KLASZTERSZÁMOK MEGHATÁROZÁSÁNAK EGY LEHETSÉGES MEGOLDÁSA[1]

RUFF FERENC
Szent István Egyetem

A dolgozat azon problémával foglalkozik, hogy a klaszteranalízis során létrejövő lehetséges megoldások (különböző klaszterszámok) esetén melyiket fogadjuk el az adatbázisban feltételezett csoportok legjobb közelítésének. Ilyen módszerek léteznek, jelen dolgozat egy ilyen eljárás kritikai vizsgálatával, valamint annak továbbfejlesztésével foglalkozik. Az eljárás lényege, hogy a klaszterek középpontja és a középpontok közötti osztópont körüli elemsűrűségekkel definiált index segítségével jellemzi a csoportosítás pontosságát. Ez olyan népszerű algoritmusok, mint pl. a K-means – ahol az algoritmus inputjaként meg kell adni az elvárt klaszterszámot – esetében ad segítséget a döntés meghozatalában.

1 Bevezetés

„A klaszterelemzés az alakfelismerés tanító nélküli tanuló algoritmusa. Egyszerűen úgy definiáljuk, hogy a klaszterelemzés megfigyelések egyedeit bontja viszonylag homogén csoportokba p változó értékeinek hasonlósága alapján. A klaszterelemzés az egyedek olyan csoportosítását keresi, amelyekre igaz, hogy egy egyed egy és csakis egy csoporthoz tartozik, és azokhoz az egyedekhez lesz hasonló, amelyekkel egy klaszterbe került, míg a többi klaszterbe tartozó egyedektől különbözik." [6, 160. old.]

Klasszifikáló elemzésnek valamint numerikus taxonómiának is nevezik, mely az 1950-es években indult fejlődének [15]. Azóta nagyon sokféle módszert dolgoztak ki a fenti célok megvalósításának érdekében. Az irodalmak között megjelentek összefoglaló jellegű ill. egy-egy szakterület számára íródott művek [2, 9, 16]. Magyar szerzők tollából származó könyvek is találhatók ezek között [5, 6, 7]. Kimondottan a marketingkutatás területén történő alkalmazással foglalkozik Simon [14] cikke.

Dolgozatom középpontjában az a probléma áll, hogy ha az elemzőnek kell megadnia a keresett klaszterek számát (az algoritmus inputjaként), akkor a különböző klaszterszám-beállítások esetén kapott eredmények közül milyen módon választhatja ki a „legjobbat". Ezt az angol nyelvű irodalomban „cluster validation" néven találhatjuk meg, amely alatt olyan kvantitatív elemzést értenek, mely a klaszteranalízis eredményeként létrejött csoportokat vizsgálja [16]. Ennek megoldására sok eljárás született, melyeket Theodoridis és

[1]Beérkezett: 2013. január 2. E-mail: ruff.ferenc@gtk.szie.hu.

Koutroumbas [16] három típusba sorol: külső kritérium alapú, belső kritérium alapú valamint relatív kritérium alapú. Egy kicsit más csoportosítást alkalmaz Füstös et al. [6, 205. old.]: „A klaszterek érvényessége (validitása) négy kritérium alapján vizsgálható. Külső követelményként értelmezhető az, ha ismert csoportokba tartozó egyedekből veszünk mintát, és arra végezzük el a klaszterezést. Belső követelménynek tekinthetők azok a mutatók, amelyekkel az eredeti és a származtatott távolságok illeszkedését mérjük. Harmadik megközelítést jelent a megismételhetőség kritériuma, amelynek lényege a kettéosztott megfigyelések klaszterezése és a felosztások összevetése. A klaszterek érvényességének relatív kritériuma az adatmátrix több eljárás szerinti klaszterezését és a felosztások közötti egyezés mérését fogalmazza meg."

Liu et al. [13] munkájában a klaszterszámok meghatározásával kapcsolatban végrehajtott vizsgálatának célja az volt, hogy megfigyeljék, hogy a vizsgált indexek pontosságára (11 ilyen indexet teszteltek) – amelyek külső információt nem tartalmaztak – milyen hatással van az adatok szerkezete (zajos adatok, sűrűség különbségek, alcsoportok, aszimmetrikus eloszlás). Ezek közül az alcsoportok felismerése okozta a legtöbb problémát az ellenőrzés során, ezen esetben a legtöbb index nem adott helyes eredményt. Egy olyan index – az ún. *S_Dbw* index – volt a 11 között, mely mindegyik esetben helyes döntést hozott. Az eljárást Halkidi és Vazirgiannis [8] dolgozta ki, mely a klaszterek közötti sűrűségkülönbségen alapszik. Ezt fejlesztette tovább Kim és Lee [10] valamint Tong és Tan [17] abba az irányba, hogy robusztusabb[2] legyen, valamint ne csak gömbszimmetrikus klasztereket ismerjen fel. Ennek fontosságára korábban felhívta a figyelmet Legány et al. [12] is, akik megfigyelték, hogy az általuk vizsgált indexek (pl. az *S_Dbw* is) csak jól szeparált, gömbszimmetrikus klaszterek esetén nyújtottak megfelelő segítséget a klaszterek validálásához. Dolgozatomban ezen módszerek vizsgálatával és továbbfejlesztésével foglalkozom.

