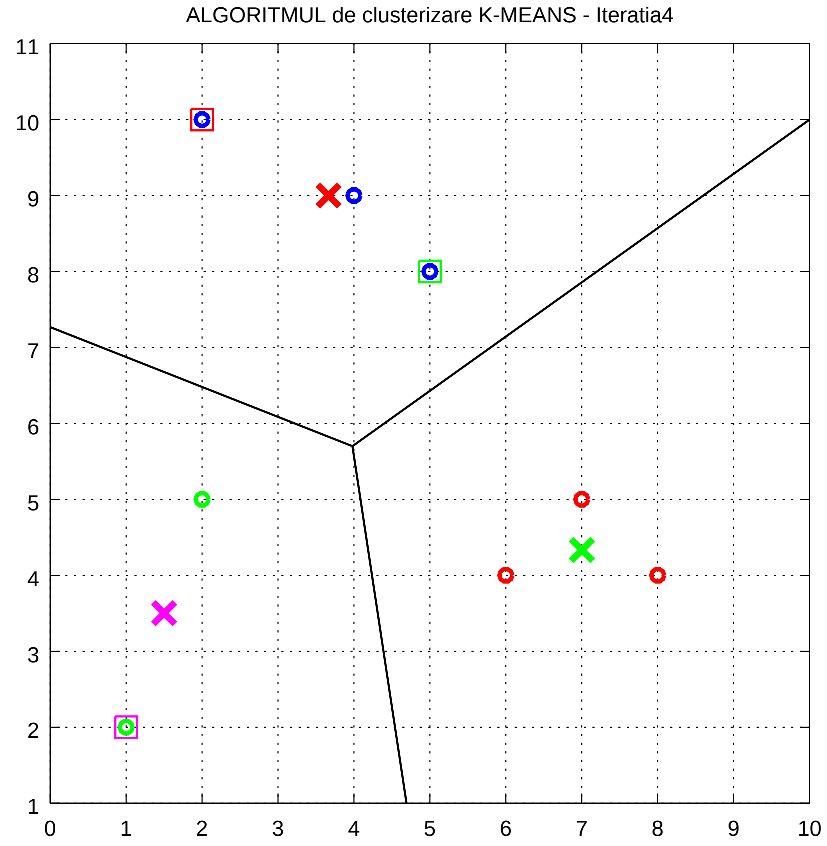


***Iteration 2*:**

$$\left.\begin{array}{l} \mu_1^2 = (3,\ 9.5) \\ \mu_2^2 = \left(\dfrac{26}{4}, \dfrac{21}{4}\right) = (6.5,\ 5.25) \\ \mu_3^2 = \mu_3^1 = (1.5,\ 3.5) \end{array}\right\} \Rightarrow \ldots \Rightarrow \left\{\begin{array}{l} C_1^2 = \{A, D, H\} \\ C_2^2 = \{C, E, F\} \\ C_3^2 = \{B, G\} \end{array}\right.$$

***Iteration 3*:**

$$\left.\begin{array}{l} \mu_1^3 = \left(\dfrac{2+4+5}{3}, \dfrac{8+9+10}{3}\right) = (11/3, 9) \\ \mu_2^3 = (7,\ 13/3) \\ \mu_3^3 = \mu_3^2 = (1.5,\ 3.5) \end{array}\right\} \Rightarrow \ldots \Rightarrow \left\{\begin{array}{l} C_1^3 = \{A, D, H\} = C_1^2 \\ C_2^3 = \{C, E, F\} = C_2^2 \\ C_3^3 = \{B, G\} = C_3^2 \end{array}\right\} \Rightarrow \textbf{\textit{Stop}}$$


ALGORITMUL de clusterizare K-MEANS - Iteratia4
1
2
3
4
5
6
7
8
9
10
11
0
1
2
3
4
5
6
7
8
9
10

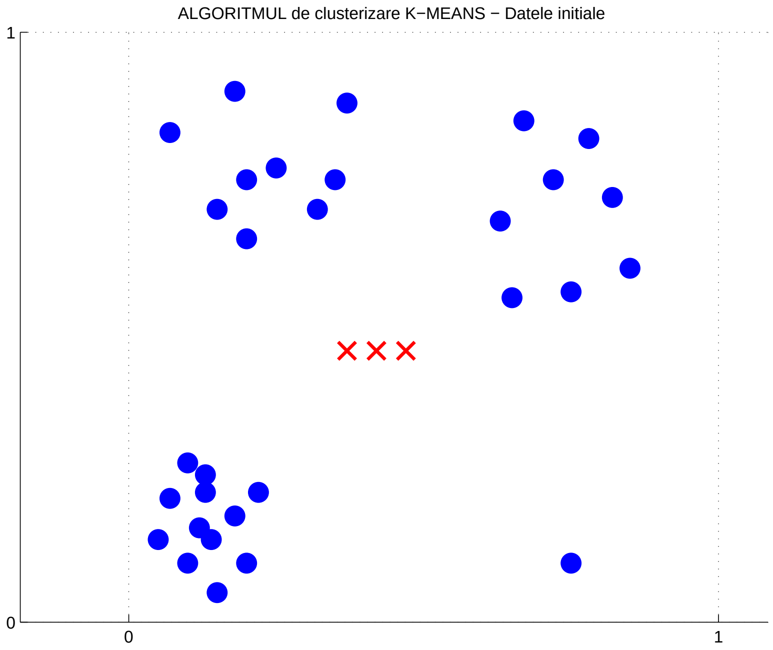
Exemplifying one property of K-means:

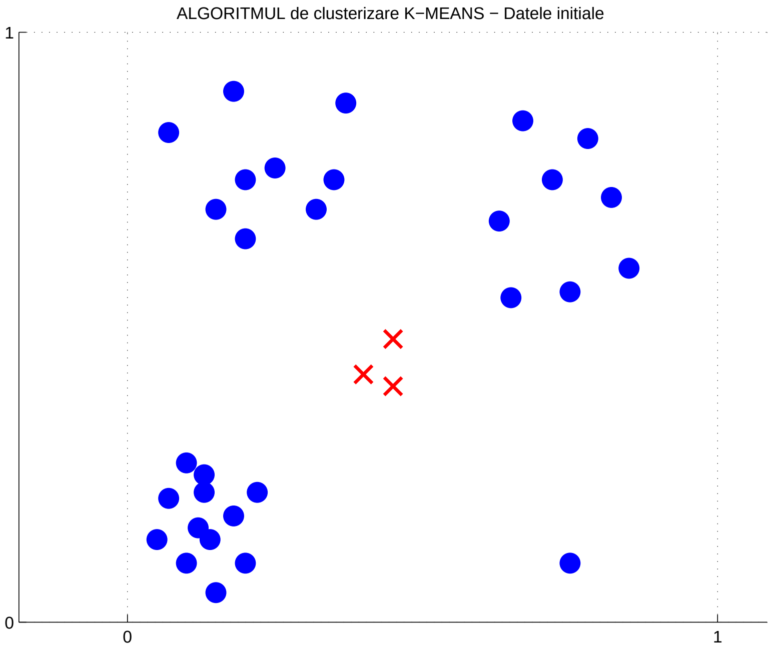
The clusterisation result depends on initialisation

CMU, 2006 spring, Carlos Guestrin, HW5, pr. 1

a-b. Consider the data set in the following figures. The • symbols indicate data points, while the crosses (×) indicate the current cluster centers. For each one of these figures, show the progress of the K-means algorithm by showing how the class centers move with each iteration until convergence. For each iteration, indicate which data points will be associated with each of the clusters, as well as the updated class centers. If during the cluster update step, a cluster center has no points associated with it, it will not move. Use/produce as many figures you need until convergence of the algorithm.

c. What does this imply about the behavior of the K-Means algorithm?

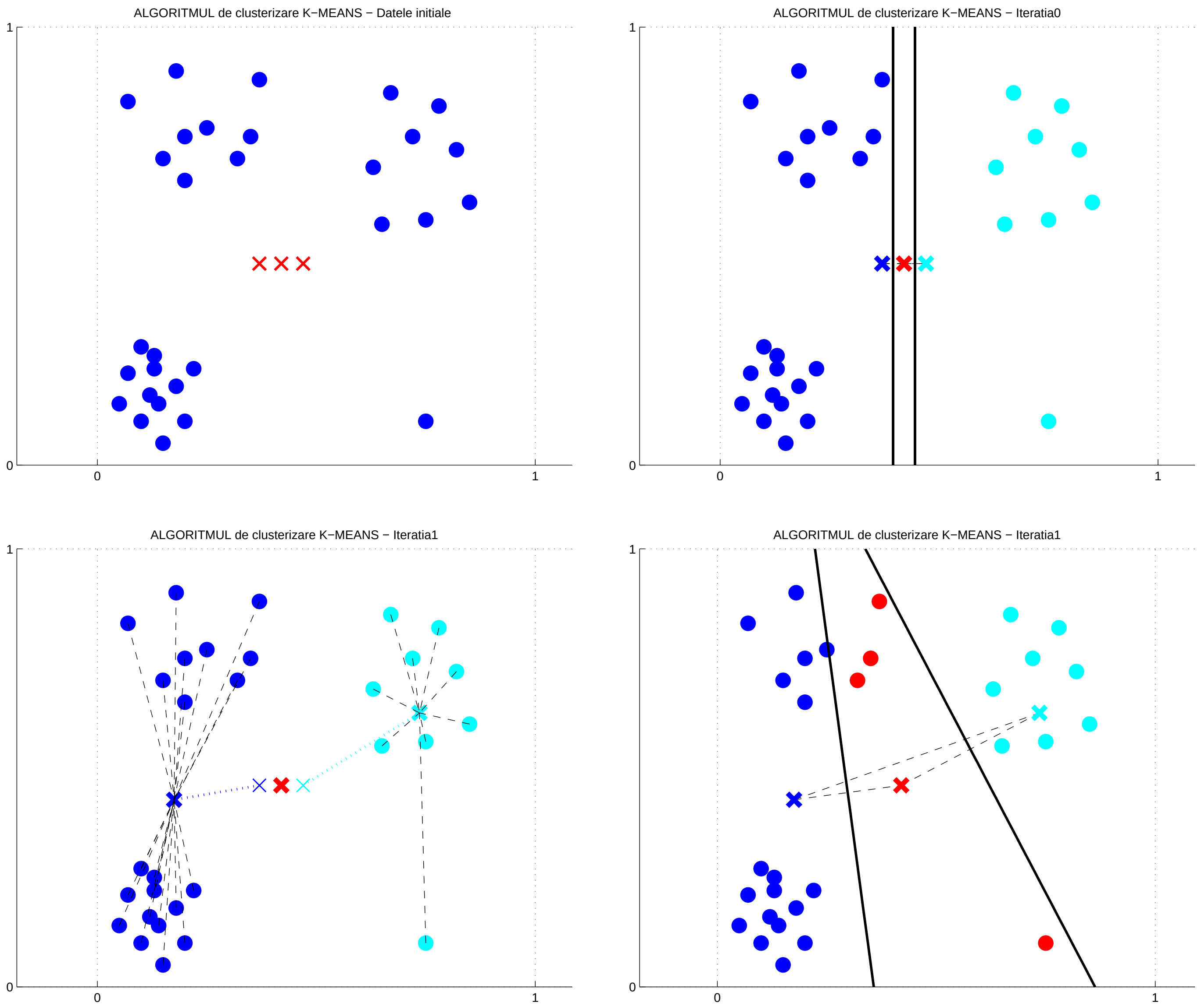

ALGORITMUL de clusterizare K−MEANS − Datele initiale
1
0
0
1


ALGORITMUL de clusterizare K−MEANS − Datele initiale
1
0
0
1

31.

Solution:

a.

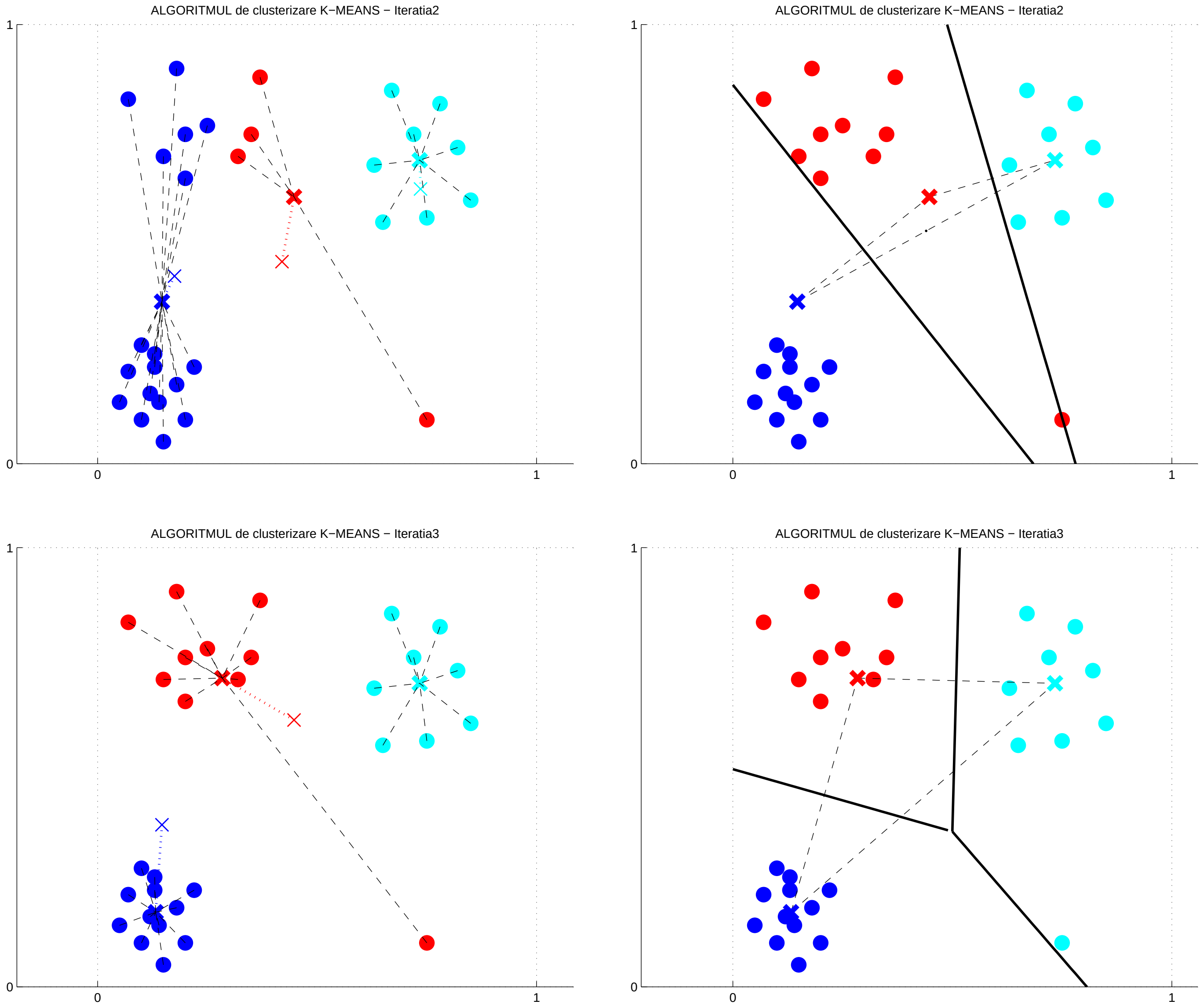


ALGORITMUL de clusterizare K−MEANS − Datele initiale

ALGORITMUL de clusterizare K−MEANS − Iteratia0

ALGORITMUL de clusterizare K−MEANS − Iteratia1

ALGORITMUL de clusterizare K−MEANS − Iteratia1

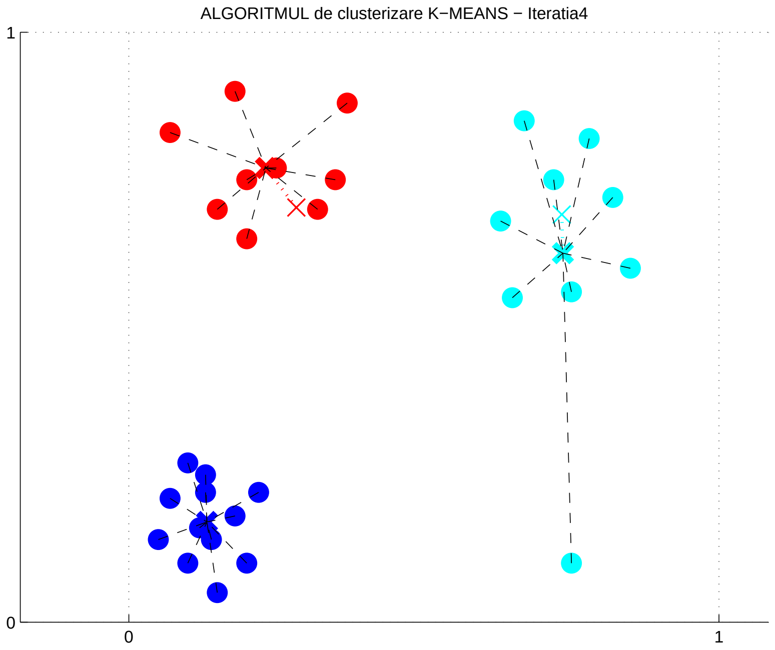


ALGORITMUL de clusterizare K−MEANS − Iteratia2

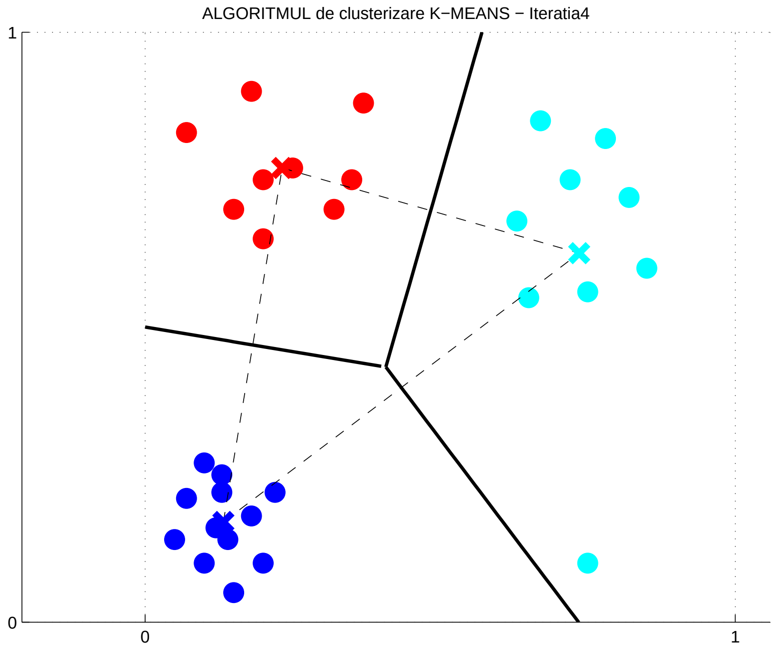
ALGORITMUL de clusterizare K−MEANS − Iteratia2

