Adatbányászat: Klaszterezés Alapfogalmak és algoritmusok

8. fejezet

Tan, Steinbach, Kumar

Bevezetés az adatbányászatba

előadás-fóliák

fordította

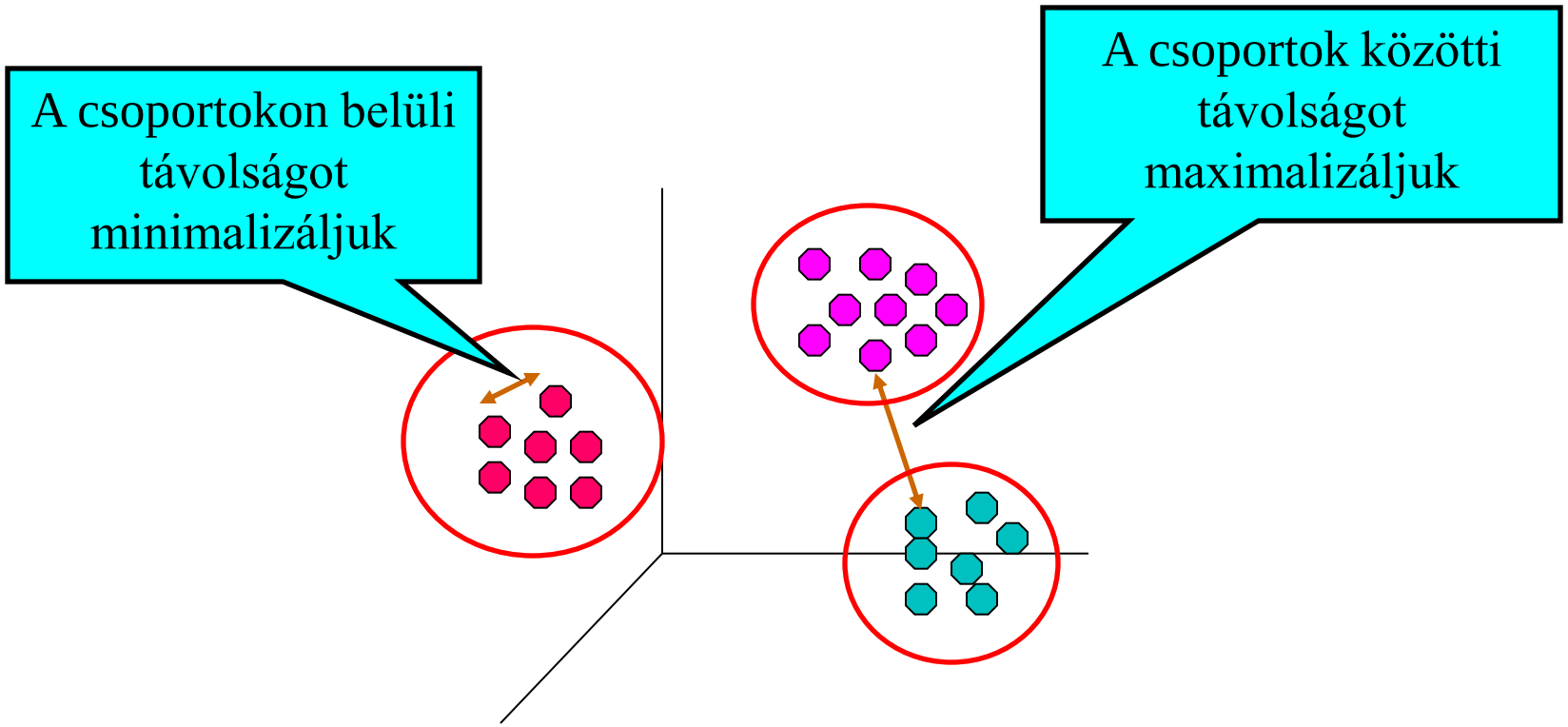
Ispány Márton

Logók és támogatás

A tananyag a TÁMOP-4.1.2-08/1/A-2009-0046 számú Kelet-magyarországi Informatika Tananyag Tárház projekt keretében készült. A tananyagfejlesztés az Európai Unió támogatásával és az Európai Szociális Alap társfinanszírozásával valósult meg.

Mi a klaszterezés (csoportosítás)?

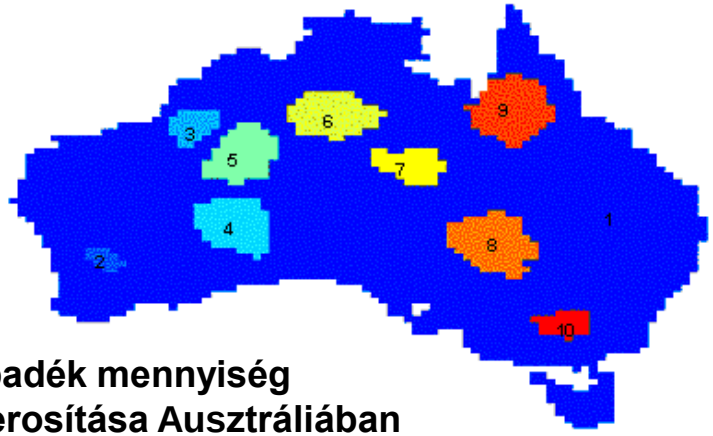
- Találjunk olyan csoportokat objektumok egy halmazában, hogy az egy csoportban lévő objektumok egymáshoz hasonlóak, míg a más csoportokban lévők pedig különbözőek.



A klaszterezés alkalmazásai

- **Megértés**
  - Böngészésnél kapott kapcsolódó dokumentumok csoportjai, hasonló funkcionalitással bíró gének és fehérjék csoportjai, hasonló ármozgású részvények csoportjai.
- **Összegzés**
  - Nagy adatállományok méretének csökkentése.

	Feltárt klaszterek	*Ipari csoport*
1	Applied-Matl-LE,Bay-Network-LE,3-COM-LE, Cabletron-Sys-LE,CISCO-LE,HP-LE, DSC-Comm-LE,INTEL-LE,LSI-Logic-LE, Micron-Tech-LE,Texas-Inst-LE,Tellabs-Inc-LE, Natl-Semiconduct-LE,Oracl-LE,SGI-LE, Sun-LE	Technológia 1-LE
2	Apple-Comp-LE,Autodesk-LE,DEC-LE, ADV-Micro-Device-LE,Andrew-Corp-LE, Computer-Assoc-LE,Circuit-City-LE, Compaq-LE, EMC-Corp-LE, Gen-Inst-LE, Motorola-LE,Microsoft-LE,Scientific-Atl-LE	Technológia 2-LE
3	Fannie-Mae-LE,Fed-Home-Loan-LE, MBNA-Corp-LE,Morgan-Stanley-LE	Bank-LE
4	Baker-Hughes-FEL,Dresser-Inds-FEL, Halliburton-HLD-FEL,Louisiana-Land-FEL, Phillips-Petro-FEL,Unocal-FEL, Schlumberger-FEL	Olaj-FEL



A csapadék mennyiség klaszterosítása Ausztráliában

Mi nem klaszterezés?

- Felügyelt osztályozás
  - Adott egy osztályozó attributum.
- Egyszerű szegmentáció
  - Osszuk fel a diákokat különböző csoportokba a vezeték nevük alapján ábécé szerint.
- Egy lekérdezés eredményei
  - A csoportok egy külső specifikáció eredményei.
- Gráf partícionálás
  - Kölcsönös fontosság vagy együttműködés az objektumok (rekordok) között, azonban különböző területeken.

A klaszter fogalma nem egyértelmű

Hány klaszter?

Hat klaszter

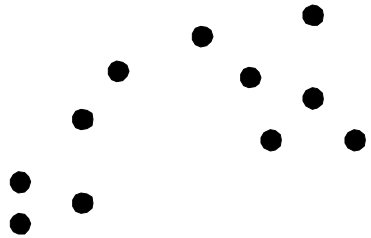
Két klaszter

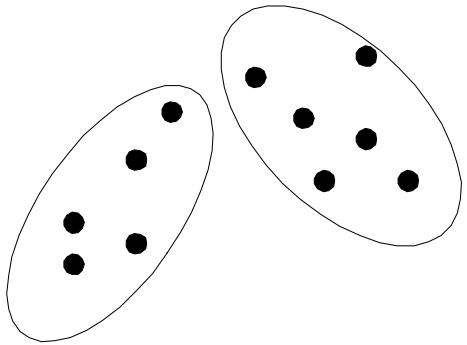
Négy klaszter

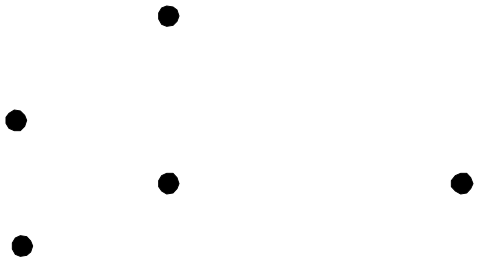
Klaszterezések fajtái

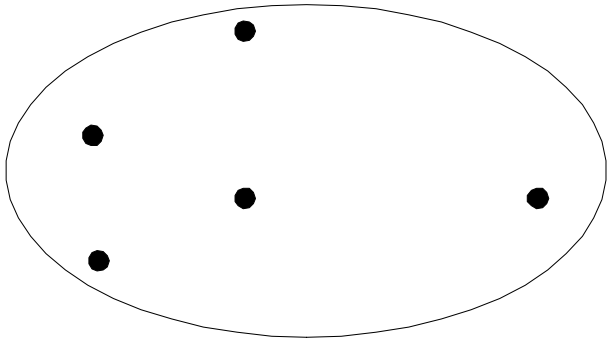
- Egy klaszterosítás klaszterek (csoportok) egy halmaza.
- Fontos különbséget tenni a hierarchikus és a felosztó klaszterezés között.
- Felosztó klaszterezés:
  - Az objektumok felosztása nem átfedő részhalmazokra (klaszterekre) úgy, hogy minden objektum pontosan egy részhalmazban szerepelhet.
- Hierarchikus klaszterezés:
  - Egymásba ágyazott klaszterek egy hierarchikus fába szervezett halmaza.

Felosztó klaszterezés





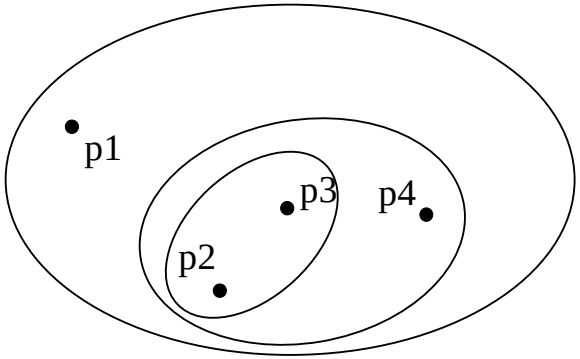




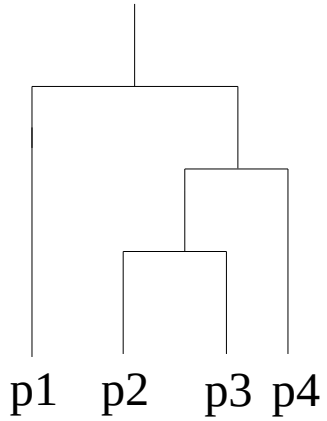
Eredeti pontok

Felosztó klaszterezés

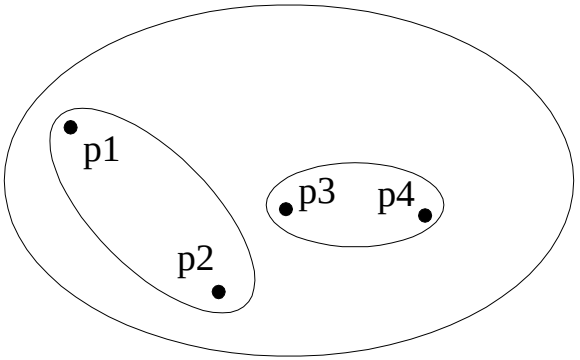
Hierarchikus klaszterezés



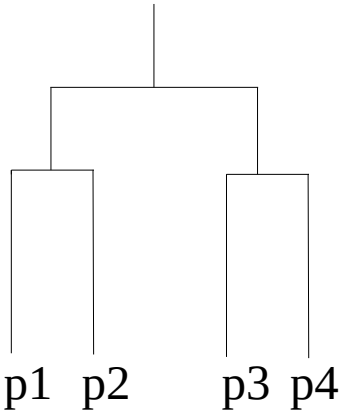
Hagyományos hierarchikus klaszterezés



Hagyományos dendrogram



Nem-hagyományos hierarchikus klaszterezés



Nem-hagyományos dendrogram

További különbségek klaszterek között

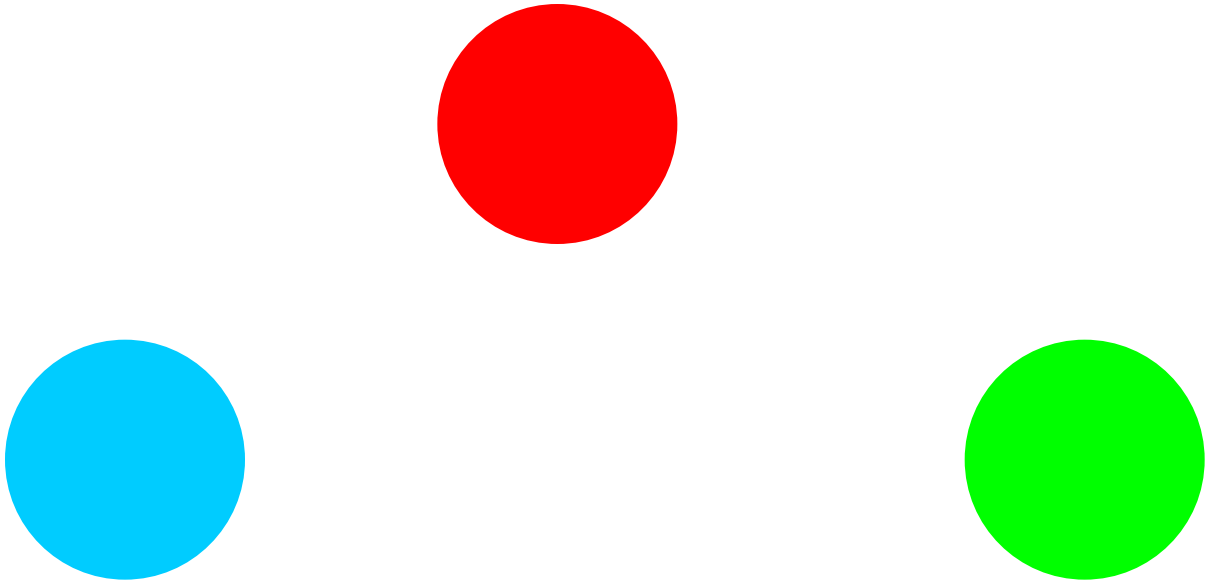
- Kizáró vagy nem-kizáró
  - A nem-kizáró kleszterezésnél a pontok több klaszterhez is tartozhatnak.
  - Egy pont, a ,,határ” pont, több osztályt is képviselhet.
- Fuzzy vagy nem-fuzzy
  - A fuzzy klaszterezésnél egy pont az összes klaszterhez tartozik 0 és 1 közötti súllyal.
  - A súlyok összege 1.
  - A valószínűségi klaszterezés hasonló tulajdonsággal bír.
- Részleges vagy teljes
  - Bizonyos esetekben az adatok egy részét akarjuk klaszterezni.
- Heterogén vagy homogén
  - Nagyon különböző méretű, alakú és sűrűségű klaszterek.

Klaszterek típusai

- Jól elválasztott klaszterek
- Középpont alapú klaszterek
- Összefüggő klaszterek
- Sűrűség alapú klaszterek
- Tulajdonság vagy fogalom alapú klaszterek
- Egy célfüggvény által leírt klaszterek

Klaszter-típusok: Jól elválasztott

- Jól elválasztott klaszterek:
  - Egy klaszter pontoknak olyan halmaza, hogy a klaszter bármely pontja közelebb van (vagy hasonlóbb) a klaszter összes további pontjához mint bármelyik nem klaszterbeli pont.



3 jól elválasztott klaszter

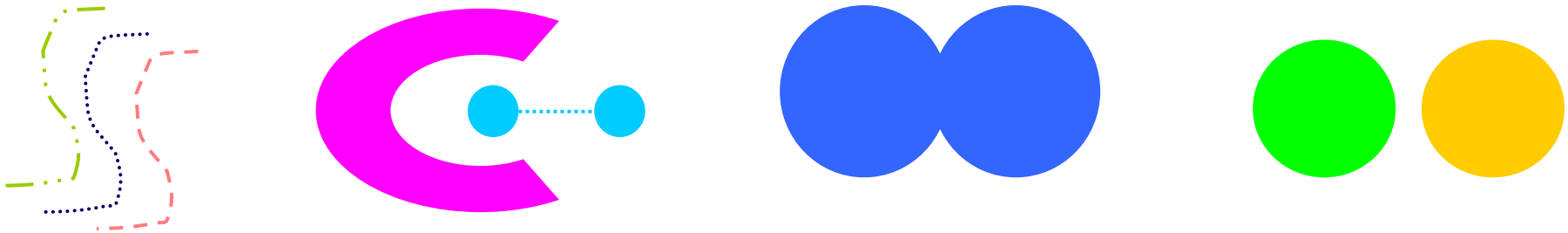
Klaszter-típusok: Középpont alapú

- Középpont alapú
  - Egy klaszter objektumoknak egy olyan részhalmaza, hogy egy klaszterbeli objektum közelebb van (hasonlóbb) a klaszter ,,középpontjához” mint bármelyik más klaszterközépponthoz.
  - Egy klaszter középpontja gyakran az ún centroid, a klaszterbeli összes pont átlaga, vagy a medoid, a klaszter legreprezentatívabb pontja.

4 középpont alapú klaszter

Klaszter-típusok: Összefüggő

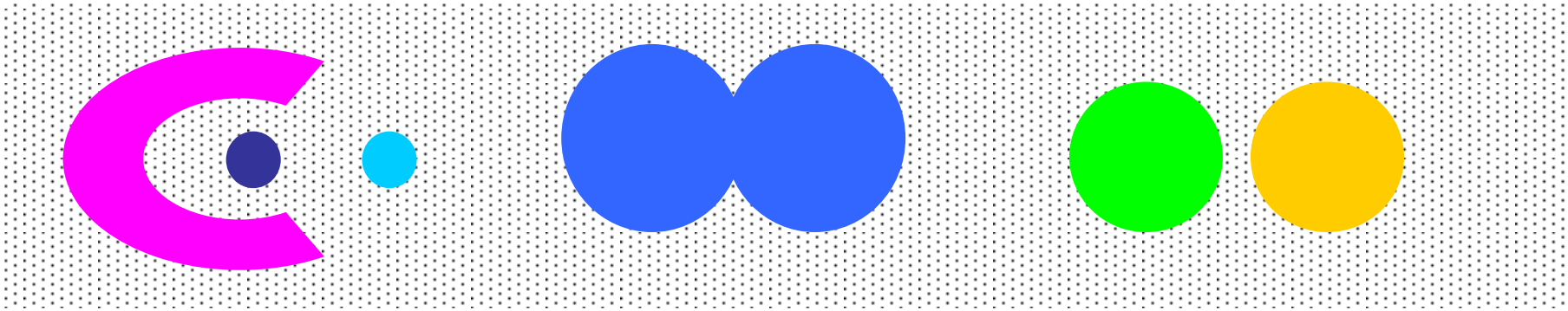
- Összefüggő klaszter (legközelebbi szomszéd)
  - Egy klaszter pontoknak olyan halmaza, hogy egy klaszterbeli pont közelebb van (hasonlóbb) a klaszter más pontjaihoz mint bármelyik nem klaszterbeli ponthoz.



8 összefüggő klaszter

Klaszter-típusok: Sűrűség alapú

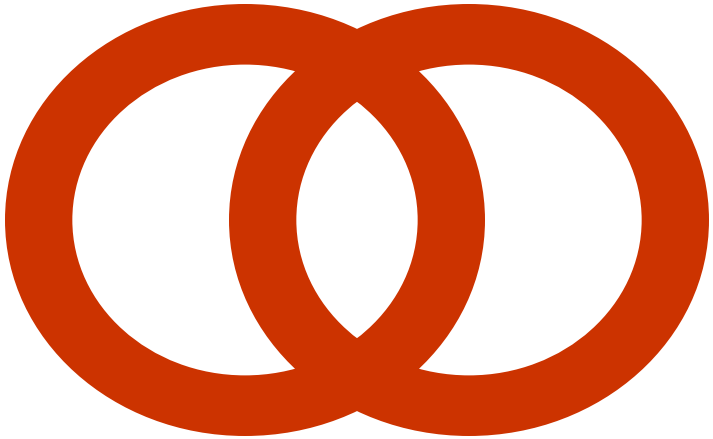
- Sűrűség alapú
  - A klaszter sűrűn elhelyezkedő pontok halmaza, amelyet alacsony sűrűségű tartományok választanak el hasonlóan nagy sűrűségű klaszterektől.
  - Akkor használjuk ha a klaszterek szabálytalanok vagy egymást átfedőek, illetve hiba vagy kiugró értékek vannak.



6 sűrűség alapú klaszter

Klaszter-típusok: Fogalom alapú

- Közös tulajdonsággal bíró vagy fogalmi alapú
  - Keressünk olyan klasztereket, amelyek valamilyen közös tulajdonságon osztoznak, illetve egy speciális fogalmat jelenítenek meg.



2 átfedő kör

Klaszter-típusok: Célfüggvény szerinti

- Egy célfüggvény által definiált klaszterek
  - Keressük meg azokat a klasztereket, amelyek egy célfüggvényt minimalizálnak vagy maximalizálnak.
  - Számoljuk össze az összes klaszterosítást és értékeljük ki minden lehetséges klaszterosítás ,,jóságát” a célfüggvény alapján. (NP-nehéz feladat)
  - Lehetnek globális és lokális célfüggvények.
    - A hierarchikus klaszterosítási algoritmusok általában több lokális célfüggvénnyel dolgoznak.
    - A felosztó módszerek általában egy globális célfüggvényt használnak.
  - A globális célfüggvényes megközelítés egy paraméteres modellt illeszt az adatokra.
    - A modell paramétereit az adatokból határozzuk meg.
    - A keverék modellek feltételezik, hogy az adatok valószínűségi eloszlások egy ,,keverékét” követik.

Klaszter-típusok: Célfüggvény szerinti

- Képezzük le a klaszterezési feladatot egy másik tartományba és oldjuk meg a kapcsolt feladatot abban a tartományban.
  - A közelségi mátrix egy súlyozott gráfot definiál, ahol a csúcsokat kell klaszterezni és a súlyozott élek reprezentálják a pontok közötti hasonlóságot.
  - A klaszterezés a gráf összefüggő komponensekre való felbontásával ekvivalens, a komponensek lesznek a klaszterek.
  - A klasztereken belüli élek súlyát minimalizálni, a klaszterek közötti élek súlyát pedig maximalizálni szeretnénk.

A fontosabb adatjellemzők

- A közelségi és sűrűségi mértékek típusai
  - Ezek származtatott mértékek, de alapvetőek a klaszterezés szempontjából.
- Ritkaság
  - Megszabja a hasonlóság típusát
  - Hozzájárul a hatékonysághoz
- Attributum-típusok
  - Megszabja a hasonlóság típusát
- Adat-típusok
  - Megszabja a hasonlóság típusát
  - Egyéb jellemzők, pl. autokorreláció
- Dimenzió probléma
- Hiba és kiugró adatok
- Eloszlás típusok

Klaszterezési algoritmusok

- K-közép módszer és változatai
- Hierarchikus klaszterezés
- Sűrűség alapú klaszterezés

K-közép (McQueen) módszer

- Felosztó megközelítés.
- Minden klaszterhez egy középpontot (centroid) rendelünk.
- Minden pontot ahooz a klaszterhez rendelünk, amelynek a középpontjához a legközelebb van.
- A klaszterek száma, K, adott kell, hogy legyen.
- Az algoritmus egyszerű.