2 Az S_Dbw_{new} index

Vizsgálatom kiindulópontja a Halkidi és Vazirgiannis [8] által kidolgozott, majd Kim és Lee [10] valamint Tong és Tan [17] által továbbfejlesztett módszer – alapja az *S_Dbw* (Scatter and Density between clusters) index – mely a sűrűségkülönbségek és a szórások alapján rendel hozzá egy adott csoportosításhoz egy valós számot. A különböző csoportosításokhoz tartozó értékek alapján lehet a legjobban illeszkedő megoldást kiválasztani. Itt most csak a legutolsó változattal foglalkozom, mert ez jobb eredményeket ért el a tesztelések során, mint az első két változat.

A módszer alapja, hogy a klaszterek közötti hasonlóságot ill. a klaszterek közötti különbséget bizonyos pontok körül kialakított tartományokon belül található megfigyelési egységek számának (mint sűrűségnek) összehasonlítása

[2] A kiugró adatokra kevésbé érzékenyen határozza meg a klaszterek számát.

alapján határozták meg [17]. Ők az indexet S_Dbw_{new}-nak nevezték (megkülönböztetésül az előzményektől), és ezt a jelölést itt is megtartom.

Legyen adott egy adatbázis, amely N számú egyedet, mint megfigyelési egységet tartalmaz. Az egyedek tulajdonságait k db változóval írjuk le. Ezen adatok egy $N \times k$ méretű mátrixba rendezhetők. Ezen adatbázison futtassunk le egy klaszterező módszert, így kapjuk a megfigyelési egységeink egy csoportosítását (c db klasztert). Ezen csoportosításhoz fogunk hozzárendelni egy számot, amely az S_Dbw_{new} index egy lehetséges értéke. Itt most az említett cikk [17] bemutatása történik.[3]

Az indexnek két összetevője van: $Dens_{bw}(c)$ – klaszteren belüli sűrűség, valamint $Scat(c)$ – klaszterek közötti variancia.

$$Dens_{bw}(c) = \frac{1}{c(c-1)} \sum_{i=1}^{c} \left[\sum_{\substack{j=1 \\ j \neq i}}^{c} \frac{\text{density}^*(\mathbf{m}_{ij})}{\max\{\text{density}^*(\mathbf{v}_i), \text{density}^*(\mathbf{v}_j)\}} \right] \tag{1}$$

ahol c: a kialakított klaszterek száma, $\mathbf{v}_i$: az i-edik klaszter középpontja.

$$\text{density}^*(\mathbf{m}) = \sum_{i=1}^{n_{\boldsymbol{m}}} f^*(\mathbf{x}_i, \mathbf{m}) \tag{2}$$

$\mathbf{x}_i$: az i-edik megfigyelési egység, $\mathbf{m}$: egy tetszőleges megfigyelési egység, n_m: a figyelembe vett megfigyelési egységek száma.

$$f^*(\mathbf{x}_i, \mathbf{m}) = \begin{cases} 1, & \text{ha } CI_-^p \leq d(x_i^p, m^p) \leq CI_+^p \,, \; \forall p \in \{1, 2, 3, \ldots, k\} \\ 0, & \text{egyébként} \end{cases} \tag{3}$$

k: a megfigyelési változók száma, továbbá

$$CI_\pm^p = v_l^p \pm \left(1.96 \cdot \frac{\sigma_l^p}{\sqrt{n_l}} \right) \tag{4}$$

v_l^p, σ_l^p, n_l: a figyelembe vett klaszter p-edik változójának átlaga ill. szórása, valamint a klaszter elemszáma. Legyen továbbá $\mathbf{m}_{ij}$ az i-edik és j-edik klaszter középpontját összekötő szakasz olyan osztópontja, mely a két klasztert „elválasztja", és melynek p-edik komponense:

$$m_{ij}^p = 0.7 \cdot \left(\frac{n_j \cdot v_i^p + n_i \cdot v_j^p}{n_i + n_j} \right) + 0.3 \cdot \left(\frac{\text{density}^*(\mathbf{v}_i) \cdot v_i^p + \text{density}^*(\mathbf{v}_j) \cdot v_j^p}{\text{density}^*(\mathbf{v}_i) + \text{density}^*(\mathbf{v}_j)} \right) \tag{5}$$

n_i: az i-edik klaszter elemszáma. Az $\mathbf{m}_{ij}$ számításakor figyelembe veszi a két klaszter elemszámait, valamint a két klaszter középpontja körüli sűrűséget, és a kettő kombinációja[4] adja az osztópontot. E részindex (1. egyenlet) számításának elve tehát, hogy összehasonlítja a klaszterek középpontja körüli,

[3] Ahol az eredeti cikk jelölésrendszere nem volt egészen világos, ott ennek módosítására került sor.