ALGORITMUL de clusterizare K−MEANS − Iteratia3

ALGORITMUL de clusterizare K−MEANS − Iteratia3



ALGORITMUL de clusterizare K−MEANS − Iteratia4

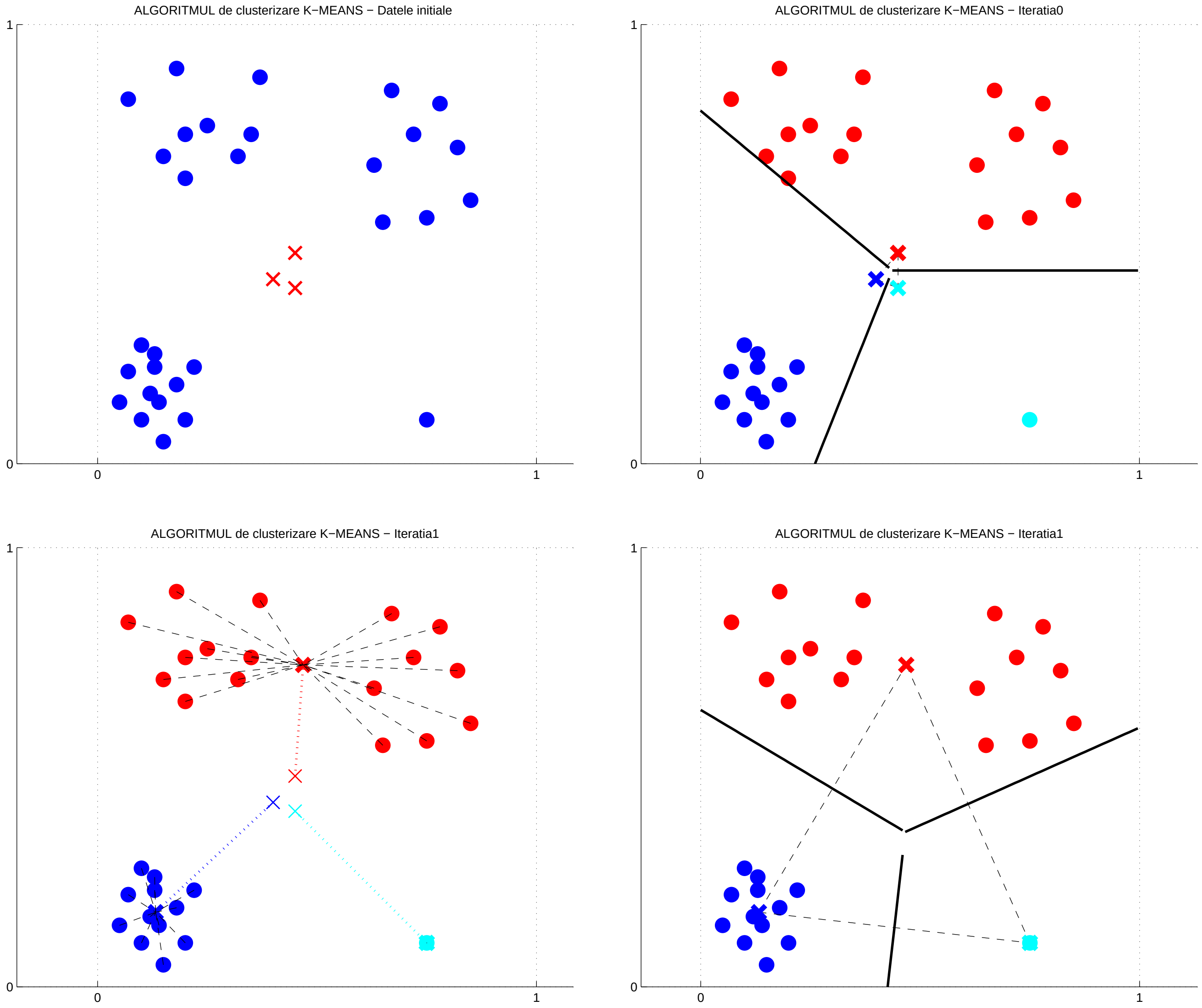


ALGORITMUL de clusterizare K−MEANS − Iteratia4

34.

Solution:

b.



ALGORITMUL de clusterizare K−MEANS − Datele initiale

ALGORITMUL de clusterizare K−MEANS − Iteratia0

ALGORITMUL de clusterizare K−MEANS − Iteratia1

ALGORITMUL de clusterizare K−MEANS − Iteratia1

Solution [in Romanian]: c.

Este evident din acest exerciţiu că rezultatul algoritmului K-means depinde de poziţionarea iniţială a centroizilor. În cazul iniţializării de la punctul a au fost necesare 4 iteraţii până a se ajunge la convergenţă, pe când la punctul b algoritmul a convers după doar o iteraţie.

Mai este încă ceva important *de remarcat*: faptul că la prima variantă de iniţializare, punctul din dreapta jos, care este un *outlier* (rom., excepţie, caz particular, aberaţie) este până la urmă asociat clusterului format de punctele din partea dreaptă (sus), în vreme ce la cea de-a doua variantă de iniţializare el constituie un cluster aparte/“singleton”, obligând în mod indirect grupările de puncte din stânga-sus şi dreapta-sus să formeze împreună un singur cluster (ccea ce, foarte probabil, nu era de dorit). Detecţia outlier-elor este un capitol important din învăţarea automată. Ignorarea lor poate conduce la rezultate eronate sau chiar aberante ale modelărilor.

Some proofs

K-means as an optimisation algorithm: The monotonicity of the J_K criterion

[CMU, 2009 spring, Ziv Bar-Joseph, HW5, pr. 2.1]

Algoritmul K-means (S. P. Lloyd, 1957)

```
Input: $x_1, \ldots, x_n \in \mathbb{R}^d$, cu $n \geq K$.
Output: o anumită $K$-partiţie pentru $\{x_1, \ldots, x_n\}$.
Procedură:
    [Iniţializare/Iteraţia 0:] $t \leftarrow 0$;
        se fixează în mod arbitrar $\mu_1^0, \ldots, \mu_K^0$, centroizii iniţiali ai clusterelor, şi
        se asignează fiecare instanţă $x_i$ la centroidul cel mai apropiat, formând astfel
        clusterele $C_1^0, \ldots, C_K^0$.
    [Recursivitate:] Se execută iteraţia $++\,t$:
        Pasul 1: se calculează noile poziţii ale centroizilor:[a]
            $\mu_j^t = \frac{1}{|C_j^{t-1}|} \sum_{x_i \in C_j^{t-1}} x_i$ pentru $j = \overline{1, K}$;
        Pasul 2:
            se reasignează fiecare $x_i$ la [clusterul cu] centroidul cel mai apropiat, adică
            se stabileşte noua componenţă a clusterelor la iteraţia $t$: $C_1^t, \ldots, C_K^t$;
    [Terminare:] până când o anumită condiţie este îndeplinită
        (de exemplu: până când poziţiile centroizilor — sau: componenţa clusterelor
        — nu se mai modifică de la o iteraţie la alta).
```

[a] Formula aceasta corespunde folosirii distanţei euclidiene. Pentru alte măsuri de distanţă, este posibil să fie necesare alte formule.

a. Demonstraţi că, de la o iteraţie la alta, algoritmul K-means măreşte *coeziunea de ansamblu* a clusterelor. I.e., considerând funcţia

$$J(C^t, \mu^t) \overset{def.}{=} \sum_{i=1}^{n} ||x_i - \mu^t_{C^t(x_i)}||^2 \overset{def.}{=} \sum_{i=1}^{n} \left(x_i - \mu^t_{C^t(x_i)}\right) \cdot \left(x_i - \mu^t_{C^t(x_i)}\right),$$

unde:

$C^t = (C^t_1, C^t_2, \ldots, C^t_K)$ **este colecţia de clustere (i.e., K-partiţia) la momentul t,**

$\mu^t = (\mu^t_1, \mu^t_2, \ldots, \mu^t_K)$ **este colecţia de centroizi ai clusterelor (K-configuraţia) la momentul t,**

$C^t(x_i)$ **desemnează clusterul la care este asignat elementul x_i la iteraţia t,**

operatorul $\cdot$ desemnează produsul scalar al vectorilor din $\mathbb{R}^d$,

arătaţi că $J(C^t, \mu^t) \geq J(C^{t+1}, \mu^{t+1})$ pentru orice t.

Ideea demonstraţiei

Inegalitatea de mai sus rezultă din două inegalităţi (care corespund paşilor 1 şi 2 de la iteraţia t):

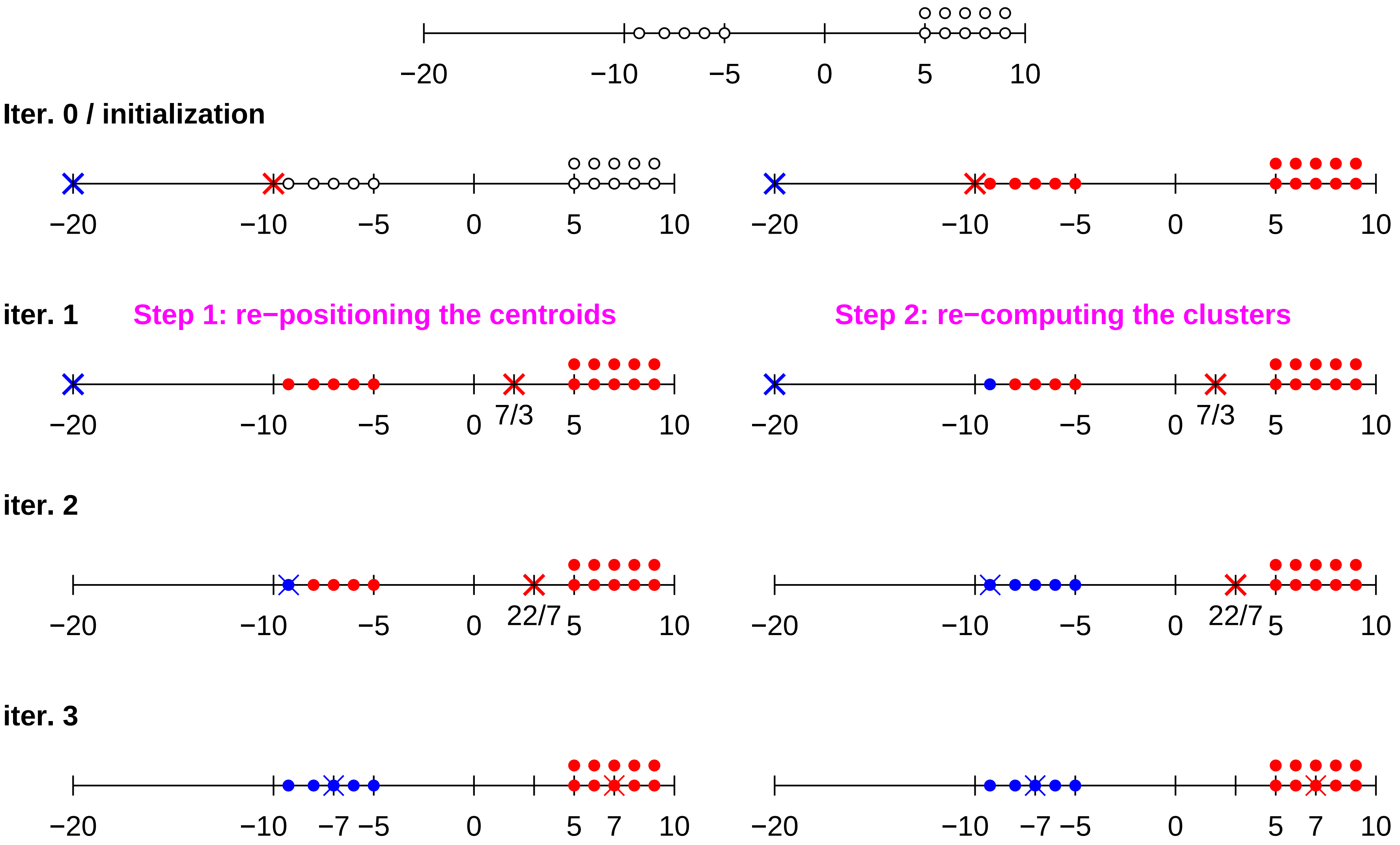
$$J(C^t, \mu^t) \overset{(1)}{\geq} J(C^t, \mu^{t+1}) \overset{(2)}{\geq} J(C^{t+1}, \mu^{t+1})$$

La prima inegalitate (cea corespunzătoare pasului 1) se poate considera că parametrul C^t este fixat iar μ este variabil, în vreme ce la a doua inegalitate (cea corespunzătoare pasului 2) se consideră μ^t fixat şi C variabil.

Prima inegalitate se poate obţine însumând o serie de inegalităţi, şi anume câte una pentru fiecare cluster C_j^t. A doua inegalitate se demonstrează imediat.

Ilustrarea acestei idei, pe un exemplu particular:

Vezi următoarele 3 slide-uri
[Edinburgh, 2009 fall, C. Williams, V. Lavrenko, HW4, pr. 3]



Iter. 0 / initialization

iter. 1

Step 1: re−positioning the centroids

Step 2: re−computing the clusters

iter. 2

iter. 3

Pentru acest exemplu de aplicare a algoritmului K-means, scriem expresiile numerice pentru valoarea criteriului $J_2(C^t, \mu^t)$ pentru fiecare iteraţie ($t = 0, 1, 2, 3$).

iter.	$J_2(C^t, \mu^t)$	
0.	$0 + \{(-9-(-10))^2 + \ldots + (-5-(-10))^2$ $+2[(5-(-10))^2 + \ldots + (9-(-10))^2]\}$	$\geq$
1.	$(-9-(-20))^2 + \{(-8-7/3))^2 + \ldots (-5-7/3)^2$ $+2[(5-7/3)^2 + \ldots + (9-7/3)^2]\}$	$\geq$
2.	$(-9-(-9))^2 + \ldots (-5-(-9))^2$ $+2[(5-22/7)^2 + \ldots + (9-22/7)^2]$	$\geq$
3.	$(-9-(-7))^2 + \ldots (-5-(-7))^2$ $+2[(5-7)^2 + \ldots + (9-7)^2]$	

***Observaţie*: La prima vedere, este greu să dovedim aceste inegalităţi ($J_2(C^{t-1}, \mu^{t-1}) \geq J_2(C^t, \mu^t)$, pentru $t = 1, 2, 3$) ...altfel decât calculând efectiv valoarea expresiilor care se compară. Însă, introducând nişte termeni intermediari, inegalităţile acestea se vor demonstra într-un mod foarte elegant...**

iter.	$J_2(C^{t-1}, \mu^t)$	$J_2(C^t, \mu^t)$
0.		$0 + \{(-9-(-10))^2 + \ldots + (-5-(-10))^2$ $+2[(5-(-10))^2 + \ldots + (9-(-10))^2]\}$ $\geq$
1.	$0 + \{(-9-7/3)^2 + (-8-7/3)^2 + \ldots + (-5-7/3))^2$ $+2[(5-7/3)^2 + \ldots + (9-7/3)^2]\}$ $\geq$	$(-9-(-20))^2 + \{(-8-7/3))^2 + \ldots + (-5-7/3)^2$ $+2[(5-7/3)^2 + \ldots + (9-7/3)^2]\}$ $\geq$
2.	$(-9-(-9))^2 + \{(-8-22/7))^2 + \ldots + (-5-22/7)^2$ $+2[(5-22/7)^2 + \ldots + (9-22/7)^2]\}$ $\geq$	$(-9-(-9))^2 + (-8-(-9))^2 + \ldots + (-5-(-9))^2$ $+2[(5-22/7)^2 + \ldots + (9-22/7)^2]$ $\geq$
3.	$(-9-(-7))^2 + \ldots + (-5-(-7))^2$ $+2[(5-7)^2 + \ldots + (9-7)^2]$ $=$	$(-9-(-7))^2 + \ldots + (-5-(-7))^2$ $+2[(5-7)^2 + \ldots + (9-7)^2]$

Explicaţii:

1. Inegalităţile pe orizontală ($J_2(C^{t-1}, \mu^t) \geq J_2(C^t, \mu^t)$, pentru $t = 1, 2, 3$) sunt uşor de demonstrat, pe baza corespondenţei termen cu termen. (Ele corespund eventualelor micşorări ale distanţelor atunci când se face reasignarea instanţelor la centroizi.)

2. Restul inegalităţilor ($J_2(C^t, \mu^t) \geq J_2(C^t, \mu^{t+1})$, pentru $t = 1, 2, 3$) se rezolvă printr-o metodă de optimizare simplă. De exemplu, pentru $t = 1$ este imediat că funcţia $(-9-x)^2 + (-8-x)^2 + \ldots + (-5-x)^2 + 2[(5-x)^2 + \ldots + (9-x)^2]$ îşi atinge minimul pentru $x = 7/3$, deci $J_2(C^1, \mu^2) \geq J_2(C^1, \mu^1)$.

Step by step $(t = 1)$

iter.	$J_2(C^{t-1}, \mu^t)$	$J_2(C^t, \mu^t)$
0.		$0 + \{(-9-(-10))^2 + \ldots + (-5-(-10))^2$ $+2[(5-(-10))^2 + \ldots + (9-(-10))^2]\} \quad \geq$
1.	$0 + \{(-9-7/3)^2 + (-8-7/3)^2 + \ldots + (-5-7/3))^2$ $+2[(5-7/3)^2 + \ldots + (9-7/3)^2]\} \quad \geq$	$(-9-(-20))^2 + \{(-8-7/3))^2 + \ldots + (-5-7/3)^2$ $+2[(5-7/3)^2 + \ldots + (9-7/3)^2]\} \quad \geq$

Details to be filled in class

Step by step ($t = 2$)

iter.	$J_2(C^{t-1}, \mu^t)$		$J_2(C^t, \mu^t)$	
1.			$(-9-(-20))^2 + \{(-8-7/3))^2 + \ldots + (-5-7/3)^2$ $+2[(5-7/3)^2 + \ldots + (9-7/3)^2]\}$	$\geq$
2.	$(-9-(-9))^2 + \{(-8-22/7))^2 + \ldots + (-5-22/7)^2$ $+2[(5-22/7)^2 + \ldots + (9-22/7)^2]\}$	$\geq$	$(-9-(-9))^2 + (-8-(-9))^2 + \ldots + (-5-(-9))^2$ $+2[(5-22/7)^2 + \ldots + (9-22/7)^2]$	$\geq$

Details to be filled
in class / at home

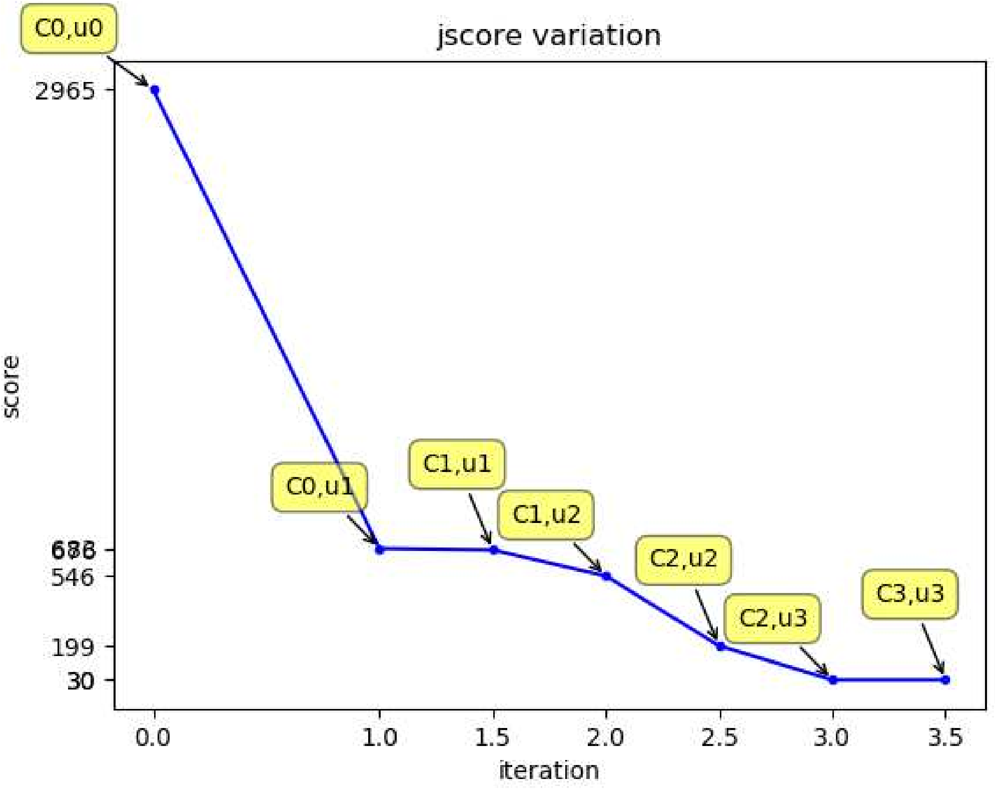
Step by step ($t = 3$)

iter.	$J_2(C^{t-1}, \mu^t)$		$J_2(C^t, \mu^t)$	
2.			$(-9-(-9))^2 + (-8-(-9))^2 + \ldots + (-5-(-9))^2$ $+2[(5-22/7)^2 + \ldots + (9-22/7)^2]$	$\geq$
3.	$(-9-(-7))^2 + \ldots + (-5-(-7))^2$ $+2[(5-7)^2 + \ldots + (9-7)^2]$	$=$	$(-9-(-7))^2 + \ldots + (-5-(-7))^2$ $+2[(5-7)^2 + \ldots + (9-7)^2]$	

**Details to be filled
in class / at home**

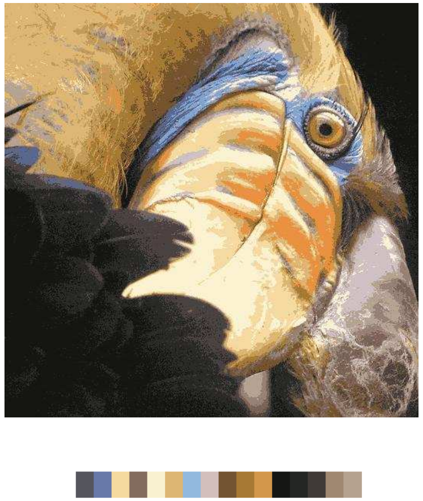
Graphical illustration

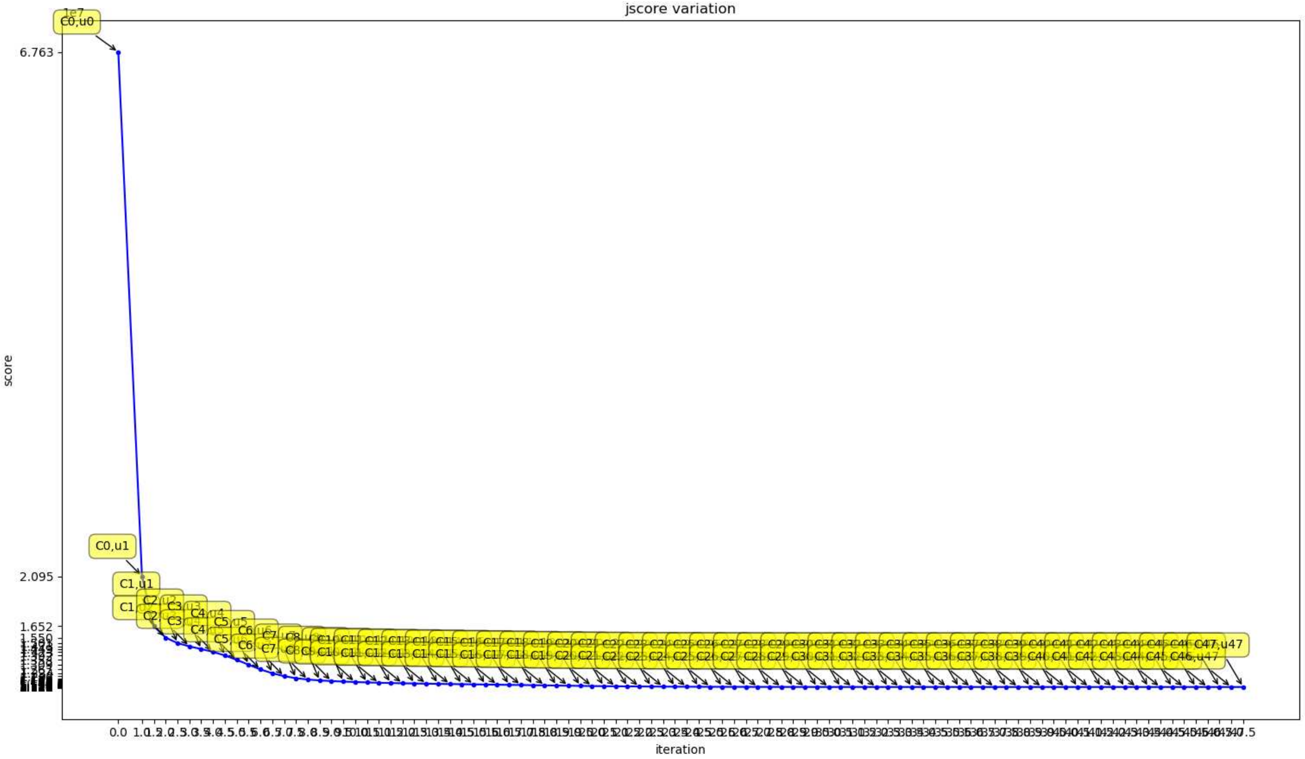
by Gheorghe Balan, FII student, 2017 fall



For Edinburgh, 2009 fall, C. Williams, V. Lavrenko, HW4, pr. 3

Graphical exemplification (cont’d)





© 2010-2012 Andrew Ng, Stanford University.

For Stanford, A. Ng, 2012 spring, HW9 (image segmentation and compression)

Demonstraţie, pentru cazul general

Observaţie: **Pentru convenienţă, ne vom limita la cazul $d = 1$. Extinderea demonstraţiei la cazul $d > 1$ nu comportă dificultăţi.**

Demonstrarea inegalităţii (1): $J(C^t, \mu^t) \geq J(C^t, \mu^{t+1})$

(Vezi pasul 1 al iteraţiei t.)

Fixăm $j \in \{1, \ldots, K\}$. Dacă notăm cu $C_j^t = \{x_{i_1}, x_{i_2}, \ldots, x_{i_l}\}$, unde $l \overset{not.}{=} |C_j^t|$, atunci

$$J(C_j^t, \mu_j^t) = \sum_{p=1}^{l} \left(x_{i_p} - \mu_j^t\right)^2, \text{ deci } J(C^t, \mu^t) = \sum_{j=1}^{K} J(C_j^t, \mu_j^t).$$

Dacă se consideră C_j^t fixat, iar μ_j^t variabil, atunci putem minimiza imediat funcţia

$$f(\mu) \overset{def.}{=} J(C_j^t, \mu) = l\mu^2 - 2\mu \sum_{p=1}^{l} x_{i_p} + \sum_{p=1}^{l} x_{i_p}^2 \Rightarrow \arg\min_{\mu} J(C_j^t, \mu) = \frac{1}{l} \sum_{p=1}^{l} x_{i_p} \overset{def.}{=} \mu_j^{t+1}.$$

Aşadar, $J(C_j^t, \mu) \geq J(C_j^t, \mu_j^{t+1})$, pentru $\forall \mu$. În particular, pentru $\mu = \mu_j^t$ vom avea: $J(C_j^t, \mu_j^t) \geq J(C_j^t, \mu_j^{t+1})$. Inegalitatea aceasta este valabilă pentru toate clusterele $j = 1, \ldots, K$. Dacă sumăm toate aceste inegalităţi, rezultă: $J(C^t, \mu^t) \geq J(C^t, \mu^{t+1})$.

Demonstrarea inegalităţii (2): $J(C^t, \mu^{t+1}) \geq J(C^{t+1}, \mu^{t+1})$

(Vezi pasul 2 al iteraţiei t.)

La acest pas, o instanţă oarecare x_i, unde $i \in \{1, \ldots, n\}$, este reasignată de la clusterul cu centroidul μ_j^{t+1}, la un alt centroid μ_q^{t+1}, dacă

$||x_i - \mu_{j'}^{t+1}||^2 \geq ||x_i - \mu_q^{t+1}||^2 \Leftrightarrow (x_i - \mu_{j'}^{t+1})^2 \geq (x_i - \mu_q^{t+1})^2,$ **pentru orice** $j' = 1, \ldots, K.$

În contextul iteraţiei t, acest lucru implică

$$\left(x_i - \mu_{C^t(x_i)}^{t+1}\right)^2 \geq \left(x_i - \mu_{C^{t+1}(x_i)}^{t+1}\right)^2.$$

Sumând membru cu membru inegalităţile de acest tip obţinute pentru $i = \overline{1, n}$, rezultă: $J(C^t, \mu^{t+1}) \geq J(C^{t+1}, \mu^{t+1})$, ceea ce era de demonstrat.

b. Ce puteţi spune despre oprirea algoritmului K-means?

Termină oare acest algoritm într-un număr finit de paşi, sau (întrucât numărul de K-partiţii care pot fi formate cu cele n instanţe de antrenament, $x_1, \ldots, x_n$, este finit, şi anume K^n) este posibil ca el să reviziteze de o infinitate de ori o K-configuraţie anterioară, $\mu = (\mu_1, \ldots, \mu_K)$?)

Răspuns:

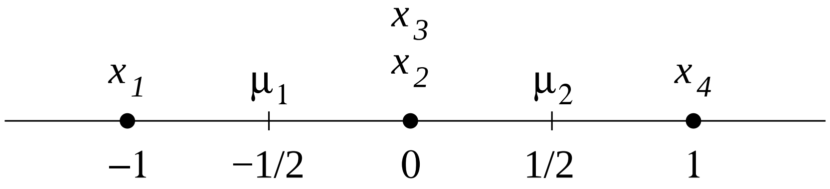
Dacă algoritmul revizitează o K-partiţie, atunci rezultă că pentru un anumit t avem $J(C^{t-1}, \mu^t) = J(C^t, \mu^{t+1})$. Este posibil ca acest fapt să se întâmple, şi anume atunci când:

– există instanţe multiple (i.e., $x_i = x_j$, deşi $i \neq j$),

– criteriul de oprire al algoritmului K-means este de forma "până când componenţa clusterelor nu se mai modifică",

– se presupune că, în cazul în care o instanţă x_i este situată la egală distanţă faţă de doi sau mai mulţi centroizi, ea poate fi asignată în mod aleatoriu la oricare dintre ei.

Aşa se întâmplă în *exemplul* din figura alăturată dacă se consideră că la o iteraţie t avem $x_2 = 0 \in C_1^t$ şi $x_3 = 0 \in C_2^t$, iar la iteraţia următoare alegem ca $x_3 = 0 \in C_1^{t+1}$ şi $x_2 = 0 \in C_2^{t+1}$ şi, din nou, invers la iteraţia $t + 2$.



Observaţii

- **Dacă se păstrează criteriul dat ca exemplu în enunţul problemei – adică se iterează până când centroizii "staţionează" – algoritmul se poate opri fără ca la ultima iteraţie $J(C, \mu)$ să fi atins minimul posibil. În cazul exemplului de mai sus, vom avea** $\frac{1}{4} + 2 \cdot \frac{1}{4} + \frac{1}{4} = 1 > \frac{2}{3} = 2 \cdot \left(\frac{1}{3}\right)^2 + \left(\frac{2}{3}\right)^2$**.**

- **Dacă nu există instanţe multiple care să fie situate la distanţe egale faţă de doi sau mai mulţi centroizi la o iteraţie oarecare a algoritmului K-means (precum sunt x_2 şi x_3 în *exemplul* de mai sus), sau dacă se impune *restricţia* ca în astfel de situaţii instanţele identice să fie asignate la un singur cluster, este evident că algoritmul K-means se opreşte într-un număr finit de paşi.**

Concluzii

- **Algoritmul K-means explorează — pornind de la o anumită iniţializare a celor K centroizi —, doar un subset din totatul de K^n K-partiţii, asigurându-ne însă că are loc proprietatea $J(C^0, \mu^1) \geq J(C^1, \mu^2) \geq \ldots \geq J(C^{t-1}, \mu^t) \geq J(C^t, \mu^{t+1})$, conform punctului a al acestei probleme.**

- **Atingerea minimului global al funcţiei $J(C, \mu)$ — unde C este o variabilă care parcurge mulţimea tuturor K-partiţiilor care se pot forma cu instanţele $\{x_1, \ldots, x_n\}$ — nu este garantată pentru algoritmul K-means. Valoarea funcţiei J care se obţine la oprirea algoritmului K-means este dependentă de plasarea iniţială a centroizilor μ, precum şi de modul concret în care sunt alcătuite clusterele în cazul în care o instanţă oarecare se află la distanţă egală de doi sau mai mulţi centroizi, după cum am arătat în exemplul de mai sus.**

K-means algorithm: The “approximate” maximization of the “distance” between clusters

[CMU, 2010 fall, Aarti Singh, HW3, pr. 5.2]

Note: In this problem we will work with a version of the K-means algorithm which is slightly modified w.r.t. the one given in the problem CMU, 2009 spring, Ziv Bar-Joseph, HW5, pr. 2.1, where we have proved the monotonicity of the criterion J.

Let $X := \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ be our sample points, and K denote the number of clusters to use. We represent the cluster assignments of the data points by an *indicator matrix* $\gamma \in \{0,1\}^{n \times K}$ such that $\gamma_{ij} = 1$ means $\mathbf{x}_i$ belongs to cluster j. We require that each point belongs to exactly one cluster, so $\sum_{j=1}^{K} \gamma_{ij} = 1$.

[We already know that] **the K-means algorithm "estimates" γ by minimizing the following "cohesion criterion" (or, "measure of distortion", or simply "sum of squares"):**

$$J(\gamma, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_K) \ := \ \sum_{i=1}^{n} \sum_{j=1}^{K} \gamma_{ij} \|\mathbf{x}_i - \boldsymbol{\mu}_j\|^2,$$

where $\|\cdot\|$ denotes the vector 2-norm.

K-means alternates between estimating γ and re-computing $\boldsymbol{\mu}_j$'s.

The K-means algorithm (...yet another version!)

- **Initialize $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_K$, and let $C := \{1, \ldots, K\}$.**
- **While the value of J is still decreasing, repeat the following:**

  1. **Determine γ by**

     $$\gamma_{ij} \leftarrow \begin{cases} 1, & \|\mathbf{x}_i - \boldsymbol{\mu}_j\|^2 \leq \|\mathbf{x}_i - \boldsymbol{\mu}_{j'}\|^2,\ \forall j' \in C, \\ 0, & \textbf{otherwise.} \end{cases}$$

     Break ties arbitrarily.

  2. **Recompute $\boldsymbol{\mu}_j$ using the updated γ:**
     For each $j \in C$, if $\sum_{i=1}^{n} \gamma_{ij} > 0$ set

     $$\boldsymbol{\mu}_j \leftarrow \frac{\sum_{i=1}^{n} \gamma_{ij} \mathbf{x}_i}{\sum_{i=1}^{n} \gamma_{ij}}.$$

     Otherwise, don't change μ_j.

Let $\bar{\mathbf{x}}$ denote the sample mean.
Consider the following three quantities:

$$\textbf{Total variation: } T(X) = \frac{\sum_{i=1}^{n} \|\mathbf{x}_i - \bar{\mathbf{x}}\|^2}{n}$$

$$\textbf{Within-cluster variation: } W_j(X) = \frac{\sum_{i=1}^{n} \gamma_{ij} \|\mathbf{x}_i - \boldsymbol{\mu}_j\|^2}{\sum_{i=1}^{n} \gamma_{ij}} \textbf{ for } j = 1, \ldots, K$$

$$\textbf{Between-cluster variation: } B(X) = \sum_{j=1}^{K} \left(\frac{\sum_{i=1}^{n} \gamma_{ij}}{n} \right) \|\boldsymbol{\mu}_j - \bar{\mathbf{x}}\|^2.$$

What is the relation between these three quantities?

Based on this relation, show that K-means can be interpreted as minimizing *a weighted average of within-cluster variations* while approximately(!) maximizing *the between-cluster variation.* Note that the relation may contain an extra term that does not appear above.

Solution

To simplify the notation, we define $n_j = \sum_{i=1}^{n} \gamma_{ij}$**.**

We then have:

$$T(X) = \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{x}_i - \bar{\mathbf{x}}\|^2 = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{K} \gamma_{ij}\,\|\mathbf{x}_i - \bar{\mathbf{x}}\|^2 = \frac{1}{n}\sum_{j=1}^{K}\sum_{i=1}^{n} \gamma_{ij}\|\mathbf{x}_i - \bar{\mathbf{x}}\|^2$$

$$= \frac{1}{n}\sum_{j=1}^{K}\sum_{i=1}^{n} \gamma_{ij}\|\mathbf{x}_i - \boldsymbol{\mu}_j + \boldsymbol{\mu}_j - \bar{\mathbf{x}}\|^2 \tag{5}$$

$$= \frac{1}{n}\sum_{j=1}^{K}\sum_{i=1}^{n} \gamma_{ij}\left(\|\mathbf{x}_i - \boldsymbol{\mu}_j\|^2 + \|\boldsymbol{\mu}_j - \bar{\mathbf{x}}\|^2 + 2\left(\mathbf{x}_i - \boldsymbol{\mu}_j\right)\cdot\left(\boldsymbol{\mu}_j - \bar{\mathbf{x}}\right)\right)$$

$$= \sum_{j=1}^{K} \frac{n_j}{n}\frac{\sum_{i=1}^{n}\gamma_{ij}\|\mathbf{x}_i - \boldsymbol{\mu}_j\|^2}{n_j} + \sum_{j=1}^{K}\frac{n_j\|\boldsymbol{\mu}_j - \bar{\mathbf{x}}\|^2}{n} + \quad + \frac{2}{n}\sum_{j=1}^{K}\left(\boldsymbol{\mu}_j - \bar{\mathbf{x}}\right)\cdot\sum_{i=1}^{n}\gamma_{ij}\left(\mathbf{x}_i - \boldsymbol{\mu}_j\right)$$

$$= \underbrace{\sum_{j=1}^{K}\frac{n_j}{n}W_j(X)}_{\frac{J}{n}} + B(X) - \frac{2}{n}\sum_{j=1}^{K} n_j\left(\boldsymbol{\mu}_j - \bar{\mathbf{x}}\right)\cdot\left(\boldsymbol{\mu}_j - \bar{\boldsymbol{\mu}}_j\right), \ \textbf{where } \bar{\boldsymbol{\mu}}_j \overset{not.}{=} \frac{\sum_{i=1}^{n}\gamma_{ij}\mathbf{x}_i}{n_j}. \tag{6}$$

Notes

1. **In (5), the quantities $\boldsymbol{\mu}_j$ can be taken arbitrarily; however, they will be thought of in the end — see relation (6) and the Conclusion on the last slide — as the centroids of the clusters $1, \ldots, K$, as computed at Step 2 in the K-means algorithm.**

2. **The equality (6) on the previous slide holds because**

$$\begin{aligned}
\sum_{i=1}^{n} \gamma_{ij}(x_i - \mu_j) &= \left(\sum_{i=1}^{n} \gamma_{ij} x_i\right) - n_j \mu_j = n_j \frac{\sum_{i=1}^{n} \gamma_{ij} x_i}{n_j} - n_j \mu_j \\
&= n_j \left(\frac{\sum_{i=1}^{n} \gamma_{ij} x_i}{n_j} - \mu_j\right) = n_j \left(\bar{\mu}_j - \mu_j\right)
\end{aligned}$$

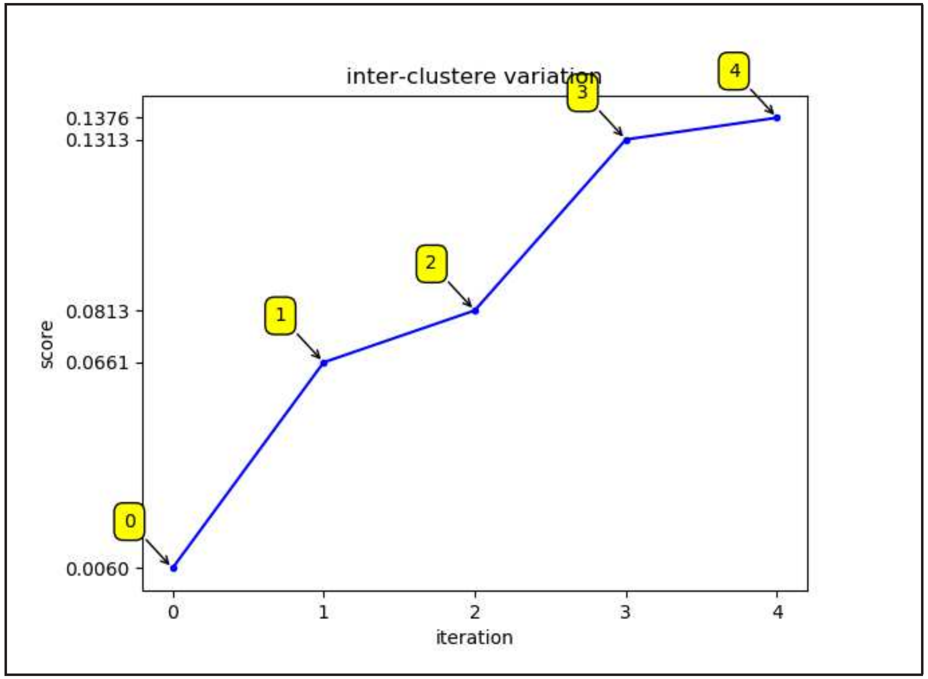
Conclusion

We already know — see CMU, 2009 spring, Ziv Bar-Joseph, HW5, pr. 2.1 — that K-means aims to minimize J, and consequently $\frac{1}{n}J$, which coincides with the first term in the expression we obtained for $T(X)$, namely $\sum_{j=1}^{K} \frac{n_j}{n} W_j(X)$.

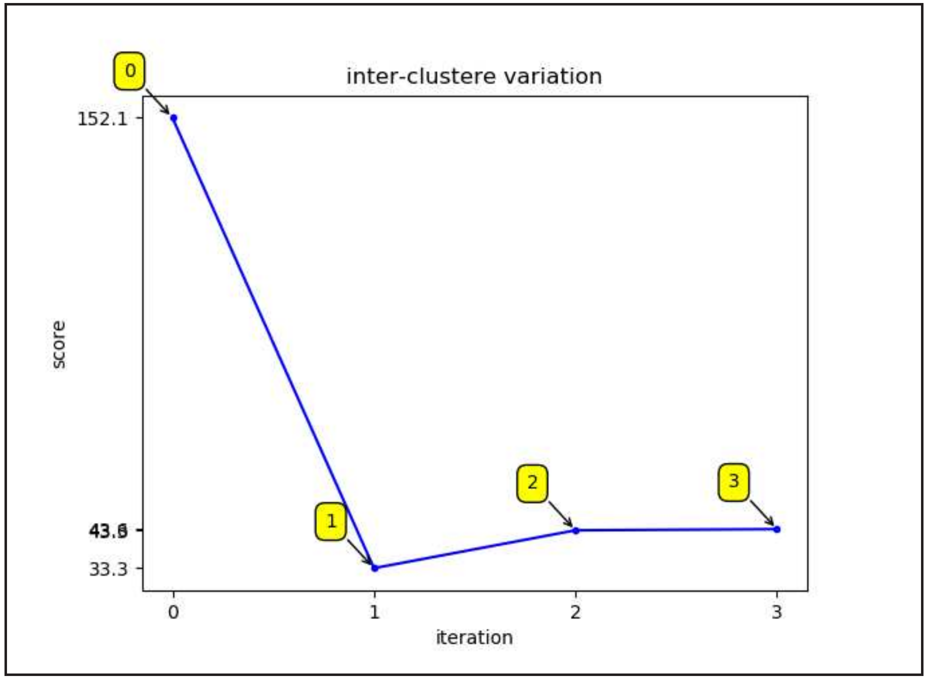
Since the total variation $T(X)$ is constant, minimizing the first term is equivalent to maximizing the sum of the other two terms, which is expected to be dominated by the between-cluster variation $B(X)$ since a good $\boldsymbol{\mu}_j$ should be close to $\bar{\boldsymbol{\mu}}_j$, making the third term small in absolute value.

Graphical exemplification

by Gheorghe Balan, FII student, 2017 fall

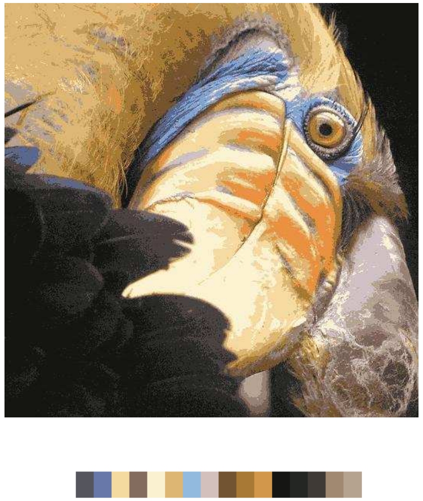


For CMU, 2006 spring, C. Guestrin, HW5, pr. 1

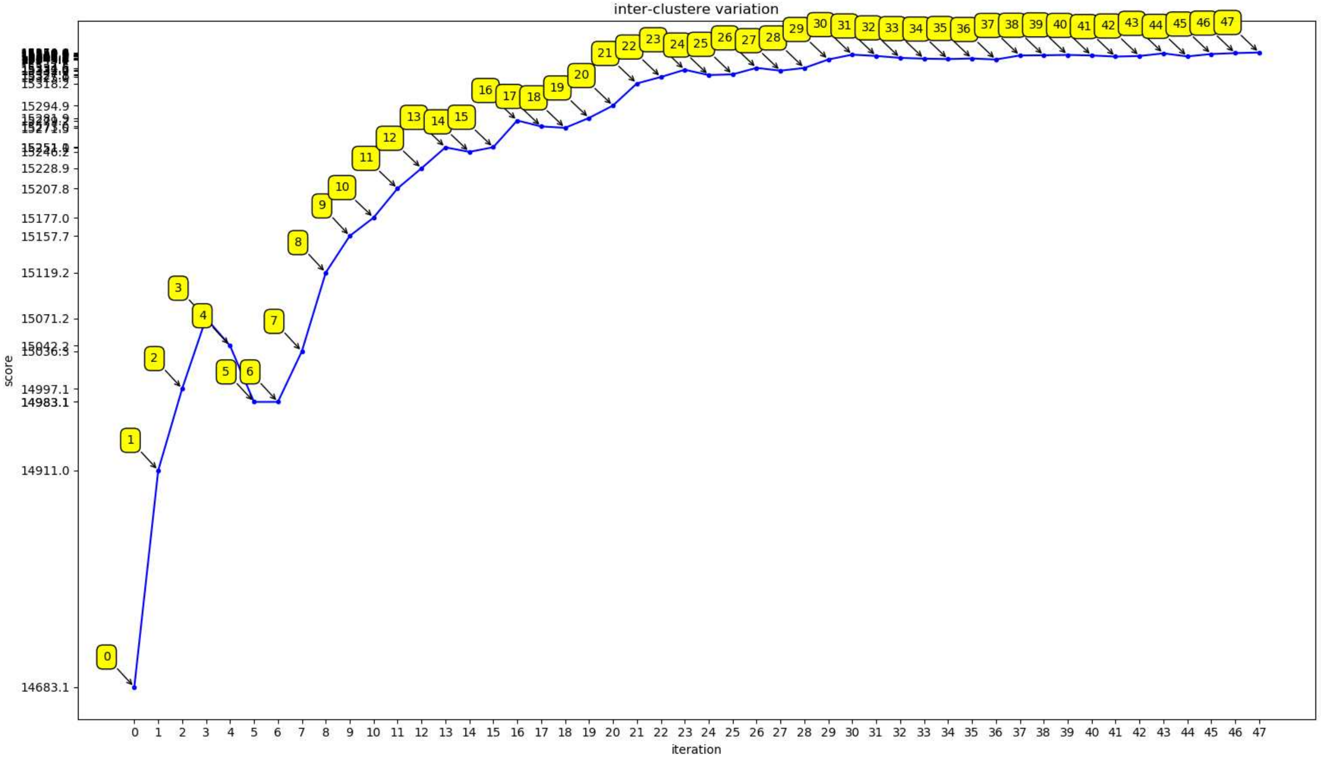


For Edinburgh, 2009 fall, C. Williams, V. Lavrenko, HW4, pr. 3

Graphical exemplification (cont’d)



© 2010-2012 Andrew Ng, Stanford University.



For Stanford, A. Ng, 2012 spring, HW9 (image segmentation and compression)

Exemplifying the application of a simple version of EM/GMM on data from $\mathbb{R}$

$(\sigma_1 = \sigma_2 = 1, \pi_1 = \pi_2 = 1/2)$

CMU, 2012 spring, Ziv Bar-Joseph, final exam, pr. 3.1

enhanced by Liviu Ciortuz

Suppose a GMM has two components with known variance and an equal prior distribution

$$\frac{1}{2}N(\mu_1, 1) + \frac{1}{2}N(\mu_2, 1).$$

The observed data are $x_1 = 0.5$ and $x_2 = 2$, and the current estimates of μ_1 and μ_2 are 1 and 2 respectively.

a. Execute the first iteration of the EM algorithm.

***Hint*: Normal densities for the standardized variable $y_{(\mu=0,\sigma=1)}$ at 0, 0.5, 1, 1.5, 2 are 0.4, 0.35, 0.24, 0.13, 0.05 respectively.**

b. Consider the log-likelihood function for the “observable” data,

$$\ell(\mu_1, \mu_2) \overset{def.}{=} \ln P(x_1, x_2|\mu_1, \mu_2) \overset{indep.}{=} \sum_{i=1}^{2} \ln P(x_i|\mu_1, \mu_2) = \sum_{i=1}^{2} \ln \left(\sum_{z_{ij}} P(x_i, z_{ij}|\mu_1, \mu_2) \right),$$

where $z_{ij} \in \{0, 1\}$ and $\sum_{j=1}^{2} z_{ij} = 1$ for all $i \in \{1, 2\}$.

Compute the values of ℓ function at the beginning and also at the end of the first iteration of the EM algorithm.
What do you see?

Solution (a.)

The E-step:

$$
\begin{aligned}
E[Z_{i1}] = P(Z_{i1}=1|x_i,\mu) &\overset{B.Th.}{=} \frac{P(x_i|Z_{i1}=1,\mu_1)P(Z_{i1}=1)}{P(x_i|Z_{i1}=1,\mu_1)P(Z_{i1}=1)+P(x_i|Z_{i2}=1,\mu_2)P(Z_{i2}=1)} \\
&= \frac{P(x_i|Z_{i1}=1,\mu_1)\cdot\frac{1}{2}}{P(x_i|Z_{i1}=1,\mu_1)\cdot\frac{1}{2}+P(x_i|Z_{i2}=1,\mu_2)\cdot\frac{1}{2}} \\
&= \frac{P(x_i|Z_{i1}=1,\mu_1)}{P(x_i|Z_{i1}=1,\mu_1)+P(x_i|Z_{i2}=1,\mu_2)} \text{ for } i\in\{1,2\}.
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
P(Z_{11}=1|x_1,\mu) &= \frac{N(0.5;1,1)}{N(0.5;1,1)+N(0.5;2,1)} = \frac{N(0.5;0,1)}{N(0.5;0,1)+N(1.5;0,1)} = \frac{0.35}{0.35+0.13} = \frac{35}{48} \\
P(Z_{21}=1|x_2,\mu) &= \frac{N(2;1,1)}{N(2;1,1)+N(2;2,1)} = \frac{N(1;0,1)}{N(1;0,1)+N(0;0,1)} = \frac{0.24}{0.24+0.4} = \frac{0.24}{0.64} = \frac{3}{8}
\end{aligned}
$$

Similarly,

$$
\begin{aligned}
P(Z_{12}=1|x_1,\mu) &= P(Z_{11}=0|x_1,\mu) = 1-P(Z_{11}=1|x_1,\mu_1) = \frac{13}{48} \\
P(Z_{22}=1|x_2,\mu) &= P(Z_{21}=0|x_2,\mu) = 1-P(Z_{21}=1|x_2,\mu_1) = \frac{5}{8}
\end{aligned}
$$

The M-step:

$$\mu_j^{(t+1)} = \frac{\sum_{i=1}^{2} E[Z_{ij}]\; x_i}{\sum_{i=1}^{2} E[Z_{ij}]} = \frac{\sum_{i=1}^{2} P(Z_{ij}=1|x_i,\mu^{(t)})\; x_i}{\sum_{i=1}^{2} P(Z_{ij}=1|x_i,\mu^{(t)})}$$

Therefore,

$$\mu_1^{(1)} = \frac{\dfrac{35}{48}\cdot 0.5 + \dfrac{3}{8}\cdot 2}{\dfrac{35}{48} + \dfrac{3}{8}} = \frac{107}{106} \approx 1.009 \quad \textbf{and} \quad \mu_2^{(1)} = \frac{\dfrac{13}{48}\cdot 0.5 + \dfrac{5}{8}\cdot 2}{\dfrac{13}{48} + \dfrac{5}{8}} = \frac{133}{86} \approx 1.54$$

Solution (b.)

$$\begin{aligned}
\ell(\mu_1,\mu_2) &\overset{def.}{=} \sum_{i=1}^{2} \ln P(x_i|\mu_1,\mu_2) = \sum_{i=1}^{2} \ln \left(\sum_{z_{ij}} P(x_i, z_{ij}|\mu_1,\mu_2) \right) \\
&= \sum_{i=1}^{2} \ln \left(\sum_{z_{ij}} P(x_i|z_{ij},\mu_1,\mu_2) \cdot \underbrace{P(z_{ij}|\mu_1,\mu_2)}_{1/2} \right) \\
&= \sum_{i=1}^{2} \ln \left(\frac{1}{2} \sum_{z_{ij}} P(x_i|z_{ij},\mu_1,\mu_2) \right) = \sum_{i=1}^{2} \left[-\ln 2 + \ln \left(\sum_{j=1}^{2} P(x_i|z_{ij}=1,\mu_j) \right) \right]
\end{aligned}$$

z_{ij}	$P(x_i \mid z_{ij},\mu_1,\mu_2)$
$z_{11}=1,\ z_{12}=0$	$\dfrac{1}{\sqrt{2\pi}}\exp(-\dfrac{1}{2}(x_1-\mu_1)^2)$
$z_{11}=0,\ z_{21}=1$	$\dfrac{1}{\sqrt{2\pi}}\exp(-\dfrac{1}{2}(x_1-\mu_2)^2)$
$z_{21}=1,\ z_{21}=0$	$\dfrac{1}{\sqrt{2\pi}}\exp(-\dfrac{1}{2}(x_2-\mu_1)^2)$
$z_{21}=0,\ z_{21}=1$	$\dfrac{1}{\sqrt{2\pi}}\exp(-\dfrac{1}{2}(x_2-\mu_2)^2)$

Therefore,

$$\begin{aligned}\ell(\mu_1,\mu_2) &= -2\ln 2 + \ln\left(\frac{1}{\sqrt{2\pi}}\left(\exp(-\frac{1}{2}(x_1-\mu_1)^2) + \exp(-\frac{1}{2}(x_1-\mu_2)^2)\right)\right)\\ &\quad + \ln\left(\frac{1}{\sqrt{2\pi}}\left(\exp(-\frac{1}{2}(x_2-\mu_1)^2) + \exp(-\frac{1}{2}(x_2-\mu_2)^2)\right)\right)\\ &= -2\ln 2 - \ln(2\pi) + \ln\left(\exp(-\frac{1}{2}(x_1-\mu_1)^2) + \exp(-\frac{1}{2}(x_1-\mu_2)^2)\right)\\ &\quad + \ln\left(\exp(-\frac{1}{2}(x_2-\mu_1)^2) + \exp(-\frac{1}{2}(x_2-\mu_2)^2)\right)\end{aligned}$$

and

$$\ell(\mu_1^{(0)},\mu_2^{(0)}) = \ell(1,2) = -2.561833 \leq \ell(\mu_1^{(1)},\mu_2^{(1)}) = \ell\left(\frac{107}{106},\frac{133}{86}\right) = -2.462877,$$

meaning that the value of the log-likelihood function increases, which is in line with the theoretical result concerning the correctness of the EM algorithm.

Derivation of the EM algorithm for a mixture of K uni-variate Gaussians: the general case (i.e., when all parameters π, μ, σ^2 are free)

following Dahua Lin,
An Introduction to Expectation-Maximization
(MIT, ML 6768 course, 2012 fall)

Note

We will first consider $K = 2$.
Generalization to $K > 2$ will be shown afterwards.

Estimation (E) Step:

$$
\begin{aligned}
p_{ij} &\overset{not.}{=} P(Z_{ij}=1 \mid X_i,\mu,\sigma,\pi) \overset{calcul}{=} E[Z_{ij} \mid X_i,\mu,\sigma,\pi] \\
&= \frac{P(X_i=x_i \mid Z_{ij}=1,\mu,\sigma,\pi)\cdot P(Z_{ij}=1 \mid \mu,\sigma,\pi)}{\sum_{j'=1}^{2} P(X_i=x_i \mid Z_{ij'}=1,\mu,\sigma,\pi)\cdot P(Z_{ij'}=1 \mid \mu,\sigma,\pi)} \\
&= \frac{\dfrac{1}{\sqrt{2\pi}\sigma_j}\cdot\exp\left(-\dfrac{(x_i-\mu_j)^2}{2\sigma_j^2}\right)\cdot\pi_j}{\dfrac{1}{\sqrt{2\pi}\sigma_1}\cdot\exp\left(-\dfrac{(x_i-\mu_1)^2}{2\sigma_1^2}\right)\cdot\pi_1+\dfrac{1}{\sqrt{2\pi}\sigma_2}\cdot\exp\left(-\dfrac{(x_i-\mu_2)^2}{2\sigma_2^2}\right)\cdot\pi_2}
\end{aligned}
$$

Therefore, for $t > 0$ we will have:

$$
p_{ij}^{(t)} = \frac{\dfrac{\pi_j^{(t-1)}}{\sigma_j^{(t-1)}}\cdot\exp\left(-\dfrac{(x_i-\mu_j^{(t-1)})^2}{2(\sigma_j^{(t-1)})^2}\right)}{\dfrac{\pi_1^{(t-1)}}{\sigma_1^{(t-1)}}\cdot\exp\left(-\dfrac{(x_i-\mu_1^{(t-1)})^2}{2(\sigma_1^{(t-1)})^2}\right)+\dfrac{\pi_2^{(t-1)}}{\sigma_2^{(t-1)}}\cdot\exp\left(-\dfrac{(x_i-\mu_2^{(t-1)})^2}{2(\sigma_2^{(t-1)})^2}\right)}
$$

The likelihood of a "complete" instance (x_i, z_{i1}, z_{i2}):

$$
\begin{aligned}
& P(X_i = x_i, Z_{i1} = z_{i1}, Z_{i2} = z_{i2} \mid \mu, \sigma, \pi) \\
& \quad = P(X_i = x_i \mid Z_{i1} = z_{i1}, Z_{i2} = z_{i2}, \mu_i, \sigma_i, \pi_i) \cdot P(Z_{i1} = z_{i1}, Z_{i2} = z_{i2} \mid \mu_i, \sigma_i, \pi_i) \\
& \quad = \frac{1}{\sqrt{2\pi}\sigma_j} \cdot \exp\left(-\frac{(x_i - \mu_j)^2}{2\sigma_j^2}\right) \cdot \pi_j, \ \textbf{where } z_{ij} = 1 \textbf{ and } z_{ij'} = 0 \textbf{ for } j' \neq j \\
& \quad = \frac{1}{\sqrt{2\pi}\ \sigma_1^{z_{i1}}\ \sigma_2^{z_{i2}}} \cdot \exp\left(-\frac{1}{2} \sum_{j \in \{1,2\}} z_{ij} \frac{(x_i - \mu_j)^2}{\sigma_j^2}\right) \cdot \pi_1^{z_{i1}} \pi_2^{z_{i2}}
\end{aligned}
$$

The log-likelihood of the same "complete" instance will be:

$$
\begin{aligned}
& \ln P(X_i = x_i, Z_{i1} = z_{i1}, Z_{i2} = z_{i2} \mid \mu, \sigma, \pi) \\
& \quad = -\frac{1}{2} \ln(2\pi) - \sum_{j=1}^{2} z_{ij} \ln \sigma_j - \frac{1}{2} \sum_{j=1}^{2} z_{ij} \frac{(x_i - \mu_j)^2}{\sigma_j^2} + \sum_{j=1}^{2} z_{ij}\ \ln \pi_j
\end{aligned}
$$

Given the dataset $X = \{x_1, \ldots, x_n\}$, the log-likelihood function will be:

$$l(\mu, \sigma, \pi) \overset{def.}{=} \ln P(X, Z_1, Z_2 \mid \mu, \sigma, \pi) \overset{i.i.d.}{=} \ln \prod_{i=1}^{n} P(X_i = x_i, Z_{i1}, Z_{i2} \mid \mu, \sigma, \pi)$$

$$= \sum_{i=1}^{n} \ln P(X_i = x_i, Z_{i1}, Z_{i2} \mid \mu, \sigma, \pi)$$

$$= -\frac{n}{2} \ln(2\pi) - \sum_{i=1}^{n} \sum_{j=1}^{2} Z_{ij} \ln \sigma_j - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{2} Z_{ij} \frac{(x_i - \mu_j)^2}{\sigma_j^2} + \sum_{i=1}^{n} \sum_{j=1}^{2} Z_{ij} \ln \pi_j$$

The expectation of the log-likelihood function:

$$E[\ln P(X, Z_1, Z_2 \mid \mu, \sigma, \pi)] =$$

$$-\frac{n}{2}\ln(2\pi) - \sum_{i=1}^{n}\sum_{j=1}^{2} E[Z_{ij}]\ln\sigma_j - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{2} E[Z_{ij}]\frac{(x_i-\mu_j)^2}{\sigma_j^2} + \sum_{i=1}^{n}\sum_{j=1}^{2} E[Z_{ij}]\,\ln\pi_j$$

Here above, the probability function w.r.t. which the expectation was computed was left unspecified. Now we will make it explicit:

$$\begin{aligned} Q(\mu, \sigma, \pi \mid \mu^{(t)}, \sigma^{(t)}, \pi^{(t)}) &\overset{not.}{=} E_{Z|X,\,\mu^{(t)},\sigma^{(t)},\pi^{(t)}}[\ln P(X, Z_1, Z_2 \mid X, \mu, \sigma, \pi)] \\ &= -\frac{n}{2}\ln(2\pi) - \sum_{i=1}^{n}\sum_{j=1}^{2} E[Z_{ij} \mid X_i, \mu^{(t)}, \sigma^{(t)}, \pi^{(t)}]\ln\sigma_j \\ &\quad -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{2} E[Z_{ij} \mid X_i, \mu^{(t)}, \sigma^{(t)}, \pi^{(t)}]\frac{(x_i-\mu_j)^2}{\sigma_j^2} \\ &\quad +\sum_{i=1}^{n}\sum_{j=1}^{2} E[Z_{ij} \mid X_i, \mu^{(t)}, \sigma^{(t)}, \pi^{(t)}]\,\ln\pi_j \end{aligned}$$

$$p_{ij}^{(t)} \overset{not.}{=} E[Z_{ij} \mid X, \mu^{(t-1)}, \sigma^{(t-1)}, \pi^{(t-1)}] \Rightarrow$$

$$Q(\mu, \sigma, \pi \mid \mu^{(t)}, \sigma^{(t)}, \pi^{(t)}) =$$

$$-\frac{n}{2}\ln(2\pi) - \sum_{i=1}^{n}\sum_{j=1}^{2} p_{ij}^{(t)} \ln\sigma_j - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{2} p_{ij}^{(t)} \frac{(x_i - \mu_j)^2}{\sigma_j^2} + \sum_{i=1}^{n}\sum_{j=1}^{2} p_{ij}^{(t)} \ln\pi_j$$

Since $K = 2$ and $\pi_1 + \pi_2 = 1$, we get

$$Q(\mu, \sigma, \pi \mid \mu^{(t)}, \sigma^{(t)}, \pi^{(t)}) =$$

$$-\frac{n}{2}\ln 2\pi - \sum_{i=1}^{n}\sum_{j=1}^{2} p_{ij}^{(t)} \ln\sigma_j - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{2} p_{ij}^{(t)} \frac{(x_i - \mu_j)^2}{\sigma_j^2} + \sum_{i=1}^{n}(p_{i1}^{(t)} \ln\pi_1 + p_{i2}^{(t)} \ln(1-\pi_1))$$

Maximization (M) Step:

[For $K = 2$**:]**

$$\frac{\partial}{\partial \pi_1} Q(\mu, \sigma, \pi \mid \mu^{(t)}, \sigma^{(t)}, \pi^{(t)}) = 0 \Leftrightarrow \frac{1}{\pi_1} \sum_{i=1}^{n} p_{i1}^{(t)} = \frac{1}{1-\pi_1} \sum_{i=1}^{n} p_{i2}^{(t)} \Leftrightarrow$$

$$\sum_{i=1}^{n} p_{i1}^{(t)} = \pi_1 (\sum_{i=1}^{n} p_{i1}^{(t)} + \sum_{i=1}^{n} p_{i2}^{(t)}) \Leftrightarrow \sum_{i=1}^{n} p_{i1}^{(t)} = \pi_1 \sum_{i=1}^{n} (\underbrace{p_{i1}^{(t)} + p_{i2}^{(t)}}_{1}) \Leftrightarrow \sum_{i=1}^{n} p_{i1}^{(t)} = n\pi_1$$

$$\Rightarrow \pi_1^{(t+1)} \leftarrow \frac{1}{n} \sum_{i=1}^{n} p_{i1}^{(t)}$$

Taking into account that $\pi_1^{(t+1)} + \pi_2^{(t+1)} = 1$ **and** $p_{i1}^{(t)} + p_{i2}^{(t)} = 1$ **for** $i = 1, \ldots, n$**,**

$$\Rightarrow \pi_2^{(t+1)} \leftarrow \frac{1}{n} \sum_{i=1}^{n} p_{i2}^{(t)}$$

$$\frac{\partial}{\partial \mu_1} Q(\mu, \sigma, \pi \mid \mu^{(t)}, \sigma^{(t)}, \pi^{(t)}) = 0 \Leftrightarrow \frac{1}{\sigma_1^2} \sum_{i=1}^{n} p_{i1}^{(t)} (x_i - \mu_1) = 0 \Leftrightarrow \sum_{i=1}^{n} p_{i1}^{(t)} (x_i - \mu_1) = 0$$

$$\Rightarrow \mu_1^{(t+1)} \leftarrow \frac{\sum_{i=1}^{n} p_{i1}^{(t)} x_i}{\sum_{i=1}^{n} p_{i1}^{(t)}}$$

Similarly, $\mu_2^{(t+1)} \leftarrow \frac{\sum_{i=1}^{n} p_{i2}^{(t)} x_i}{\sum_{i=1}^{n} p_{i2}^{(t)}}$

$$\frac{\partial}{\partial \sigma_1} Q(\mu, \sigma, \pi \mid \mu^{(t)}, \sigma^{(t)}, \pi^{(t)}) = 0 \Leftrightarrow -\frac{1}{\sigma_1} \sum_{i=1}^{n} p_{i1}^{(t)} + \frac{1}{\sigma_1^3} \sum_{i=1}^{n} p_{i1}^{(t)} (x_i - \mu_1)^2 = 0,$$

$$\Rightarrow \left(\sigma_1^{(t+1)}\right)^2 \leftarrow \frac{\sum_{i=1}^n p_{i1}^{(t)} (x_i - \mu_1^{(t+1)})^2}{\sum_{i=1}^n p_{i1}^{(t)}}$$

Similarly, $\left(\sigma_2^{(t+1)}\right)^2 \leftarrow \frac{\sum_{i=1}^n p_{i2}^{(t)} (x_i - \mu_2^{(t+1)})^2}{\sum_{i=1}^n p_{i2}^{(t)}}$

***Note*: One could relatively easily prove that these solutions (namely, $\pi^{(t+1)}, \mu^{(t+1)}, \sigma^{(t+1)}$) of the partial derivatives of the *auxiliary function* Q designate the values for which Q reaches its *maximum*.**

Generalization to $K > 2$

In this case, the Bernoulli distribution is replaced by a categorical one. The only one change needed in the above proof concerns updating the parameters of this distribution.

Since $\pi_1 + \ldots + \pi_K = 1$, we must solve the following *constraint optimization problem*:

$$\max_{\pi,\mu,\sigma} Q(\pi, \mu, \sigma | \pi^{(t)}, \mu^{(t)}, \sigma^{(t)})$$

$$\text{subject to } \sum_{i=1}^{K} \pi_j = 1 \textbf{ and } \pi_j \geq 0,\ \forall j = 1, \ldots, K.$$

By letting asside the $\geq$ constraints, and using the *Lagrangean multiplier* $\lambda \in \mathbb{R}$, this problem becomes:

$$\max_{\pi,\mu,\sigma} \left(Q(\pi, \mu, \sigma | \pi^{(t)}, \mu^{(t)}, \sigma^{(t)}) + \lambda(1 - \sum_{i=1}^{K} \pi_j) \right).$$

For $j = 1, \ldots, K$**:**

$$\frac{\partial}{\partial \pi_j} Q(\mu, \sigma, \pi \mid \mu^{(t)}, \sigma^{(t)}, \pi^{(t)}) = 0 \Leftrightarrow \sum_{i=1}^{n} p_{ij}^{(t)} \frac{1}{\pi_j} = \lambda \Leftrightarrow \pi_j^{(t+1)} = \frac{1}{\lambda} \sum_{i=1}^{n} p_{ij}^{(t)}.$$

Because $\sum_{j=1}^{K} \pi_j^{(t+1)} = 1$**, it follows that**

$$\lambda = \sum_{j=1}^{K} \sum_{i=1}^{n} \pi_j^{(t+1)} = \sum_{i=1}^{n} \underbrace{\sum_{j=1}^{K} \pi_j^{(t+1)}}_{1} = \sum_{i=1}^{n} 1 = n.$$

Therefore,

$$\pi_j^{(t+1)} \leftarrow \frac{1}{n} \sum_{i=1}^{n} p_{ij}^{(t)}.$$

***Note* that indeed** $\pi_j^{(t+1)} \geq 0$**, because the** $p_{ij}^{(t)}$ **terms designate some probabilities (see E-step).**

To summarize:

E Step:

$$p_{ij}^{(t)} \overset{not.}{=} P(z_{ij} = 1 \mid x_i; \mu^{(t)}, (\sigma^2)^{(t)}, \pi^{(t)}) = \frac{N(x_i \mid \mu_j^{(t)}, (\sigma_j^2)^{(t)}) \cdot \pi_j^{(t)}}{\sum_{l=1}^{K} N(x_i \mid \mu_l^{(t)}, (\sigma_l^2)^{(t)}) \cdot \pi_l^{(t)}}$$

where $N(x_i \mid \mu_j, \sigma_j^2) \overset{def.}{=} \frac{1}{\sqrt{2\pi}\sigma_j} \cdot \exp\left(-\frac{(x_i - \mu_j)^2}{2\sigma_j^2}\right)$.

M Step:

$$\pi_j^{(t+1)} \leftarrow \frac{1}{n}\sum_{i=1}^{n} p_{ij}^{(t)}$$

$$\mu_j^{(t+1)} \leftarrow \frac{\sum_{i=1}^{n} p_{ij}^{(t)} x_i}{\sum_{i=1}^{n} p_{ij}^{(t)}}$$

$$\left(\sigma_j^{(t+1)}\right)^2 \leftarrow \frac{\sum_{i=1}^{n} p_{ij}^{(t)} (x_i - \mu_j^{(t+1)})^2}{\sum_{i=1}^{n} p_{ij}^{(t)}}$$

Example: Modelling the waiting and eruption times for the Old Faithful geyser, (Yellowstone Park, USA)

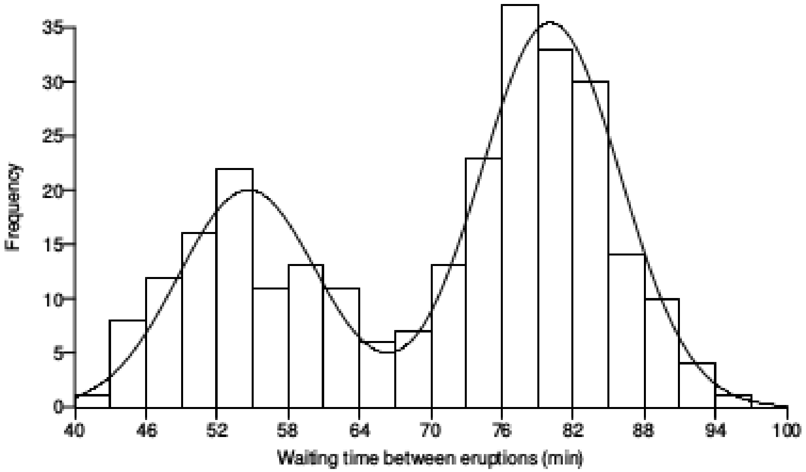
Michael Eichler (University of Chicago, Statistics course (24600) - Spring 2004)

R code

```
p< −c(0.5, 40, 90, 20, 20)
emstep<−function(Y, p) {
    EZ<−p[1] * dnorm(Y, p[2], sqrt(p[4]))/
        (p[1] * dnorm(Y, p[2], sqrt(p[4]))
        +(1 - p[1]) * dnorm(Y, p[3], sqrt(p[5])))
    p[1]<−mean(EZ)
    p[2]<−sum(EZ * Y) / sum(EZ)
    p[3]<−sum((1 - EZ)*Y) / sum(1-EZ)
    p[4]<−sum(EZ * (Y - p[2])^2) / sum(EZ)
    p[5]<−sum((1 - EZ) * (Y - p[3])^2) / sum(1 - EZ)
    p
}
emiteration<−function(Y, p, n=10) {
    for (i in (1:n)) {
        p<−emstep(Y, p)
    }
    p
}
p<−c(0.5, 40, 90, 20, 20)
p<−emiteration(Y, p, 20)
p
p<−emstep(Y, p)
p
}
```

Starting values:

$$p^{(0)} = 0.4$$
$$\mu_1^{(0)} = 40, \ \sigma_1^{(0)} = 4$$
$$\mu_1^{(0)} = 90, \ \sigma_1^{(0)} = 4$$



k	$p^{(k)}$	$\mu_1^{(k)}$	$\mu_2^{(k)}$	$\sigma_1^{(k)}$	$\sigma_2^{(k)}$
1	0.3508	54.22	79.91	5.465	5.999
2	0.3539	54.38	79.94	5.671	6.013
3	0.3562	54.46	79.99	5.744	5.969
4	0.3578	54.51	80.02	5.787	5.935
5	0.3588	54.55	80.05	5.815	5.912
6	0.3595	54.57	80.06	5.834	5.897
7	0.3600	54.59	80.07	5.846	5.887
8	0.3603	54.60	80.08	5.855	5.880
9	0.3605	54.60	80.08	5.860	5.876
10	0.3606	54.61	80.09	5.864	5.873
11	0.3607	54.61	80.09	5.866	5.871
12	0.3608	54.61	80.09	5.868	5.870
13	0.3608	54.61	80.09	5.869	5.869
14	0.3608	54.61	80.09	5.870	5.869
15	0.3609	54.61	80.09	5.870	5.868
20	0.3609	54.61	80.09	5.871	5.868
25	0.3609	54.61	80.09	5.871	5.868

Exemplifying

some methodological issues regarding the application of the EM algorithmic schema

(using a simple EM/GMM algorithm on data from $\mathbb{R}$ $(\pi_1 = \pi_2 = 1/2)$)

CMU, 2007 spring, Eric Xing, final exam, pr. 1.8

A long time ago there was a village amidst hundreds of lakes. Two types of fish lived in the region, but only one type in each lake.

These types of fish both looked exactly the same, smelled exactly the same when cooked, and had the exact same delicious taste – except one was poisonous and would kill any villager who ate it. The only other difference between the fish was their effect on the pH (acidity) of the lake they occupy.

The pH for lakes occupied by the non-poisonous type of fish was distributed according to a Gaussian with unknown mean (μ_{safe}) and variance (σ^2_{safe}) and the pH for lakes occupied by the poisonous type was distributed according to a different Gaussian with unknown mean (μ_{deadly}) and variance (σ^2_{deadly}). (Poisonous fish tended to cause slightly more acidic conditions).

Naturally, the villagers turned to machine learning for help. However, there was much debate about the right way to apply EM to their problem. For each of the following procedures, indicate whether it is an accurate implementation of Expectation-Maximization and will provide a reasonable estimate for parameters μ and σ^2 for each class.

a.

Guess initial values of μ and σ^2 for each class.
(1) For each lake, find the most likely class of fish for the lake.
(2) Update the μ and σ^2 values using their maximum likelihood estimates based on these predictions.
Iterate (1) and (2) until convergence.

b.

For each lake, guess an initial probability that it is safe.
(1) Using these probabilities, find the maximum likelihood estimates for the μ and σ values for each class.
(2) Use these estimates of μ and σ to reestimate lake safety probabilities.
Iterate (1) and (2) until convergence.

c.

Compute the mean and variance of the pH levels across all lakes.
Use these values for the μ and σ^2 value of each class of fish.
(1) Use the μ and σ^2 values of each class to compute the belief that each lake contains poisonous fish.
(2) Find the maximum likelihood values for μ and σ^2.
Iterate (1) and (2) until convergence.

Solution

a. It'll do ok if we give sensible enough μ and σ^2 initial values.

b. Ok, this is the same as a after the first M-step.
(See the general EM algorithmic schema on the next slide.)

c. This will be stuck at the initial μ and σ^2:

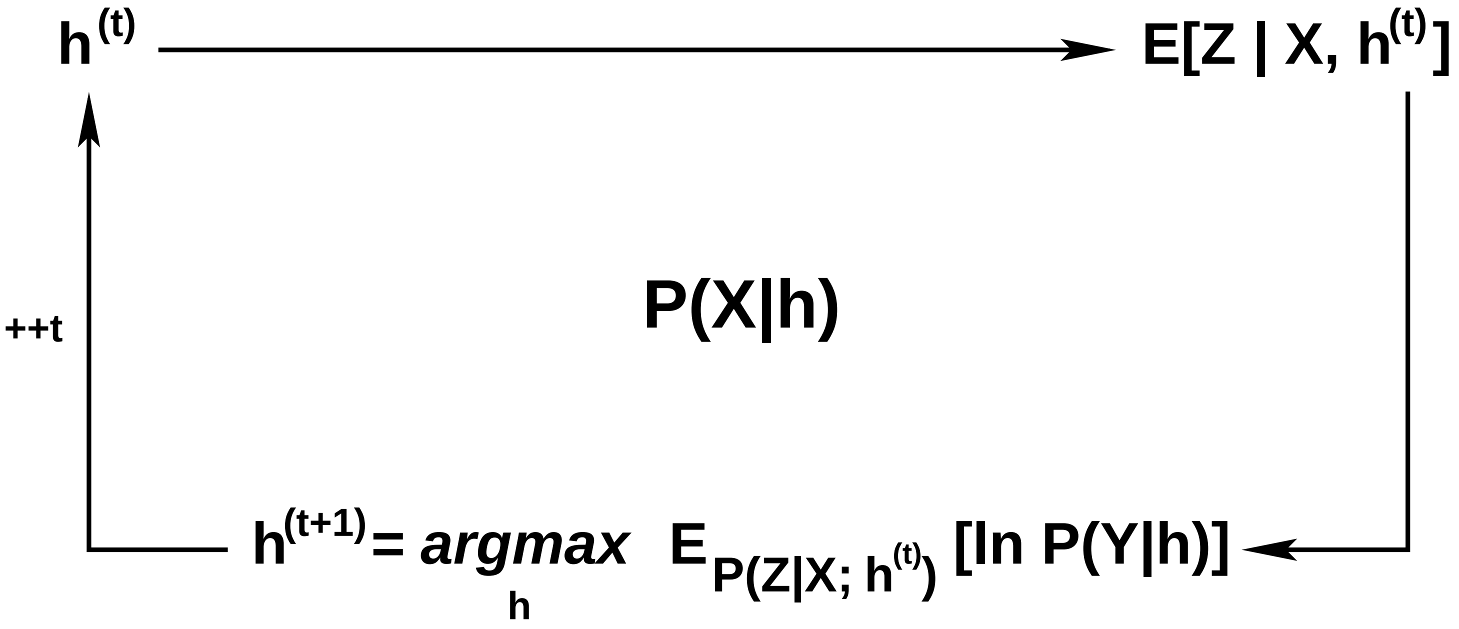
In the E-step we'll get:

$$P(safe|x) = \frac{P(x|safe) \cdot P(safe)}{P(x|safe) \cdot P(safe) + P(x|deadly) \cdot P(deadly)} = \frac{1}{2}$$

since on one side we assume $P(safe) = P(poison) = \dfrac{1}{2}$ **and on the other side** $P(x|safe) = P(x|poison)$ **because** $\mu_{safe} = \mu_{deadly}$ **and** $\sigma^2_{safe} = \sigma^2_{deadly}$**.**

In the M-step μ and σ^2 will not change since we are again letting them be calculated from all lakes (weighted equally).

The [general] EM algorithmic schema



$h^{(t)}$

$E[Z \mid X, h^{(t)}]$

$P(X|h)$

++t

$$h^{(t+1)} = \underset{h}{\operatorname{\mathit{argmax}}}\ E_{P(Z|X;\, h^{(t)})}\,[\ln P(Y|h)]$$

The EM algorithm for modeling mixtures of multi-variate Gaussians with diagonal covariance matrices

Utah University, Piyush Rai
ML course (CS5350/6350), 2009 fall,
lecture notes, *Gaussian Mixture Models*

[adapted by Liviu Ciortuz]

Fie mixtura de distribuţii gaussiene

$$gmm(x) = \sum_{j=1}^{K} \pi_j \, \mathcal{N}(x; \mu_j, \sigma^2 I),$$

unde

$x \in \mathbb{R}^d$**,**

probabilităţile a priori de selecţie $\pi_j \in \mathbb{R}$ **satisfac (ca de obicei) restricţiile** $\pi_j \geq 0$ **pentru** $j = 1, \ldots, K$ **şi** $\sum_{j=1}^{K} \pi_i = 1$**;**

mediile gaussienelor $j = 1, \ldots, K$ **sunt vectorii** $\mu_j \in \mathbb{R}^d$**, iar**

matricele de covarianţă ale acestor gaussine sunt identice, ba chiar au forma particulară $\sigma^2 I$**, cu** $\sigma \in \mathbb{R}$ **şi** $\sigma > 0$**, matricea** I **fiind matricea identitate.**

Se consideră instanţele $x_1, \ldots, x_n \in \mathbb{R}^d$ **generate cu distribuţia probabilistă de mai sus (**gmm**).**

În acest exerciţiu veţi deduce *regulile de actualizare* din cadrul [pasului E şi al pasului M al] algoritmului EM care face estimarea parametrilor $\pi \overset{not.}{=} (\pi_1, \ldots, \pi_K), \mu \overset{not.}{=} (\mu_1, \ldots, \mu_K)$ **şi** σ**.**

a. Se ştie că expresia funcţiei de densitate a distribuţiei gaussiene multivariate (d-dimensionale) de medie $\mu \in \mathbb{R}^d$ şi matrice de covarianţă $\Sigma \in \mathbb{R}^{d\times d}$ este

$$\frac{1}{(\sqrt{2\pi})^d \sqrt{|\Sigma|}} \exp\left(-\frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu)\right),$$

unde x şi μ sunt consideraţi vectori-coloană din $\mathbb{R}^d$, iar operatorul $\top$ desemnează operaţia de transpunere a vectorilor/matricelor.

Aduceţi expresia de mai sus la forma cea mai simplă pentru cazul $\Sigma = \sigma^2 I$.

Vă recomandăm să folosiţi faptul că $\|x-\mu\|^2 = (x-\mu)^\top(x-\mu) = (x-\mu)\cdot(x-\mu)$, unde operatorul $\cdot$ desemnează produsul scalar al vectorilor.

***Observaţie*: Prima din ultimele două egalităţi implică un uşor *abuz* (sau, mai degrabă, o convenţie) *de notaţie*: o matrice reală de dimensiune 1×1 este identificată cu un număr real, care este chiar singurul ei element. Acelaşi tip de abuz/convenţie a intervenit şi în scrierea expresiei $\exp(\ldots)$ de mai sus.**

Answer

$$\Sigma = \sigma^2 I \Rightarrow \ \Sigma^{-1} = \frac{1}{\sigma^2} I,$$

and also

$$|\Sigma| = (\sigma^2)^d \Rightarrow \sqrt{|\Sigma|} = \sigma^d \ \textbf{since} \ \sigma > 0.$$

Therefore,

$$\begin{aligned}
\mathcal{N}(x; \mu, \Sigma = \sigma^2 I) &= \frac{1}{(\sqrt{2\pi})^d \ \sqrt{|\Sigma|}} \exp\left(-\frac{1}{2}(x-\mu)^\top \Sigma^{-1} (x-\mu)\right) \\
&= \frac{1}{(\sqrt{2\pi})^d \sigma^d} \exp\left(-\frac{1}{2}(x-\mu)^\top \frac{1}{\sigma^2}(x-\mu)\right) \\
&= \frac{1}{(\sqrt{2\pi}\sigma)^d} \exp\left(-\frac{1}{2\sigma^2}\|x-\mu\|^2\right)
\end{aligned}$$

b. Vom asocia fiecărei instanţe x_i un vector-indicator (mai precis, un vector de variabile aleatoare $z_i \in \{0,1\}^K$, cu $z_{ij} = 1$ dacă şi numai dacă x_i a fost generat de gaussiana $\mathcal{N}(x; \mu_j, \sigma^2 I)$.

Pentru *pasul E* al algoritmului EM, veţi demonstra mai întâi că media $E[z_{ij}] \overset{not.}{=} E[z_{ij}|x_i; \pi, \mu, \sigma]$, unde $x_i = (x_{i,1}, \ldots, x_{i,d}) \in \mathbb{R}^d$, $\mu \overset{not.}{=} (\mu_1, \ldots, \mu_K) \in (\mathbb{R}^d)^K$ şi $\sigma \in \mathbb{R}_+$, are valoarea $P(z_{ij} = 1|x_i; \pi, \mu, \sigma)$, iar apoi veţi elabora formula de calcul a acestei probabilităţi, folosind teorema lui Bayes.

Answer

For the sake of simplicity, we will designate $E[z_{ij}|x_i;\pi\mu,\sigma]$ **as** $E[z_{ij}]$**.**
So,

$$
\begin{aligned}
E[z_{ij}] & \overset{not.}{=} E[z_{ij}|x_i;\pi\mu,\sigma] \overset{def.}{=} 0\cdot P(z_{ij}=0|x_i;\pi,\mu,\sigma)+1\cdot P(z_{ij}=1|x_i;\pi,\mu,\sigma) \\
& = P(z_{ij}=1|x_i;\pi,\mu,\sigma) \\
& \overset{Bayes\ F.}{=} \frac{P(x_i|z_{ij}=1;\pi,\mu,\sigma)\cdot P(z_{ij}=1;\pi,\mu,\sigma)}{P(x_i;\pi,\mu,\sigma)} \\
& = \frac{P(x_i|z_{ij}=1;\pi,\mu,\sigma)\cdot P(z_{ij}=1;\pi,\mu,\sigma)}{\sum_{j'=1}^{K} P(x_i|z_{ij'}=1;\pi,\mu,\sigma)P(z_{ij'}=1;\pi,\mu,\sigma)} \\
& \overset{a.}{=} \frac{\dfrac{1}{(\sqrt{2\pi}\sigma)^d}\exp\left(-\dfrac{1}{2\sigma^2}\|x_i-\mu_j\|^2\right)\ \pi_j}{\sum_{j'=1}^{K}\dfrac{1}{(\sqrt{2\pi}\sigma)^d}\exp\left(-\dfrac{1}{2\sigma^2}\|x_i-\mu_{j'}\|^2\right)\ \pi_{j'}} \\
& = \frac{\pi_j\ \exp\left(-\dfrac{1}{2\sigma^2}\|x_i-\mu_j\|^2\right)}{\sum_{j'=1}^{K}\pi_{j'}\ \exp\left(-\dfrac{1}{2\sigma^2}\|x_i-\mu_{j'}\|^2\right)} \qquad (7)
\end{aligned}
$$

c. Arătaţi că expresia funcţiei de log-verosimilitate a datelor ,,complete“ în raport cu parametrii π, μ şi σ este

$$\ln p(x, z | \pi, \mu, \sigma) = \sum_{i=1}^{n} \sum_{j=1}^{K} z_{ij} (\ln \pi_j + \ln \mathcal{N}(x_i; \mu_j, \sigma^2 I)),$$

unde $x \overset{not.}{=} (x_1, \ldots, x_n)$ şi $z \overset{not.}{=} (z_1, \ldots, z_n)$.

Deduceţi apoi expresia ,,funcţiei auxiliare“

$$Q(\pi, \mu, \sigma | \pi^{(t)}, \mu^{(t)}, \sigma^{(t)}) \overset{def.}{=} E[\ln p(x, z | \pi, \mu, \sigma)],$$

cu precizarea că media aceasta este calculată în raport cu distribuţia / distribuţiile $P(z_{ij} | x_i, \pi^{(t)}, \mu^{(t)}, \sigma^{(t)})$.

Answer

The log-verosimility of the complete data is

$$\begin{aligned}
\ln p(x,z|\pi,\mu,\sigma) &\overset{def.}{=} \ln p((x_1,z_1),\ldots,(x_n,z_n)|\pi,\mu,\sigma) \overset{i.i.d.}{=} \ln \prod_{i=1}^{n} p(x_i,z_i|\pi,\mu,\sigma)\\
&= \sum_{i=1}^{n} \ln p(x_i,z_i|\pi,\mu,\sigma) \overset{mult.\ rule}{=} \sum_{i=1}^{n} \ln p(x_i|z_i;\pi,\mu,\sigma)\cdot \underbrace{p(z_i|\pi,\mu,\sigma)}_{\pi_j}\\
&= \sum_{i=1}^{n}\sum_{j=1}^{K} z_{ij}\left[\ln \mathcal{N}(x_i|\mu_j,\sigma)+\ln\pi_j\right],
\end{aligned}$$

since $z_i=(z_{i,1},\ldots,z_{i,j},\ldots,z_{i,K})$**, with** $z_{i,j}=1$ **and** $z_{i,j'}=0$ **for all** $j'\neq j$**.**
Furthermore, using the result we got at part a**, we can write**

$$\ln p(x,z|\pi,\mu,\sigma)=\sum_{i=1}^{n}\sum_{j=1}^{K} z_{ij}\left[-\frac{d}{2}\ln(2\pi\sigma^2)-\frac{1}{2\sigma^2}\|x_i-\mu_j\|^2+\ln\pi_j\right].$$

Finally, using the linearity of expectation,

$$\begin{aligned}
Q(\pi,\mu,\sigma|\pi^{(t)},\mu^{(t)},\sigma^{(t)}) &\overset{def.}{=} E[\ln p(x,z|\pi,\mu,\sigma)]\\
&= \sum_{i=1}^{n}\sum_{j=1}^{K} E[z_{ij}]\left[-\frac{d}{2}\ln(2\pi\sigma^2)-\frac{1}{2\sigma^2}\|x_i-\mu_j\|^2+\ln\pi_j\right].
\end{aligned}$$

Pentru *Pasul M*, în contextul precizat în enunţ, vom avea de rezolvat următoarea *problemă de optimizare*:

$$(\pi^{(t+1)}, \mu^{(t+1)}, \sigma^{(t+1)}) = \operatorname*{argmax}_{\pi,\mu,\sigma} Q(\pi, \mu, \sigma | \pi^{(t)}, \mu^{(t)}, \sigma^{(t)}), \tag{8}$$

$$\textit{cu restricţiile } \pi_j^{(t+1)} \geq 0 \textit{ pentru } j = 1, \ldots, K \textit{ şi } \sum_{j=1}^{K} \pi_j^{(t+1)} = 1.$$

Această problemă se rezolvă optimizând funcţia ei obiectiv în mod separat în raport cu variabilele π, μ şi σ.

d. Aplicaţi metoda multiplicatorilor lui Lagrange pentru a rezolva problema de optimizare cu restricţii (8) în raport (doar) cu variabilele π.

Answer

For now, we will ignore the constraints $\pi_j \geq 0$. Thus, the *Lagrangean functional* associated to our optimisation problem is:

$$(\pi^{(t+1)}, \mu^{(t+1)}, \sigma^{(t+1)}) = \underset{\pi,\mu,\sigma,\lambda}{\mathrm{argmax}} (Q(\pi, \mu, \sigma | \pi^{(t)}, \mu^{(t)}, \sigma^{(t)}) - \lambda(1 - \sum_{j=1}^{K} \pi_j)), \tag{9}$$

with $\lambda \in \mathbb{R}$ playing the role of *Lagrangean multiplier*. (After solving this optimisation problem it will be seen that indeed $\pi_j \geq 0$ for all $j = 1, \ldots, K$.)
By taking the partial derivative of the Lagrangean functional with respect to π_j (with $j \in \{1, \ldots, K\}$) and then solving for it,

$$\frac{\partial}{\partial \pi_j} (Q(\pi, \mu, \sigma | \pi^{(t)}, \mu^{(t)}, \sigma^{(t)}) - \lambda(1 - \sum_{j=1}^{K} \pi_j)) = 0 \Leftrightarrow$$

$$\frac{\partial}{\partial \pi_j} \left(\sum_{i=1}^{n} \sum_{j=1}^{K} E[z_{ij}] \left[-\frac{d}{2} \ln(2\pi\sigma^2) - \frac{1}{2\sigma^2} \|x_i - \mu_j\|^2 + \ln \pi_j \right] \right) - \lambda = 0 \Leftrightarrow \sum_{i=1}^{n} E[z_{ij}] \frac{1}{\pi_j} = \lambda$$

we will obtain the solution

$$\pi_j^{(t+1)} = \frac{1}{\lambda} \sum_{i=1}^{n} E[z_{ij}]. \tag{10}$$

Moreover, since these solutions should satisfy the constraint $\sum_{j=1}^{K} \pi_j^{(t+1)} = 1$**, it follows that**

$$\sum_{j=1}^{K} \frac{1}{\lambda} \sum_{i=1}^{n} E[z_{ij}] = 1 \Leftrightarrow \frac{1}{\lambda} \sum_{j=1}^{K} \sum_{i=1}^{n} E[z_{ij}] = 1 \overset{b.}{\Leftrightarrow} \frac{1}{\lambda} \sum_{i=1}^{n} \sum_{j=1}^{K} P(z_{ij} = 1 | x_i, \pi, \mu, \sigma) = 1 \Leftrightarrow$$

$$\frac{1}{\lambda} \sum_{i=1}^{n} \underbrace{\sum_{j=1}^{K} P(z_{ij} = 1 | x_i, \pi, \mu, \sigma)}_{1} = 1 \Leftrightarrow \frac{1}{\lambda} \sum_{i=1}^{n} 1 = 1 \Leftrightarrow \lambda = \frac{1}{n}.$$

By replacing this value of λ **back into relation** (10)**, we will get**

$$\pi_j^{(t+1)} = \frac{1}{n} \sum_{i=1}^{n} E[z_{ij}]. \tag{11}$$

It is obvious that $\pi_j^{(t+1)} \geq 0$**.**

e. Optimizaţi funcţia Q (i.e., rezolvaţi problema de optimizare (8)) în raport cu variabilele μ.

Answer:

Remember the notation $\mu = (\mu_1, \ldots, \mu_K) \in (\mathbb{R}^d)^K$, with $\mu_j \in \mathbb{R}^d$ for $j = 1, \ldots, K$.

$$\begin{aligned}
&\nabla_{\mu_j} Q(\pi, \mu, \sigma | \pi^{(t)}, \mu^{(t)}, \sigma^{(t)}) \\
&= \nabla_{\mu_j} \sum_{i=1}^{n} \sum_{j'=1}^{K} E[z_{ij'}] \left[-\frac{d}{2} \ln(2\pi\sigma^2) - \frac{1}{2\sigma^2} \|x_i - \mu_j'\|^2 + \ln \pi_{j'} \right] \\
&= \sum_{i=1}^{n} E[z_{ij}] \left[-\frac{1}{2\sigma^2} \nabla_{\mu_j} (x_i - \mu_j)^2 \right] = -\frac{1}{2\sigma^2} \sum_{i=1}^{n} E[z_{ij}] 2 (x_i - \mu_j)(-1) \\
&= \frac{1}{\sigma^2} \sum_{i=1}^{n} E[z_{ij}](x_i - \mu_j) = \frac{1}{\sigma^2} \left[\left(\sum_{i=1}^{n} E[z_{ij}] x_i \right) - \left(\sum_{i=1}^{n} E[z_{ij}] \right) \mu_j \right]
\end{aligned}$$

After equating this expression to zero (in fact, the column-vector $(0, \ldots, 0)^\top \in \mathbb{R}^d$), we get the solution:

$$\mu^{(t+1)} = \frac{\sum_{i=1}^{n} E[z_{ij}] x_i}{\sum_{i=1}^{n} E[z_{ij}]}. \tag{12}$$

f. Optimizaţi funcţia Q (i.e., rezolvaţi problema de optimizare (8)) în raport cu variabila σ.

Answer:

The partial derivative of $Q(\pi, \mu, \sigma | \pi^{(t)}, \mu^{(t)}, \sigma^{(t)})$ with respect to σ is:

$$
\begin{aligned}
&\frac{\partial}{\partial \sigma} Q(\pi, \mu, \sigma | \pi^{(t)}, \mu^{(t)}, \sigma^{(t)}) \\
&= \frac{\partial}{\partial \sigma} \sum_{i=1}^{n} \sum_{j=1}^{K} E[z_{ij}] \left[-\frac{d}{2} \ln(2\pi\sigma^2) - \frac{1}{2\sigma^2} \|x_i - \mu_j\|^2 + \ln \pi_j \right] \\
&= \sum_{i=1}^{n} \sum_{j=1}^{K} E[z_{ij}] \frac{\partial}{\partial \sigma} \left[-\frac{d}{2} \ln(2\pi\sigma^2) - \frac{1}{2\sigma^2} \|x_i - \mu_j\|^2 \right] \\
&= \sum_{i=1}^{n} \sum_{j=1}^{K} E[z_{ij}] \left[-\frac{d}{2} \cdot \frac{2\sigma}{\sigma^2} - \frac{-2}{2\sigma^3} \|x_i - \mu_j\|^2 \right] = \sum_{i=1}^{n} \sum_{j=1}^{K} E[z_{ij}] \left[-\frac{d}{\sigma} + \frac{1}{\sigma^3} \|x_i - \mu_j\|^2 \right] \\
&= \frac{1}{\sigma^3} (-d\sigma^2 \sum_{i=1}^{n} \underbrace{\sum_{j=1}^{K} E[z_{ij}]}_{1} + \sum_{i=1}^{n} \sum_{j=1}^{K} E[z_{ij}] \|x_i - \mu_j\|^2) = \frac{1}{\sigma^3} (-nd\sigma^2 + \sum_{i=1}^{n} \sum_{j=1}^{K} E[z_{ij}] \|x_i - \mu_j\|^2)
\end{aligned}
$$

By equating $\dfrac{\partial}{\partial\sigma} Q(\pi, \mu, \sigma | \pi^{(t)}, \mu^{(t)}, \sigma^{(t)})$ **to** 0 **we get the solution:**

$$(\sigma^{(t+1)})^2 = \frac{1}{nd} \sum_{i=1}^{n} \sum_{j=1}^{K} E[z_{ij}] \|x_i - \mu_j^{(t+1)}\|^2 \geq 0. \tag{13}$$

It is not too difficult to see that this solution leads to the maximization of Q**. (Note that** $\sigma > 0$ **and the expression** $-nd\sigma^2 + \sum_{i=1}^{n} \sum_{j=1}^{K} E[z_{ij}] \|x_i - \mu_j\|^2$ **as a function of** σ^2**, is linear and decreasing.)**

g. Sumarizaţi rezultatele obţinute la punctele de mai sus (b şi d-f), redactând în pseudocod algoritmul EM pentru rezolvarea mixturii de gaussiene din enunţ.

Answer:

By gathering the relations (7), (11), (12) and (13), we are now able to write the pseudocode of our algorithm:

- Initialize the a priori probabilities π, the means μ, and the variance σ.
- Iterate until a certain *termination condition* is met:

  Step E: Compute the expectation (i.e., a posterori probabilities) of z variables:

$$p_{ij}^{(t+1)} \overset{not.}{=} E[z_{ij}|x_i;\pi^{(t)},\mu^{(t)},\sigma^{(t)}] = \frac{\pi_j^{(t)}\,\mathcal{N}(x;\mu_j^{(t)},(\sigma^{(t)})^2 I)}{\sum_{j'=1}^K \pi_{j'}^{(t)}\,\mathcal{N}(x;\mu_{j'}^{(t)},(\sigma^{(t)})^2 I)}$$

  Step M: Compute new values for π, μ şi σ:

$$\pi_j^{(t+1)} = \frac{1}{n}\sum_{i=1}^n p_{ij}^{(t)} \qquad \mu_j^{(t+1)} = \frac{\sum_{i=1}^n p_{ij}^{(t)}\;x_i}{\sum_{i=1}^n p_{ij}^{(t)}} \qquad \left(\sigma^{(t+1)}\right)^2 = \frac{1}{dn}\sum_{i=1}^n\sum_{j=1}^K p_{ij}^{(t)}\,\|x_i-\mu_j^{(t)}\|^2$$

Important Remark: a slight *generalization*

Suppose that our mixture model is made of Gaussians with unrestricted diagonal covariance matrices, i.e.,

$$Z_i \sim \mathit{Categorical}(p_1, \ldots, p_K)$$

$$X_i|Z_i = j \sim \mathcal{N}\left(\begin{bmatrix} \mu_{j,1} \\ \vdots \\ \mu_{j,d} \end{bmatrix}, \begin{bmatrix} (\sigma_{j,1})^2 & 0 & \ldots & 0 \\ 0 & (\sigma_{j,2})^2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \ldots & 0 & (\sigma_{j,d})^2 \end{bmatrix}\right)$$

It can be easily be proven (by going along the lines of the above proof) that the only *update relation* that changes in the formulation of the EM algorithm is (13). It becomes:

$$(\sigma_{jk}^{(t+1)})^2 = \frac{\sum_{i=1}^{n} E[z_{ij}](x_{i,k} - \mu_{j,k}^{(t+1)})^2}{\sum_{i=1}^{n} E[z_{ij}]} \geq 0 \textbf{ for } j = 1, \ldots, K \textbf{ and } k = 1, \ldots, d. \quad (14)$$

This is indeed what we would expect, given on one hand the fact that the Gaussian components are mutually independent, and on the other hand the updating formulas [that we have already obtained] for the EM algorithm for solving mixtures of uni-variate Gaussians [of unrestricted form].

The EM algorithm for modeling mixtures of multi-variate Gaussians

Stanford University, Prof. Andrew Ng

ML course, 2009, lecture notes, parts VIII and IX

[adapted by Liviu Ciortuz]

Suppose that we are given the *instances* $x_1, \ldots, x_n \in \mathbb{R}^d$ (all seen as column-vectors). We wish to *model* these data by specifying a joint distribution $p(x_i, z_i) = p(x_i|z_i) \cdot p(z_i)$. Here,

$z_i \sim \text{Categorical}(\pi)$**,**
K **denotes the number of values that the** z_i**'s can take on, namely** $\pi_j \overset{not.}{=} p(z_i = j)$ **for** $j = 1, \ldots, K$**, with** $\sum_{j=1}^{K} \pi_j = 1$**, and**
the [conditional] distribution $x_i|z_i = j$ **is a Gaussian of mean vector** μ_j **and covariance matrix** Σ_j**.**

Thus, our model posits that each x_i **was *generated* by randomly choosing** z_i **from** $\{1, \ldots, K\}$**, and then** x_i **was drawn from one of the** K **Gaussians, depending on** z_i**. This is called *the mixture of* [*multi-variate*] *Gaussians* model. Remember that**

$$\mathcal{N}(x; \mu, \sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma^{-1}|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu)\right)$$

***Note* that the** z_i**'s are *latent* random variables, meaning that they're hidden/unobserved.**

[Use the EM general scheme (see Tom Mitchell's *Machine Learning* book, 1997, pag. 194-195) to] prove that the EM algorithm for estimating the parameters π, μ and Σ of our mixture of multi-variate Gaussian distributions has the following *update rules*:

E-step:

$$w_{ij} \overset{not.}{=} E[z_i = j | x_i; \pi', \mu', \Sigma'] = \frac{\mathcal{N}(x_i; \mu', \Sigma')\ \pi_j}{\sum_{l=1}^{K} \mathcal{N}(x_i; \mu', \Sigma')\ \pi_l} \tag{15}$$

M-step:

$$\begin{align} \pi_j &= \frac{1}{n} \sum_{i=1}^{n} w_{ij}, \tag{16} \\ \mu_j &= \frac{\sum_{i=1}^{n} w_{ij} x_i}{\sum_{i=1}^{n} wij}. \tag{17} \\ \Sigma_j &= \frac{\sum_{i=1}^{n} w_{ij} (x_i - \mu_j)(x_i - \mu_j)^{\top}}{\sum_{i=1}^{n} w_{ij}}. \tag{18} \end{align}$$

where π', μ' and Σ' represent the values of our parameters at initialization, and respectively the previous iteration of the EM algorithm.

Hint: **You may find useful the following formulas (from *Matrix Identities*, by Sam Roweis, 1999):**

(1e) $(A^{-1})^\top = (A^\top)^{-1}$

(2b) $|A^{-1}| = \dfrac{1}{|A|}$

(4a) $\dfrac{\partial}{\partial X}|AXB| = |AXB|(X^{-1})^\top = |AXB|(X^\top)^{-1}$

(4b) $\dfrac{\partial}{\partial X}\ln|X| = (X^{-1})^\top = (X^\top)^{-1}$

(5a) $\dfrac{\partial}{\partial X}a^\top X = \dfrac{\partial}{\partial X}X^\top a = a$

(5b) $\dfrac{\partial}{\partial X}X^\top AX = (A + A^\top)X$

(5c) $\dfrac{\partial}{\partial X}a^\top Xb = ab^\top$

(5e) $\dfrac{\partial}{\partial X}a^\top Xa = \dfrac{\partial}{\partial X}a^\top X^\top a = aa^\top$

(5g) $\dfrac{\partial}{\partial X}(Xa+b)^\top C(Xa+b) = (C + C^\top)(Xa+b)a^\top$

Solution

The *E-step* is easy (use Bayes rule):

$$
\begin{aligned}
w_{ij} &\overset{not.}{=} E[z_i = j | x_i; \pi', \mu', \Sigma'] = p(z_i = j | x_i; \pi', \mu', \Sigma') = \\
&= \quad \frac{p(x_i | z_i = j; \mu', \Sigma')\, p(z_i = j; \pi')}{\sum_{l=1}^{K} p(x_i | z_i = l; \mu', \Sigma')\, p(z_i = l; \pi')} = \frac{\mathcal{N}(x_i; \mu', \Sigma')\, \pi'_j}{\sum_{l=1}^{K} \mathcal{N}(x_i; \mu', \Sigma')\, \pi_l}
\end{aligned}
$$

We will now concentrate on the *M-step*:

According to *the general EM scheme*, we need to maximize, with respect to our parameters π, μ, Σ, the "auxiliary" function

$$
\begin{aligned}
Q(\pi, \mu, \Sigma | \pi', \mu', \Sigma') &\overset{def.}{=} E_{p(z_i = j | x_i; \pi', \mu', \Sigma')} \ln p(x, z; \mu, \Sigma, \pi) \\
&= \sum_{i=1}^{n} \sum_{j=1}^{K} p(z_i = j | x_i; \pi', \mu', \Sigma') \ln p(x_i, z_i = j; \mu, \Sigma, \pi) \\
&= \sum_{i=1}^{n} \sum_{j=1}^{K} p(z_i = j | x_i; \pi', \mu', \Sigma') \ln (p(x_i | z_i = j; \mu, \Sigma) \, p(z_i = j; \pi)) \\
&= \sum_{i=1}^{n} \sum_{j=1}^{K} w_{ij} \ln \frac{1}{(2\pi)^{d/2} |\Sigma_j|^{1/2}} \cdot \exp(-\frac{1}{2} (x_i - \mu_j)^\top \, \Sigma_j^{-1} (x_i - \mu_j)) \cdot \pi_j \\
&= \sum_{i=1}^{n} \sum_{j=1}^{K} w_{ij} \left[- \ln ((2\pi)^{d/2} |\Sigma_j|^{1/2}) - \frac{1}{2} (x_i - \mu_j)^\top \, \Sigma_j^{-1} (x_i - \mu_j) + \ln \pi_j \right] \qquad (19)
\end{aligned}
$$

First, let's derive the M-step update rule for μ_l, with $l = 1, \ldots, K$.
We have to maximize (19) with respect to μ_l, so let's compute the corresponding derivative:

$$\frac{\partial}{\partial \mu_l} \sum_{i=1}^{n} \sum_{j=1}^{K} w_{ij} \left[-\ln((2\pi)^{d/2} |\Sigma_j|^{1/2}) - \frac{1}{2} (x_i - \mu_j)^\top \Sigma_j^{-1} (x_i - \mu_j) + \ln \pi_j \right]$$

$$= -\frac{\partial}{\partial \mu_l} \sum_{i=1}^{n} \sum_{j=1}^{K} w_{ij} \frac{1}{2} (x_i - \mu_j)^\top \Sigma_j^{-1} (x_i - \mu_j) = -\frac{1}{2} \sum_{i=1}^{n} w_{ij} \frac{\partial}{\partial \mu_l} \sum_{i=1}^{n} (x_i - \mu_j)^\top \Sigma_j^{-1} (x_i - \mu_j)$$

$$\overset{(5g)}{=} -\frac{1}{2} \sum_{i=1}^{n} w_{ij} \frac{\partial}{\partial \mu_l} \sum_{i=1}^{n} (\Sigma_j^{-1} + (\Sigma_j^{-1})^\top)\, (x_i - \mu_j)$$

$$\overset{(1e)}{=} \frac{1}{2} \sum_{i=1}^{n} w_{ij} \sum_{i=1}^{n} (\Sigma_j^{-1} + (\underbrace{\Sigma_j^\top}_{\Sigma_j})^{-1})\, (x_i - \mu_j) = \frac{1}{2} \sum_{i=1}^{n} w_{ij} \sum_{i=1}^{n} 2\Sigma_j^{-1}\, (x_i - \mu_j)$$

$$= \sum_{i=1}^{n} w_{il} \left(\Sigma_l^{-1} x_i - \Sigma_l^{-1} \mu_l \right) = \sum_{i=1}^{n} w_{il} \Sigma_l^{-1} x_i - \sum_{i=1}^{n} w_{il} \Sigma_l^{-1} \mu_l.$$

Setting this to zero and solving for μ_l therefore yields the update rule

$$\mu_l = \frac{\sum_{i=1}^{n} w_{il} x_i}{\sum_{i=1}^{n} w_{il}}.$$

Secondly, we'll derive the M-step updates to Σ_j, for $j = 1, \ldots, K$.
Grouping together only the terms that depend on Σ_j in (19), we find that we need to maximize

$$\sum_{i=1}^{n} \sum_{j=1}^{K} w_{ij} \left[\ln \frac{1}{|\Sigma_j|^{1/2}} - \frac{1}{2}(x_i - \mu_j)^\top \Sigma_j^{-1}(x_i - \mu_j) \right]$$

$$\overset{(2b)}{=} \sum_{i=1}^{n} \sum_{j=1}^{K} w_{ij} \left[\ln |\Sigma_j^{-1}|^{1/2} - \frac{1}{2}(x_i - \mu_j)^\top \Sigma_j^{-1}(x_i - \mu_j) \right].$$

We use the usual trick of working with the precision matrix $\Lambda_j \overset{not.}{=} \Sigma_j^{-1}$, where Σ_j is assumed invertible.

When maximizing the above quantity with respect to Λ_j by taking derivatives, we find:

$$
\begin{aligned}
&\frac{\partial}{\partial \Lambda_j} \sum_{i=1}^{n} w_{ij} \left[\ln |\Lambda_j|^{1/2} - \frac{1}{2}(x_i - \mu_j)^\top \Lambda_j (x_i - \mu_j) \right] \\
&\quad = \frac{1}{2} \sum_{i=1}^{n} w_{ij} \frac{\partial}{\partial \Lambda_j} \ln |\Lambda_j| - \frac{1}{2} \sum_{i=1}^{n} w_{ij} \frac{\partial}{\partial \Lambda_j} \left[(x_i - \mu_j)^\top \Lambda_j (x_i - \mu_j) \right] \\
&\quad \overset{(4b),(5c)}{=} \frac{1}{2} \sum_{i=1}^{n} w_{ij} \Lambda_j^{-1} - \frac{1}{2} \sum_{i=1}^{n} w_{ij} (x_i - \mu_j)(x_i - \mu_j)^\top \\
&\quad = \frac{1}{2} \Lambda_j^{-1} \sum_{i=1}^{n} w_{ij} - \frac{1}{2} \sum_{i=1}^{n} w_{ij} (x_i - \mu_j)(x_i - \mu_j)^\top .
\end{aligned}
$$

Setting this to zero and solving, we get:

$$
\Sigma_j = \Lambda_j^{-1} = \frac{\sum_{i=1}^{n} w_{ij} (x_i - \mu_j)(x_i - \mu_j)^\top}{\sum_{i=1}^{n} w_{ij}} .
$$

Finally, let's derive the M-step update for the parameters π_j.
Grouping together only the terms that depend on π_j in (19), we find that we need to maximize

$$\sum_{i=1}^{n} \sum_{j=1}^{K} w_{ij} \ln \pi_j .$$

However, there is an additional constraint that the π_j's sum to 1, since they represent the probabilities $\pi_j = p(z_i = j; \pi)$. To deal with the constraint that $\sum_{j=1}^{K} \pi_j = 1$, we construct the Lagrangian

$$\mathcal{L}(\pi) = \sum_{i=1}^{n} \sum_{j=1}^{K} w_{ij} \ln \pi_j + \beta \left(\sum_{j=1}^{K} \pi_j - 1 \right),$$

where β is the Lagrange multiplier.

***Note*: We don't need to worry about the constraint that $\pi_j \geq 0$, because as we'll shortly see, the solution we'll find from this derivation will automatically satisfy that anyway.**

Taking derivatives of $\mathcal{L}(\pi)$, we find:

$$\frac{\partial}{\partial \pi_j}\mathcal{L}(\pi) = \sum_{i=1}^{n} \frac{w_{ij}}{\pi_j} + \beta = \frac{1}{\pi_j}\sum_{i=1}^{n} w_{ij} + \beta.$$

Setting this to zero and solving, we get $\pi_j = \frac{\sum_{i=1}^{n} w_{ij}}{-\beta}$.

By using the constraint $\sum_j \pi_j = 1$**, and given the fact that** $\sum_j w_{ij} = 1$ **since** $w_{ij} \overset{not.}{=} p(z_i = j | x_i; \pi', \mu', \Sigma')$**, we easily find**

$$-\beta = \sum_{i=1}^{n}\sum_{j=1}^{K} w_{ij} = \sum_{i=1}^{n} 1 = n.$$

We therefore have our M-step derivation for the parameters π_j**:**

$$\pi_j = \frac{1}{n}\sum_{i=1}^{n} w_{ij},$$

and, obviously, $\pi_j \geq 0$**.**

Remarks

1. Let's contrast the update rules in the M-step with the formulas we would have when the z_is were known exactly (see the MLE of the parameters of a single multi-variate Gaussean distribution, CMU, 2010 fall, Aarti Singh, HW1, pr. 3.2.1):

$$\begin{aligned}
\pi_j &= \frac{1}{n}\sum_{i=1}^{n} 1\{z_i = j\}, \\
\mu_j &= \frac{\sum_{i=1}^{n} 1\{z_i = j\} x_i}{\sum_{i=1}^{n} 1\{z_i = j\}}, \\
\Sigma_j &= \frac{\sum_{i=1}^{n} 1\{z_i = j\}(x_i - \mu_j)(x_i - \mu_j)^\top}{\sum_{i=1}^{n} 1\{z_i = j\}},
\end{aligned}$$

with $1\{z_i = j\}$ ("indicator functions") indicating from which Gaussian each datapoint had come.

They are identical, except that instead of the indicator functions $1\{z_i = j\}$ indicating from which Gaussian each datapoint had come, we now have the w_{ij}'s.

Remarks (cont'd)

2. The EM-algorithm is reminiscent of the K-means clustering algorithm, except that instead of the "hard" cluster assignments $c(i)$, we have the "soft" assignments w_{ij}.

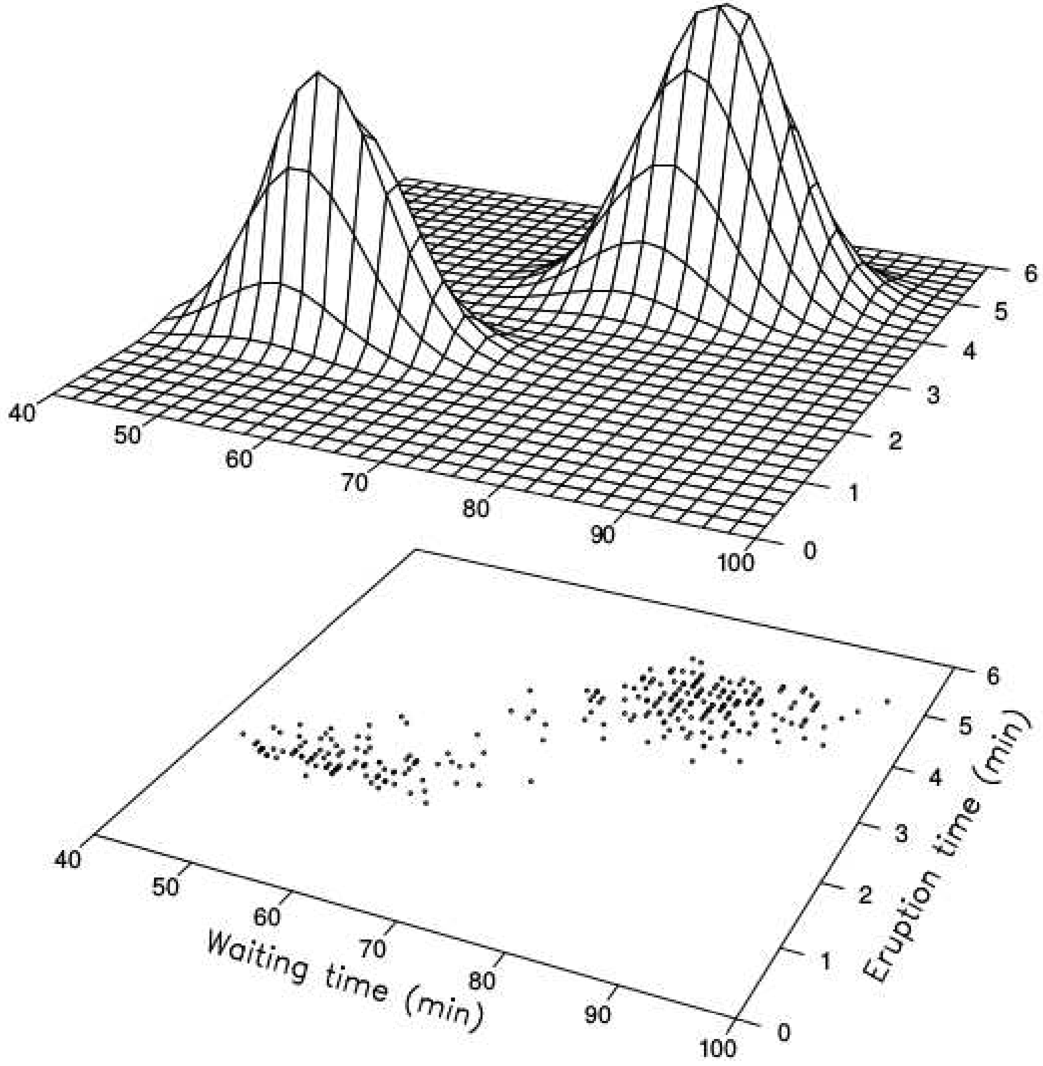
3. Similar to K-means, the EM algorithm is also susceptible to local optima, so reinitializing at several different initial parameters may be a good idea.

4. It's clear that the EM algorithm has a very natural interpretation of repeatedly trying to guess the unknown z_i's.

Example:

Modelling the waiting and eruption times for *the Old Faithful geyser*, (Yellowstone Park, USA)

Michael Eichler
University of Chicago
Statistics course (24600),
Spring 2004



A link between K-means and EM/GMM (the multi-variate case)

CMU, 2008 fall, Eric Xing, HW4, pr. 2.2

(see also CMU, 2010 fall, Aarti Singh, HW4, pr. 1.2)

Given N data points x_i, $(i = 1, \ldots, N)$, K-means will group them into K clusters by minimizing the *distortion* function

$$J = \sum_{i=1}^{N} \sum_{j=1}^{K} \gamma_{ij} \|x_i - \mu_j\|^2,$$

where μ_j is the centroid of the j-th cluster, and
$\gamma_{ij} = 1$ if x_i belongs to the j-th cluster and $\gamma_{ij} = 0$ otherwise.

In this exercise, we will use the following procedure for K-means:

- **Initialize [randomly] the cluster centroids μ_j, $j = 1, \ldots, K$;**
- **Iterate until *convergence*:**
  - **for every data point x_n, update its *cluster assignment*:**
    $\gamma_{ij} = 1$ if $j = \arg\min_k \|x_i - \mu_k\|^2$, and $\gamma_{ij} = 0$ otherwise.
  - **for each cluster j, update its centroid: $\mu_j = \dfrac{\sum_{i=1}^{N} \gamma_{ij} x_i}{\sum_{i=1}^{N} \gamma_{ij}}$**

Remember that in GMM, $p(x) = \sum_{j=1}^{K} \pi_k \, \mathcal{N}(x|\mu_j, \Sigma_j)$, where π_j is the prior [probability] for the j^{th} component, μ_j and Σ_j are the mean and covariance matrix for the j^{th} component respectively.
In the E-step of the EM algorithm, we will update

$$p(z_{ij} = 1|x_i) = \frac{\pi_j \, \mathcal{N}(x_i|\mu_j, \Sigma_j)}{\sum_{k=1}^{K} \pi_k \, \mathcal{N}(x_i|\mu_k, \Sigma_k)}$$

Now, suppose that

i. $\Sigma_k = \sigma^2 I$, **for some** $\sigma > 0$, **and for all** $k = 1, \ldots, K$

ii. $\pi_k \neq 0$ **for** $k = 1, \ldots, K$ **[LC: at any iteration of the EM algoritm], and**

iii. $\|x_i - \mu_{k'}\| \neq \|x_i - \mu_k\|$ **for any** $k' \neq k$ **[at any iteration of the EM algoritm].**

Under the above assumptions, prove that when $\sigma \rightarrow 0_+$ we will get $p(z_{ij} = 1|x_i) \rightarrow \gamma_{ij}$, where γ_{ij} is the *cluster assignment* used in K-means.

Answer:

$$
\begin{aligned}
p(z_{ij}=1|x_i) &= \frac{\pi_j\ \mathcal{N}(x_i|\mu_j,\Sigma_j)}{\sum_{k=1}^K \pi_k\ \mathcal{N}(x_i|\mu_k,\Sigma_k)} = \frac{\pi_j\ \exp\left(-\frac{1}{2\sigma^2}\|x_i-\mu_j\|^2\right)}{\sum_{k=1}^K \pi_k \exp\left(-\frac{1}{2\sigma^2}\|x_i-\mu_k\|^2\right)} \\
&= \frac{1}{1+\sum_{k\neq j}\frac{\pi_k}{\pi_j}\exp\left(\frac{1}{2\sigma^2}(\|x_i-\mu_j\|^2-\|x_i-\mu_k\|^2)\right)}
\end{aligned}
$$

Case 1:
If $\|x_i-\mu_j\| = \min_k \|x_i-\mu_k\|$, **then for each** $k \neq j$ **we have** $\|x_i-\mu_j\|^2 - \|x_i-\mu_k\|^2 < 0$. **Since** $\sigma \to 0_+$, **it will follow that** $\exp\left(\frac{1}{2\sigma^2}(\|x_i-\mu_j\|^2-\|x_i-\mu_k\|^2)\right) \to 0$. **So,** $p(z_{ij}=1|x_i) \to 1$.

Case 2:
If $\|x_i-\mu_j\| \neq \min_k \|x_i-\mu_k\|$, **then**

- **for all** k **such that** $\|x_i-\mu_j\| < \|x_i-\mu_k\|$ **it will follow (exactly as above) that** $\exp\left(\frac{1}{2\sigma^2}(\|x_i-\mu_j\|^2-\|x_i-\mu_k\|^2)\right) \to 0$;
- **for all** k **such that** $\|x_i-\mu_j\| > \|x_i-\mu_k\|$ **we will have** $\exp\left(\frac{1}{2\sigma^2}(\|x_i-\mu_j\|^2-\|x_i-\mu_k\|^2)\right)$ $\to +\infty$.

Therefore, $p(z_{ij}=1|x_i) \to \frac{1}{1+\infty} = 0$.