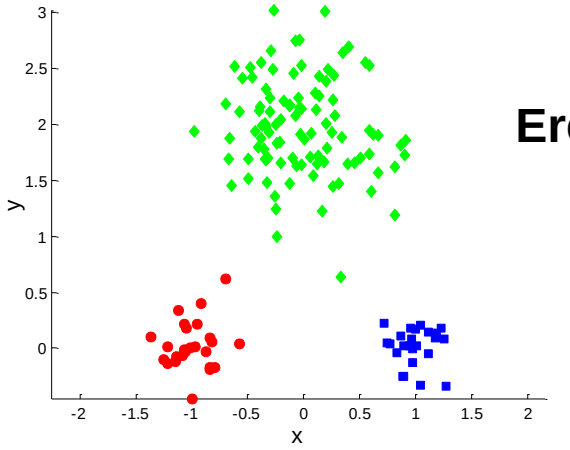
Algoritmus. Alap *K*-közép módszer

1. Válasszunk ki *K* kezdeti középpontot.
2. **repeat**
3. Hozzunk létre *K* klasztert a pontoknak a legközelebbi középpontokhoz való hozzárendelésével.
4. Számoljuk újra a középpontot minden klaszternél.
5. **until** A középpontok nem változnak.

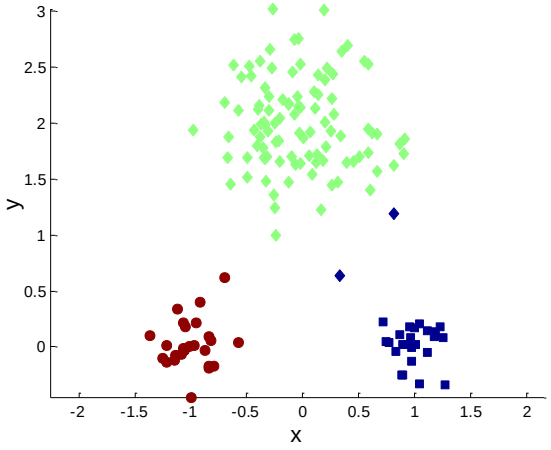
K-közép módszer – Részletek

- A kezdeti középpontok általában véletlenszerűek.
  - A kapott klaszterek futásról futásra változhatnak.
- A középpont (általában) a klaszterbeli pontok átlaga.
- A ,,közelséget” mérhetjük az euklideszi távolsággal, koszinusz hasonlósággal, korrelációval stb.
- A K-közép módszer konvergál a fenti általános hasonlósági mértékekre.
- A konvergencia legnagyobb része az első néhány iterációban megtörténik.
  - Általában a leállási feltétel arra módosul, hogy ,,Viszonylag kevés pont vált klasztert”
- Komplexitás: O(n * K * I * d)
  - n = pontok száma, K = klaszterek száma,
    I = iterációk száma, d = attributumok száma

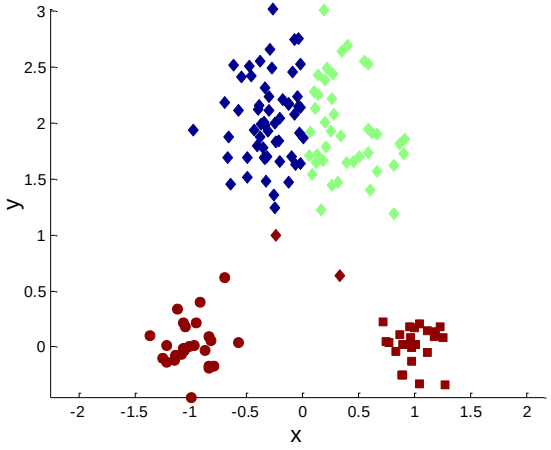
Két különböző K-közép klaszterezés



Eredeti pontok

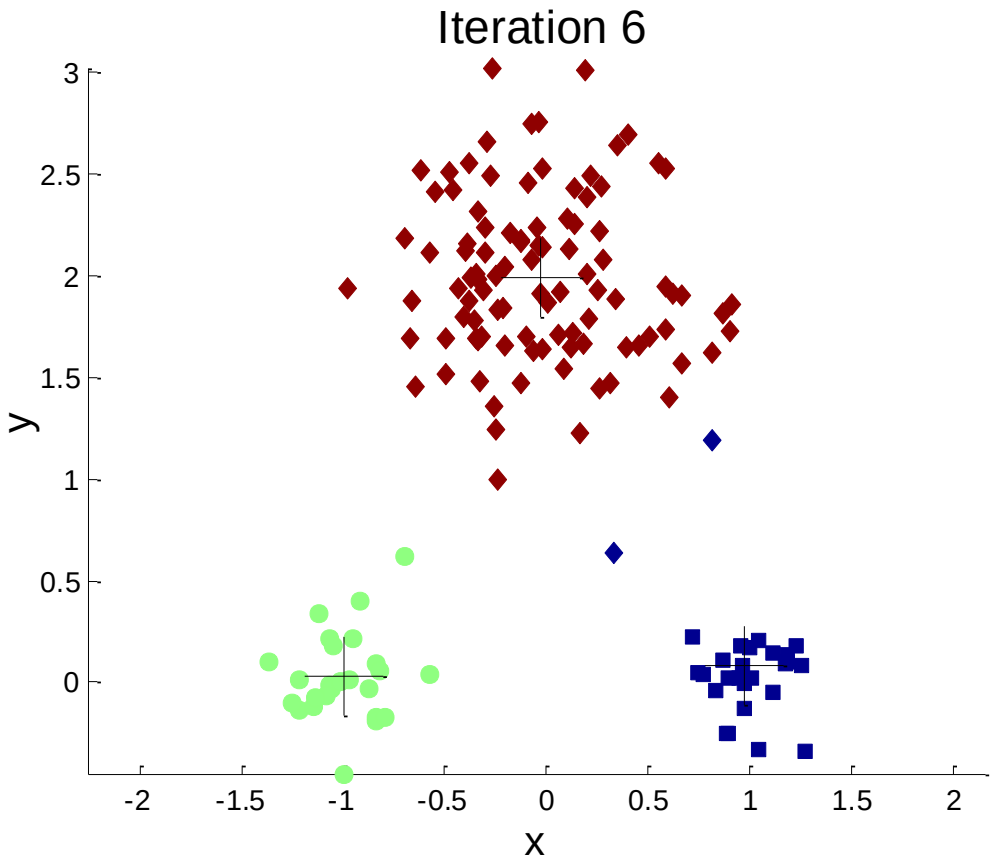


Optimális megoldás

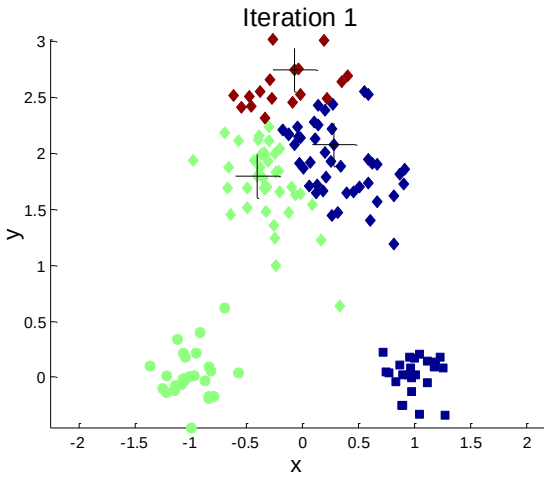


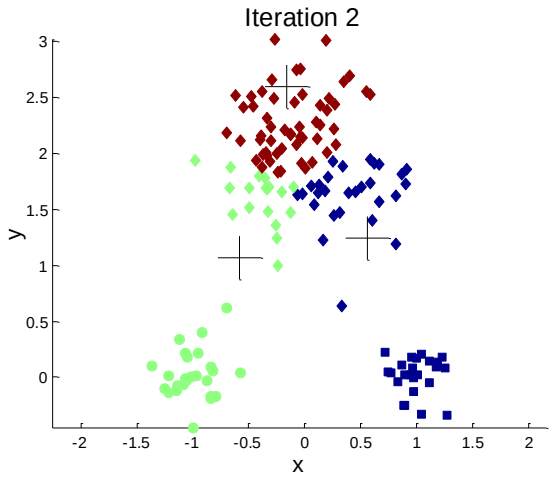
Lokális megoldás

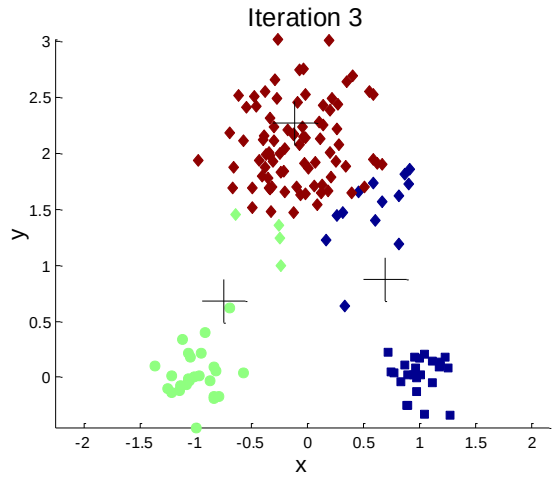
Kezdeti középpontok megválasztása

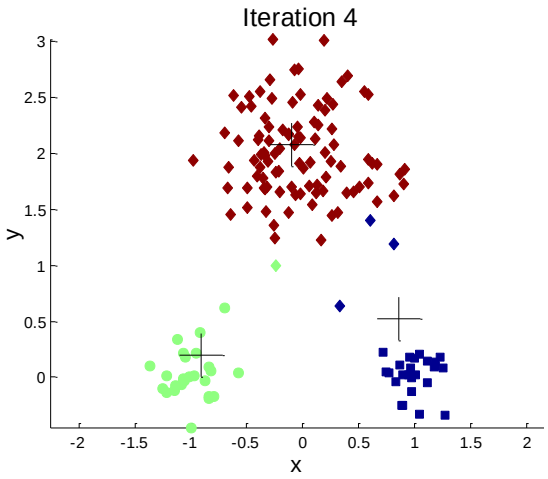


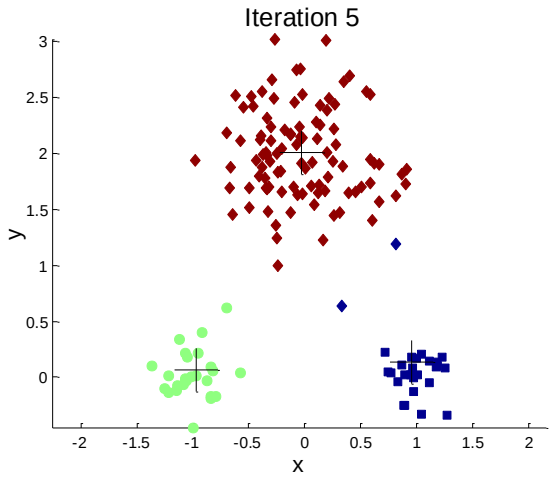
Kezdeti középpontok megválasztása

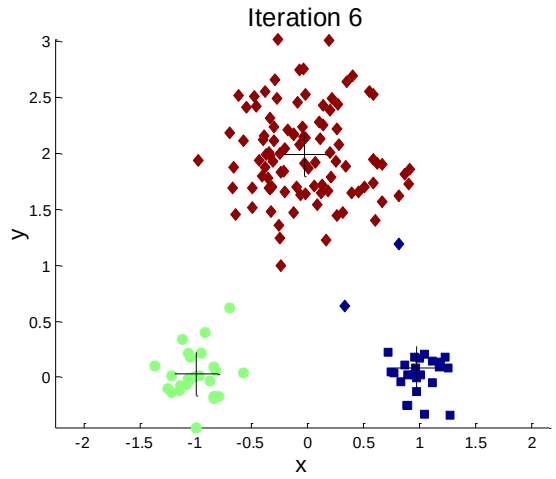












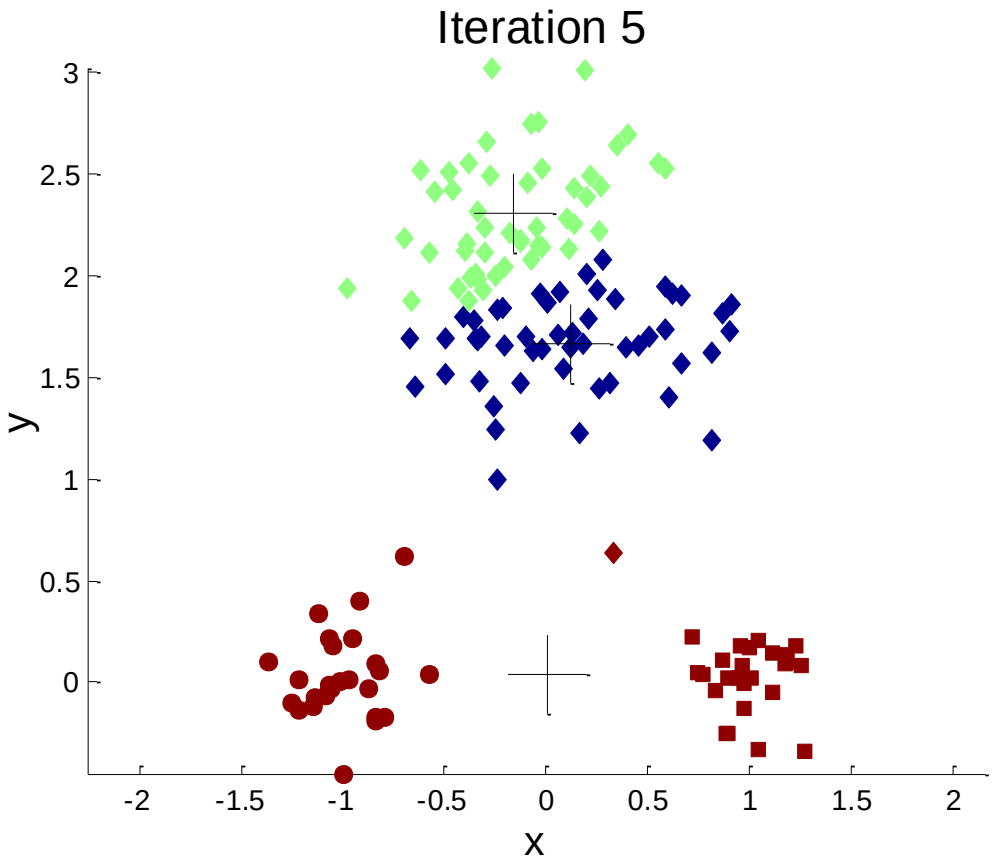
K-közép módszer kiértékelése

- Az általános mérőszám: hiba négyzetösszeg (SSE - Sum of Squared Error)
  - Minden pontra a hiba a legközelebbi klasztertől való távolság.
  - Az SSE ezen hibák négyzetének összege:

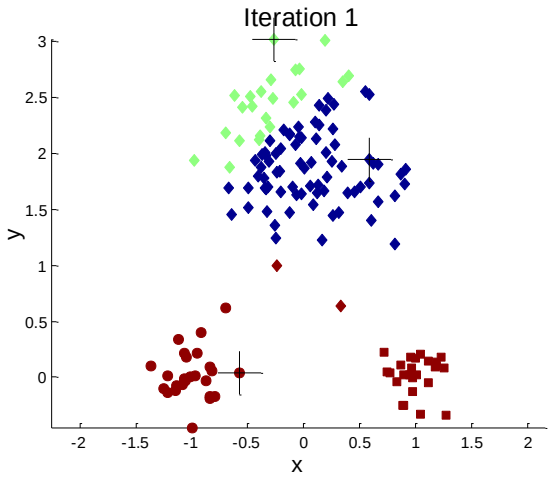
$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist^2(m_i, x)$$

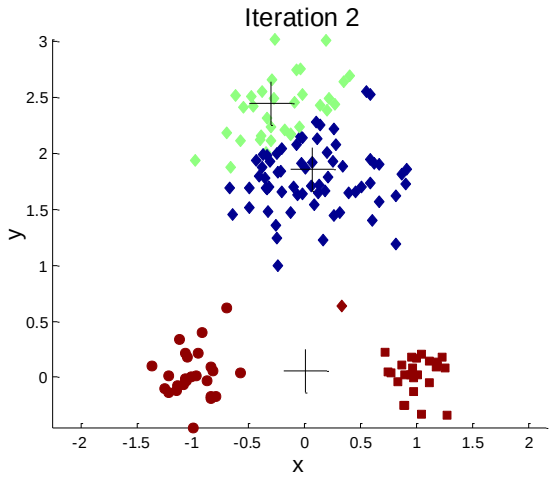
  - x egy pont a C_i klaszterben, m_i a C_i klaszter reprezentánsa
    - általában m_i a klaszter középpontja (átlaga)
  - Két klaszterezés közül azt választjuk, amelyiknek kisebb a hibája.
  - A legegyszerűbb módja az SSE csökkentésének a K növelése.
    - Egy jó klaszterezésnek kisebb K mellett lehet kisebb SSE-je mint egy rosszabb klaszterezésnek nagyobb K mellett.

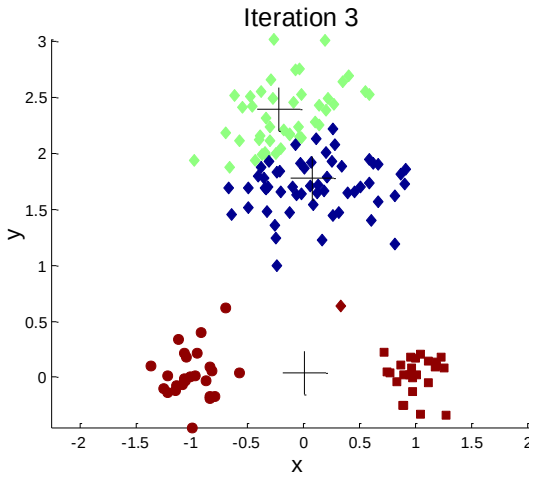
Kezdeti középpontok megválasztása

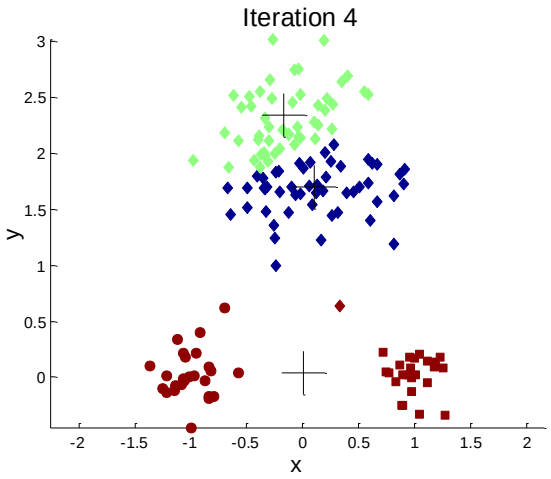


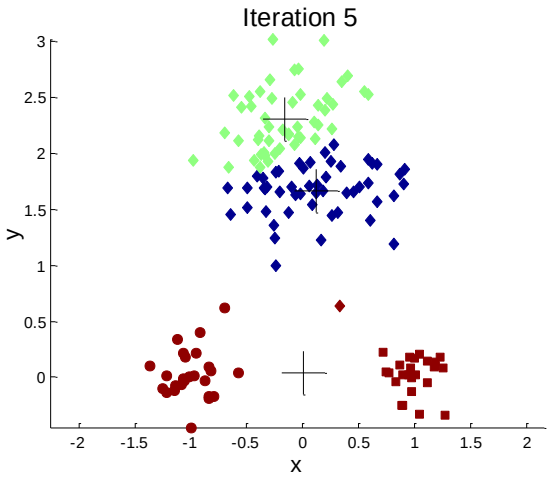
Kezdeti középpontok megválasztása











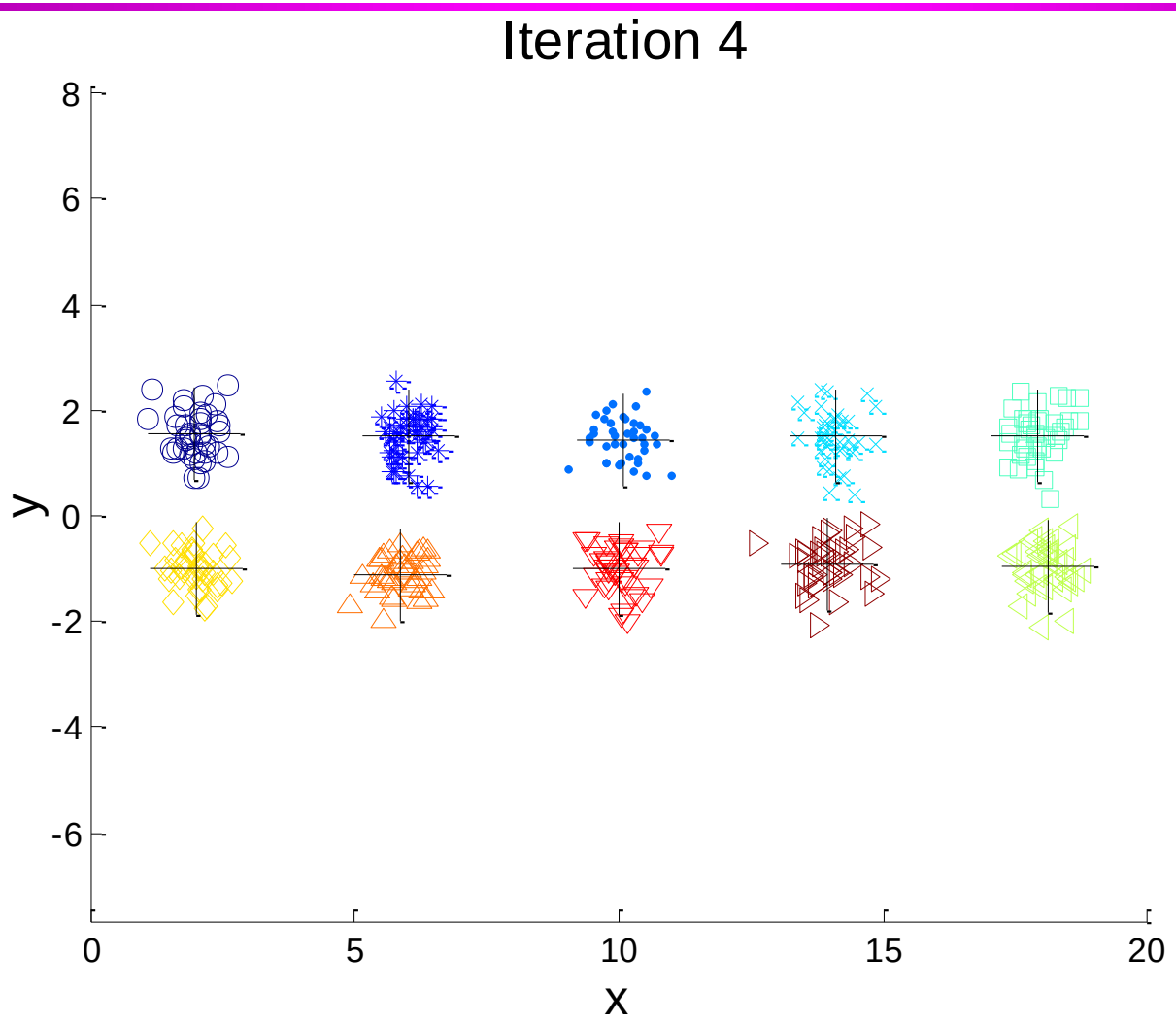
A kezdeti középpontok problémája

- Ha adott K ,,igazi” klaszter, akkor annak esélye, hogy minden klaszterből választunk középpontot kicsi.
  - Ez az esély viszonylag kicsi ha K nagy
  - Ha a klaszterek ugyanolyan méretűek, pl n, akkor a klasszikus formula alapján (jó esetek száma/összes eset)

$$P = \frac{K!n^K}{(Kn)^K} = \frac{K!}{K^K}$$

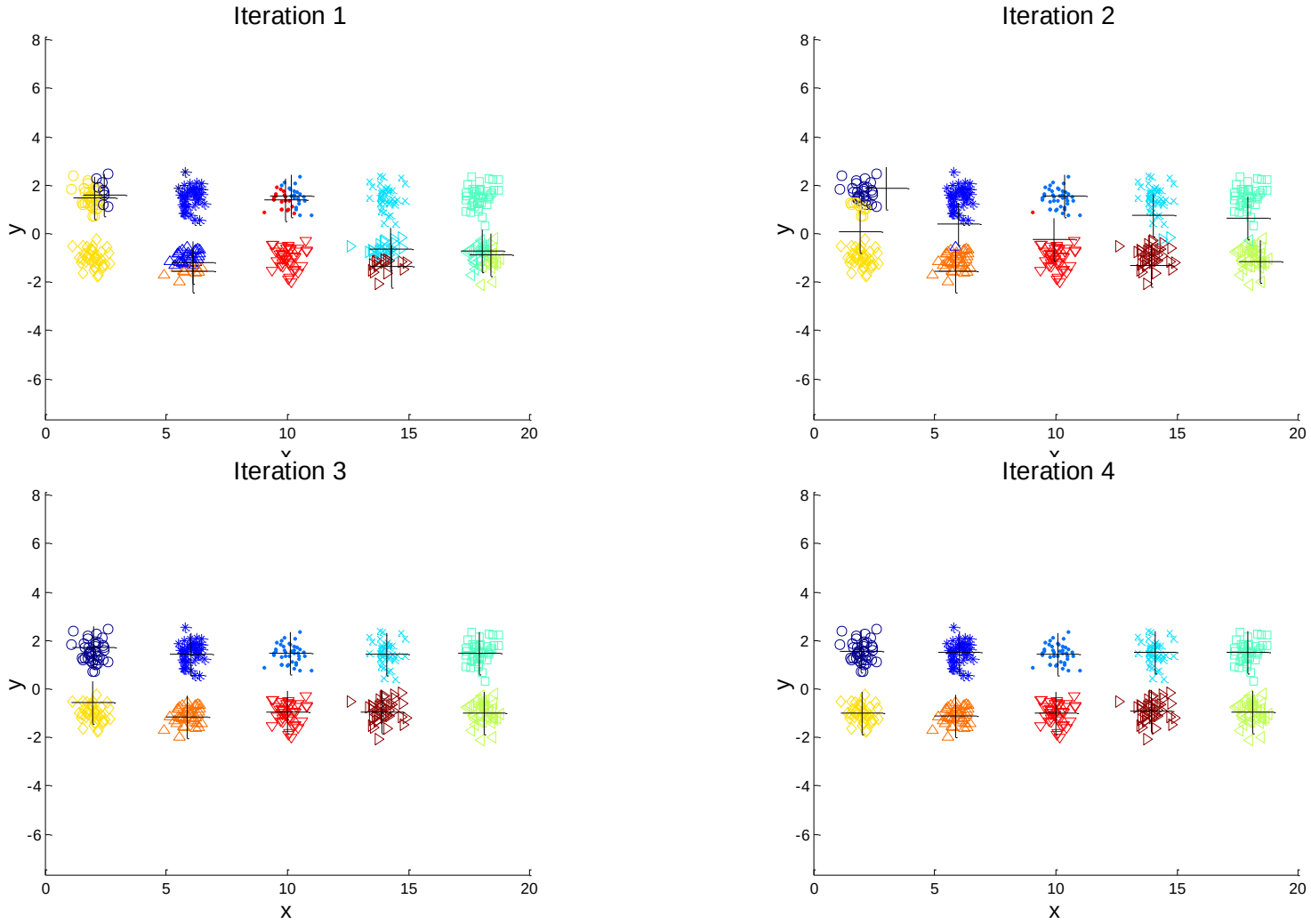
  - Például ha K = 10 akkor ez a valószínűség = $10!/10^{10}$ = 0.00036
  - Néha a kezdeti középpontok hozzáigazulnak a ,,helyes” módhoz néha azonban nem.
  - Tekintsünk példaként 5 klaszterpárt (10 klasztert).

10 klaszterből álló példa



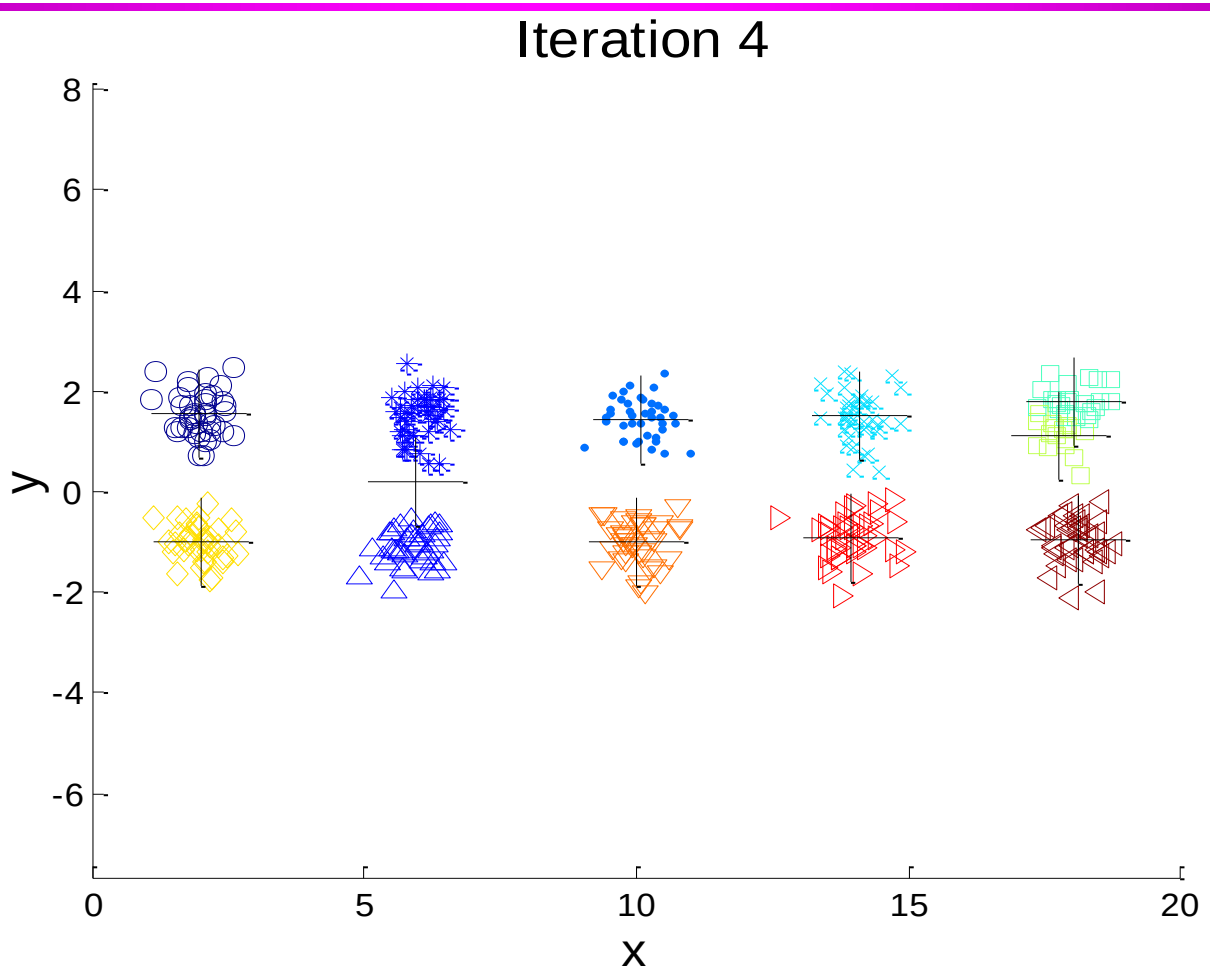
Két kezdeti középponttal indulva minden pár egyik klaszteréből.

10 klaszterből álló példa



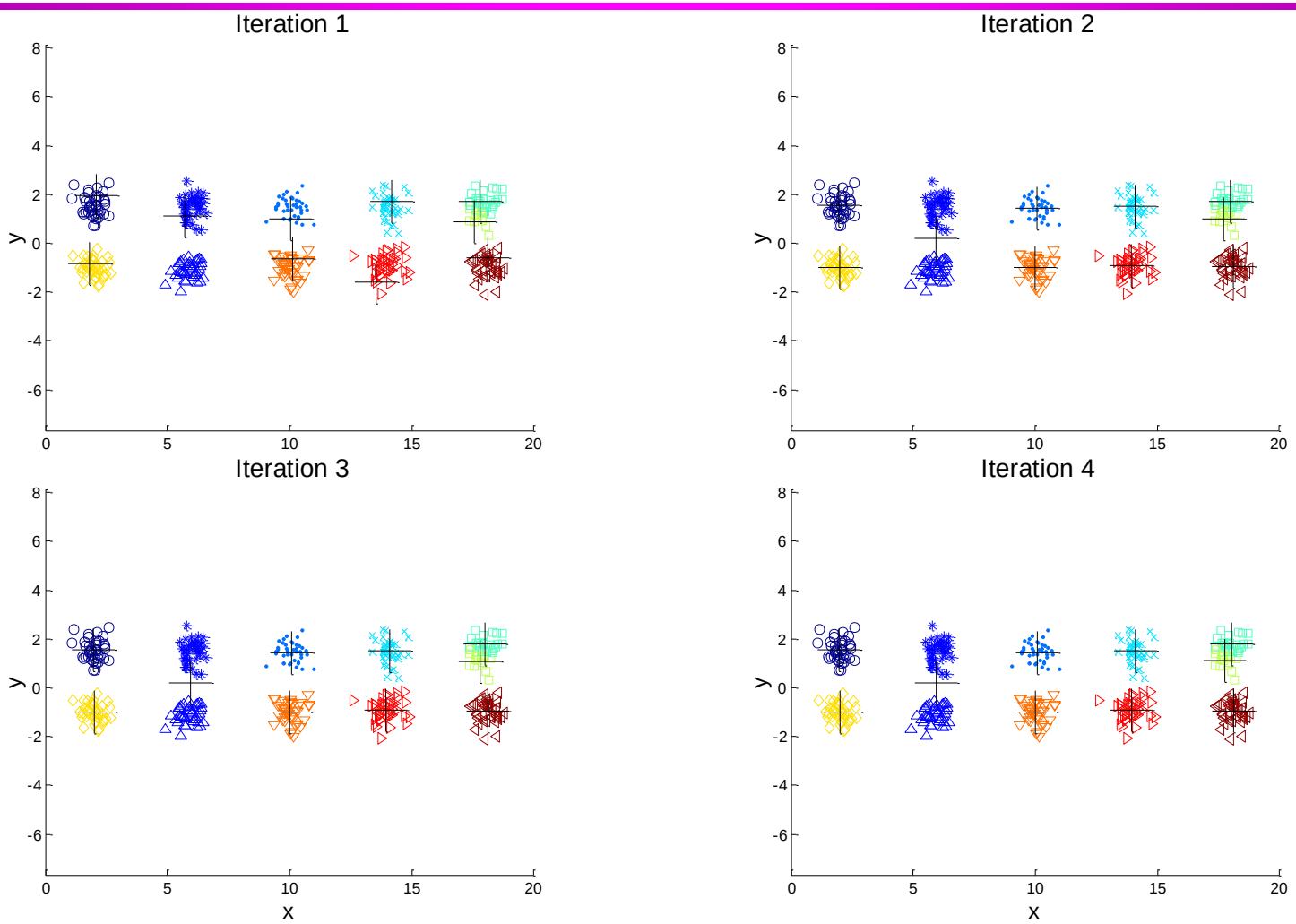
Két kezdeti középponttal indulva minden pár egyik klaszteréből.

10 klaszterből álló példa



Olyan klaszterpárokkal indítva, melyeknek 3 kezdeti középpontja van, míg a többi klaszternak csak egy.

10 klaszterből álló példa



Olyan klaszterpárokkal indítva, melyeknek 3 kezdeti középpontja van, míg a többi klaszternak csak egy.

A kezdeti középpont probléma megoldása

- Többszöri futtatás
  - Segíthet, azonban a valószínűség nem a mi oldalunkon áll.
- Mintavétel után alkalmazzunk hierarchikus klaszterezést a kezdeti középpontok meghatározására.
- Válasszunk több mint *k* kezdeti középpontot majd válogassunk közülük.
  - Válasszuk ki a legjobban elkülönülőket.
- Utófeldolgozás
- Felező *K*-közép módszer
  - Nem annyira érzékeny az inicializálási problémákra.

Üres klaszterek kezelése

- Az alap *K*-közép algoritmus üres klasztereket is adhat.

- Több stratégia
  - Válasszuk ki azt a pontot, amely az SSE legnagyobb részét adja.
  - Válasszunk egy olyan pontot, amely a legnagyobb SSE-t adja.
  - Ha több üres klaszter van, akkor a fentieket többször kell megismételni.

A középpontok járulékos frissítése

- Az alap K-közép algoritmusban a középpontokat akkor számoljuk újra, ha már minden pontot hozzárendeltünk egy középponthoz.
- Egy alternatíva az ha a középpontokat minden egyes hozzárendelés után frissítjük (járulékos megközelítés).
  - Minden hozzárendelés 0 vagy 2 középpontot frissít.
  - Költségesebb.
  - Bejön a sorrendtől való függőség is.
  - Sohasem kapunk üres klasztert.
  - Súlyokat is használhatunk a hatás megváltoztatására.

Elő- és utófeldolgozás

- Előfeldolgozás
  - Normalizáljuk (standardizáljuk) az adatokat.
  - Távolítsuk el a kiugróakat.
- Utófeldolgozás
  - Távolítsuk el a kis klasztereket, amelyekben kiugró adatok lehetnek.
  - Vágjuk ketté a ,,széteső” klasztereket, azaz amelyeknek viszonylag nagy az SSE-jük.
  - Vonjuk össze azokat a klasztereket, amelyek ,,közel” vannak egymáshoz és viszonylag kicsi az SSE-jük.
  - Ezeket a lépéseket használhatjuk a klaszterezés során is
    - ISODATA

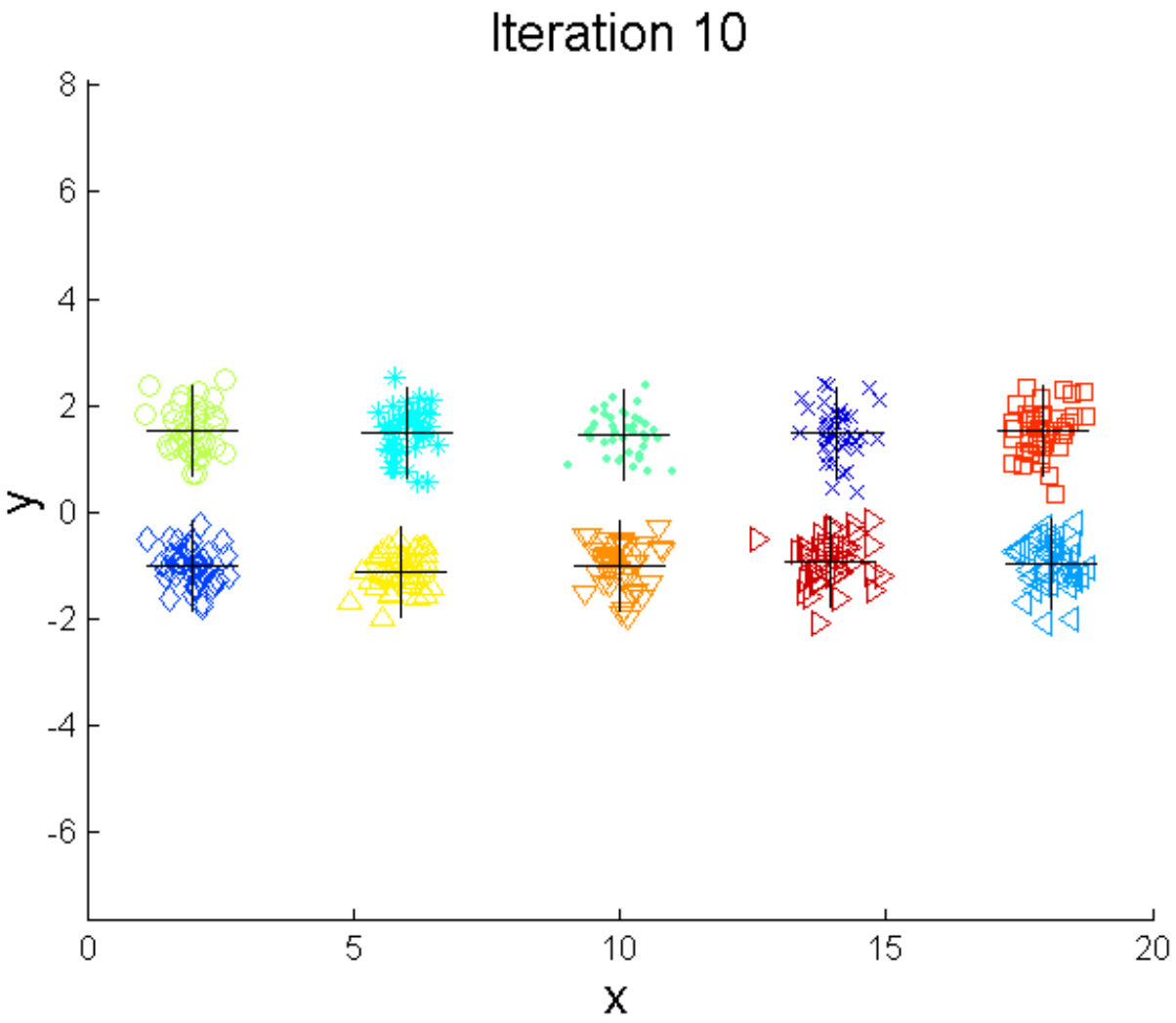
Felező *K*-közép módszer

- Felező *K*-közép algoritmus
  - Egy olyan *K*-közép variáns, amely felosztó illetve hierarchikus klaszterezésre egyaránt alkalmazható.

Algoritmus. Felező K-közép módszer

1. **Inicializálás.** A klaszterlista tartalmazzon egy klasztert, amelynek az összes pont legyen az eleme.
2. **repeat**
3. Válasszunk ki egy klasztert a listából. (Többször is próbálkozhatunk.)
4. **for** i=1 **to** iterációk_száma **do**
5. Osszuk fel a kiválasztott klasztert az alap K-közép algoritmussal.
6. **end for**
7. Adjuk hozzá azt a két klasztert a klaszterlistához, amelyeknek a legkisebb az SSE-jük.
8. **until** Ismételjünk addig, amíg a klaszterlista *K* klasztert nem tartalmaz.

Példa felező K-közép módszerre

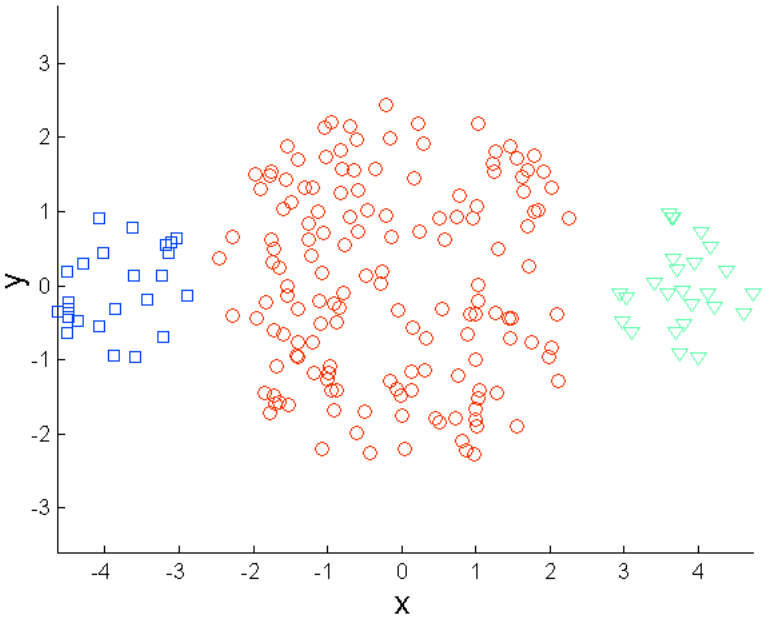


A *K*-közép módszer korlátai

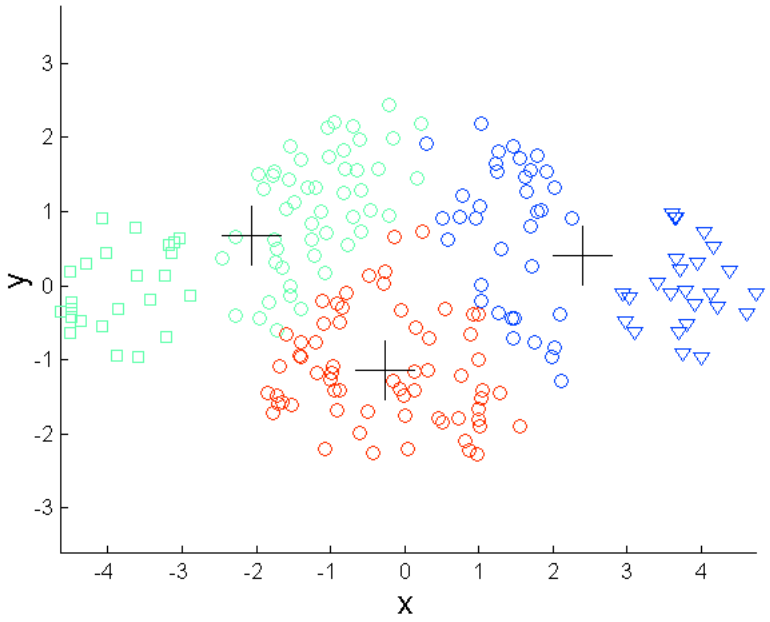
- A K-közép módszerrel gond van amennyiben a klaszteréknek eltérő a
  - méretük,
  - sűrűségük,
  - vagy nem gömb alakúak.

- A K-közép módszerrel gond van amennyiben az adatok között vannak kiugróak.

K-közép korlátok: különböző méretek

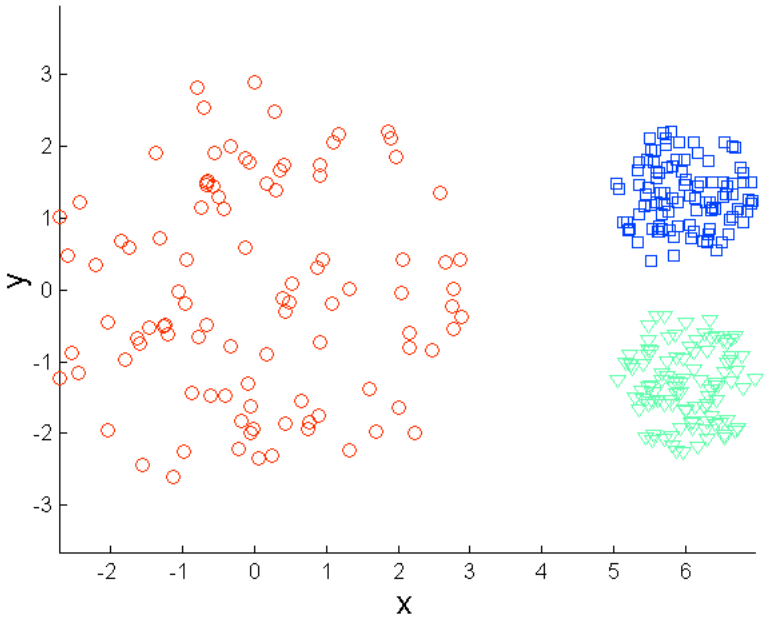


Eredeti pontok

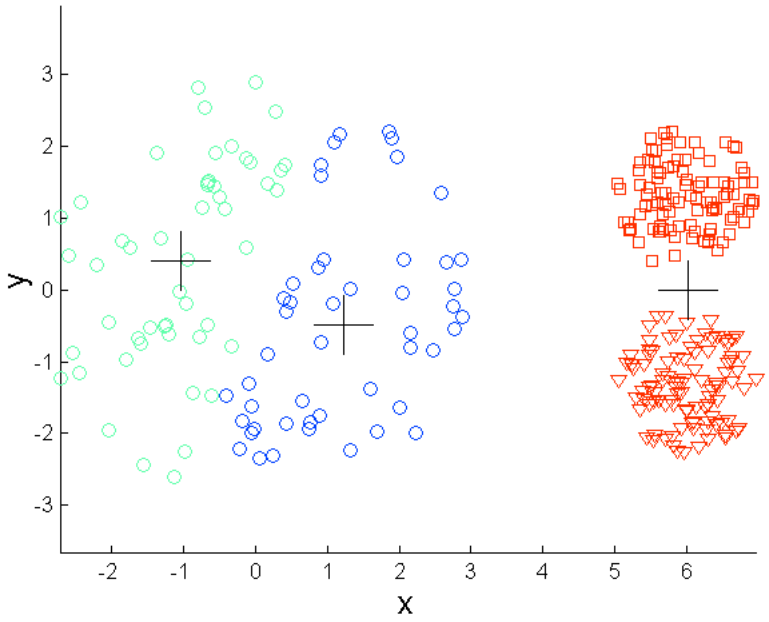


***K*-közép (3 klaszterek)**

K-közép korlátok: eltérő sűrűségek

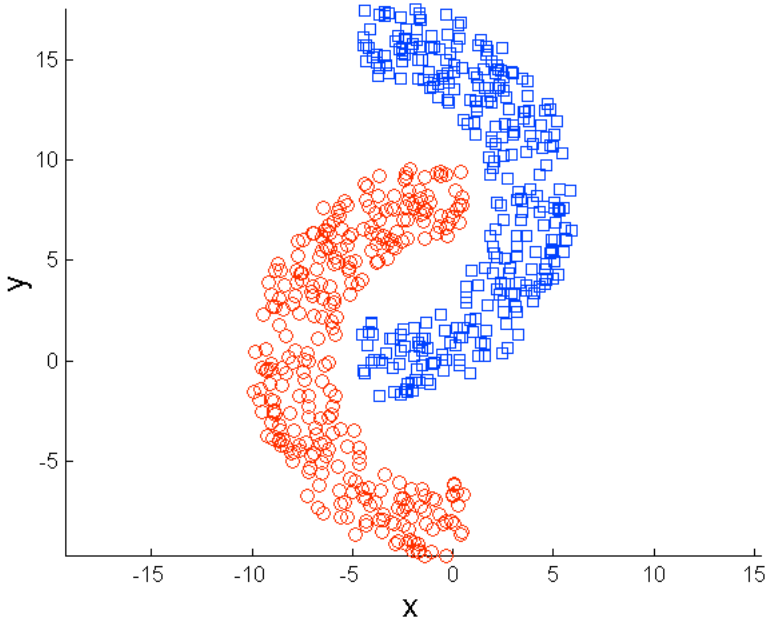


Eredeti pontok

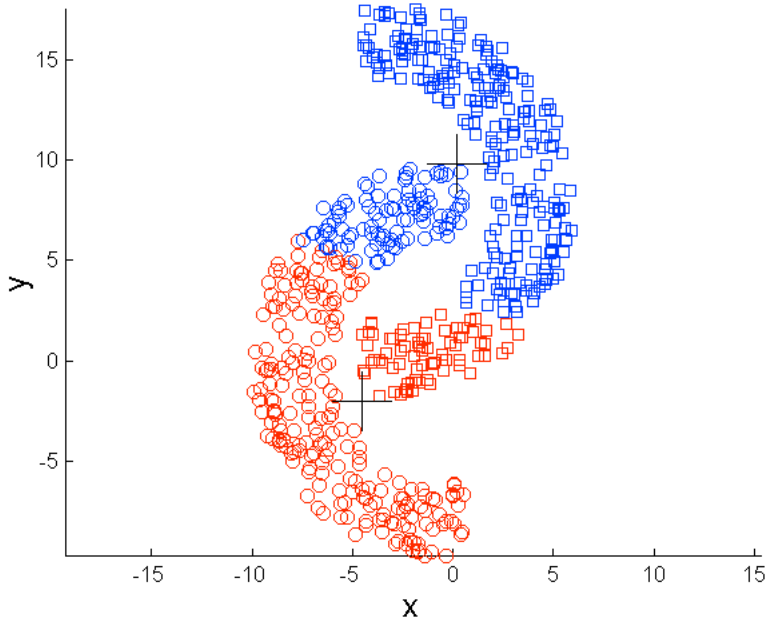


***K*-közép (3 klaszter)**

A K-közép korlátai: nem gömbszerű alak

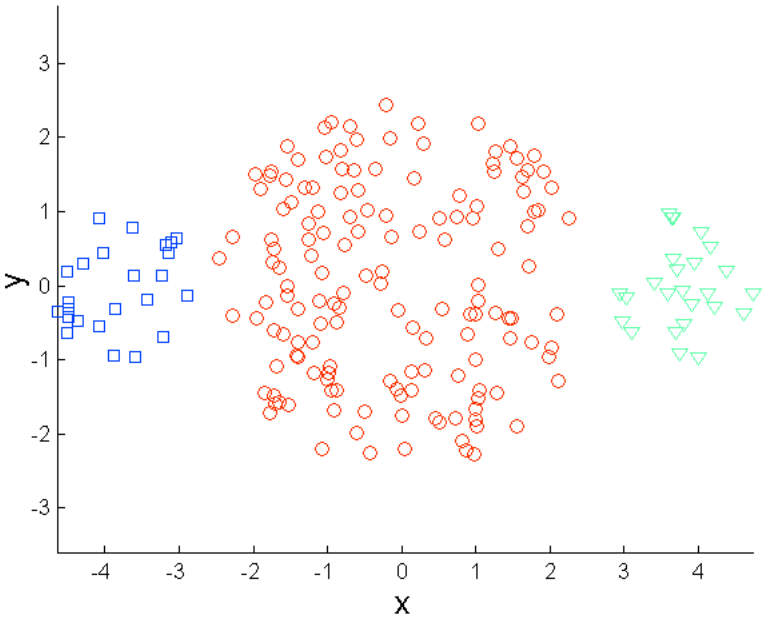


Eredeti pontok

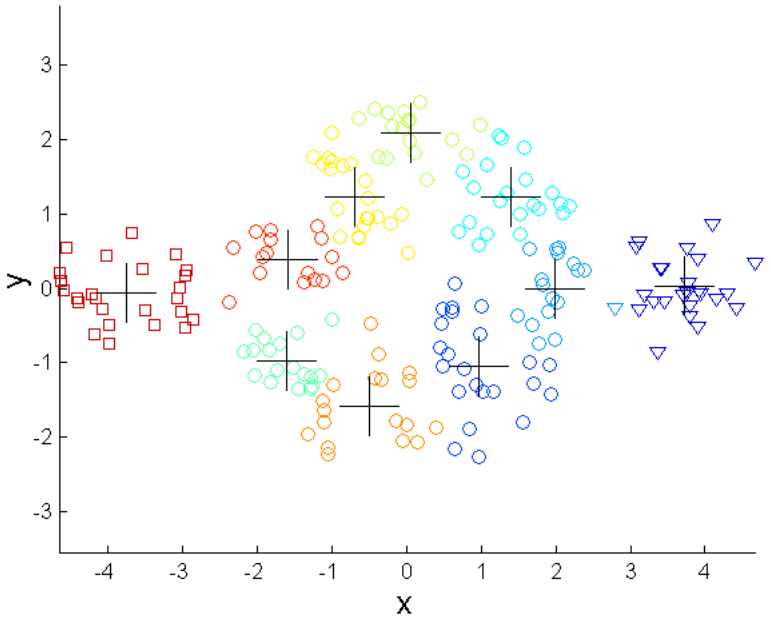


***K*-közép (2 klaszter)**

A *K*-közép korlátainak legyőzése



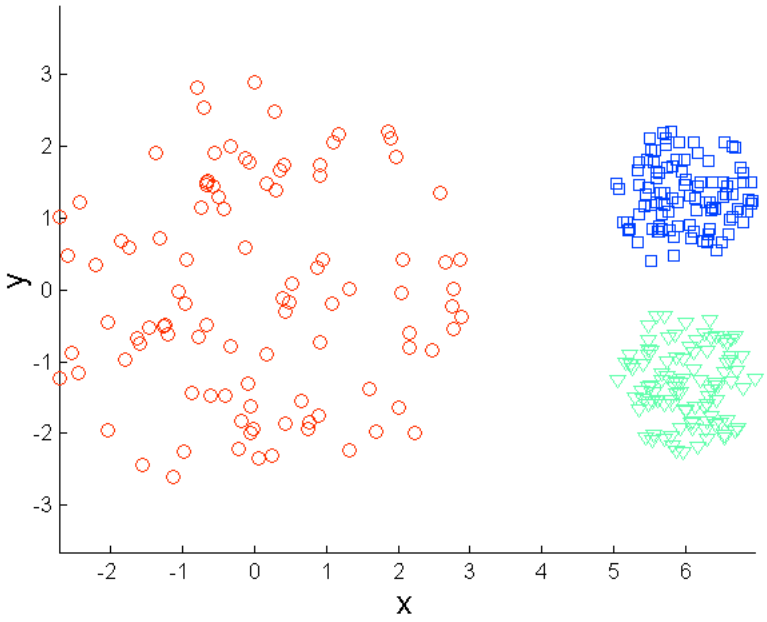
Eredeti pontok



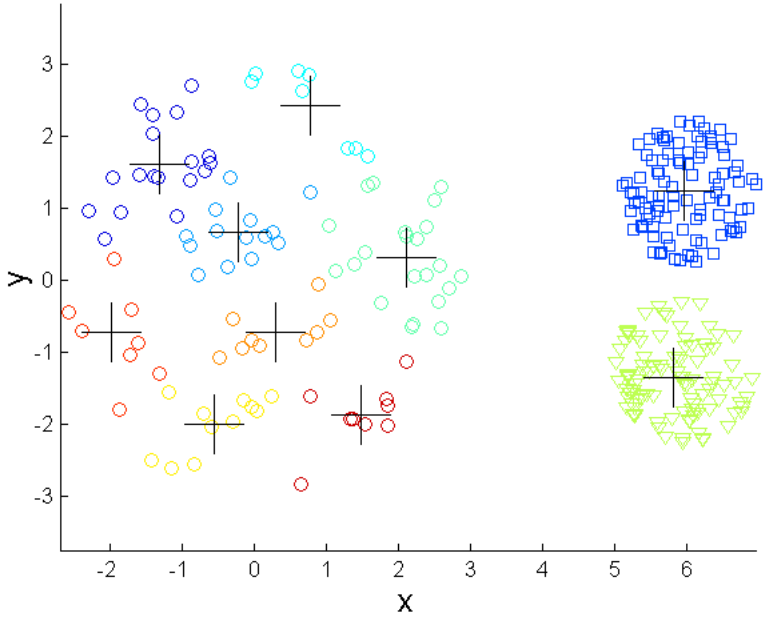
***K*-közép klaszterek**

Egy lehetséges megoldás ha több klasztert használunk. Találjuk meg a klaszterek részeit majd ügyeljünk az összevonásukra.

A *K*-közép korlátainak legyőzése

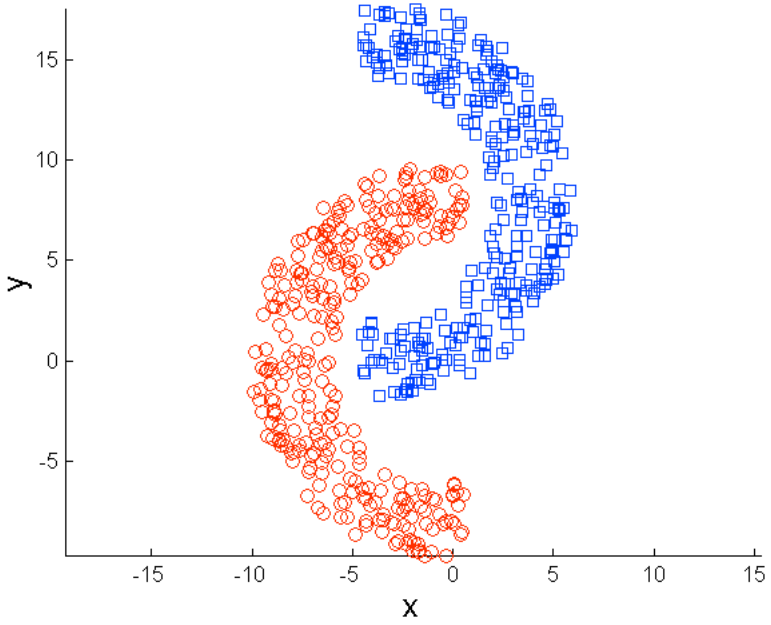


Eredeti pontok

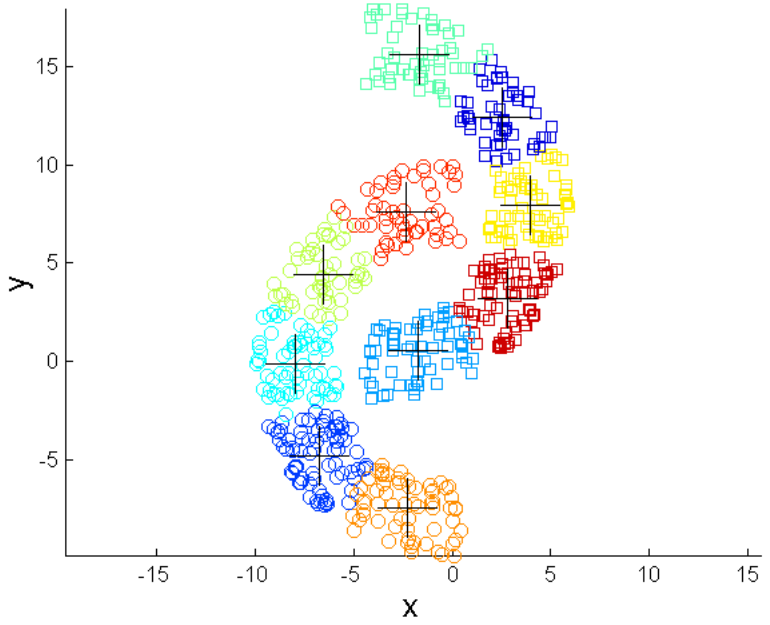


***K*-közép klaszterek**

A *K*-közép korlátainak legyőzése



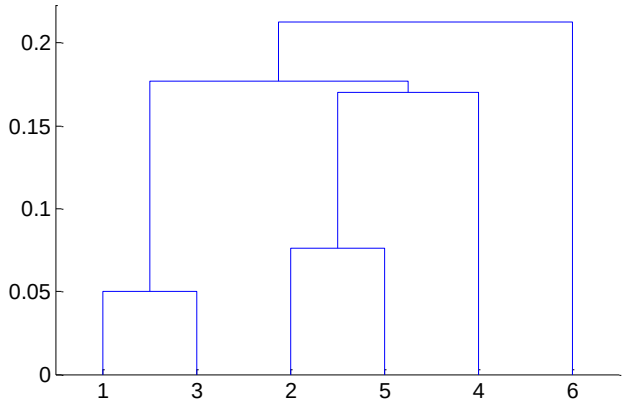
Eredeti pontok

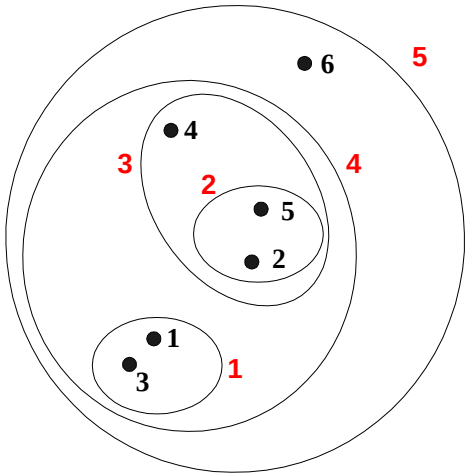


***K*-közép klaszterek**

Hierarchikus klaszterezés

- Egymásba ágyazott klaszterek egy hierarchikus fába szervezett halmazát állítja elő.
- Egy ún. dendrogrammal jeleníthetjük meg.
  - Ez egy fa alakú diagram, amely a rekordokat összevonások vagy szétvágások sorozataivá rendezi.





A hierarchikus klaszterezés előnyei

- Nem kell feltételezni semmilyen konkrét klaszter-számot előre.
  - Bármilyen elvárt klaszterszámot megkaphatunk a dendrogram egy megfelelő szinten való ,,elvágásával”.

- Értelmes osztályozásoknak (taxonómiáknak) is megfelelhet.
  - Példák a biológia területén (állatvilág, filogenetikus rekonstrukció).

Hierarchikus klaszterezés

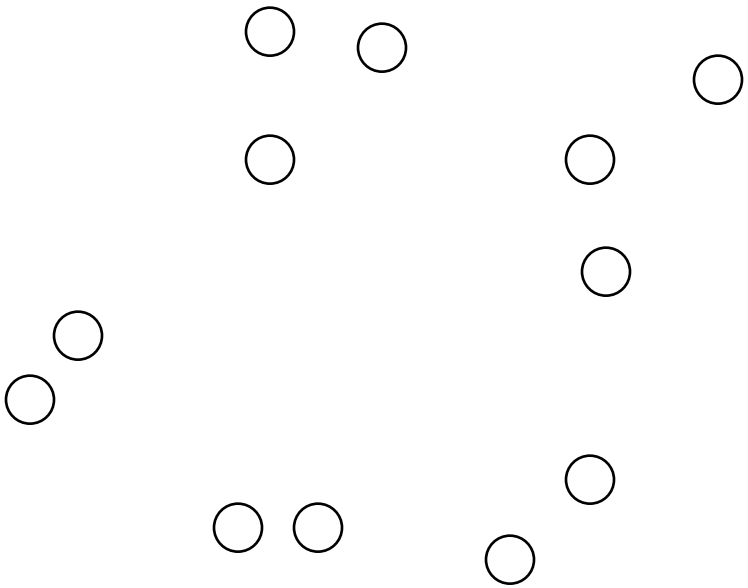
- A hierarchikus klaszterezés két fő típusa
  - Összevonó:
    - Induljunk minden pontot külön klaszterként kezelve.
    - Minden lépésnél vonjuk össze a két legközelebbi klasztert amíg csak egy (vagy *k*) klaszter nem marad.
  - Felosztó:
    - Induljunk egy minden pontot tartalmazó klaszterből.
    - Minden lépésnél vágjunk ketté egy klasztert amíg minden klaszter csak egy pontot nem tartalmaz (vagy amíg *k* klasztert nem kapunk).
- A hagyományos hierarchikus algoritmusok hasonlósági vagy távolság mátrixot használnak.
  - Egyszerre egy klasztert vonjunk össze vagy vágjunk szét.

Összevonó klaszterezési algoritmus

- A népszerűbb hierarchikus klaszterezési módszer.
- Az alap algoritmus egyszerű
  1. Számoljuk ki a közelségi mátrixot.
  2. Legyen minden egyes pont egy önálló klaszter.
  3. **Repeat**
  4. Vonjuk össze a két legközelebbi klasztert.
  5. Frissítsük a közelségi mátrixot.
  6. **Until** Ismételjük amíg csak egy klaszter nem marad.
- Az alapvető művelet két klaszter közelségének a kiszámolása.
  - A klaszterek közötti távolság definíciójának különböző megközelítései más-más algoritmusokhoz vezetnek.

Kiinduló helyzet

- Induljunk ki minden pontot külön klaszterként kezelve a közelségi mátrixból.



	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

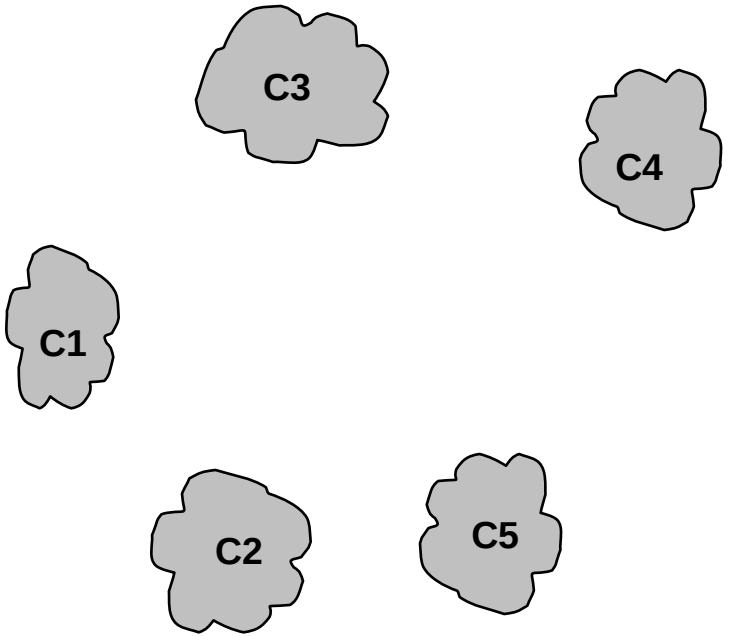
Közelségi mátrix

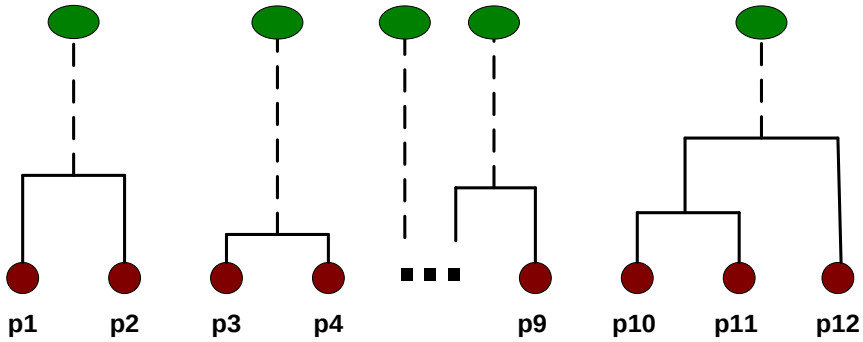
Közbenső helyzet

- Néhány összevonás után az alábbi klasztereket kapjuk.

	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

Közelségi mátrix



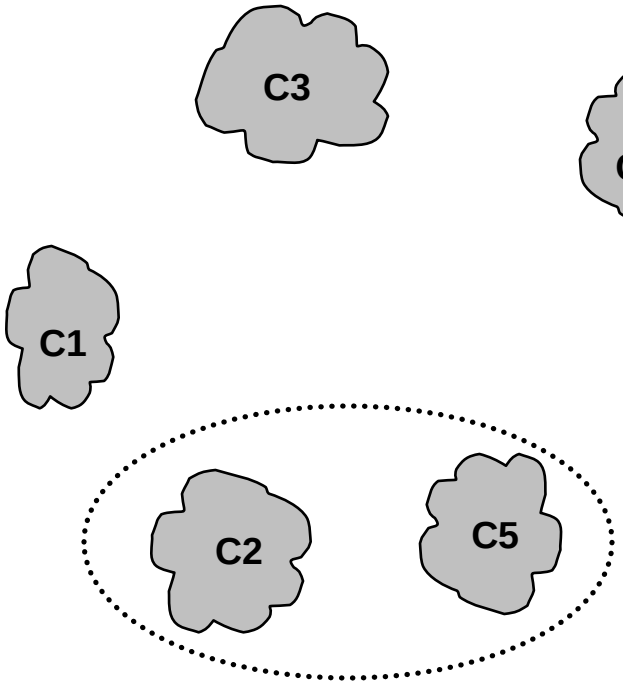


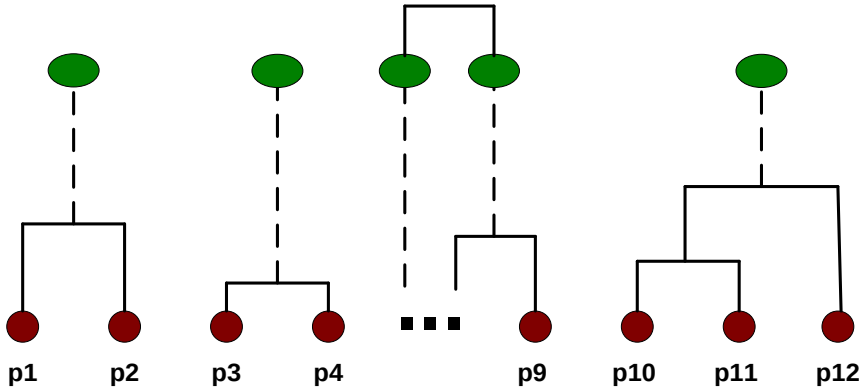
Közbenső helyzet

- Össze akarjuk vonni a két legközelebbi klasztert (C2 és C5) majd frissíteni szeretnénk a közelségi mátrixot.

	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

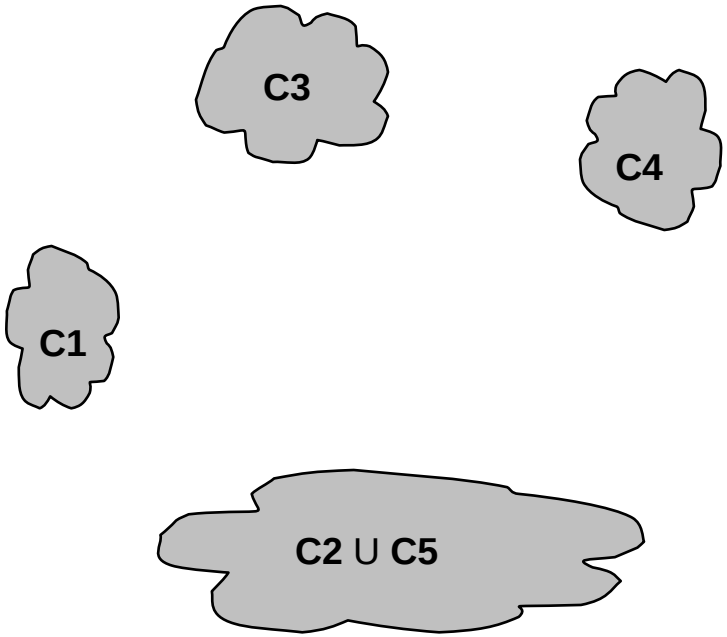
Közelségi mátrix





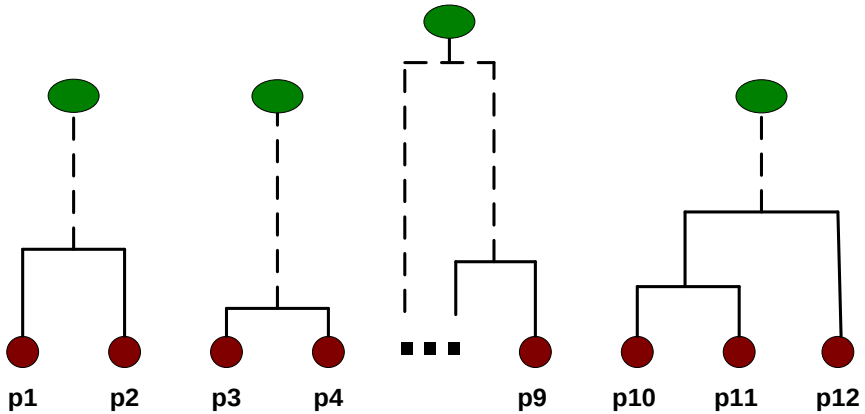
Összevonás után

- A kérdés a következő: ,,Hogyan frissítsük a közelségi mátrixot?”

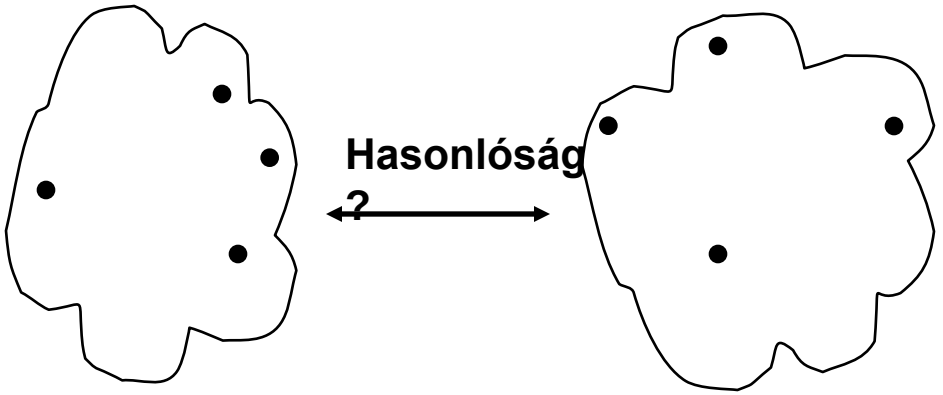


	C1	C2 U C5	C3	C4
C1		?		
C2 U C5	?	?	?	?
C3		?		
C4		?		

Közelségi mátrix



Hogyan definiáljuk a klaszterek közötti a hasonlóságot?

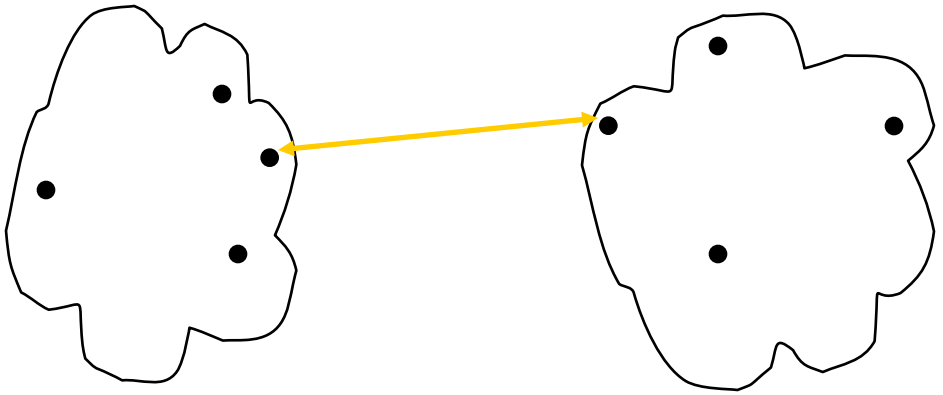


	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						

Közelségi mátrix

- MIN
- MAX
- Csoport-átlag
- Középpontok közötti távolságok
- Más, célfüggvény által meghatározott módszer
  - A Ward módszer négyzetes hibát használ

Hogyan definiáljuk a klaszterek közötti a hasonlóságot?



	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						

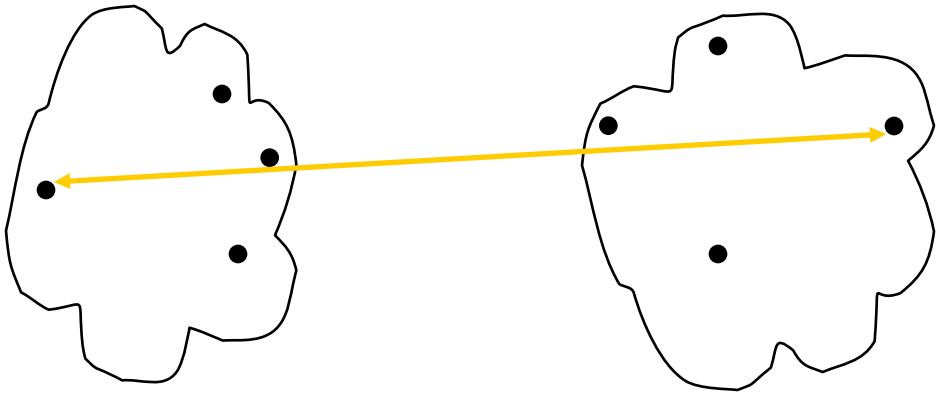
.

.

Közelségi mátrix

- MIN
- MAX
- Csoport-átlag
- Középpontok közötti távolságok
- Más, célfüggvény által meghatározott módszer
  - A Ward módszer négyzetes hibát használ

Hogyan definiáljuk a klaszterek közötti a hasonlóságot?



	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						

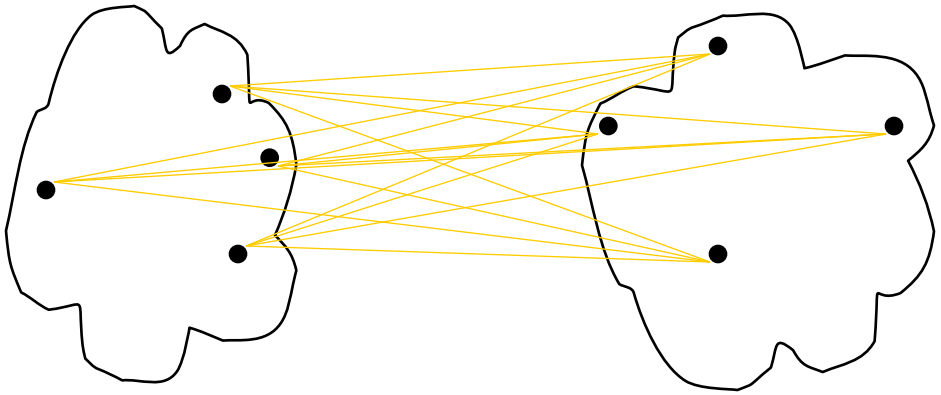
.

.

Közelségi mátrix

- MIN
- MAX
- Csoport-átlag
- Középpontok közötti távolságok
- Más, célfüggvény által meghatározott módszer
  - A Ward módszer négyzetes hibát használ

Hogyan definiáljuk a klaszterek közötti a hasonlóságot?



	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						

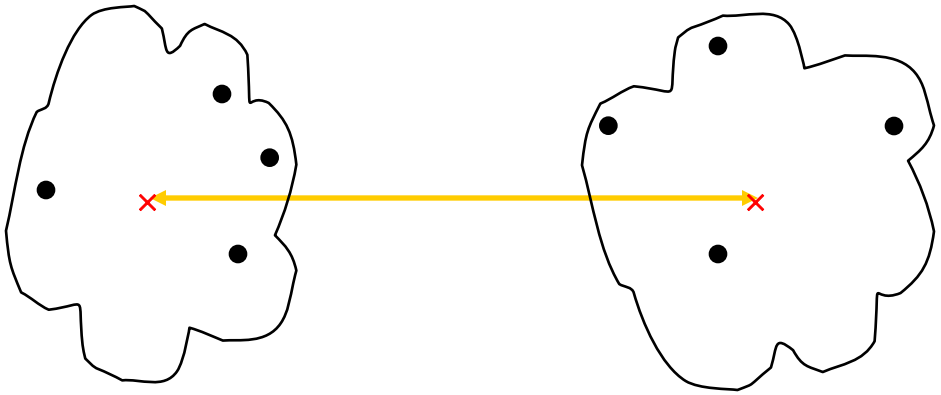
.

.

Közelségi mátrix

- MIN
- MAX
- Csoport-átlag
- Középpontok közötti távolságok
- Más, célfüggvény által meghatározott módszer
  - A Ward módszer négyzetes hibát használ

Hogyan definiáljuk a klaszterek közötti a hasonlóságot?



	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						

. .

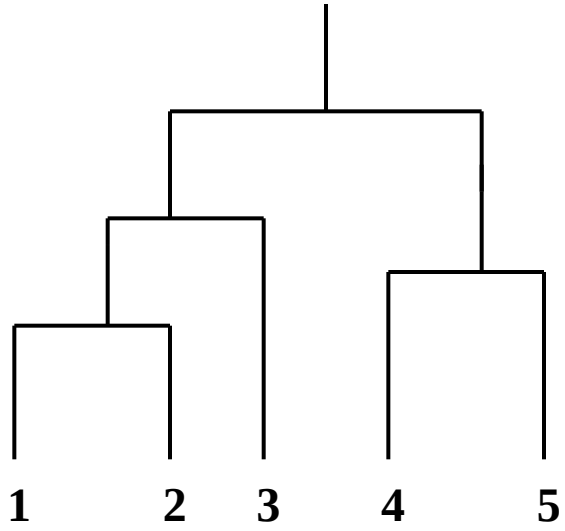
Közelségi mátrix

- MIN
- MAX
- Csoport-átlag
- Distance Between Centroids
- Más, célfüggvény által meghatározott módszer
  - A Ward módszer négyzetes hibát használ

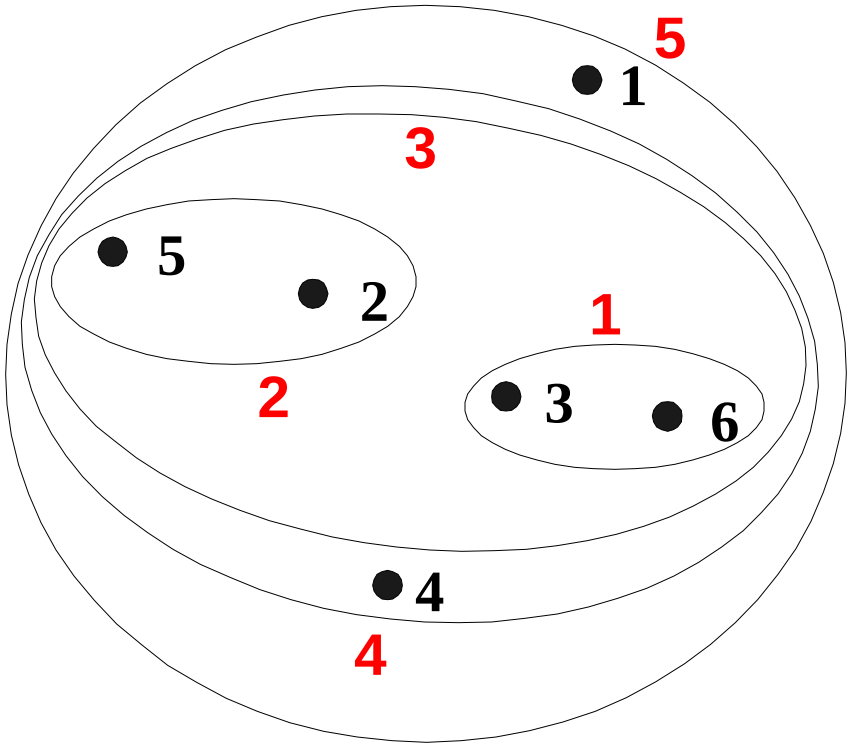
Klaszter-hasonlóság: MIN vagy egyszerű kapcsolás

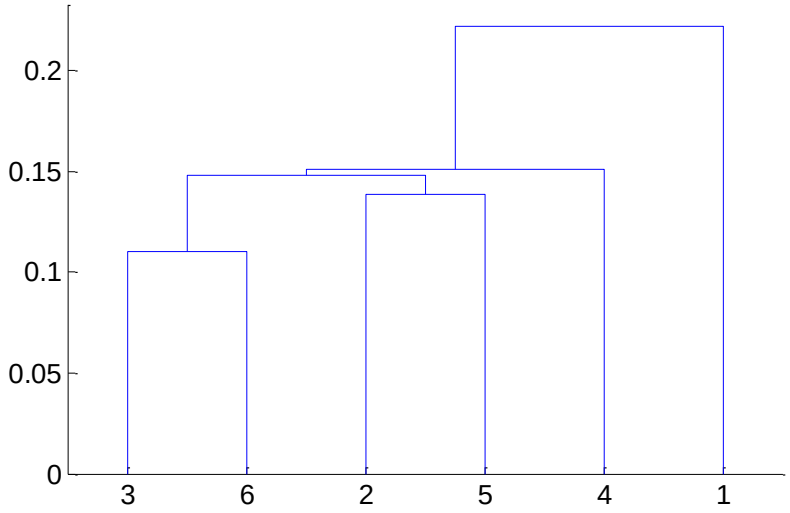
- Két klaszter hasonlósága a klaszterekbeli két leghasonlóbb (legközelebbi) ponton alapszik.
  - Egy pontpár által, azaz a közelségi gráfban egy kapcsolat által (a többitől függetlenül) meghatározott.

	I1	I2	I3	I4	I5
I1	1.00	0.90	0.10	0.65	0.20
I2	0.90	1.00	0.70	0.60	0.50
I3	0.10	0.70	1.00	0.40	0.30
I4	0.65	0.60	0.40	1.00	0.80
I5	0.20	0.50	0.30	0.80	1.00



Hierarchikus klaszterezés: MIN módszer

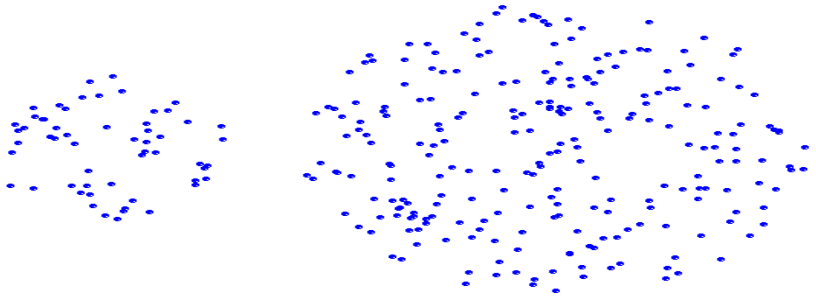




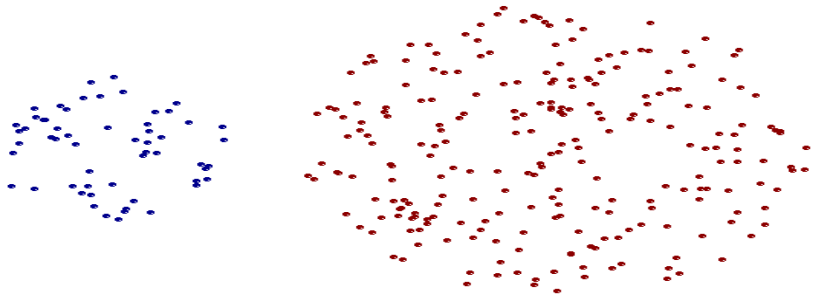
Egymásba ágyazott klaszterek

Dendrogram

A MIN módszer előnyei



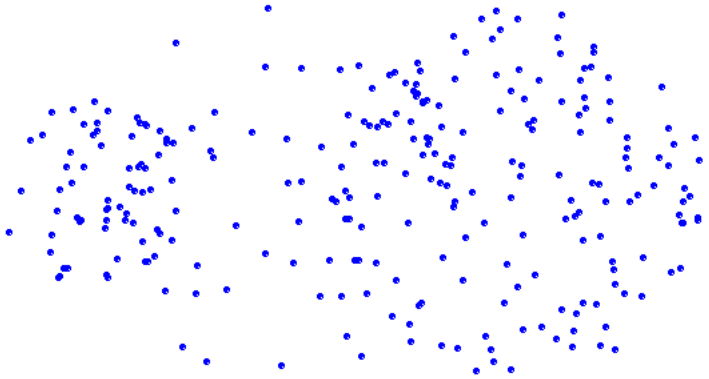
Eredeti pontok



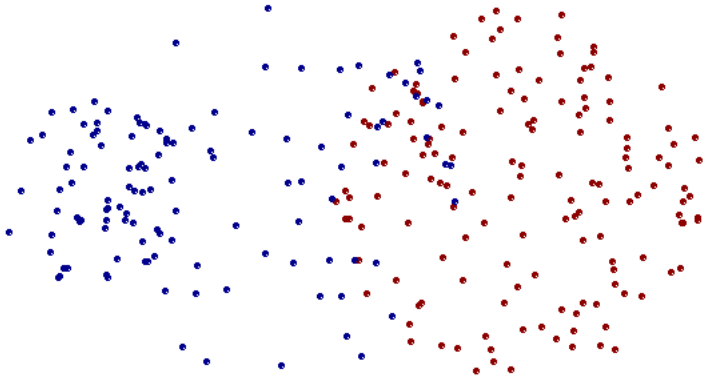
Két klaszter

- **Nem elliptikus alakokat is tud kezelni.**

A MIN módszer korlátai



Eredeti pontok



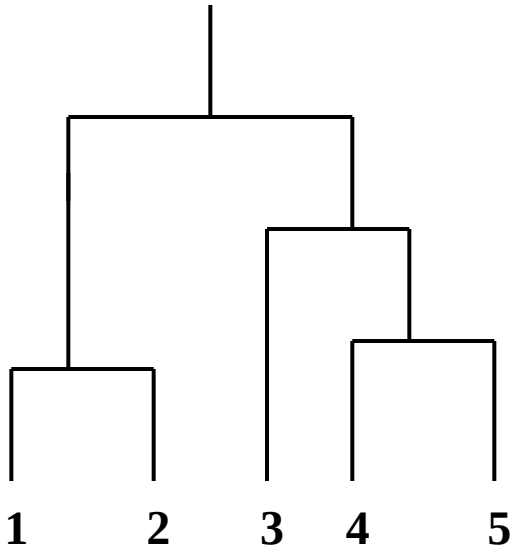
Két klaszter

- **Érzékeny a hibára és a kiugró adatokra.**

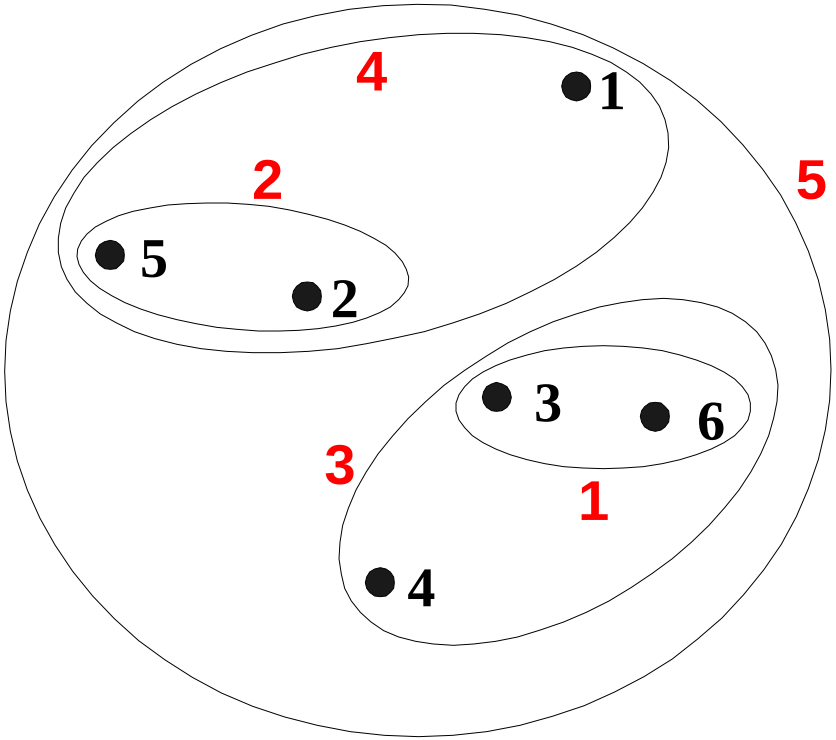
Klaszter-hasonlóság: MAX vagy teljes kapcsolás

- Két klaszter közötti hasonlóság a klaszterekbeli két legkevésbé hasonló (legtávolabbi) ponttól függ.
  - A két klaszter összes pontja által meghatározott.

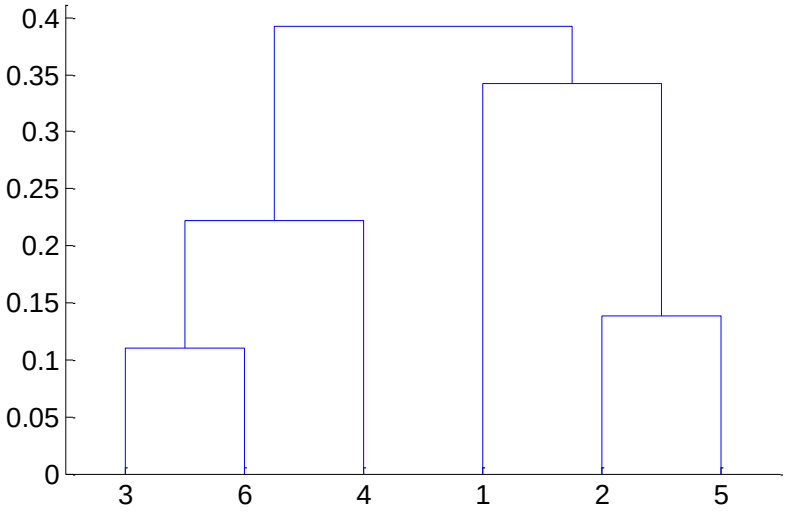
	I1	I2	I3	I4	I5
I1	1.00	0.90	0.10	0.65	0.20
I2	0.90	1.00	0.70	0.60	0.50
I3	0.10	0.70	1.00	0.40	0.30
I4	0.65	0.60	0.40	1.00	0.80
I5	0.20	0.50	0.30	0.80	1.00



Hierarchikus klaszterezés: MAX módszer

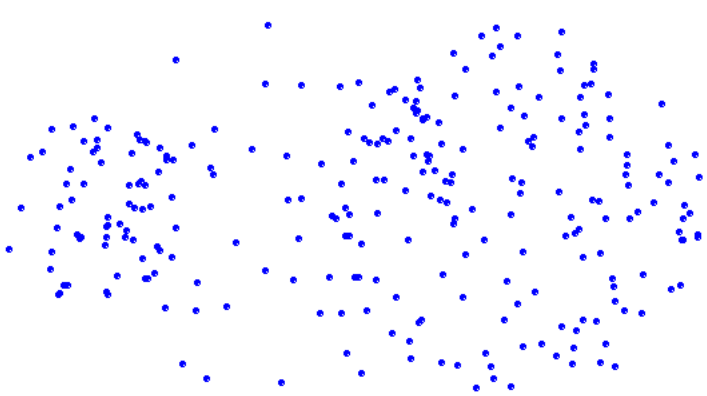


Egymásba ágyazott klaszterek

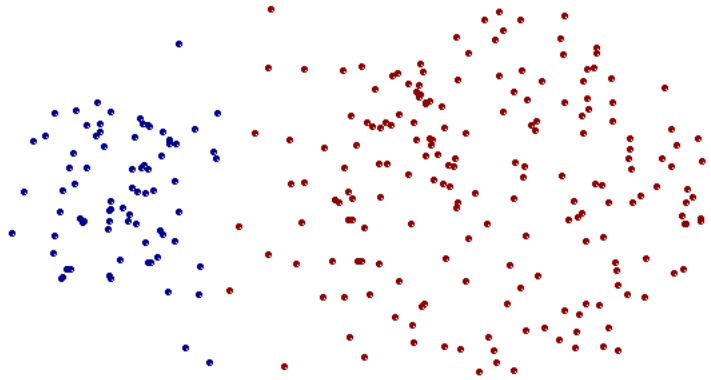


Dendrogram

A MAX módszer előnyei



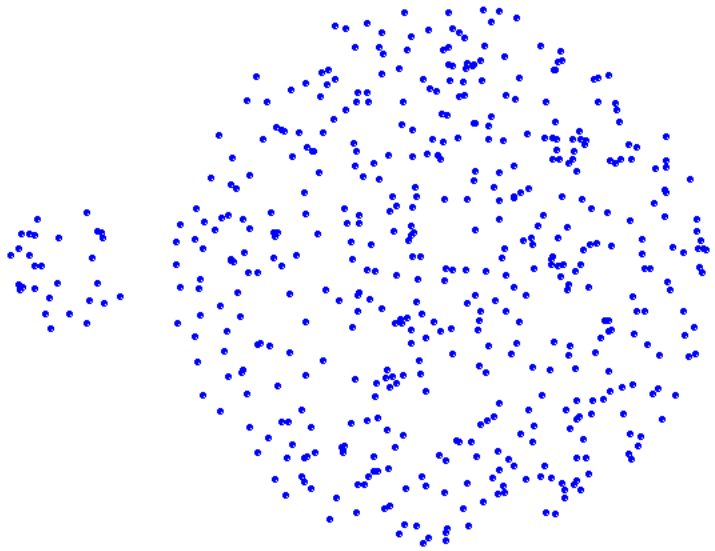
Eredeti pontok



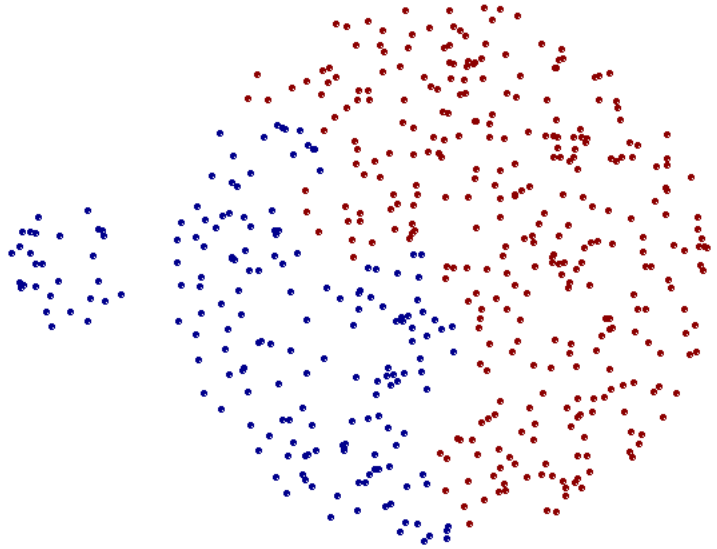
Két klaszter

- **Kevésbé érzékeny a hibára és a kiugró adatokra.**

A MAX módszer korlátai



Eredeti pontok



Két klaszter

- Hajlamos a nagy klasztereket ketté vágni.
- Torzít a gömbölyű klaszterek irányában.

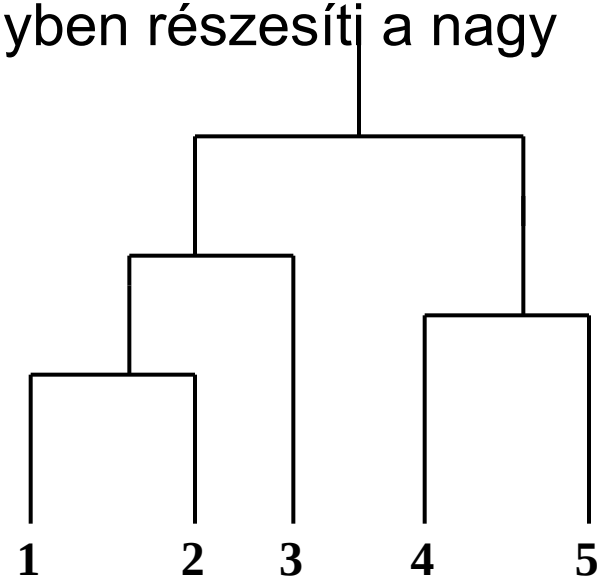
Klaszter-hasonlóság: csoport-átlag

- Két klaszter közötti közelség a klaszterekbeli pontok közötti mérőszámok átlaga.

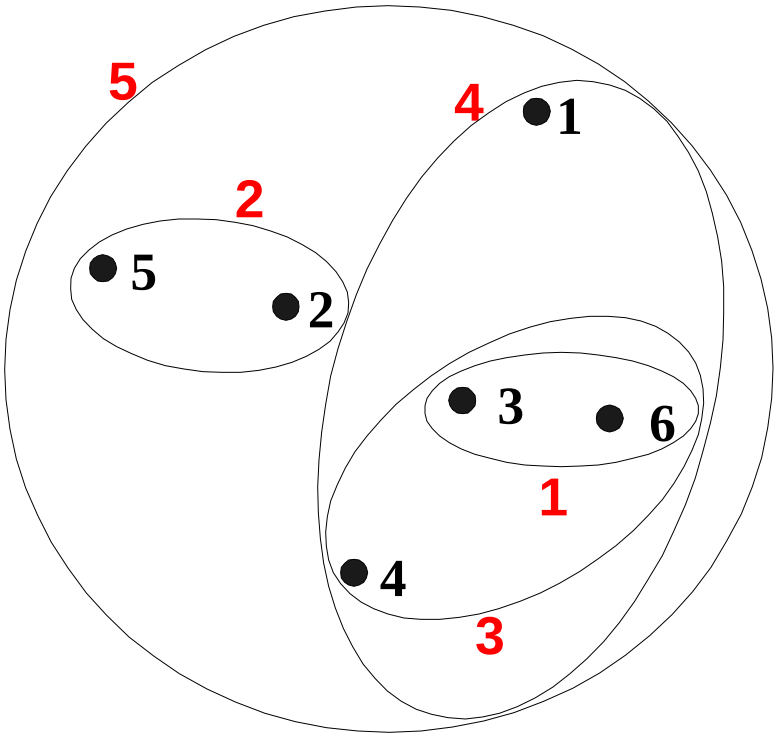
$$\text{közelség}(\text{Klaszter}_i, \text{Klaszter}_j) = \frac{\sum_{\substack{p_i \in \text{Klaszter}_i \\ p_j \in \text{Klaszter}_j}} \text{közelség}(p_i, p_j)}{|\text{Klaszter}_i| * |\text{Klaszter}_j|}$$

- Azért kell a skálázhatóság miatt átlagos összekapcsolhatóságot használni, mert a teljes közelség előnyben részesíti a nagy klasztereket.

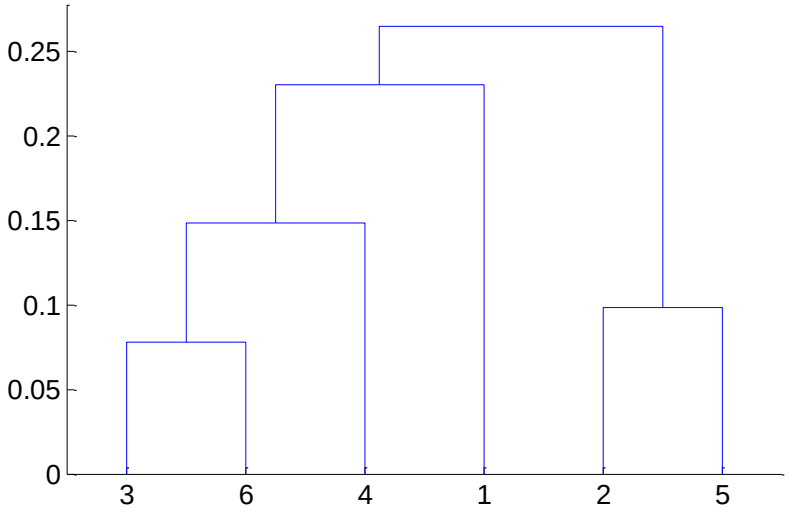
	I1	I2	I3	I4	I5
I1	1.00	0.90	0.10	0.65	0.20
I2	0.90	1.00	0.70	0.60	0.50
I3	0.10	0.70	1.00	0.40	0.30
I4	0.65	0.60	0.40	1.00	0.80
I5	0.20	0.50	0.30	0.80	1.00



Hierarchikus klaszterezés: csoport-átlag



Egymásba ágyazott klaszterek



Dendrogram

Hierarchikus klaszterezés: csoport-átlag

- Az egyszerű és a teljes kapcsolás közötti kompromisszum.

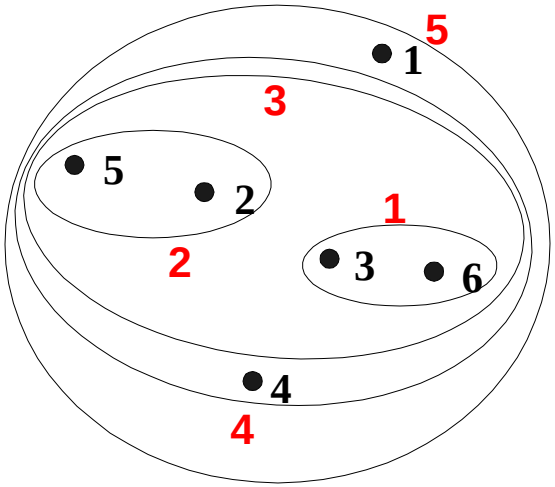
- Erősségek
  - Kevésbé érzékeny a hibára és a kiugró adatokra.

- Korlátok
  - Torzít a gömbölyű klaszterek irányában.

Klaszter-hasonlóság: Ward módszer

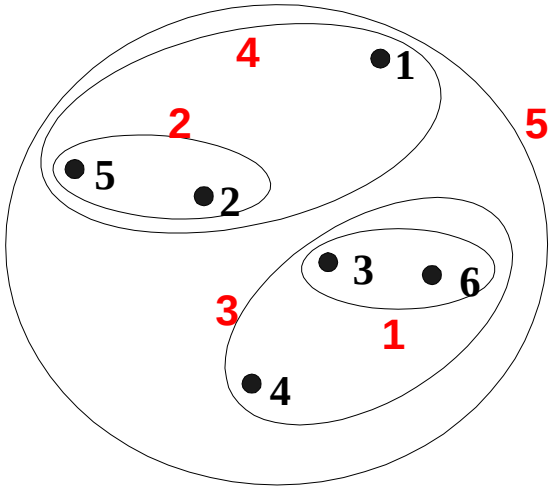
- Két klaszter közötti hasonlóság azon alapszik, hogy az összevonásuk után mennyivel nő a négyzetes hiba.
  - Hasonló a csoport-átlaghoz amennyiben a pontok közötti távolság a négyzetes euklideszi távolság.
- Kevésbé érzékeny a hibára és a kiugró adatokra.
- Torzít a gömbölyű klaszterek irányában.
- A *K*-közép módszer hierarchikus változata
  - A *K*-közép inicializálására is használható.

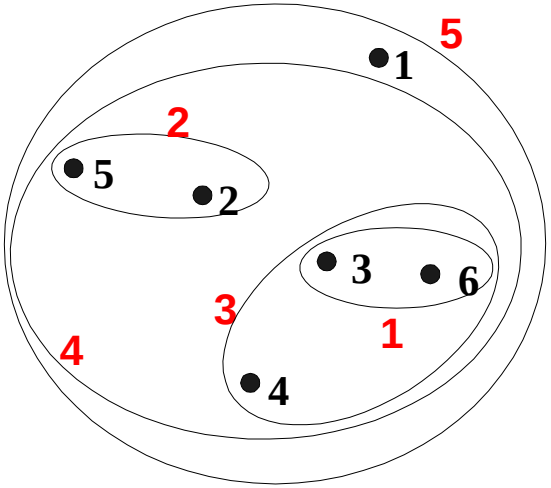
Hierarchikus klaszterezés: összehasonlítás



MIN

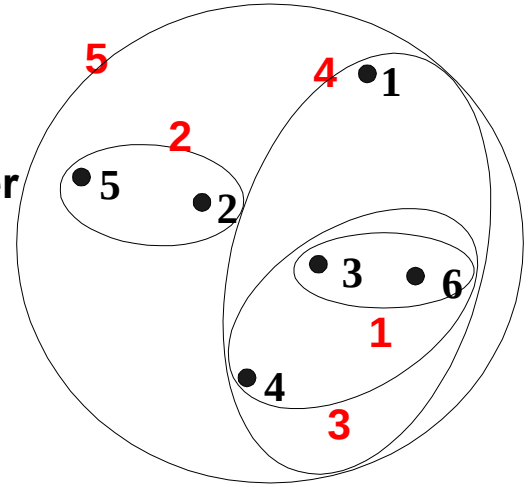
MAX





Csoport-átlag

Ward módszer



Hierarchikus klaszterezés: idő és tár korlátok

- $O(N^2)$ tárigény mivel a közelségi mátrixot használja.
  - N a pontok száma.

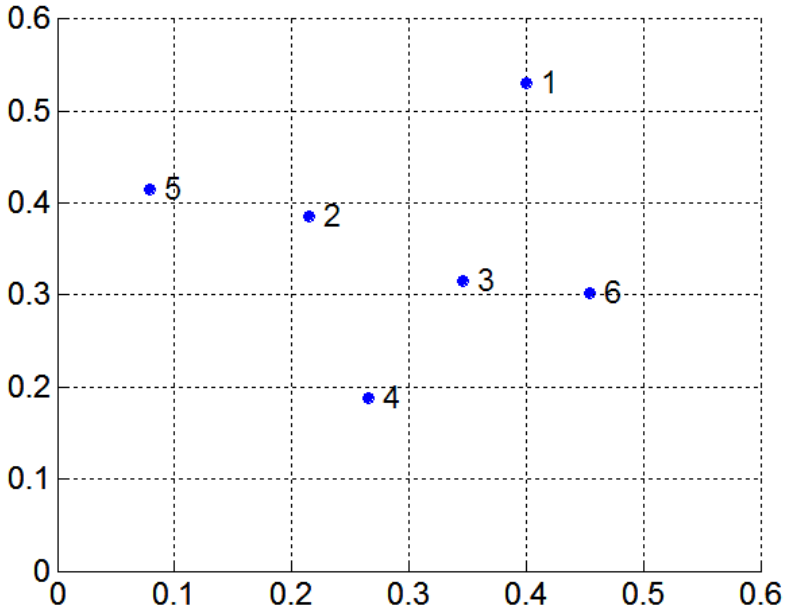
- $O(N^3)$ időigény az esetek többségében.
  - N lépést kell végrehajtani és minden egyes lépésben egy N^2 méretű közelségi mátrixot kell frissíteni és kell benne keresni.
  - Egyes megközelítéseknél az időigény $O(N^2 \log(N))$-re redukálható.

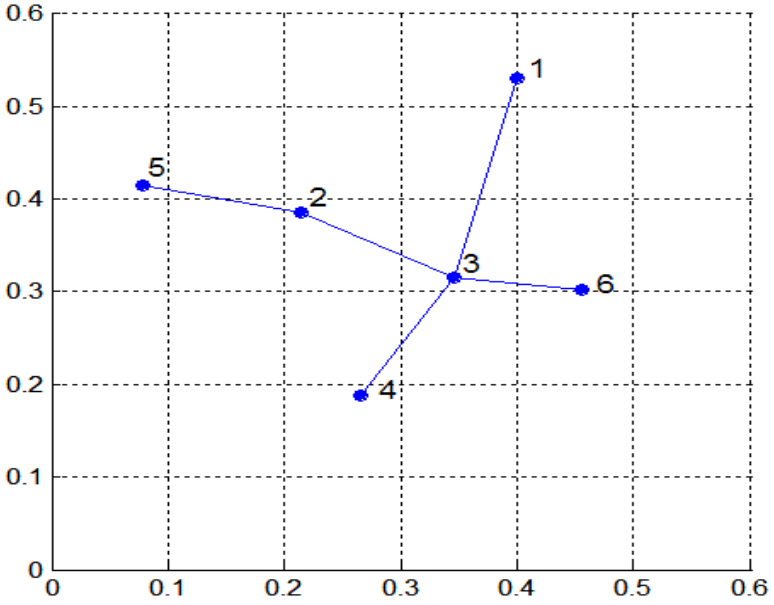
Hierarchikus klaszterezés: problémák és korlátok

- Ha egyszer döntést hozunk arról, hogy két klasztert összevonunk, akkor azt már nem lehet meg nem történtté tenni.
- Nincs célfüggvény, melyet közvetlenül minimalizálunk.
- A különböző eljárásoknál az alábbi problémák közül léphet fel egy vagy több:
  - Érzékenység a hibára és a kiugró adatokra.
  - Nehéz kezelni a különböző méretű klasztereket és konvex alakzatokat.
  - Hajlam nagy klaszterek szétvágására.

MFF: Felosztó hierarchikus klaszterezés

- Építsünk egy MFF-t (Minimális feszítő fa)
  - Induljunk egy tetszőleges pontból álló fából.
  - Egymás utáni lépésekben keressük meg a legközelebbi olyan *(p, q)* pontpárt, amelynél *p* eleme a fának *q* pedig nem.
  - Adjuk hozzá *q*-t a fához és húzzuk be a *p* és *q* közötti élt.





MFF: Felosztó hierarchikus klaszterezés

- Használjuk az MFF-t klaszterek hierarchiájának előállítására.

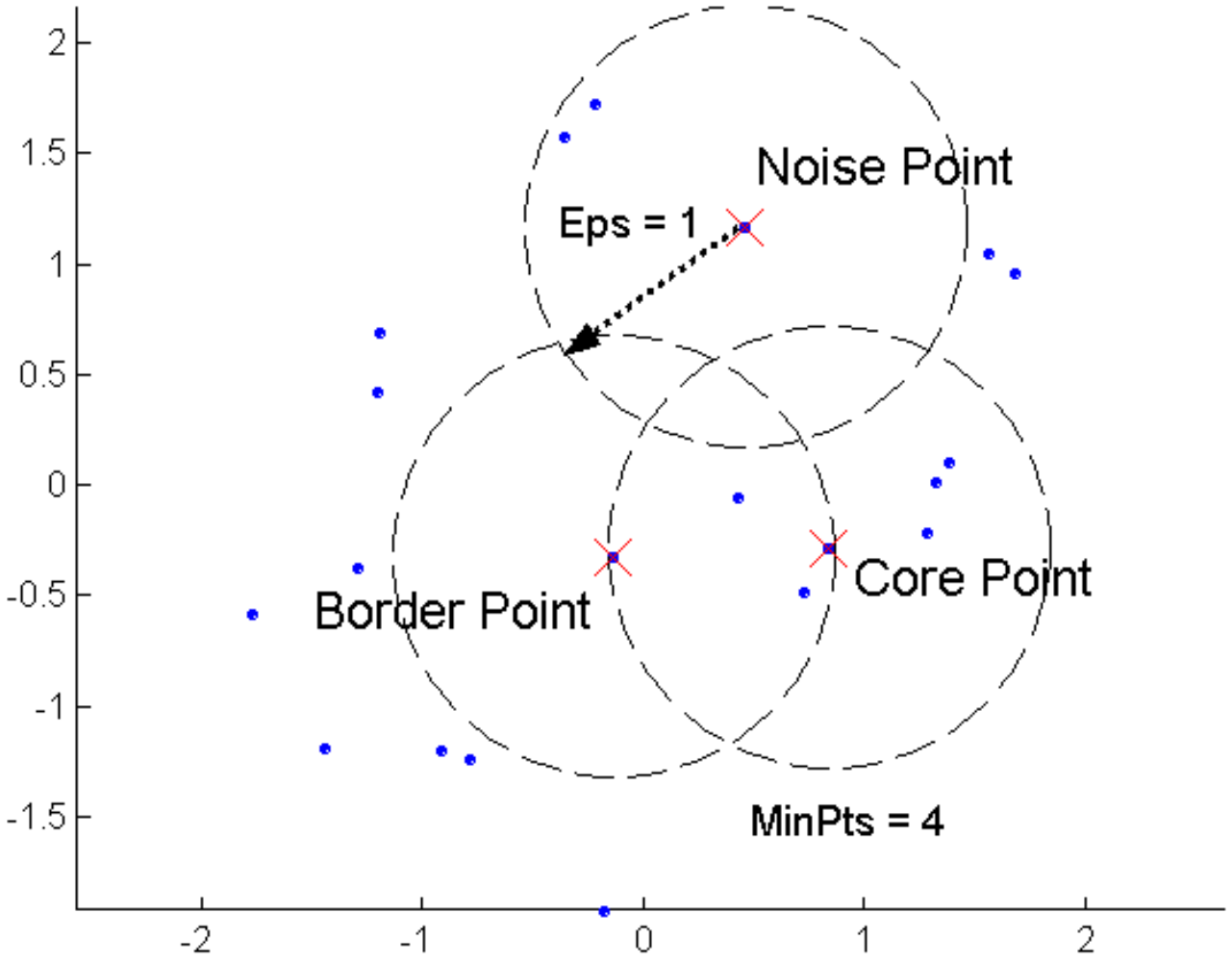
Algoritmus. MFF felosztó hierarchikus klaszterezés.

1. Határozzuk meg a minimális feszítő fát a közelségi gráfra.
2. **repeat**
3. Hozzunk létre egy új klasztert a legnagyobb távolságnak (legkisebb hasonlóságnak) megfelelő kapcsolat törlésével.
4. **until** Amíg egy elemű klaszterek nem maradnak.

DBSCAN

- DBSCAN (Density-Based Spatial Clustering of Applications with Noise – Sűrűség alapú térbeli klaszterezés hiba mellett)
- A DBSCAN egy sűrűség alapú algoritmus.
  - Sűrűség = egy rögzített sugáron (Eps) belüli pontok száma
  - Egy pont belső pont ha egy előírtnál (MinPts) több pont van Eps sugarú környezetében.
    - Ezek lesznek egy klaszter belsejének pontjai.
  - A határ pontnak az Eps sugarú környezetben MinPts-nél kevesebb pontja van, azonban van belső pont ebben a környezetben.
  - A zajos pont az összes olyan pont, amelyik nem belső illetve határ pont.

DBSCAN: belső, határ és zajos pont



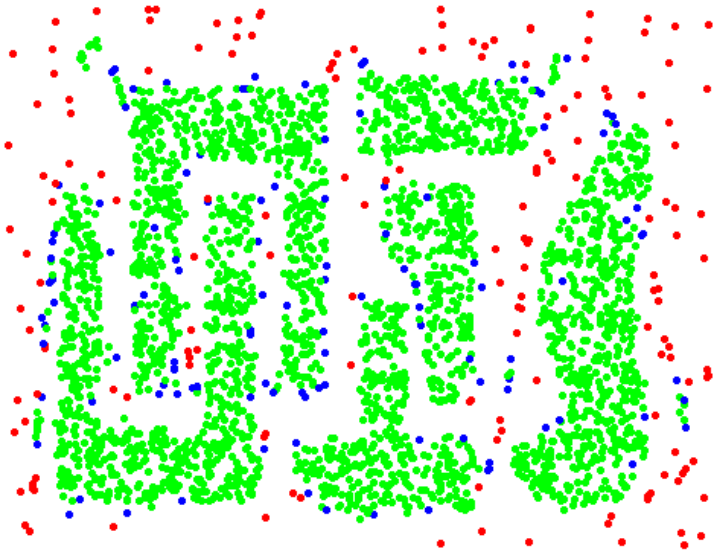
DBSCAN algoritmus

- Töröljük a zajos pontokat.
- Hajtsuk végre a klaszterosítást a fennmaradóakon.

```
current_cluster_label ← 1
for all core points do
  if the core point has no cluster label then
    current_cluster_label ← current_cluster_label + 1
    Label the current core point with cluster label current_cluster_label
  end if
  for all points in the Eps-neighborhood, except i^th the point itself do
    if the point does not have a cluster label then
      Label the point with cluster label current_cluster_label
    end if
  end for
end for
```

DBSCAN: belső, határ és zajos pont

Eredeti pontok

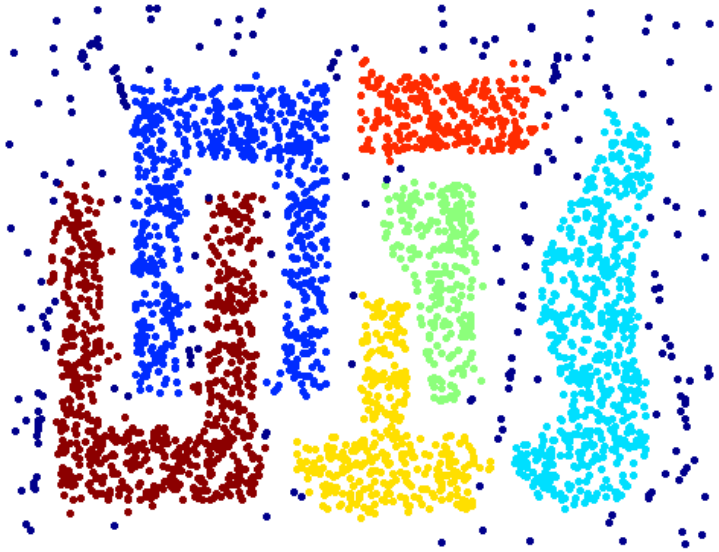


Pont típusok: belső, határ and zajos

Eps = 10, MinPts = 4

Amikor a DBSCAN jól működik

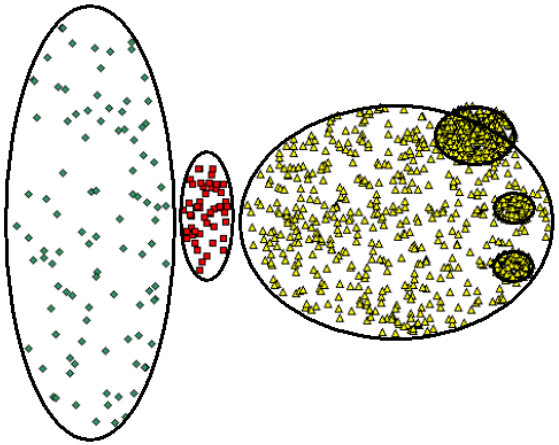
Eredeti pontok



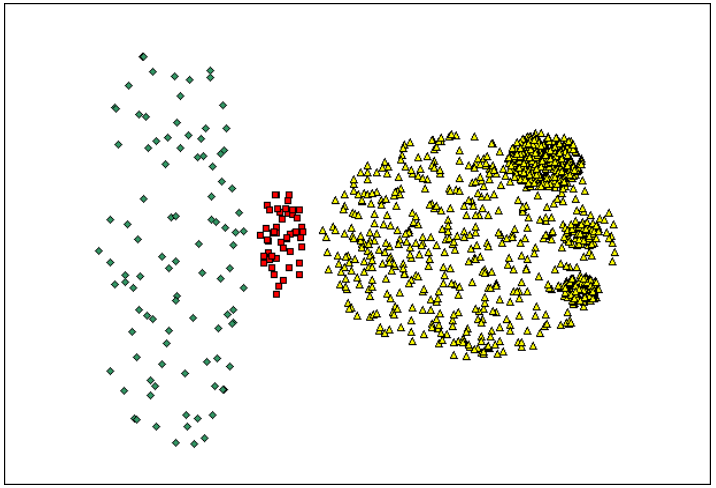
Klaszterek

- Ellenálló a zajjal szemben.
- Különböző méretű és alakú klasztereket egyaránt tud kezelni

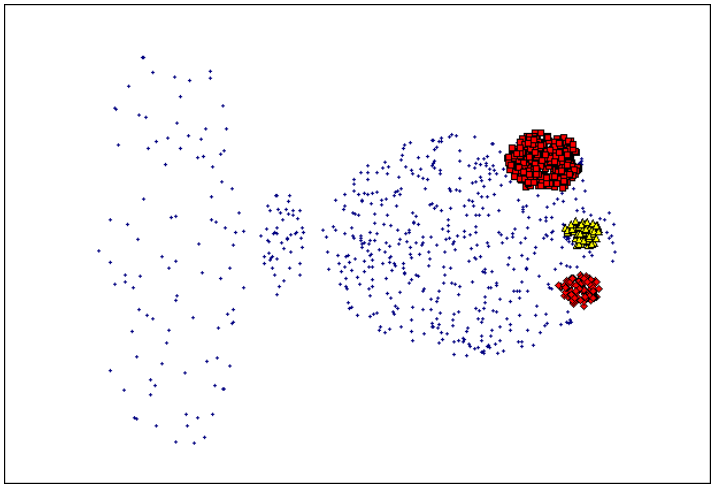
Amikor a DBSCAN nem működik jól



Eredeti pontok



(MinPts=4, Eps=9.75).

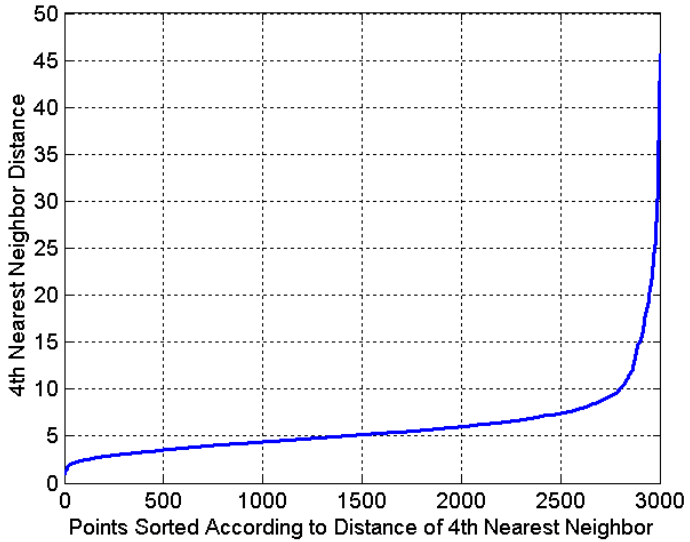


(MinPts=4, Eps=9.92)

- **Változó sűrűség**
- **Magas dimenziójú adatok**

DBSCAN: EPS és MinPts meghatározása

- Az ötlet az, hogy a klaszterbeli pontok durván ugyanakkora távolságra vannak a k^{th} –adik legközelebbi szomszédjuktól.
- A zajos pontoknak a k^{th} –adik legközelebbi szomszédja messze van.
- Ábrázoljuk a pontok és a k^{th} –adik legközebbi szomszédjuk közötti rendezett távolságokat. Ha egy viszonylag nagy platót kapunk (ld. ábra) az a jó.

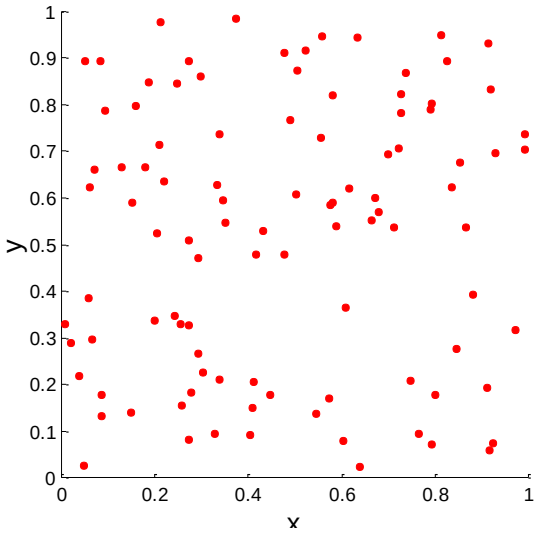


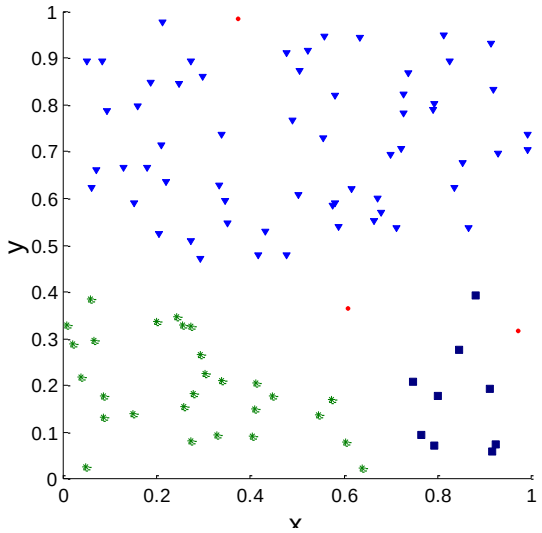
Klaszter validálás

- Felügyelt tanításnál számos mérőszám van annak mérésére, hogy mennyire jó egy modell.
  - Például: pontosság, visszahívhatóság
- Klaszterezésnél az analóg kérdés: hogyan mérhetjük az eredményül kapott klaszterek jóságát?
- Azonban a klaszter csak az azt néző szemében van!
- Ennek ellenére miért akarjuk mégis kiértékelni?
  - Hogy elkerüljük a zajban való mintázat keresést.
  - Hogy összehasonlítsunk klaszterező algoritmusokat.
  - Hogy összehasonlítsunk két klaszterezést.
  - Hogy összehasonlítsunk két klasztert.

Klaszterek véletlen adatokban

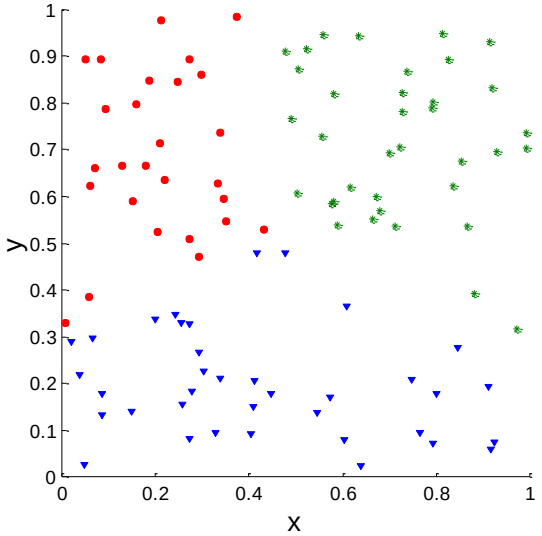
Véletlen pontok

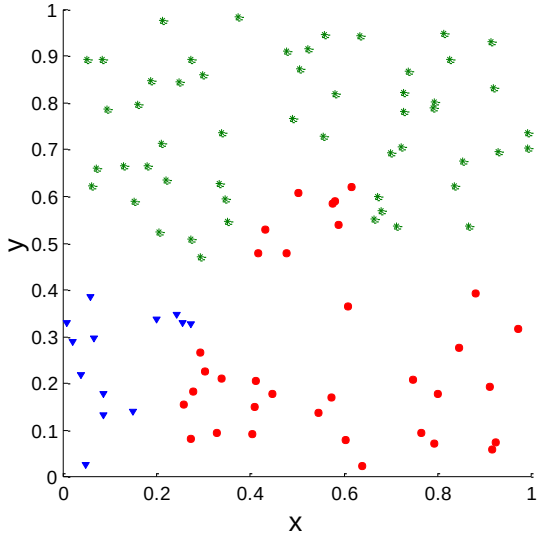




DBSCAN

K-közép





Teljes kapcsolás

A klaszter validálás különböző szempontjai

1. Adatok egy halmazán az ún. klaszterosodási hajlam meghatározása, azaz annak eldöntése, hogy van-e nem véletlen struktúra az adatokban.
2. A klaszterezés eredményeinek összehasonlítása külső ismert eredményekkel, pl. adott osztálycímkékkel.
3. Annak kiértékelése hogyan illeszkednek a klaszterezés eredményei az adatokra külső információkra való hivatkozás nélkül.

   - Csak az adatokat használjuk!
4. Két klaszterezés különböző eredményeinek összehasonlítása annak megállapítására hogy melyik a jobb.
5. A ,,helyes” klaszterszám meghatározása.

A 2., 3., és 4. pontoknál további szempontok lehetnek, hogy az egész klaszterezést vagy csak egyedi klasztereket akarunk kiértékelni.

A klaszter validálás mérőszámai

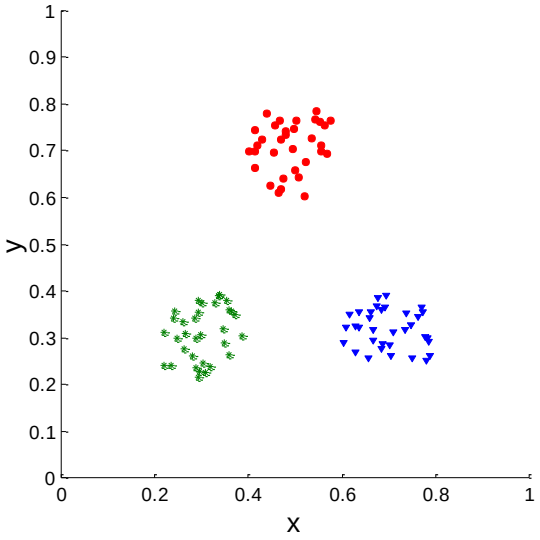
- A klaszter validálás különböző szempontjainak igazolására szolgáló numerikus mérőszámokat a következő 3 típusba sorolhatjuk.
  - Külső index: Annak mérésére használható, hogy melyik klaszter-címke illeszkedik egy külső osztálycímkére.
    - Entrópia
  - Belső index: A klaszterezés jóságának mérésére használható anélkül, hogy külső információkra támaszkodnánk.
    - Négyzetes hiba (SSE)
  - Relatív index: Két különböző klaszterezés vagy klaszter összehasonlítására használható.
    - Gyakran külső v. belső indexet használunk erre a célra, pl. SSE v. entrópia.
- Esetenként ezekre kritériumként hivatkozunk index helyett.
  - Azonban néha a kritérium általános eljárást jelöl és az index pedig egy numerikus mérőszámot, melyet az eljárás implementálásával kapunk.

Klaszter validálás korreláció útján

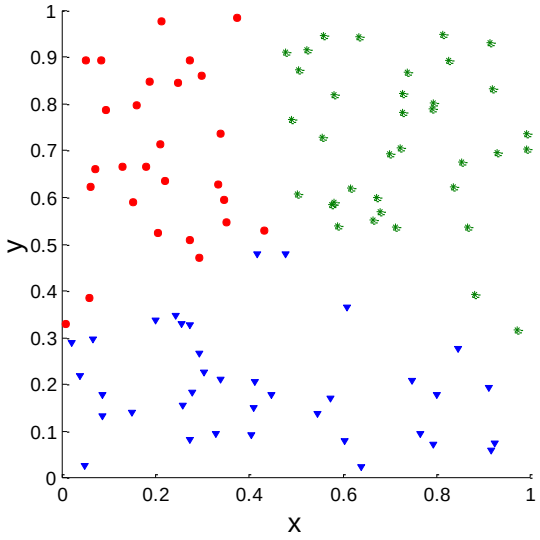
- Két mátrix
  - Közelségi mátrix
  - ,,Incidencia” mátrix
    - Egy sor és egy oszlop minden rekordra.
    - A mátrix eleme 1 ha a hozzárendelt pár ugyanabba a klaszterbe került.
    - A mátrix eleme 0 ha a hozzárendelt pár különböző klaszterbe került.
- Számoljuk ki a korrelációt a két mátrix között.
  - Mivel a mátrixok szimmetrikusak csak a n(n-1) / 2 számú elem közötti korrelációt kell kiszámolni.
- A magas korreláció arra utal, hogy azok a pontok, amelyek egy klaszterbe kerülnek, közel vannak egymáshoz.
- Nem elég jó mérőszám bizonyos sűrűségen alapuló vagy összefüggő klaszterek esetén.

Klaszter validálás korreláció útján

- Az incidencia és a közelségi mátrix korrelációja K-közép klaszterezés esetén a következő két adatállományra.



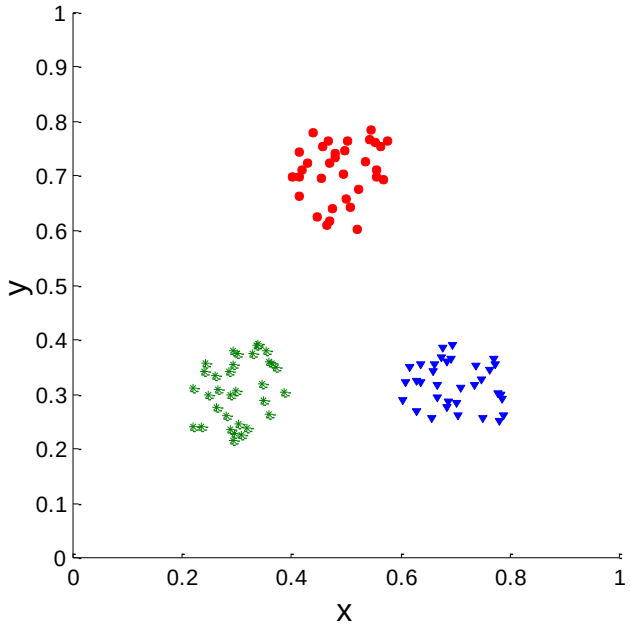
Corr = -0.9235

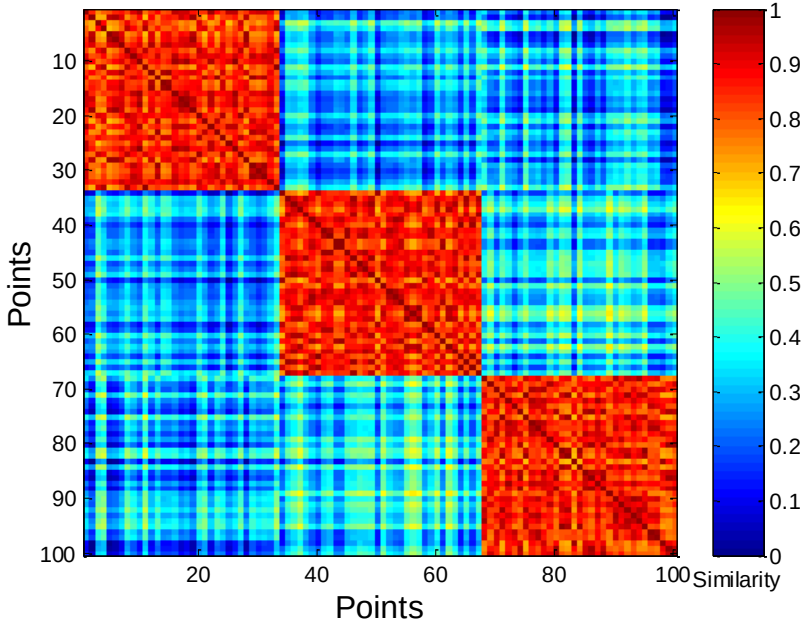


Corr = -0.5810

Klaszter validálás hasonlósági mátrixszal

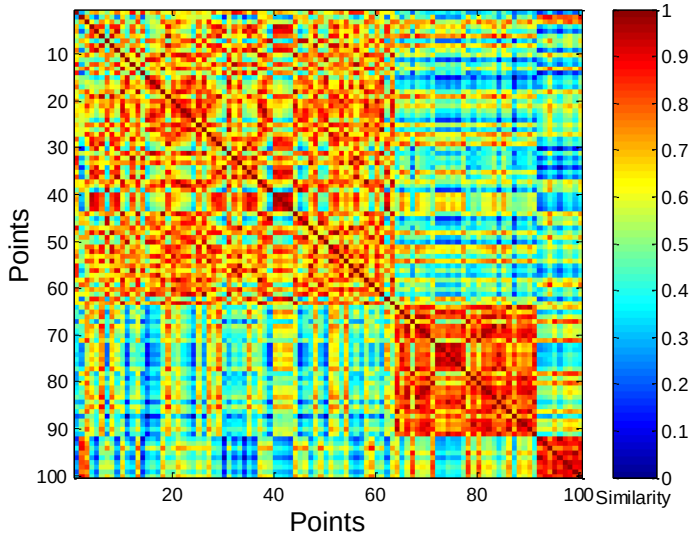
- Rendezzük a hasonlósági mátrixot a klasztercímkéknek megfelelően és vizsgáljuk grafikusan.

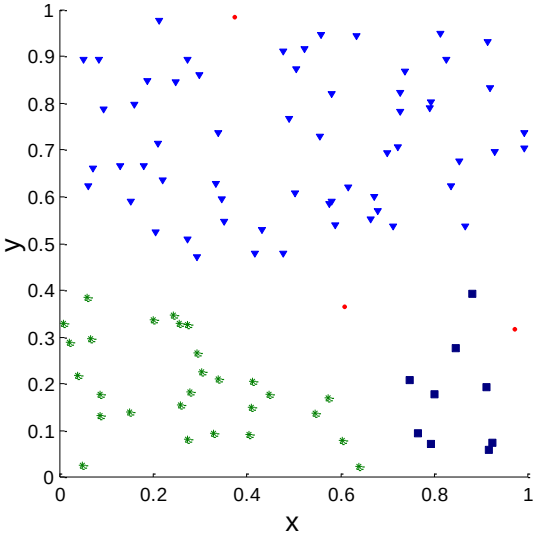




Klaszter validálás hasonlósági mátrixszal

- A véletlen adatokban a klaszterek nem olyan élesek.

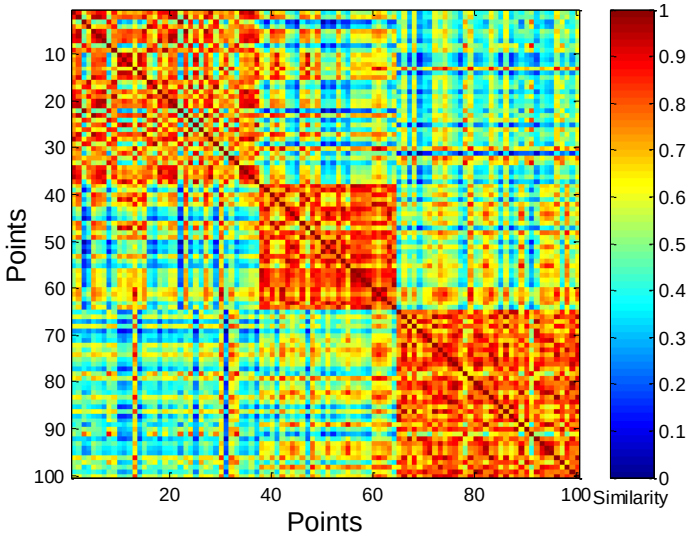


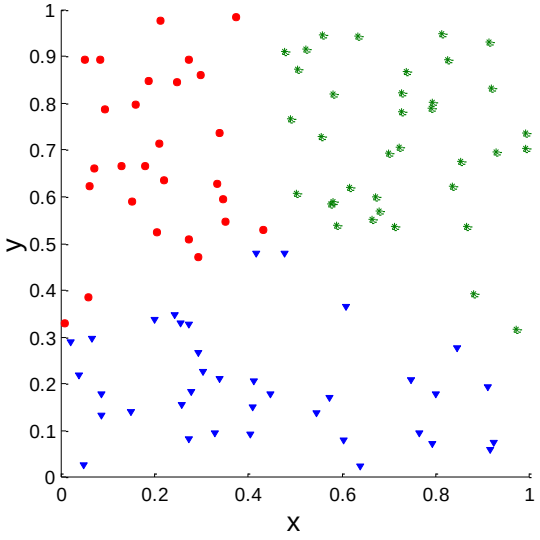


DBSCAN

Klaszter validálás hasonlósági mátrixszal

- A véletlen adatokban a klaszterek nem olyan élesek.

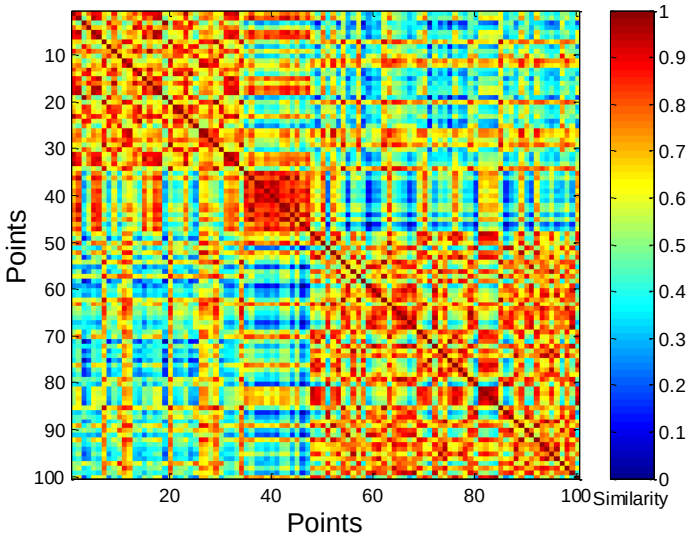


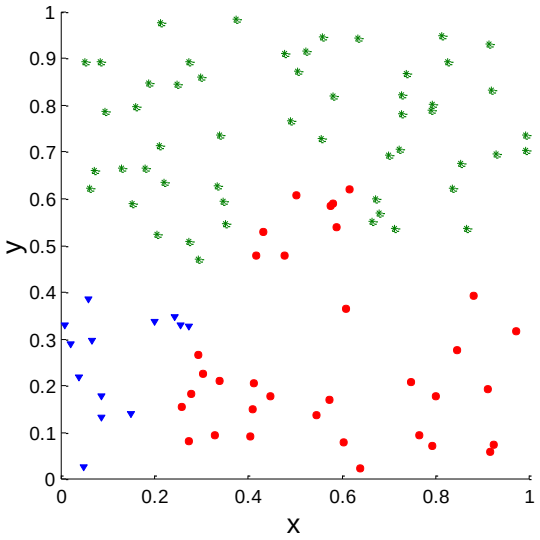


K-közép

Klaszter validálás hasonlósági mátrixszal

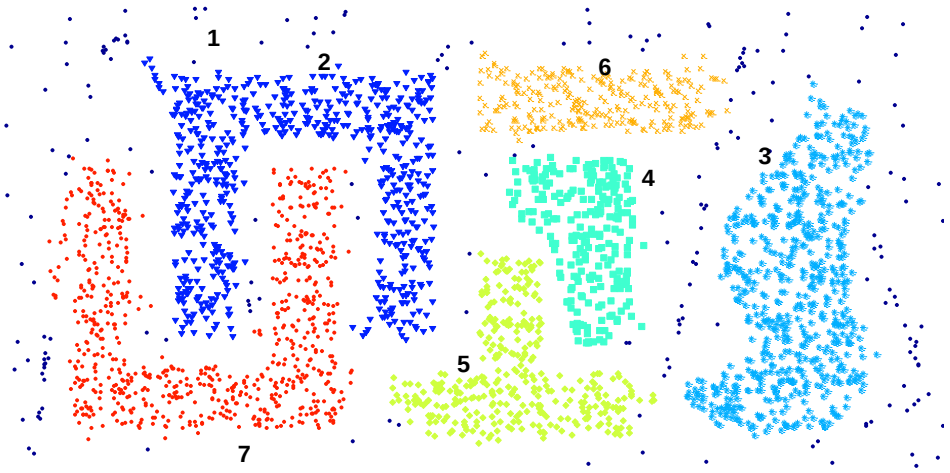
- A véletlen adatokban a klaszterek nem olyan élesek.

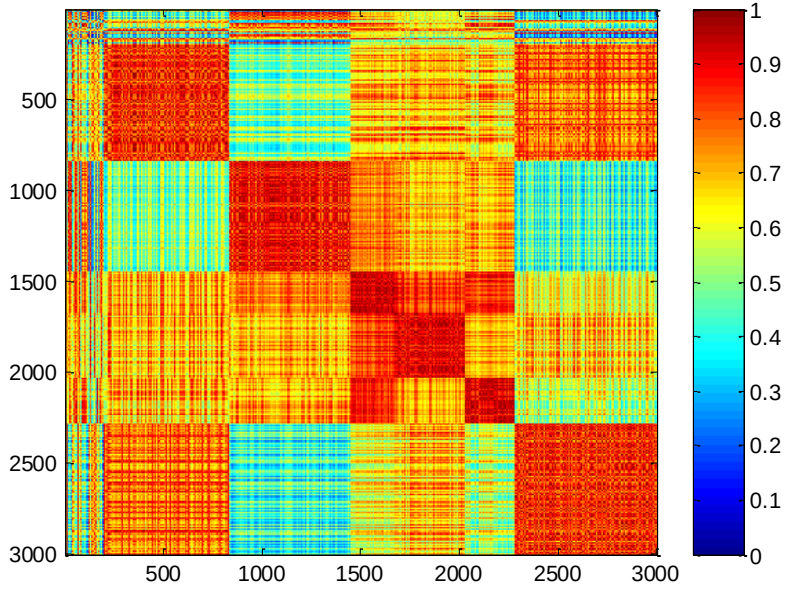




Teljes kapcsolás

Klaszter validálás hasonlósági mátrixszal

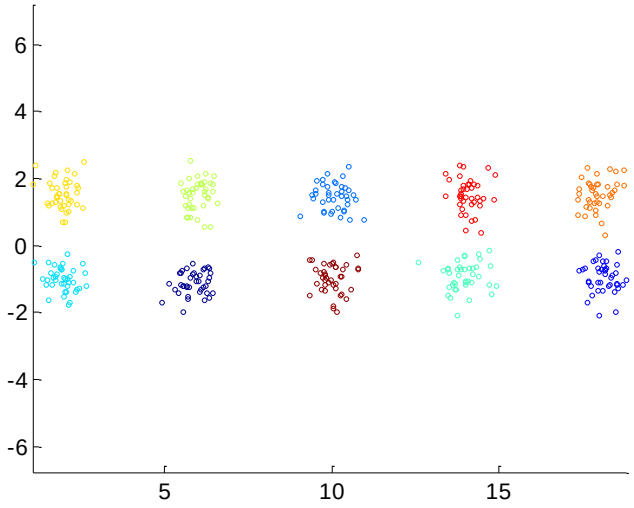


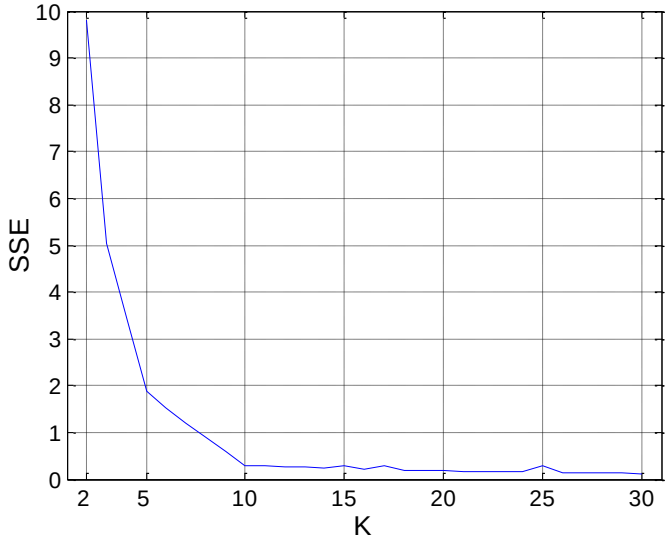


DBSCAN

Belső mérték: SSE

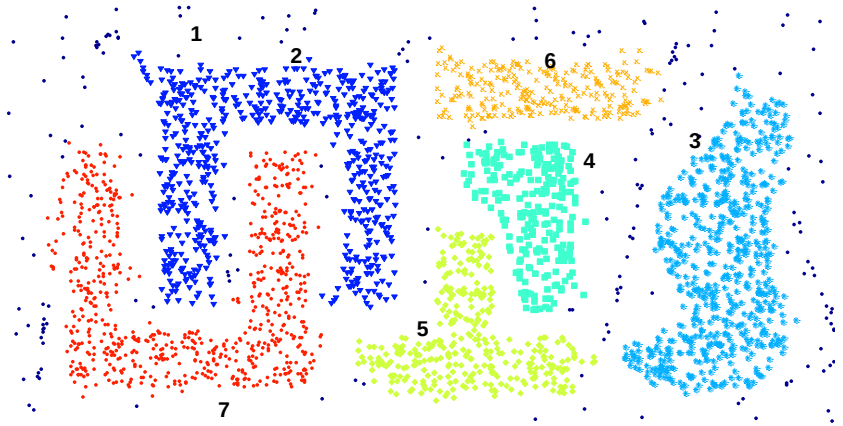
- Összetett alakzatoknál a klaszterek nem jól szeparálódnak.
- Belső index: A klaszterezés jóságának mérésére használható anélkül, hogy külső információkra támaszkodnánk.
  - SSE
- Az SSE jó két klaszterosítás vagy két klaszter összehasonlítására (átlagos SSE).
- Szintén használható a klaszterek számának becslésére.

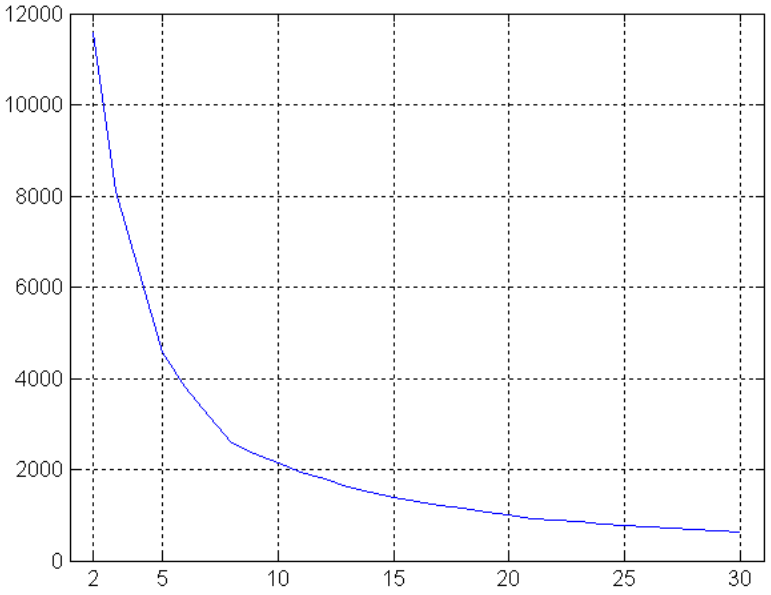




Belső mérték: SSE

- Az SSE görbe összetett adatállományra.





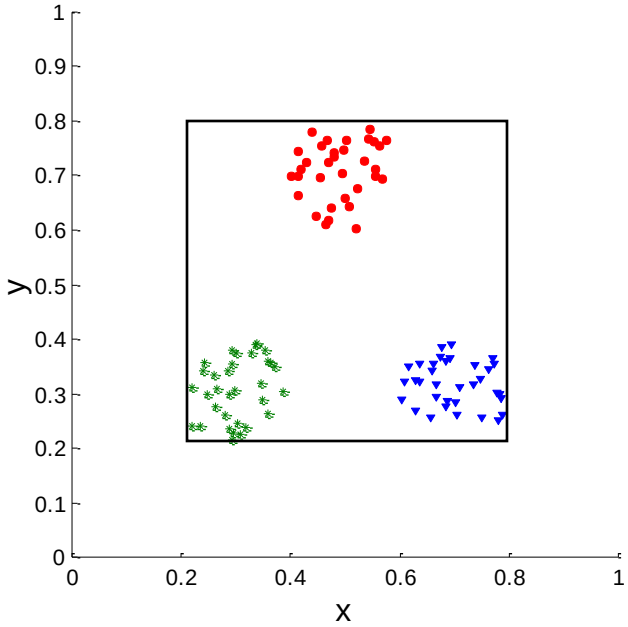
SSE a K-közép által talált klaszterekre

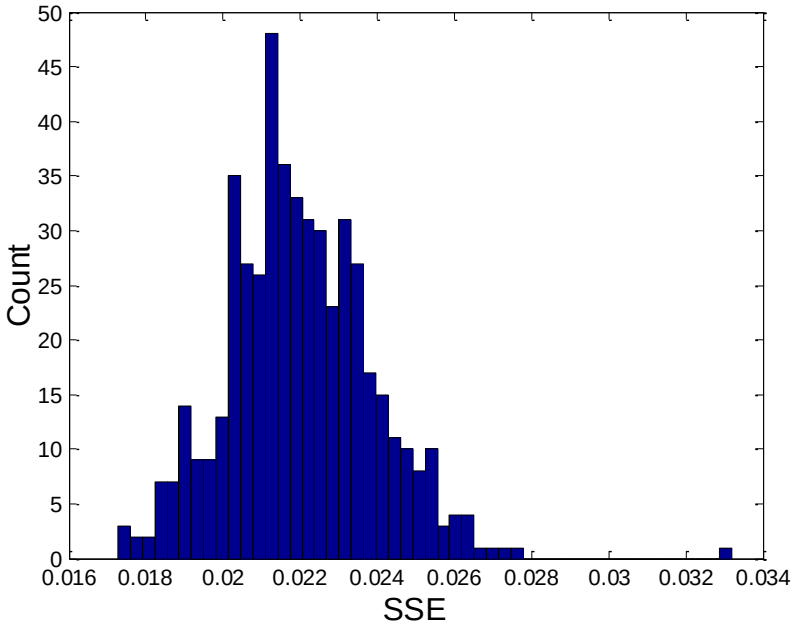
A klaszter validálás kerete

- Egy keret szükséges bármilyen mérték értelmezésére.
  - Például, ha egy kiértékelésnél kapott mérőszám 10, akkor az jó, rossz vagy elégtelen?
- A statisztika szolgáltat keretet a validálásra
  - Minél kevésbé ,,tipikus” egy klaszterezés végeredménye annál valószínűbb, hogy az létező struktúrát jelenít meg az adatokban.
  - Össze tudjuk hasonlítani azokat az idexértékeket, amelyek egyrészt véletlen adatok, másrészt valós klaszterek klaszterosításaiból jönnek.
    - Ha egy index értéke valószínűtlen, akkor a klaszterezés eredménye érvényes.
  - Ezek a megközelítések bonyolultabbak és nehezebben megérthetőek.
- Két különböző klaszterezés végeredményének össze-hasonlítására már kevésbé szükséges a fenti keret.
  - Azonban fennmarad az a kérdés, hogy a két index közötti különbség szignifikáns-e.

Statisztikus keret az SSE-re

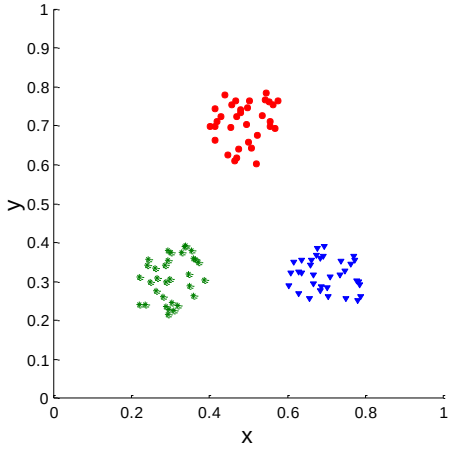
- Példa
  - Hasonlítsuk össze a baloldali adatokra kapott SSE=0.005 értéket véletlen adatoknál kapott 3 klaszter indexével.
  - A hisztogram véletlen adatok 500 elemű halmazára 100-szor végzett 3 klaszterrre való klaszterosítás SSE indexének eloszlását mutatja, amely a (0.2,0.8) intervallumba esik.

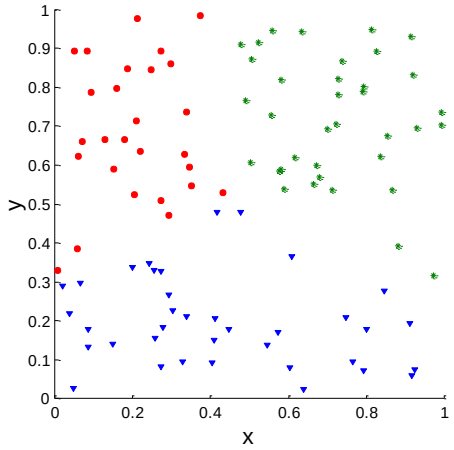


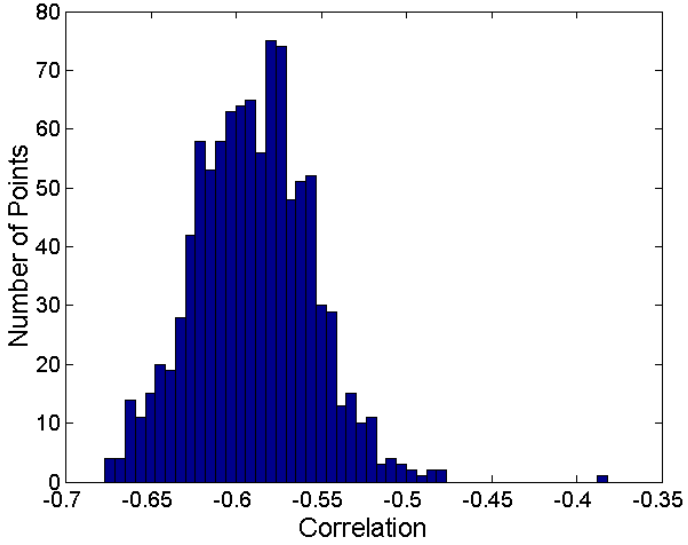


Statisztikus keret az SSE-re

- Az alábbi két adatállományra végrehajtott K-közép klaszterezésnél az incidencia mátrix és a közelségi mátrix korrelációja.







Corr = -0.9235

Corr = -0.5810

Belső mértékek: összetartás és elválasztás

- Klaszter összetartás: azt méri, hogy milyen szorosan kapcsolódnak a klaszterbeli objektumok.
  - Példa: SSE
- Klaszter elválasztás: azt méri, hogy mennyire különbözik egy klaszter a többitől.
- Példa: Négyzetes hiba
  - Az összetartást a klasztereken belüli négyzetösszeggel mérjük

$$WSS = \sum_i \sum_{x \in C_i} (x - m_i)^2$$

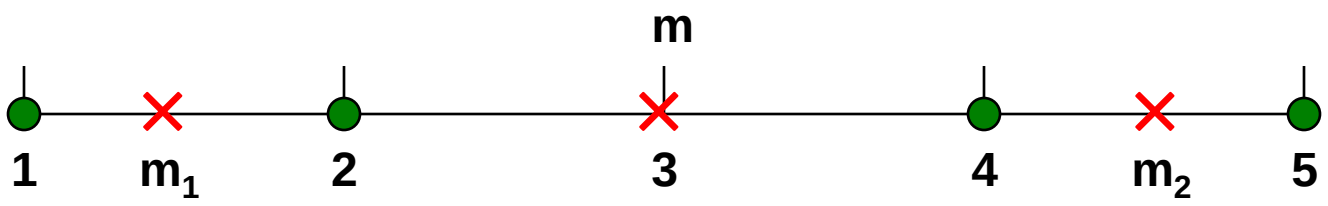
  - Az elválasztást a klaszterek közötti négyzetösszeggel mérjük

$$BSS = \sum_i |C_i| (m - m_i)^2$$

    - ahol $|C_i|$ az i-edij klaszterbeli elemek száma.

Belső mértékek: összetartás és elválasztás

- Példa: SSE
  - BSS + WSS = konstans



K=1 klaszter:

$$WSS = (1-3)^2 + (2-3)^2 + (4-3)^2 + (5-3)^2 = 10$$

$$BSS = 4 \times (3-3)^2 = 0$$

$$Összes = 10 + 0 = 10$$

K=2 klaszter:

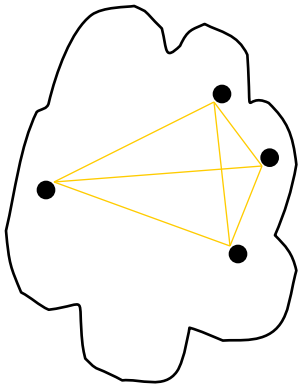
$$WSS = (1-1.5)^2 + (2-1.5)^2 + (4-4.5)^2 + (5-4.5)^2 = 1$$

$$BSS = 2 \times (3-1.5)^2 + 2 \times (4.5-3)^2 = 9$$

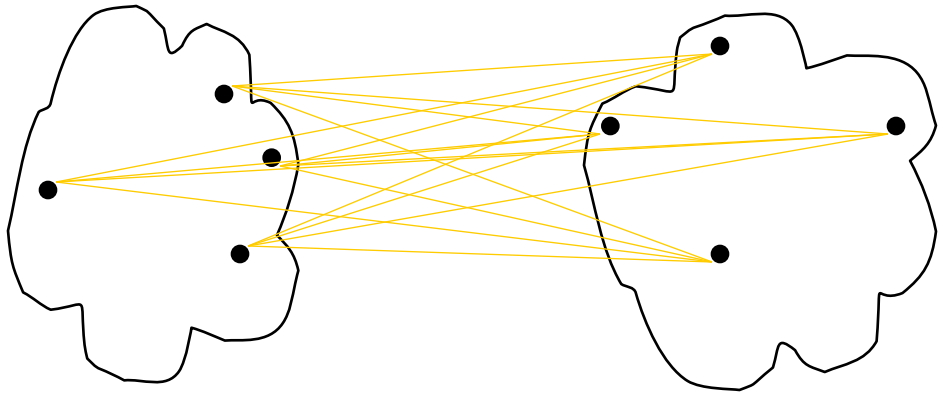
$$Összes = 1 + 9 = 10$$

Belső mértékek: összetartás és elválasztás

- A közelségi gráfon alapuló megközelítést egyaránt használhatjuk összetartás és elválasztás mérésére.
  - Egy klaszter összetartása a klaszteren belüli összes kapcsolat súlyának az összege.
  - Egy klaszter elkülönülése a klaszter elemei és más klaszteren kívüli elemek közötti kapcsolatok súlyainak az összege.



összetartás



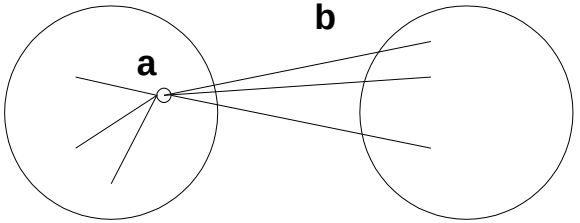
elválasztás

Belső mértékek: árnykép együttható

- Az árnykép együttható az összetartás és az elválasztás ötletét kombinálja mind egyedi pontokra, mind pedig klaszterekre és klaszterezésekre.
- Egy *i* egyedi pontra
  - Számoljuk ki ***a*** = az *i* átlagos távolságát a klaszteren belüli pontoktól.
  - Számoljuk ki ***b*** = min (az *i* átlagos távolságát más klaszterektől)
  - Egy pont árnykép együtthatóját az alábbi módon definiáljuk:

    $s = 1 - a/b$ ha $a < b$, (vagy $s = b/a - 1$ ha $a \geq b$, nem szokványos eset)

  - Általában 0 és 1 közé esik.
  - Minél közelebb van az 1-hez annál jobb.



- Az átlagos árnykép szélességet ezután klaszterekre és klaszterezésekre is ki tudjuk terjeszteni.

Klaszter validálás külső mértékei: entrópia és tisztaság

Table 5.9. K-means Clustering Results for LA Document Data Set

Cluster	Entertainment	Financial	Foreign	Metro	National	Sports	Entropy	Purity
1	3	5	40	506	96	27	1.2270	0.7474
2	4	7	280	29	39	2	1.1472	0.7756
3	1	1	1	7	4	671	0.1813	0.9796
4	10	162	3	119	73	2	1.7487	0.4390
5	331	22	5	70	13	23	1.3976	0.7134
6	5	358	12	212	48	13	1.5523	0.5525
Total	354	555	341	943	273	738	1.1450	0.7203

entropy For each cluster, the class distribution of the data is calculated first, i.e., for cluster j we compute p_{ij}, the 'probability' that a member of cluster j belongs to class i as follows: $p_{ij} = m_{ij}/m_j$, where m_j is the number of values in cluster j and m_{ij} is the number of values of class i in cluster j. Then using this class distribution, the entropy of each cluster j is calculated using the standard formula $e_j = \sum_{i=1}^{L} p_{ij} \log_2 p_{ij}$, where the L is the number of classes. The total entropy for a set of clusters is calculated as the sum of the entropies of each cluster weighted by the size of each cluster, i.e., $e = \sum_{i=1}^{K} \frac{m_i}{m} e_j$, where m_j is the size of cluster j, K is the number of clusters, and m is the total number of data points.

purity Using the terminology derived for entropy, the purity of cluster j, is given by $purity_j = \max p_{ij}$ and the overall purity of a clustering by $purity = \sum_{i=1}^{K} \frac{m_i}{m} purity_j$.

Utolsó megjegyzés klaszter validáláshoz

,,Klaszterezések validálása a legnehezebb és legfrusztrálóbb része a klaszteranalízisnek.

Az ebbe az irányba tett jelentős erőfeszítések nélkül a klaszteranalízis továbbra is fekete mágia marad, amely csak azon igaz hívők számára érhető el, akik nagy bátorsággal és gyakorlattal bírnak.”

Jain & Dubes: *Algorithms for Clustering Data*