[4] A súlyok (0.7 - 0.3) meghatározása empirikus vizsgálatok tapasztalatai alapján történt [17].

valamint a klaszterközéppontok között kiválasztott pont ($\mathbf{m}_{ij}$) körül elhelyezkedő egyedek számát.

A másik részindex számításának módja:

$$Scat(c) = \frac{1}{c-1} \sum_{i=1}^{c} \frac{n - n_i}{n} \cdot \frac{\|\sigma^2(\mathbf{v}_i)\|}{\|\sigma^2(\mathbf{S})\|} \tag{6}$$

ahol $\sigma^2(\mathbf{v}_i)$: a $\mathbf{v}_i$ középpontú klaszter variancia vektora[5], $\sigma^2(\mathbf{S})$: az adatbázis variancia vektora, $\|.\|$: vektor euklideszi normája.

Ezekből a részindexekből a következő módon adódik az index:

$$S\_Dbw(c) = Dens_{bw}(c) + Scat(c) \tag{7}$$

Legyen $\mathbf{S}$ olyan adatbázis, mely konvex klasztereket tartalmaz. Futtassunk le ezen, különböző klaszterszám beállításával, egy klaszterező eljárást többször. Belátható, hogy az index akkor vesz fel minimális értéket, ha a klaszterező eljárás a tényleges klasztereket találta meg [8]. Természetesen nem garantált, hogy a klaszterek képzése során a tényleges klaszterek (ha léteznek) valóban előállnak megoldásként. Ekkor is az index minimumát fogadjuk el megoldásként, mivel ez jelenti a legjobb szeparációt [8].

3 Az S_Dbw_{new} index kritikája

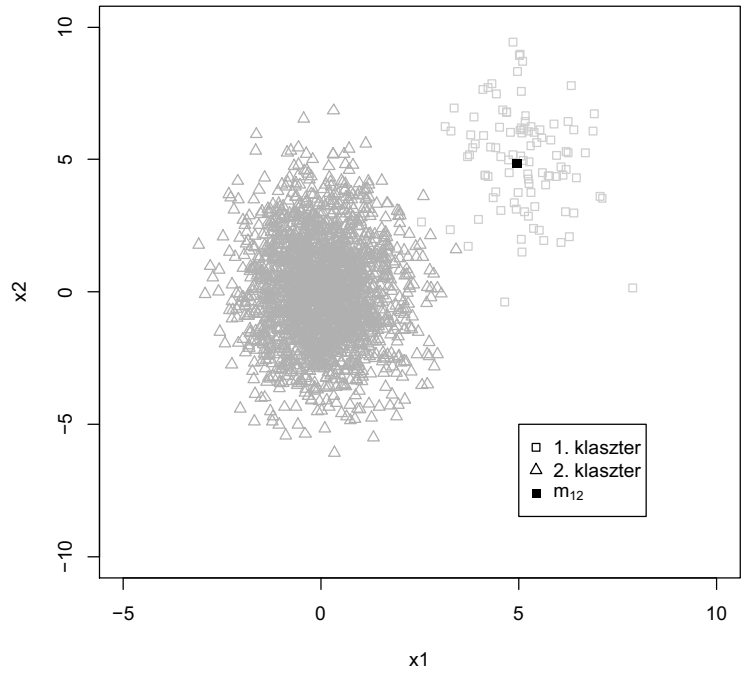
A 3. egyenlet megadja, hogy a ,,sűrűség" számításánál mely egyedeket kell figyelembe venni, és melyeket nem. Az adott pont környezetének definiálása határozza meg ezt a számot. Látható, hogy a CI hossza a klaszter számának (n) növekedésével csökken[6] (4. egyenlet). Ez pedig azt jelenti, hogy a lecsökkenő területen (még a nagy egyedszám mellett is) kevés egyed található, vagy egyáltalán nem is találunk egyedet. Ezzel pedig a mérés válik lehetetlenné, hiszen nem lesz alkalmas a sűrűbb és ritkább tartományok elkülönítésére.

Következő észrevételem, hogy az $\mathbf{m}_{ij}$ osztópont (5. egyenlet) számításánál két szempontot vettek figyelembe. Az első fele a két klaszter- középpontot összekötő szakaszt a klaszterek elemszámának arányában osztja, méghozzá úgy, hogy amelyik klaszternek nagyobb az elemszáma, attól távolabb lesz az osztópont. A második rész pedig a klaszter-középpontok körüli sűrűségek arányában osztja a szakaszt úgy, hogy amelyik klaszter esetében a sűrűség nagyobb volt, ahhoz kerül közelebb az osztópont. Ezen két hatás konvex lineáris kombinációjából állították elő $\mathbf{m}_{ij}$-t, méghozzá kísérletekből, tapasztalati úton állították be az együtthatókat (0.7 - 0.3). Kísérleteim szerint az így kialakított osztópont a két klaszter eltérő elemszáma esetén jelentősen eltolódhat a kevesebb elemet tartalmazó klaszter közelébe. Szélsőséges esetben a nagy elemszámú klaszter ,,beletolja" az osztópontot a kevesebb elemet tartalmazó klaszterbe. Ez látható az *1. ábrán*, amely esetében az elemszámok aránya 1:20. Az osztópont eltolódásának egyik oka, hogy a nagy elemszám

[5] A koordinátatengelyek irányába számolt varianciákból képzett vektor.

[6] A $\boldsymbol{CI}$ 0-hoz tart, ha $\boldsymbol{n} \rightarrow \infty$.

miatt a CI értéke olyan kicsi lett, hogy abba nem került elem, így a sűrűség értéke 0, ami azt jelenti, hogy a második része a képletnek (5. egyenlet) nem kompenzálja az első rész hatását (hiszen ezek itt éppen egymás ellen hatnának). Sőt, mint ahogyan 1. ábrán látható eset hátterében is megfigyelhető volt, ha a kisebb sűrűségű klaszter esetében az adott tartományba véletlenül belekerül egy pont, míg a nagyobb sűrűségű esetében nem, akkor az még jobban növeli a torzító hatást (ld. 5. egyenlet).



1. ábra. Klaszterek középpontja közötti Tong és Tan [17] féle osztópont eltolódása 2 változó esetén. *Forrás:* saját szerkesztés.

Ezen észrevételeket támasztják alá a következő gondolatmenetek. Az 1. ábra 1. klaszterének adatai: $\mathbf{v}_1 = (5,5)^T$, $\sigma = (1,2)^T$, $n_1 = 100$, valamint 2. klaszterének adatai: $\mathbf{v}_2 = (0,0)^T$, $\sigma = (1,2)^T$, $n_2 = 2000$. Keresem annak a valószínűségét, hogy egy megfigyelési egység a klaszterközéppont megfelelő (ld. 3. és 4. egyenlet) környezetébe esik. Jelentse ξ_{1x} ill. ξ_{1y} az első klaszterbe tartozó pont x ill. y koordinátáját (normál eloszlású valószínűségi változók). Továbbá ξ_{2x} ill. ξ_{2y} a második klaszterbe tartozó pont x ill. y koordinátáját. Az 1. klaszter esetében (x és y irányban):

$$\mathrm{P}\left(5 - 1.96 \cdot \frac{1}{\sqrt{100}} < \xi_{1x} < 5 + 1.96 \cdot \frac{1}{\sqrt{100}}\right) = 2\Phi\left(\frac{1.96}{\sqrt{100}}\right) - 1 = 0.155$$

és

$$\mathrm{P}\left(5 - 1.96 \cdot \frac{2}{\sqrt{100}} < \xi_{1y} < 5 + 1.96 \cdot \frac{2}{\sqrt{100}}\right) = 2\Phi\left(\frac{1.96}{\sqrt{100}}\right) - 1 = 0.155$$

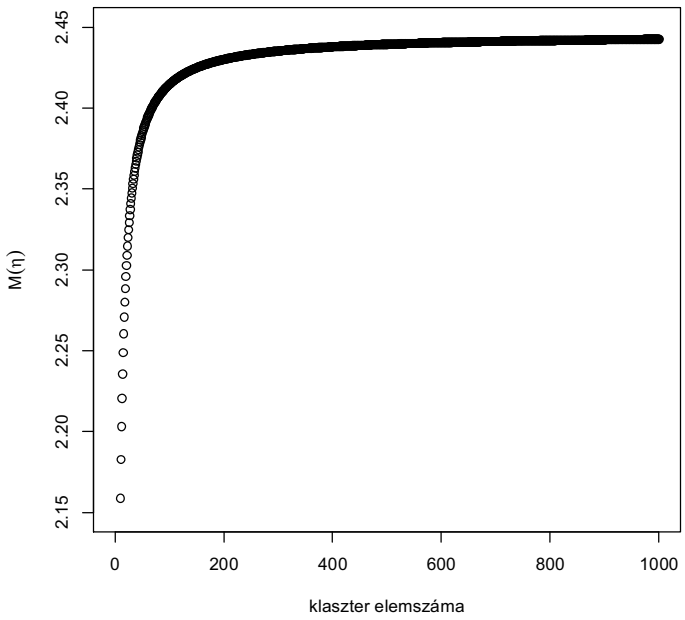
A keresett valószínűség tehát $p_1 = 0.155^2 = 0.0241$. Legyen η_1 egy diszkrét valószínűségi változó, és jelentse a középpont megadott környezetében található egyedek számát. Ennek várható értéke (binomiális eloszlás esetén): $\mathrm{M}(\eta_1) = n_1 \cdot p_1 = 100 \cdot 0.0241 = 2.41$. A 2. klaszter esetében (x és y irányban):

$$\mathrm{P}\left(0 - 1.96 \cdot \frac{1}{\sqrt{2000}} < \xi_{1x} < 0 + 1.96 \cdot \frac{1}{\sqrt{2000}}\right) = 2\Phi\left(\frac{1.96}{\sqrt{2000}}\right) - 1 = 0.034$$

és

$$\mathrm{P}\left(0 - 1.96 \cdot \frac{2}{\sqrt{2000}} < \xi_{1y} < 0 + 1.96 \cdot \frac{2}{\sqrt{2000}}\right) = 2\Phi\left(\frac{1.96}{\sqrt{2000}}\right) - 1 = 0.034$$

A keresett valószínűség tehát $p_2 = 0.034^2 = 0.0012$. Legyen η_2 egy diszkrét valószínűségi változó, és jelentse a középpont megadott környezetében található egyedek számát. Ennek várható értéke (binomiális eloszlás esetén): $\mathrm{M}(\eta_2) = n_2 \cdot p_2 = 2000 \cdot 0.0012 = 2.44$. Vagyis a 2. klaszter 20-szor annyi elemet tartalmaz, mégis, a középpont megadott környezetében található elemek száma közelítőleg annyi, mint az 1. klaszter esetében, átlagosan 2.44. Annak valószínűsége pedig, hogy egy elem sem esik a megadott környezetbe: $\mathrm{P}(\eta_2 = 0) = (1 - 0.0012)^{2000} = 0.0906$. Az 1. ábrához tartozó eset (a nagy klaszterben 0, a kicsiben nem 0 a középpont meghatározott környezetében található elemek száma) valószínűsége pedig $0.0906 \cdot (1 - (1 - 0.0241)^{100}) = 0.088$, ami nem elhanyagolható, tehát bekövetkezésével számolni kell.



2. ábra. $\mathrm{M}(\eta)$ a klaszter elemszámának függvényében a Tong-féle környezet esetén. *Forrás:* saját szerkesztés.

Érdekes következmény továbbá, hogy a fent számított két valószínűség (p_1 és p_2) közel azonos. Megvizsgáltam tehát a $\mathrm{M}(\eta)$ értékét az n (klaszterelemek száma) függvényében. A grafikon (*2. ábra*) egy monoton növekvő függvény képét mutatta. Kiszámítottam a kapott függvény határértékét a végtelenben.

$$\lim_{k\to\infty}\left(2\Phi\left(\frac{1.96}{\sqrt{k}}\right)-1\right)^2\cdot k=\lim_{k\to\infty}\frac{\left(2\Phi\left(\frac{1.96}{\sqrt{k}}\right)-1\right)^2}{k^{-1}}=$$

$$=\lim_{k\to\infty}\frac{2\left(2\Phi\left(\frac{1.96}{\sqrt{k}}\right)-1\right)\cdot 2\phi\left(\frac{1.96}{\sqrt{k}}\right)\cdot 1.96\cdot\left(-\frac{1}{2}k^{-\frac{3}{2}}\right)}{-k^{-2}}=$$

$$=\lim_{k\to\infty}\frac{4\phi\left(\frac{1.96}{\sqrt{k}}\right)\cdot 1.96\cdot\left(-\frac{1}{2}k^{-\frac{3}{2}}\right)\cdot 1.96\cdot\phi\left(\frac{1.96}{\sqrt{k}}\right)}{-\frac{1}{2}k^{-\frac{3}{2}}}+$$

$$+\lim_{k\to\infty}\frac{2\cdot\left(2\Phi\left(\frac{1.96}{\sqrt{k}}\right)-1\right)\cdot\phi'\left(\frac{1.96}{\sqrt{k}}\right)\cdot 1.96^2\cdot\left(-\frac{1}{2}k^{-\frac{3}{2}}\right)}{-\frac{1}{2}k^{-\frac{3}{2}}}=$$

$$=\lim_{k\to\infty}\left[4\cdot 1.96^2\cdot\phi^2\left(\frac{1.96}{\sqrt{k}}\right)+2\cdot\left(2\Phi\left(\frac{1.96}{\sqrt{k}}\right)-1\right)\cdot\phi'\left(\frac{1.96}{\sqrt{k}}\right)\cdot 1.96^2\right]=$$

$$=4\cdot 1.96^2\cdot\left(\frac{1}{\sqrt{2\pi}}\right)^2+0\cdot 0=\frac{2\cdot 1.96^2}{\pi}\approx 2.4456\,.$$

Vagyis növelve a klaszterek elemszámát, a középpont adott környezetében található elemek számának várható értéke lényegében konstansnak tekinthető. Ennek oka a korábban már említett terület csökkenése, mely terület a klaszter elemszámával fordítottan arányos.

Végül meghatároztam az η_2 valószínűségi változó eloszlását, és annak egy részletét tartalmazza az *1. táblázat* (a várható érték környezete[7]). Ebből is látszik, hogy az 1-3 objektum előfordulásának legnagyobb a valószínűsége, a maximuma 2-nél van. Ezáltal a fejezet elején tett megállapításaimat igazoltam.

y_i	$\mathrm{P}(\eta_2 = y_i)$
0	0.0905
1	0.2177
2	0.2613
3	0.2092
4	0.1255
5	0.0602
⋮	⋮

1. táblázat. Az η_2 valószínűségi változó eloszlásának részlete. *Forrás:* saját számítás.

[7] y_i az η_2 valószínűségi változó által felvehető értékeket jelenti. Mivel binomiális eloszlású valószínűségi változóról van szó, ezért $y_i \in \{0, 1, 2, \ldots, 2000\}$

4 Az S_Dbw_{new} index módosítása

4.1 A módosított index (S_Dbw^{**})

A kritikai részben megfogalmazott hibák miatt a tartomány megválasztásának módosítását javaslom. Az eredeti javaslat – 3. egyenlet – helyett a következőképpen definiálom az f^* függvényt, amelyet megkülönböztetésül f^{**}-nak nevezek:

$$f^{**}(\mathbf{x}_i, \mathbf{m}) = \begin{cases} 1 & \text{ha } m^p - \alpha \cdot D^p \leq x_i^p \leq m^p + \alpha \cdot D^p,\ \forall p \in \{1, 2, 3, \ldots, k\} \\ 0 & \text{egyébként.} \end{cases} \tag{8}$$

ahol $\mathbf{m}$ egy tetszőleges egyed; m^p a tetszőleges egyed p-edik változójának értéke; $D^p = \min_i(\sigma_i^p)\,,\ i \in \{1, 2, \ldots, c\}$, a klaszterelemek p-edik változójának szórásai közül a minimális; α egy alkalmasan megválasztott konstans.

A módosítás lényege, hogy az az intervallum, amelyen belül a megfigyelési egységeket keresem, már független az n-től (a klaszterelemek számától), így egy adott intervallumba eső megfigyelési egységek száma (az adott térrészben) arányos lesz a klaszterek elemszámával. Másrészt, az $\mathbf{m}_{ij}$ osztópontok esetében a korábban említett torzító hatás is megszűnik.

4.2 A klaszterek közötti mérőszám ($Dens_{bw}^{**}$) elemzése

Az 1. egyenlet adja meg a klaszterek közötti sűrűségkülönbség alapján, hogy mely klasztereket tekintünk majd különbözőnek, és melyeket nem tudunk megkülönböztetni. A következő elemzésben két klaszter egymáshoz viszonyított helyének függvényében vizsgálom az index értékét, két változó bevonása mellett.

Legyen adott két klaszter (C_1 és C_2). Középpontjaik: $\mathbf{v}_1 = (0, 0)^T$ és $\mathbf{v}_2 = (a, 0)^T$. Mindkettő legyen kör alakú, azonos átmérővel (mindkét irányú szórásuk legyen 1-1). Legyen $\alpha = 0.5$ (8. egyenlet). Az elméleti megközelítés esetében nem konkrét elemekkel megadott klasztereket vizsgálok, hanem a két klasztert két-két normál eloszlású valószínűségi változóval jellemzem (ξ_{1x}, ξ_{1y}, ξ_{2x}, ξ_{2y}). Ilyen feltételek mellett vizsgálom az alábbi három valószínűséget:

$$p_1 = \mathrm{P}\left((0 - 0.5 \cdot 1 < \xi_{1x} < 0 + 0.5 \cdot 1) \wedge (0 - 0.5 \cdot 1 < \xi_{1y} < 0 + 0.5 \cdot 1)\right)\,,$$

mely arányos a C_1 klaszter középpontja körüli $\alpha \cdot D^1$, azaz $0.5 \cdot 1$ sugarú tartományba, valamint (y irányban) a C_1 klaszter középpontja körüli $\alpha \cdot D^2$, azaz $0.5 \cdot 1$ sugarú tartományba eső pontok számával (mely tartomány egy téglalap).

$$p_2 = \mathrm{P}\left((a - 0.5 \cdot 1 < \xi_{2x} < a + 0.5 \cdot 1) \wedge (0 - 0.5 \cdot 1 < \xi_{2y} < 0 + 0.5 \cdot 1)\right)\,,$$

mely ugyanaz, mint az előbb, csak a C_2 klaszterre vonatkoztatva.

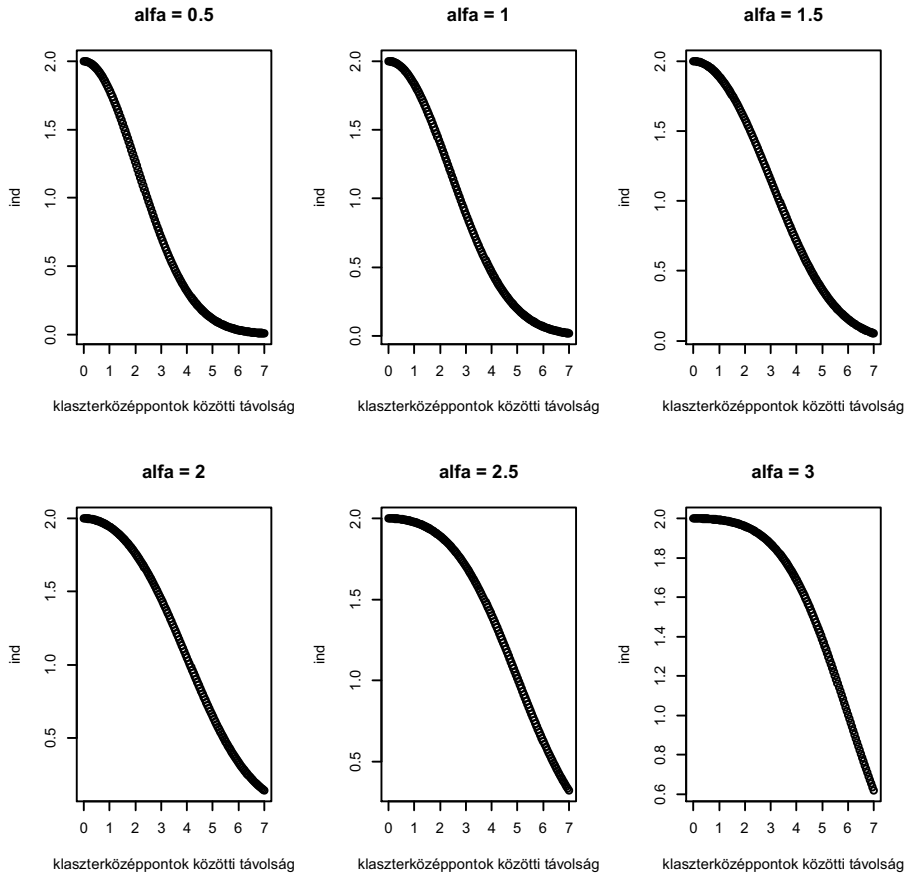
$$p_k = 2 \cdot \mathrm{P}((\frac{a}{2} - 0.5 \cdot 1 < \xi_{1x} < \frac{a}{2} + 0.5 \cdot 1) \wedge (0 - 0.5 \cdot 1 < \xi_{1y} < 0 + 0.5 \cdot 1))\,,$$

mely jelentése azonos az előzőekkel, csak a két középpontot összekötő szakasz felezőpontjára vonatkoztatva. A 2-vel való szorzás a két eloszlás azonossága miatt alkalmazható. Ezen mennyiségek segítségével definiálom a következő indexet:

$$ind := \frac{p_k}{\max(p_1, p_2)} \tag{9}$$

mely index arányos az 1. egyenletben megadott $Dens_{bw}$ indexszel. Értékkészlete a $[0, 2]$, hiszen maximális értéket akkor vesz fel, ha a két klaszter középpontja egybeesik.

A definiált index vizsgálata során azt figyeltem, hogy miként változik az index értéke a középpontok távolságának függvényében. A távolság értékét 0- tól 7-ig változtattam (a szórás értéke 1), azaz $a \in [0, 7]$. A kapott *ind*(*távolság*) függvényt a *3. ábra* első grafikonja ($\alpha = 0.5$) mutatja.

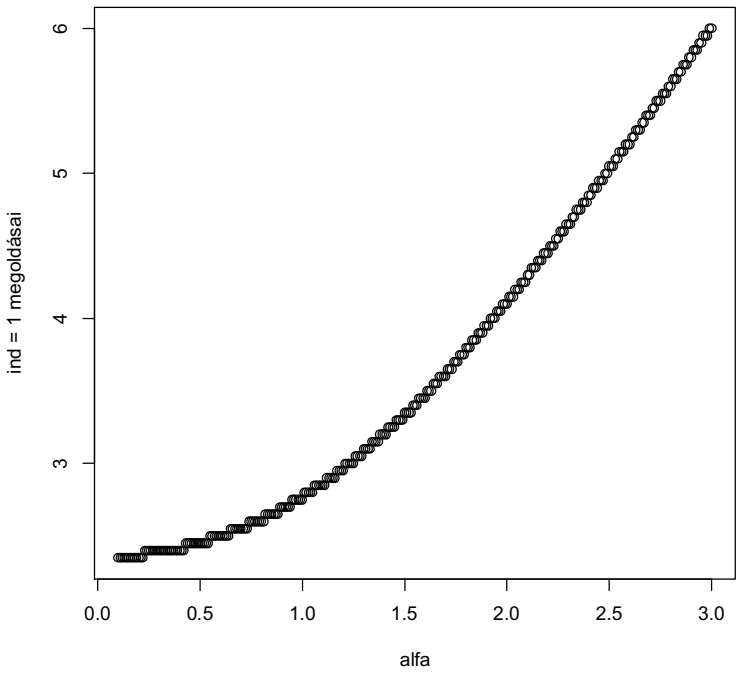


3. ábra. Az „*ind*" index a klaszterközéppontok közötti távolság függvényében különböző *alfa* paraméterek esetén. *Forrás:* saját számítás.

A függvény az 1-et, mint függvényértéket az $x = 2.4$ helyen veszi fel, ami azt jelenti, hogy a két középpont ezen távolsága esetén a két középpont adott környezetében (ld. α) ugyanannyi megfigyelési egység található, mint a két középpontot elválasztó osztópont (jelen esetben felezőpont) ugyanazon környezetében. A három pont tehát a sűrűség szempontjából egymástól nem megkülönböztethető. A távolság további növelésével az index értéke (csökkenő ütemben) tovább csökken.

A fenti kísérletben az α értékét 0.5-nek választottam. Hogy hogyan változnak a függvény értékei más α paraméterértékek esetén, a 3. ábra további grafikonjai mutatják. Látható, hogy az α értéket növelve az $ind = 1$ egyenlet megoldásai – vagyis azon klaszterközéppont távolságok, melyekre az *ind* index értéke 1 lesz – is egyre növekednek.

Vagyis célszerű minél kisebb α értéket választani. Azonban a normál eloszlás esetében az elméleti valószínűségek számolhatók akkor is, ha α nagyon kicsi, addig a konkrét adatbázis esetén ezek a kicsi intervallumok nem tartalmaznak majd megfigyelési egységeket, vagyis nem lesznek alkalmasak az összehasonlításra. Ennek további vizsgálata céljából megrajzoltam egy függvényt, mely azon klaszterközéppontok távolságát adja meg az α függvényében, melyek esetében az *ind* index értéke 1 lett (*4. ábra*).



4. ábra. Az $ind = 1$ eredményt adó klasztertávolságok az *alfa* paraméter függvényében.
Forrás: saját számítás.

A grafikon azt mutatja, hogy a változó (α) növekedésével egyre növekvő mértékben növekszik a függvényérték. Más szóval, minél nagyobb α értéket választunk, annál távolabbinak kell lennie a két klaszternek, hogy meg tudjuk különböztetni őket[8]. Mivel a függvény monoton növekvő, ezért az optimális értéket az értelmezési tartomány bal oldalán veszi fel. Itt bizonyos tartományonként konstans értéket vesz fel (ezen a részén a növekedés lassúbb), vagyis ezen a részen kell egy alkalmas α értéket kiválasztani. A továbbiakban legyen ez az érték $\alpha = 0.4$, figyelembe véve a korábbi érveket.

További kérdés azonban, hogy a fenti kísérlet eredménye 2 gömbszimmetrikus klaszter esetében lett kiértékelve. Van-e ennek hatása a 4. ábra grafikonjának jellegére? A kísérlet különböző szórás paraméterek mellett megismételve ugyanazt a jellegű görbét adta.

4.3 A módosított S_Dbw^{**} index szerkezetének vizsgálata

A vizsgálat egyik célja, hogy a teljes index értékét a két részindex változásának függvényében figyelhessük meg. Ennek modellezésére egy három klaszterből álló adatbázist készítettem, amelyben két klaszter helyét nem változtattam, a harmadikat pedig kiindulásként az egyik fix klaszterre helyeztem, majd távolítottam tőle az x tengely mentén (miközben a másik klaszterhez sem közelítettem). A két egymást átfedő klaszter egyszer egynek, majd két különböző klaszternek tekintettem, és vizsgáltam az indexek értékét mindkét változat esetében. A harmadik klaszterra azért volt szükség, hogy minden esetben legyen legalább két klaszter, amire az index számolható.

Először mindhárom (C_1, C_2, C_3) klaszter x és y irányú szórását azonosra állítom: $\sigma_{1x} = \sigma_{2x} = \sigma_{3x} = \sigma_{1y} = \sigma_{2y} = \sigma_{3y} = 1$. A $\mathbf{v_1} = (0,0)^T$, $\mathbf{v_2} = (d,0)^T$, ahol $d \in [0;7]$, továbbá $\mathbf{v_3} = (0,-7)^T$ pedig az egyes klaszterek középpontjait határozzák meg. Mindhárom klaszter 1000 megfigyelési egységet tartalmazott. Először a C_1 és a C_2 klasztert összevontam egy klaszterré, majd pedig külön klaszternek tekintettem őket, és mindkét esetben vizsgáltam az indexek értékét, miközben az d értékét 0-tól 7-ig változtattam bizonyos lépésközönként. