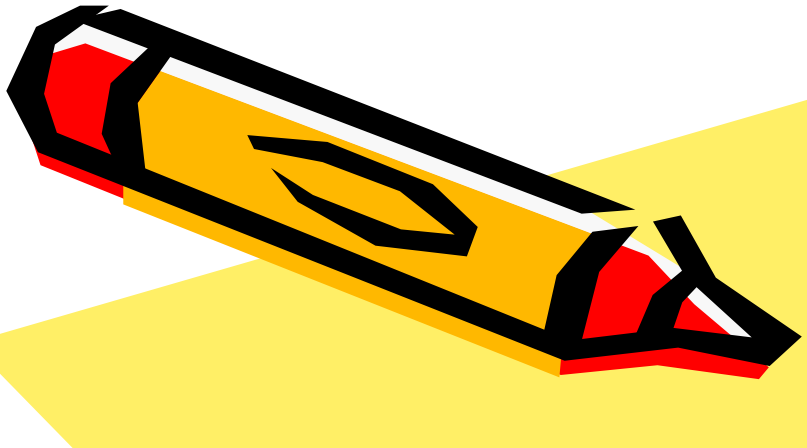


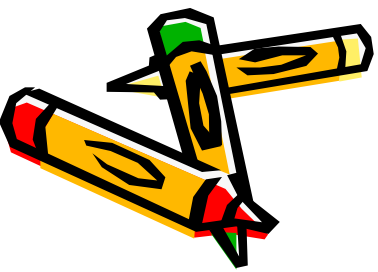
Clustering

Marina Gorunescu
mgorun@inf.ucv.ro

cluster

Clustering - ul este o tehnică de învățare nesupervizată, care determină o structură intrinsecă într-o mulțime de date. Această metodă este un proces de organizarea a obiectelor, care sunt asemănătoare (similare) dintr-un anumit punct de vedere.

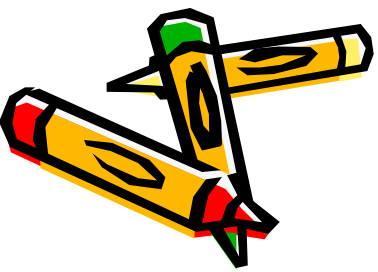
Cluster-ul este o mulțime de obiecte asemănătoare între ele și diferite de obiectele aparținând altui cluster.



Putem vorbi despre clustering, bazat pe o anumită *distanță*, în sensul că avem următorul criteriu de similaritate:

Două sau mai multe obiecte aparțin aceluiași cluster dacă sunt foarte apropiate, relativ la distanța considerată.

În *clustering*- ul *conceptual*, obiectele sunt grupate în același cluster dacă au aceeași proprietate, un cluster fiind definit de acea proprietate (concept).



probleme rezolvabile utilizand clustering-ul

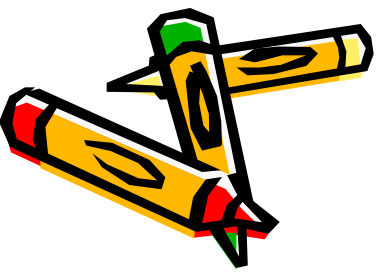
- reducerea datelor (*data reduction*), găsind reprezentanții unui grup omogen;
- determinarea unor clase convenabile („*useful data classes*");
- determinarea unor clustere naturale și descrierea proprietăților lor necunoscute („*natural data types*");
- găsirea unor obiecte mai rar întâlnite (*outlier detection*)

aplicatii

- *marketing*, pentru găsirea grupurilor de clienți cu profil asemănător, pe baza unei baze de date a acestor clienți, cu atributele lor și cumpărăturile lor anterioare;
- *biologie*, pentru clasificarea plantelor și animalelor;
- *asigurări*, pentru identificarea clienților ce prezintă riscuri mari și pentru identificarea fraudelor;
- *biblioteci*, pentru ordonarea cărților după anumite criterii;
- *internet*, pentru împărțirea documentelor în funcție de tematică.

masura de similaritate

Măsura de similaritate poate fi considerată ca fiind o metrică definită pe o anumită mulțime *M*.
Este aleasă în concordanță cu tipul de date cu care se lucrează și cu scopul propus.



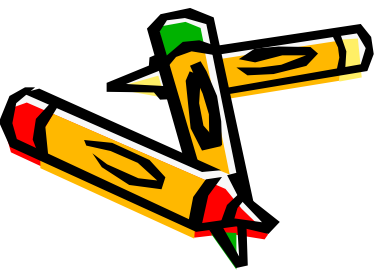
O bază de date formată din *m* vectori linie, n-dimensionali

$x_1,\ldots,x_m$, unde $x_k = (x_1^k,\ldots,x_n^k)$ poate fi reprezentată ca fiind

o matrice cu *m* linii şi *n* coloane

Vom defini mai multe metrici (distanțe) între vectorii x_k

şi x_1:



distanta euclidiana

- distanta euclidiană

$$d(x_k, x_l) = \sqrt{(x_k - x_l) \cdot (x_k - x_l)'} = \sqrt{\sum_{i=1}^{n} \left(x_i^k - x_i^l\right)^2}$$

distanta euclidiana standardizata

- distanța euclidiană standardizată

$$d_s(x_k, x_l) = \sqrt{(x_k - x_l) \cdot D^{-1} \cdot (x_k - x_l)'}$$

unde D este matricea diagonală ce are ca elemente

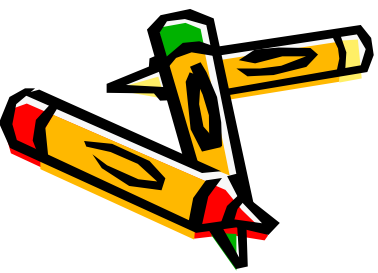
dispersia vectorului $X_i = (x_i^1, \ldots, x_i^m)$, notată d_i^2, adică:

$$d_s(x_k, x_l) = \sqrt{\sum_{i=1}^{n} \left(\frac{x_i^k - x_i^l}{d_i} \right)^2}$$

distanta city block

- distanța city block (Manhattan)

$$d_c(x_k, x_l) = \sum_{i=1}^{n} \left| x_i^k - x_i^l \right|$$



distanta Minkowski

- distanța Minkowski

$$d_M(x_k, x_l) = \left(\sum_{i=1}^{n} \left| x_i^k - x_i^l \right|^p \right)^{\frac{1}{p}}$$

Se observă că:

- dacă $p = 1$ obținem distanța *Manhattan*;
- dacă $p = 2$ avem distanța *euclidiană*.

distanta Minkowski ponderata

- Pentru ierarhizarea fiecărui atribut în funcție de scopul propus, pot fi incorporate ponderi și astfel distanța *Minkowski ponderată* este de forma:

$$d_{\alpha}(x_k, x_l) = \left(\sum_{i=1}^{n} \alpha_i \cdot \left| x_i^k - x_i^l \right|^p \right)^{\frac{1}{p}}, \ \alpha_i > 0, 1 \le i \le n,$$

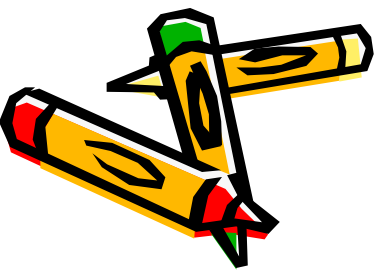
$\sum_{i=1}^{n} \alpha_i = 1$, unde α_i reprezintă ponderea atributului i.

distanta coeficientului de corelatie

- distanța *Pearson's r* (distanța coeficientului de corelație):

$$r(x_k, x_l) = 1 - \frac{\left(x_k - \overline{x_k}\right) \cdot \left(x_l - \overline{x_l}\right)'}{\sqrt{\left(x_k - \overline{x_k}\right) \cdot \left(x_k - \overline{x_k}\right)'} \cdot \sqrt{\left(x_l - \overline{x_l}\right) \cdot \left(x_l - \overline{x_l}\right)'}}$$

unde $\overline{x_k} = \frac{1}{n} \sum_{i=1}^{n} x_i^k$



distanta Mahalanobis

- Considerăm o mulțime de n vectori priviți ca un eşantion de dimensiune n a n variabile aleatoare independente.

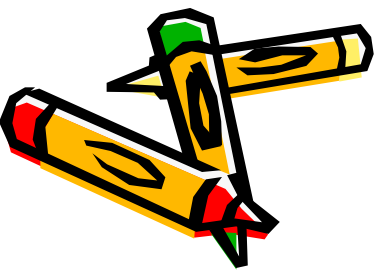
  Pentru fiecare pereche de vectori x_k şi x_l, covariança lor este definită de formula:

$$\mathrm{cov}(x_k, x_l) = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i^k \cdot x_i^l - \overline{x_k} \cdot \overline{x_l}$$

Notând cu $\mathrm{cov}(D)$ matricea de covarianță corespunzătoare eșantionului, măsura ***Mahalanobis*** este dată de formula:

$$D_M(x_k, x_l) = \sqrt{(x_k - x_l) \cdot \mathrm{cov}(D)^{-1} \cdot (x_k - x_l)'}$$

Dacă variabilele eșantionului nu sunt corelate, distanța Mahalanobis se rezumă la distanța euclidiană.



masuri fuzzy

- Măsurile *fuzzy* sunt utilizate pentru compararea de vectori ale căror componente iau valori în $[0,1]$.

  Pentru vectorul $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, $x_i \in [0,1]$, valoarea lui x_j reprezintă gradul în care vectorul $\mathbf{x}$ posedă al $j-$lea atribut (caracteristică).

  În scopul definirii măsurilor de similaritate *fuzzy* definim:

  $$s(x_i, y_i) = \max\{\min\{x_i, y_i\}, \min\{1-x_i, 1-y_i\}\}$$

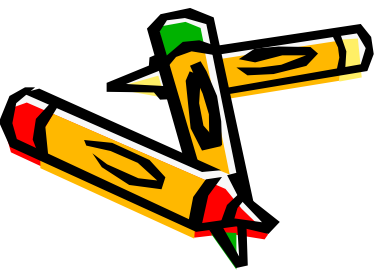
  plecând de la cazul clasic, avem măsura *fuzzy Minkowski* :

  $$d_F(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=1}^{n} s(x_i, y_i)^p \right)^{\frac{1}{p}}.$$

masura de similaritate mixta si ponderata

Obiectele au ca reprezentare un vector în care componentele au semnificații diferite, având naturi diferite: numerice, nominale, fuzzy etc.

Vom defini o măsura de similaritate *mixtă și ponderată*, pentru cuantificarea tipului de date, respectiv semnificația atributelor.



Avem de comparat două obiecte:

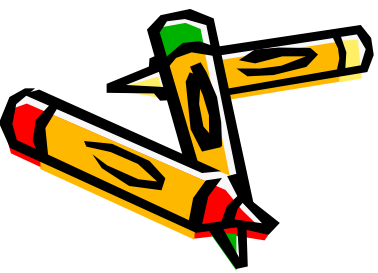
$$x = ((x_1^1,\ldots,x_{k_1}^1),\ldots,(x_1^p,\ldots,x_{k_p}^p))$$

$$y = ((y_1^1,\ldots,y_{k_1}^1),\ldots,(y_1^p,\ldots,y_{k_p}^p))$$

care au p tipuri de date de dimensiuni $k_1,\ldots,k_p$.

-vom pondera cele p secvențe cu valorile $\alpha_i > 0,\ \sum_{i=1}^{p} \alpha_i = 1$,

în funcție de importanța lor în context;



- vom aplica fiecărui cuplu de secvențe $x_j = (x_1^j, \ldots, x_{k_j}^j)$ și $y_j = (y_1^j, \ldots, y_{k_j}^j)$ măsura de similaritate specifică $d_j, 1 \le j \le p$.

Astfel:

$$d(\mathbf{x}, \mathbf{y}) = \sum_{j=1}^{p} \alpha_j \cdot d_j(x_j, y_j)$$

standardizarea caracteristicilor

Standardizarea caracteristicilor este o problemă ce o avem de rezolvat înainte de a calcula măsura de similaritate între obiecte.

Considerăm obiectele $x_1,\ldots,x_n$, unde $x_k = (x_1^k,\ldots,x_p^k)$, care pot fi reprezentate sub forma unei matrice X:

$$X = \begin{pmatrix} x_1^1 & \ldots & x_i^1 & \ldots & x_p^1 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ x_1^k & \ldots & x_i^k & \ldots & x_p^k \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ x_1^n & \ldots & x_i^n & \ldots & x_p^n \end{pmatrix}$$

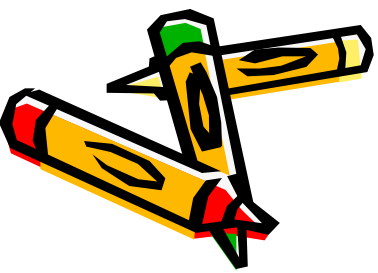
Pentru standardizare putem folosi transformările:

- $x_i^k \to \dfrac{x_i^k - \overline{x^i}}{SD(x^i)}$, unde $x^i = \begin{pmatrix} x_i^1 \\ .. \\ x_i^n \end{pmatrix}$, $1 \le k \le n, 1 \le i \le p$

- $x_i^k \to \dfrac{x_i^k - \overline{x_k}}{SD(x_k)}$, unde $x_k = (x_1^k, ..., x_p^k)$, $1 \le k \le n, 1 \le i \le p$.

Pentru fiecare vector, media este zero şi dispersia egală cu unitatea.

Metode de clustering neierarhic

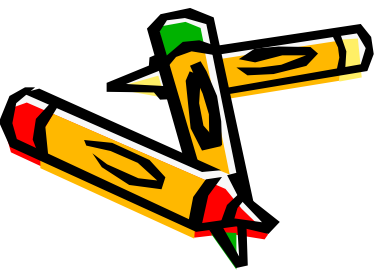


K-means

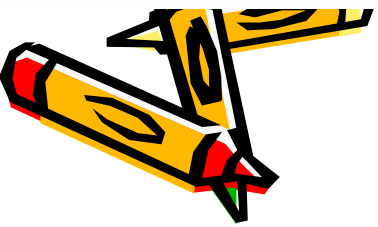
Algoritmul *k-means* (Mac Queen, 1967) clasifică o mulțime de n obiecte într-un număr k dat *apriori* de clustere.

- alegem aleator k puncte în spațiul definit de obiectele din baza de date, ce urmează a fi clusterizate. Acestea sunt *centroizii* inițiali.

De obicei încercăm să îi plasăm cât se poate de departe unul de altul.

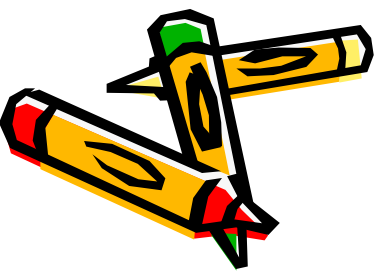


- asociem fiecare obiect din mulțimea de date celui mai apropiat centroid (în sensul distanței considerate), obținând astfel un prim grupaj.
- calculăm centrele de greutate ale clusterelor obținute, definind astfel noii centroizi.
- reluăm procedeul și astfel cei k centroizi își vor schimba locul pas cu pas, până ajung în poziția dorită.



În final, algoritmul urmărește minimizarea *funcției obiectiv* $J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2$ unde $\left\| x_i^{(j)} - c_j \right\|^2$ este distanța aleasă între punctul $x_i^{(j)}$ din baza de date și centroidul c_j al clusterului C_j.

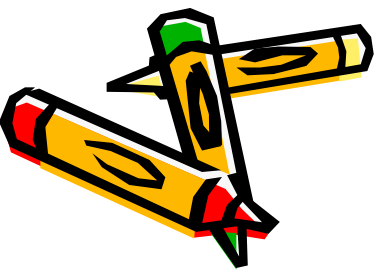
Funcția obiectiv, care este în acest caz funcția erorii pătratice, este un indicator al distanței dintre cele n puncte din baza de date și centrele clusterelor corespunzătoare.



probleme

- algoritmul nu găsește întotdeauna configurația optimă, corespunzătoare minimului global al funcției obiectiv.
- algoritmul este sensibil la alegerea inițială a centroizilor.

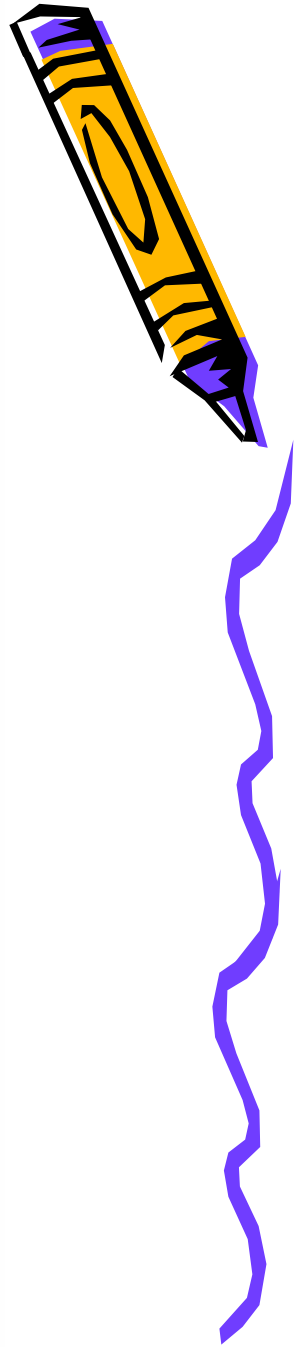
Pentru reducerea acestor efecte se rulează de mai multe ori algoritmul.



exemplu

aplicație referitoare la cancerul hepatic:
considerăm un lot de 17 pacienți dintre care 6 au cancer hepatic (HCC).
Pentru a decide dacă un pacient are sau nu cancer hepatic, studiile clinice arată că se iau în considerare anumite enzime serice (de obicei un număr de 15).
Pentru simplificarea calculelor, le vom lua în considerare pe cele mai importante din punct de vedere clinic:
alkaline phosphatase (FA), g-glutamyl transferase (gGT), leucine amino peptidase (LAP) și colesterol (C)

	FA	gCT	LAP	C	diagnostic
P1	162	255	74	258	non HCC
P2	210	324	93	220	non HCC
P3	259	208	115	266	non HCC
P4	120	114	89	171	non HCC
P5	246	173	98	210	non HCC
P6	138	189	76	165	non HCC
P7	132	177	48	138	non HCC
P8	152	183	105	178	non HCC
P9	186	220	140	148	non HCC
P10	180	119	114	171	non HCC
P11	236	270	88	150	non HCC
P12	422	488	183	292	HCC
P13	607	259	65	275	HCC
P14	446	283	176	309	HCC
P15	460	1053	145	221	HCC
P16	680	381	280	275	HCC
P17	561	450	180	164	HCC



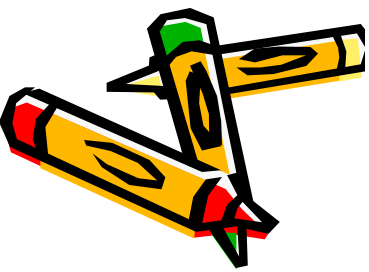
Alegem aleator doi centroizi:

$C1(150,100,100,100)$ şi $C2(400,400,200,200)$;

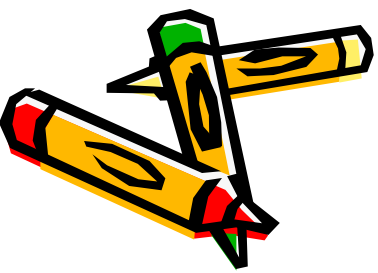
calculăm distanţele dintre aceştia şi punctele Pi (pacienţii din baza de date), împărţindu-le în două submulţimi după regula:

„dacă $d(C1,Pi) < d(C2,Pi)$ pacientul Pi aparţine mulţimii $M1$;

dacă $d(C2,Pi) < d(C1,Pi)$ pacientul Pi aparţine mulţimii $M2$."



	$d(C1, Pi)$	$d(C2, Pi)$	mulțimea
P1	231.3	153.9	M2
P2	276.9	165.3	M2
P3	254.4	122.9	M2
P4	75.8	164.1	M1
P5	196.8	115.5	M2
P6	119	143.4	M1
P7	105.3	179.1	M1
P8	125.3	110	M2
P9	160.3	83	M2
P10	109.5	123.2	M1
P11	223.6	145.7	M2
P12	545.8	375.4	M2
P13	560.5	439.2	M2
P14	450.1	282.5	M2
P15	1026.9	893.6	M2
P16	691.6	524.5	M2
P17	587.8	441	M2



Am obținut mulțimile

$M1 = \{P4, P6, P7, P10\}$

și

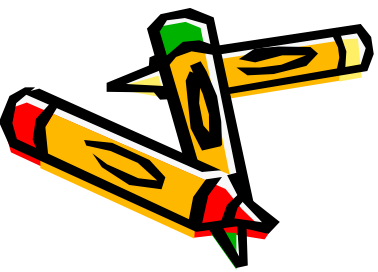
$M2 = \{P1, P2, P3, P5, P8, P9, P11, P12, P13, P14, P15, P16, P17\}$

Calculăm centroizii acestor mulțimi:

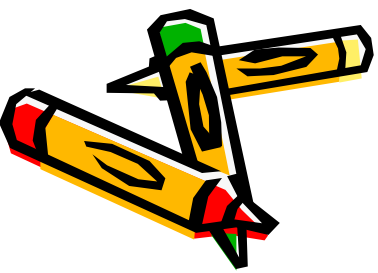
$C_1(142.5, 149.7, 81.7, 161.2)$

$C_2(355.9, 349.7, 134, 228.1)$

și distanțele dintre vectori (pacienți) și aceştia.



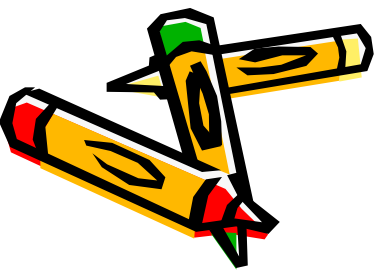
	$d(C_1, Pi)$	$d(C_2, Pi)$	mulțimea
P1	144.5	225.9	M_1'
P2	196.2	153.9	M_2'
P3	170.4	176.8	M_1'
P4	43.9	341.3	M_1'
P5	117.9	211.9	M_1'
P6	40.1	283.9	M_1'
P7	50.2	308.9	M_1'
P8	44.9	296.6	M_1'
P9	102	228	M_1'
P10	59	296.3	M_1'
P11	152.9	170..1	M_1'
P12	468.9	173.1	M_2'
P13	490.8	279.7	M_2'
P14	374.9	144.4	M_2'
P15	961.4	711.1	M_2'
P16	628.2	359.9	M_2'
P17	524.3	241.1	M_2'



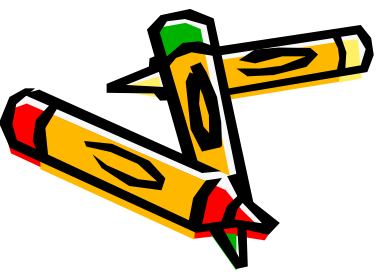
Calculăm centroizii mulțimilor M_1 și M_2

$C_1'(181.1, 190.8, 94.7, 185.5)$

$C_2'(485, 462.5, 160.2, 250.8)$



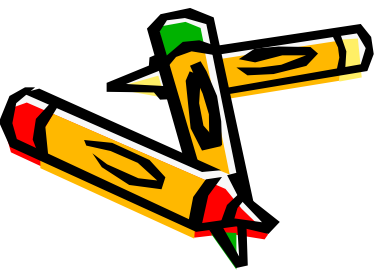
	$d(C_1', Pi)$	$d(C_2', Pi)$	mulțimea
P1	100.8	393.5	M_1''
P2	140.6	316.6	M_1''
P3	115.1	343.6	M_1''
P4	99.3	515.8	M_1''
P5	71.6	382.7	M_1''
P6	51.2	457.8	M_1''
P7	83.8	481	M_1''
P8	32.7	444.2	M_1''
P9	65.8	398.9	M_1''
P10	75.7	468.5	M_1''
P11	102.9	338.2	M_1''
P12	406.8	82.6	M_2''
P13	441.5	286.7	M_2''
P14	317	193.3	M_2''
P15	908.2	591.9	M_2''
P16	572.2	244.1	M_2''
P17	468.2	117.7	M_2''



Continuând procedeul, centroizii mulțimilor obținute sunt:

$C_1''(183.7, 202.9, 94.5, 188.6)$

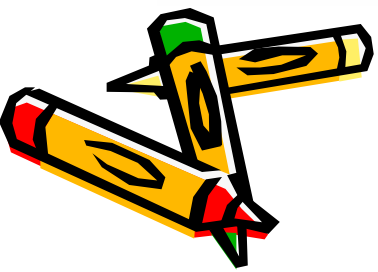
$C_2''(355.9, 349.7, 134, 228.1)$.



	$d(C_1'', Pi)$	$d(C_2'', Pi)$	mulțimea
P1	91.7	445.7	M_1''
P2	127.8	369.4	M_1''
P3	110	392.7	M_1''
P4	110.9	566.4	M_1''
P5	72.4	431.6	M_1''
P6	56.3	532.4	M_1''
P7	89.9	532.4	M_1''
P8	40.2	495.5	M_1''
P9	63.3	449.5	M_1''
P10	87.9	517.6	M_1''
P11	93.6	389.3	M_1''
P12	395.7	115.2	M_2''
P13	436.6	282.4	M_2''
P14	310.4	225.9	M_2''
P15	895.8	573.4	M_2''
P16	565.6	212.9	M_2''
P17	459.7	103.5	M_2''

Am obținut aceleași mulțimi, ca în tabelul precedent, ceea ce ne indică faptul că procesul s-a încheiat. Acest fapt constituie condiția de stop a algoritmului.

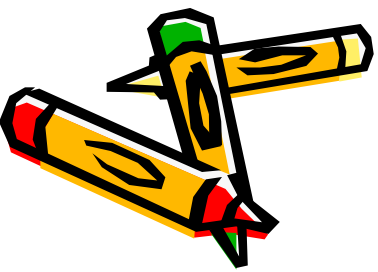
Am împărțit mulțimea de obiecte în două clustere, și testând cu mulțimea noastră de date, rezultatul este optim.



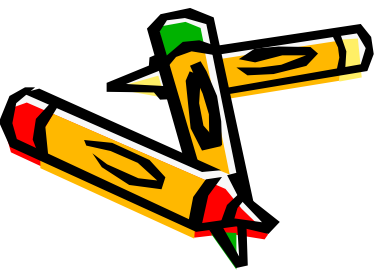
Algoritmul este sensibil la alegerea centroizilor inițiali.
în funcție de date
alegem următorii centroizi:

$C1(150,100,100,100)$ și $C2(400,400,200,200)$

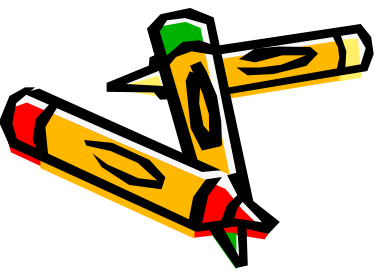
și obținem un rezultat asemănător cu rezultatul
din cel de-al doilea tabel.



	$d(C1, Pi)$	$d(C2, Pi)$	mulțimea
P1	231.3	311.3	M1
P2	276.9	231.7	M2
P3	254.4	261.3	M1
P4	75.8	416.3	M1
P5	196.8	292.8	M1
P6	119	360.2	M1
P7	105.3	385.3	M1
P8	125.3	343.6	M1
P9	160.3	290.6	M1
P10	109.5	368.2	M1
P11	223.6	242.5	M1
P12	545.8	130.3	M2
P13	560.5	294.2	M2
P14	450.1	168.3	M2
P15	1026.9	658.3	M2
P16	691.6	301.3	M2
P17	587.8	173.5	M2



Metoda *k-means* nu este aplicabilă la date ne-numerice, deoarece definirea mediei devine o problemă.



metoda euristica de clusterizare rapida

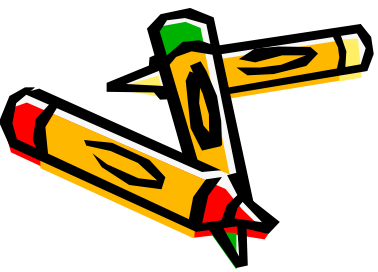
Prezentăm o metodă euristică de clusterizare rapidă care dă rezultate bune, dar nu optime.
Avem o mulțime de obiecte, care urmează a fi împărțită în *k* clustere:

- se începe cu un singur cluster al cărui centroid este media obiectelor din baza de date.
- se calculează distanțele dintre obiectele din baza de date și centroid și stabilind euristic o valoare -treshhold, .se împarte clusterul în două submulțimi.

- calculăm mediile obiectelor din cele două submulțimi și obținem doi centroizi.
- în funcție de distanțele dintre obiectele considerate și cei doi centroizi, obținem două clustere.
- calculăm centroizii celor două clustere și obținem eventual o altă grupare a obiectelor, prin determinarea distanțelor dintre acestea și acești centroizi.
- această parte a procesului se oprește când nu mai apar modificări în componența clusterelor.

- se continuă procedeul împărțind fiecare cluster în două, până obținem numărul de clustere cerut.

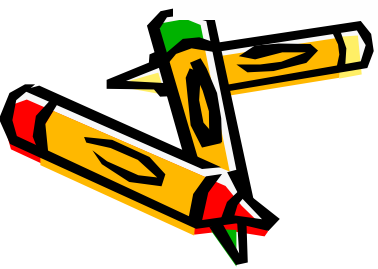
Dacă numărul k nu este o putere a lui 2, ne oprim la cea mai mică putere a lui 2 mai mare decât k și renunțăm la clusterele ce sunt mai puțin importante.



Reluăm exemplul prezentat anterior: aplicația cu determinarea bolnavilor cu cancer hepatic, folosind cele 4 enzime serice.

Avem două clase:

- HCC = bolnavi cu cancer hepatic
- Non-HCC = bolnavi cu alte afecțiuni hepatice.



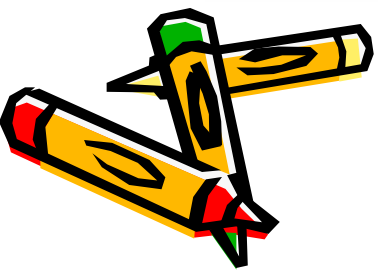
Pentru început vom calcula centroidul ce are drept componente mediile fiecărui atribut (enzimă serică):

$$c1 = mean(FA) = 305.7\,; \quad c2 = mean(gGT) = 214.3\,;$$

$$c3 = mean(LAP) = 97.2\,; \; c4 = mean(C) = 193.2\,.$$

centroidul inițial este punctul de coordonate:

$$C(195.8, 214.3, 97.2, 193.2)\,.$$



Calculăm distanța dintre fiecare vector (pacient) și acest centroid, notând $di = d(C, Pi)$:

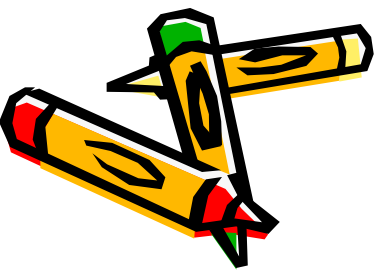
$d1 = 165.1;\quad d2 = 102.4;\quad d3 = 118.6;\quad d4 = 269.9;$

$d5 = 144.7;\quad d6 = 213;\quad d7 = 238.6;\quad d8 = 198.5;$

$d9 = 160.1;\quad d10 = 226.5;\quad d11 = 104.6;\quad d12 = 240.7;$

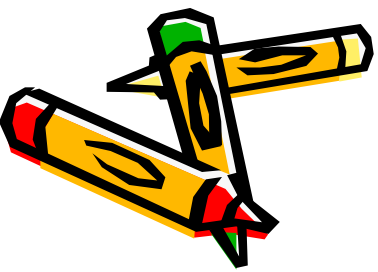
$d13 = 315.8;\quad d14 = 763.7;\quad d15 = 766.4;\quad d16 = 418.5;$

$d17 = 304.3$



Vom împărți lotul de pacienți în două submulțimi:

- M1: pacienții pentru care distanța di este mai mică decât 200 (aleasă euristic) $\{P1, P2, P3, P5, P8, P9, P11\}$
- M2: restul pacienților.



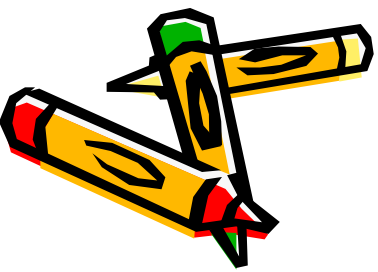
Calculăm centrele de greutate a celor două mulțimi:

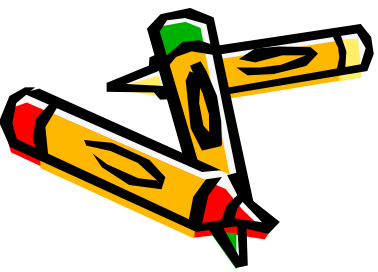
$C1(257.9, 244.9, 104.8, 200.4)$

$C2(528.6, 572.3, 200.3, 268.3)$.

Evaluăm distanțele dintre Pi și centroizii găsiți:

dacă $d(C1, Pi) < d(C2, Pi)$ pacientul Pi va aparține clusterului de centru $C1$.





	$d(C1, Pi)$	$d(C2, Pi)$	cluster
P1	116.4	501.1	M_1'
P2	95.2	420.7	M_1'
P3	75.9	461.1	M_1'
P4	193.04	631.5	M_1'
P5	73.8	503.1	M_1'
P6	139.9	570.6	M_1'
P7	166.1	594.7	M_1'
P8	124.7	557.3	M_1'
P9	98.8	509.5	M_1'
P10	151.2	586.4	M_1'
P11	62.7	451.2	M_1'
P12	317.1	139	M_2'
P13	359.4	350.2	M_2'
P14	231.7	304.5	M_1'
P15	834.2	490.9	M_2'
P16	482.6	256.7	M_2'
P17	375.3	165.2	M_2'

Am obținut mulțimile:

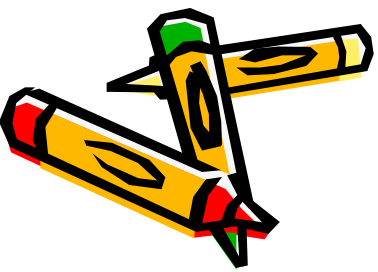
$$M_1' = \{P1, P2, P3, P4, P5, P6, P7, P8, P9, P10, P11, P14\}$$

$$M_2' = \{P12, P13, P15, P16, P17\}$$

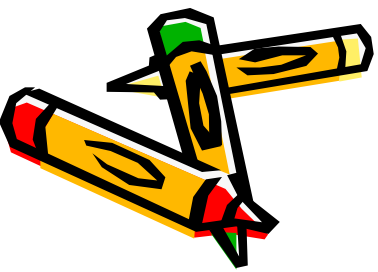
Centroizii acestor mulțimi sunt:

$$C'1 = (205.5, 209.5, 101.3, 198.6)$$

$$C'2 = (546, 526.2, 170.6, 245.4)$$



	$d(C'1, Pi)$	$d(C'2, Pi)$	cluster
P1	90.9	480	M'_1
P2	115.7	400.5	M'_1
P3	87	432.5	M'_1
P4	131.8	602.9	M'_1
P5	55.7	470.3	M'_1
P6	82.2	543.6	M'_1
P7	112	565.5	M'_1
P8	63.3	530.9	M'_1
P9	67.4	483.5	M'_1
P10	98.9	555.4	M'_1
P11	84.3	421.5	M'_1
P12	373.8	138.4	M'_2
P13	413.1	295.2	M'_2
P14	284.4	270.5	M'_2
P15	882.3	534.9	M'_2
P16	540.5	237.7	M'_2
P17	437.6	112.8	M'_2



Avem mulțimile:

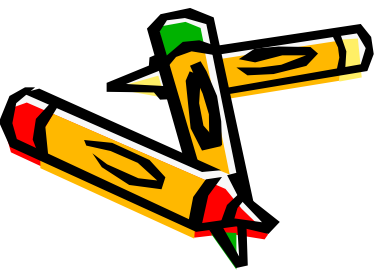
$$M_1'' = \{P1, P2, P3, P4, P5, P6, P7, P8, P9, P10, P11\}$$

$$M_2'' = \{P12, P13, P15, P16, P17\},$$

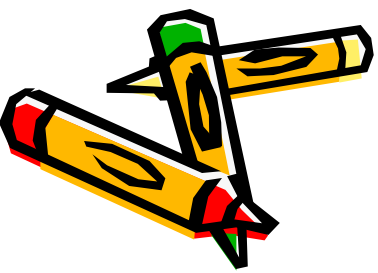
cu centroizii:

$$C''1(183.7, 202.9, 94.5, 188.6)$$

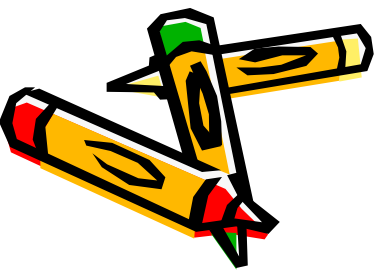
$$C''2(529.3, 485.6, 171.5, 256)$$



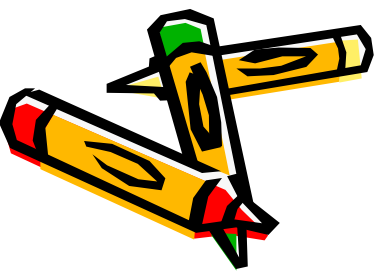
	$d(C''1, Pi)$	$d(C''2, Pi)$	cluster
P1	91.7	444.5	M'_1
P2	127.8	368.1	M'_1
P3	109.9	391.7	M'_1
P4	110.9	565.4	M'_1
P5	72.3	430.7	M'_1
P6	56.4	508.4	M'_1
P7	89.8	531.3	M'_1
P8	40.3	494,4	M'_1
P9	63.3	448.4	M'_1
P10	88	516.7	M'_1
P11	93.6	388.2	M'_1
P12	395.6	113.8	M'_2
P13	436.6	262.8	M'_2
P14	310.3	225.4	M'_2
P15	895.8	573.2	M'_2
P16	565.5	213..9	M'_2
P17	459.6	103.9	M'_2



Am obținut aceleași clustere, ceea ce înseamnă încheierea procesului.



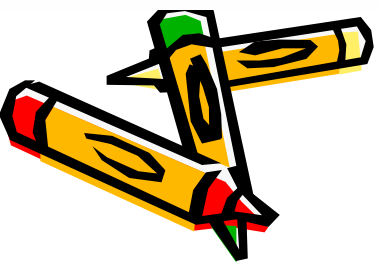
Clustering ierarhic



Metoda de *clustering ierarhic* constă în gruparea obiectelor în mod iterativ, utilizând o anumită legătură - *linkage*.

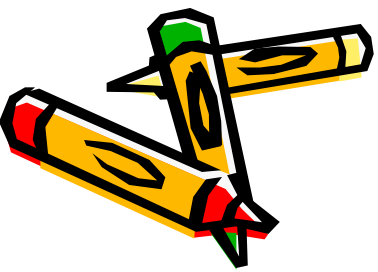
Există două tipuri de metode:

- *aglomerativă*, care constă într-o serie de fuziuni a celor *n* obiecte în grupuri (cea mai utilizată)
- *de divizare*, care separă succesiv cele *n* obiecte în grupuri mai fine.



Putem defini distanța dintre două clustere în mai multe moduri:

- *Single linkage*
- *Complete linkage*
- *Average linkage*
- *Centroid linkage*
- *Ward linkage*

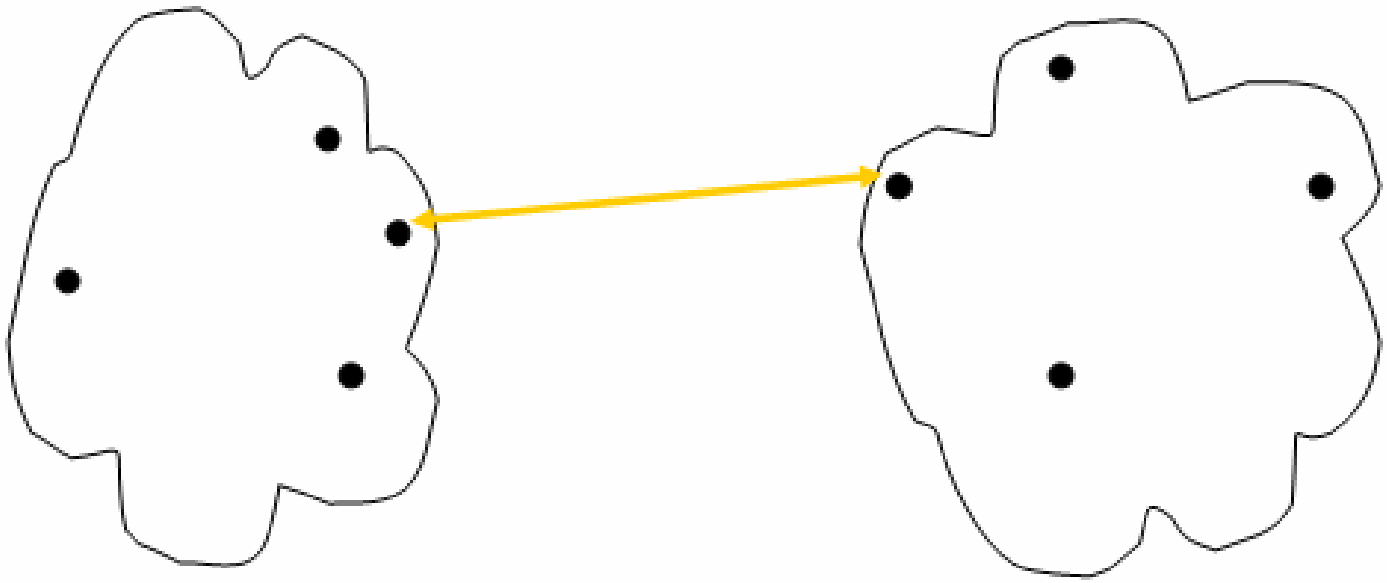


single linkage

- *Single linkage*, caz în care distanța dintre clustere este determinată de distanța între cele mai apropiate obiecte:

$$d(A,B) = \min\{d(\mathbf{x}_k, \mathbf{y}_j), \mathbf{x}_k \in A, \mathbf{y}_j \in B\},$$

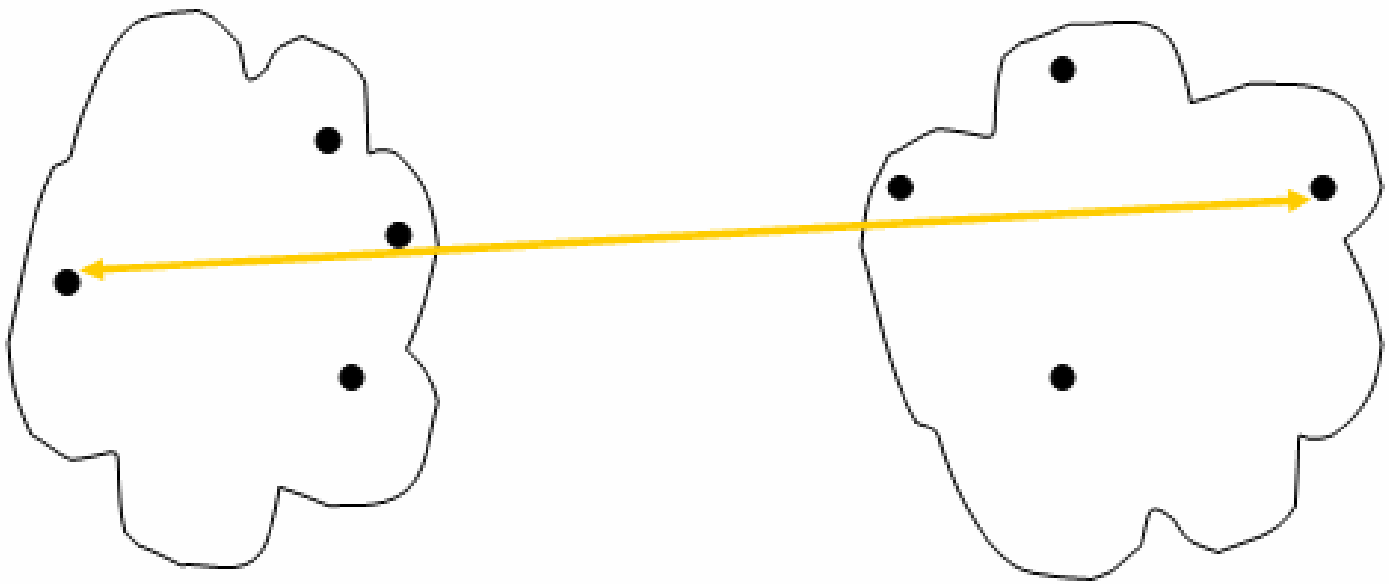
unde am notat cu *A* și *B* cele două clustere



complete linkage

- *Complete linkage*, caz în care distanța dintre clustere este determinată de distanța între cele mai îndepărtate obiecte:

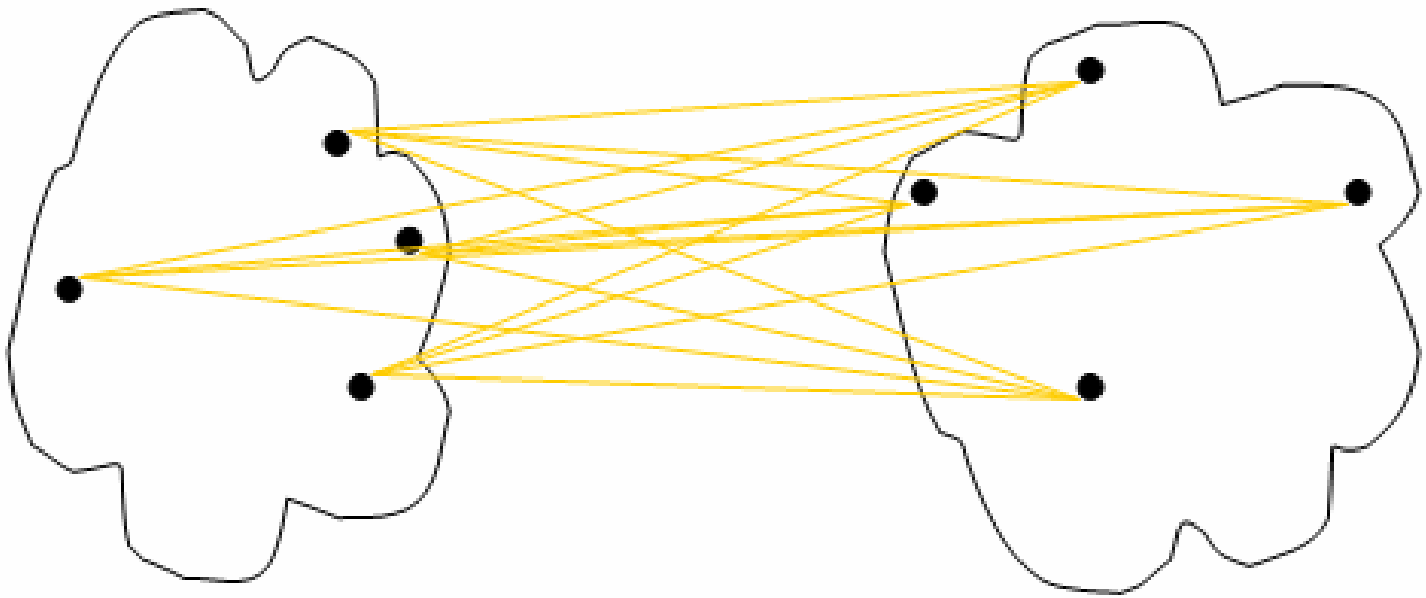
$$d(A, B) = \max\{d(\mathrm{x}_k, \mathrm{y}_j), \mathrm{x}_k \in A, \mathrm{y}_j \in B\}$$



average linkage

- *average linkage*, caz în care distanța este calculată

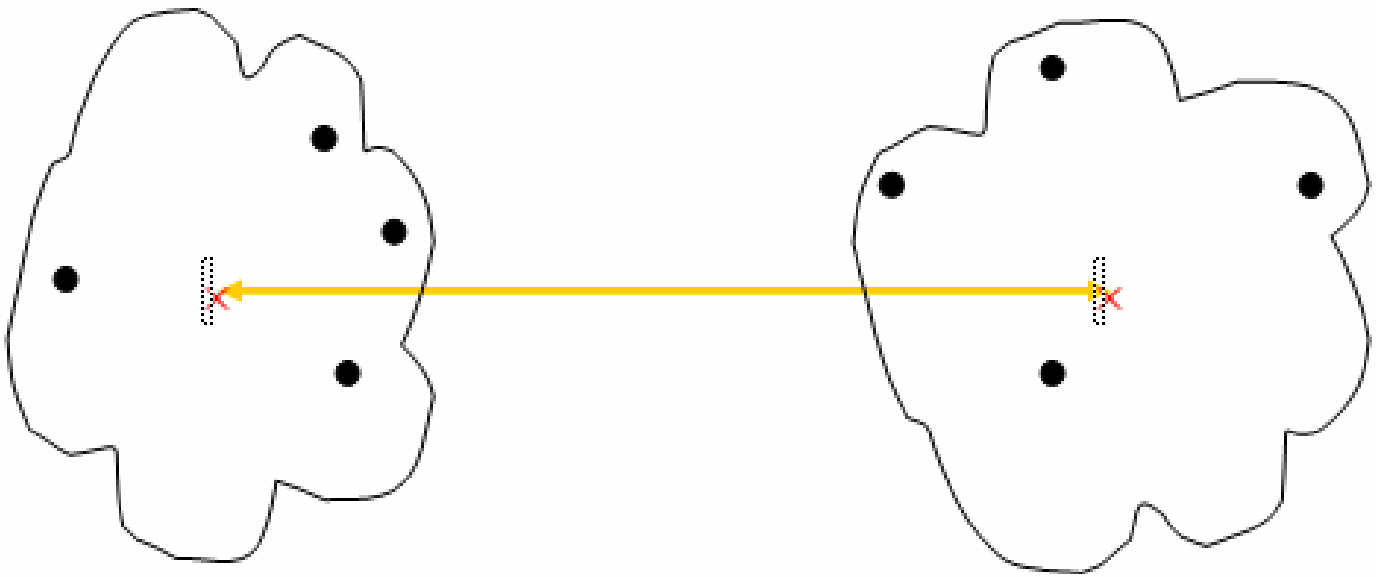
conform formulei: $d(A,B) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} d(x_i, y_j)}{m \cdot n}$,



centroid linkage

- *centroid linkage* este distanța dintre centroizi. Este de mentionat faptul că aceasta poate fi folosita doar dacă se utilizează distanța euclidiană.

$d_{cen}(A,B) = d(\bar{x}_A, \bar{x}_B)$ unde $\bar{x}_A = \frac{1}{n}\sum_{i=1}^{n} x_A^i$.



ward linkage

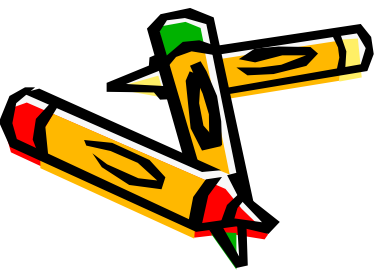
Suma pătratelor distanțelor din interiorul unui cluster este definită ca fiind suma pătratelor distanțelor dintre obiectele existente în cluster şi centroidul acestuia. Ward linkage se defineşte prin formula:

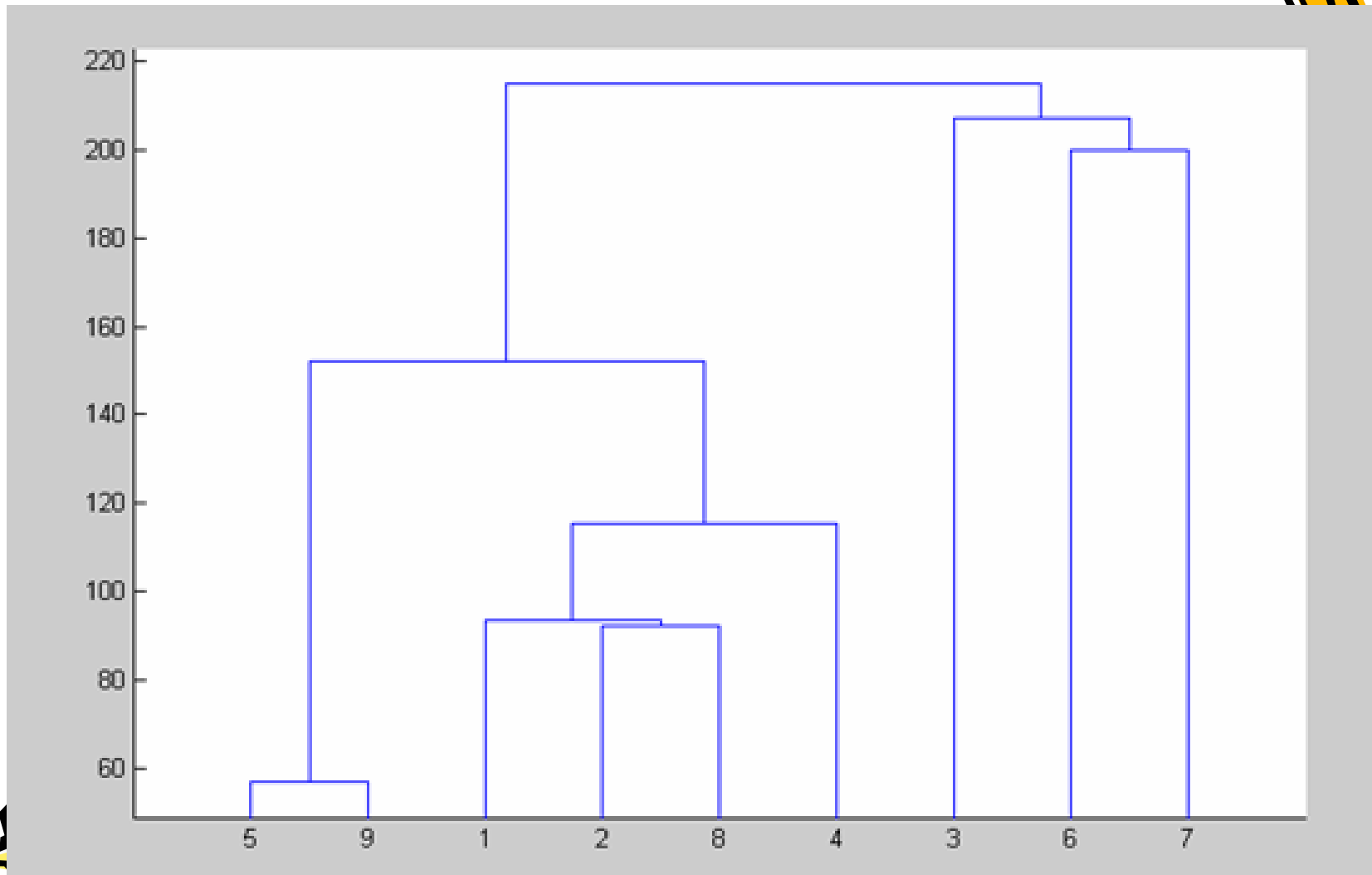
$$d_W(A,B) = \frac{n \cdot m d_{cen}(A,B)}{n+m}$$

unde clusterul A conține m obiecte şi clusterul B conține m obiecte

dendrograma

Clustering-ul ierarhic permite o reprezentare printr-o diagramă bidimensională, numită *dendrogramă*, care ilustrează fuziunile sau divizările din fiecare pas al procesului efectuat.




220
200
180
160
140
120
100
80
60
5
9
1
2
8
4
3
6
7

exemplu

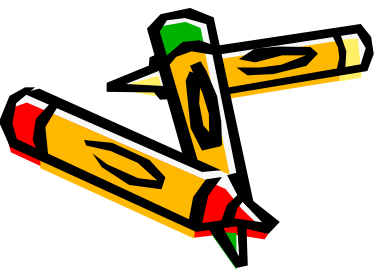
Vom prezenta o aplicație referitoare la cancerul hepatic.:
Considerăm un lot de pacienți dintre care 3 au cancer hepatic (HCC). Pentru a decide dacă un pacient are sau nu cancer hepatic, studiile clinice arată că se iau în considerare anumite enzime serice (de obicei un număr de 15).
Pentru simplificarea calculelor, le vom lua în considerare pe cele mai importante din punct de vedere clinic:
alkaline phosphatase (FA), g-glutamyl transferase (gGT), leucine amino peptidase (LAP) și colesterol (C)

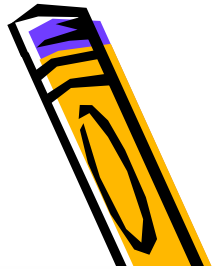
	FA	gCT	LAP	C	diagnostic
P1	162	255	74	258	non HCC
P2	210	324	93	220	non HCC
P3	422	488	183	292	HCC
P4	259	208	115	266	non HCC
P5	129	114	89	171	non HCC
P6	607	259	65	275	HCC
P7	446	283	176	309	HCC
P8	236	270	88	150	non HCC
P9	180	119	114	171	non HCC

Avem 9 clustere pentru început.

Calculăm distanțele între vectorii standardizați și vom alcătui o matrice, matricea distanțelor, unde elementul $a_{ij} = d(Pi, Pj)$.

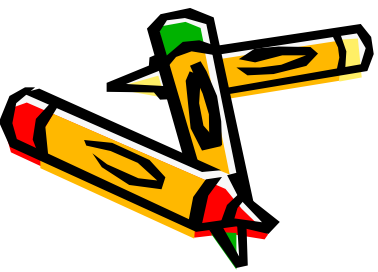
Vom determina astfel vectorii cei mai apropiați, în sensul distanței, pe care îi vom grupa în clustere:





0	94.2	367.3	115.6	169.6	445,4	307.3	132.5	162.3
94.2	0	291.7	135.8	230.4	407	268.7	92.3 *	211.9
367.3	291.7	0	322	499.2	317.6	207,2	333.6	460.2
115.6	135.8	322	0	188.2	355.3	214.8	136.2	146.2
169.6	230.4	499.2	188.2	0	510.8	394.5	190.3	57.3 *
445,4	407	317.6	355.3	510.8	0	199.9 *	392.3	461.7
307.3	268.7	207,2	214.8	394.5	199.9 *	0	278	343.7
132.5	92.3 *	333.6	136.2	190.3	392.3	278	0	166
162.3	211.9	460.2	146.2	57.3 *	461.7	343.7	166	0

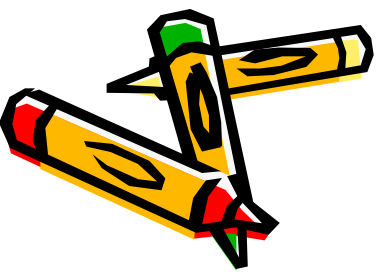
Alăturând pacienții 2&8, 5&9, 6&7, obținem 6 clustere (deocamdată pacienții 1,3 și 4 formează clustere separate).



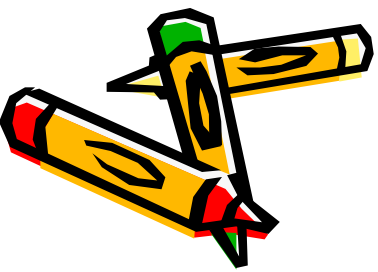
Avem următorii vectori:

- P1(162, 255, 74, 258);
- P3 (422, 488, 183, 292);
- P4(259, 208,115,266);
- C1(223, 297, 90.5, 185) corespunzător clusterului 2&8;
- C2(154.5, 116.5, 101.5, 176) corespunzător clusterului 5&9;
- C3(526.5, 271, 120.5, 292), corespunzător clusterului 6&7

Prezentăm matricea distanțelor dintre aceşti vectori:



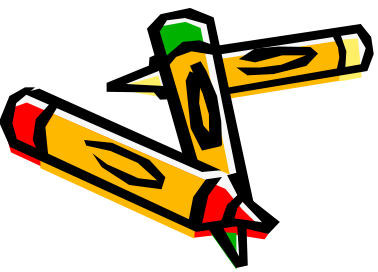
$$\begin{pmatrix} 0 & 367.3 & 115.6 & 105.2^{*} & 163.5 & 369.4 \\ 367.3 & 0 & 322 & 310 & 479.2 & 248.8^{*} \\ 115.6 & 322 & 0 & 128 & 166 & 276.1 \\ 105.2^{*} & 310 & 128 & 0 & 193.6 & 324.2 \\ 163.5 & 479.2 & 166 & 193.6 & 0 & 419.6 \\ 369.4 & 248.8^{*} & 276.1 & 324.2 & 419.6 & 0 \end{pmatrix}$$



Formăm astfel noi clustere:1&2&8, 3&6&7, 5&9 şi pacientul 4

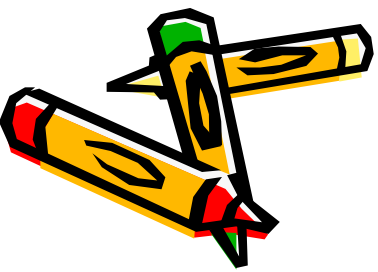
Continuăm procedeul cu următorii vectori:

- P4(259, 208,115,266);
- C2(154.5, 116.5, 101.5, 176) ;
- C4(192.5 276. 82.2 221.5) corespunzător clusterului 1&2&8;
- C5 (474.2 379.5 151.7 292) corespunzător clusterului 3&6&7.



Calculăm matricea distanțelor:

$$\begin{pmatrix} 0 & 219.4 & 52.2\,* & 285.5 \\ 219.4 & 0 & 171.2 & 432.9 \\ 52.2\,* & 171.2 & 0 & 316 \\ 285.5 & 432.9 & 316 & 0 \end{pmatrix}$$



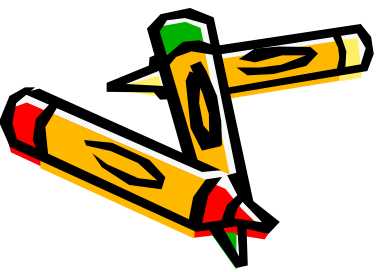
$$\begin{pmatrix} 0 & 432.8 & 159.4^{*} \\ 432.8 & 0 & 292.9 \\ 159.4^{*} & 292.9 & 0 \end{pmatrix}$$

și astfel obținem cele două clustere, corespunzătoare celor doua boli hepatice

(1&2&4&5&8&9) și (3&6&7)

Testând cu mulțimea de date, observăm că rezultatul este validat 100%.

În final, este necesară aprecierea validității modelului, lucru realizabil prin repetarea procesului folosind alte măsuri de similaritate și eventual alte tehnici de clustering.



Clusteringul ierarhic este o metodă care studiază modul de grupare a obiectelor dintr-o mulțime, prin crearea unui cluster tree (rbore de cluster).

Acest arbore nu este o mulțime de clustere, ci o ierarhie pe mai multe nivele în care clusterele de pe un anume nivel se unesc cu alte clustere la nivelul superior următor, fapt ce ne permite să decidem care nivel de clusterizare este corespunzător aplicației noastre.

Folosind aceste tehnici de clusterizare descoperim clase „naturale" în care se plasează obiectele din baza de date şi nu realizăm o nouă ordine în structura datelor.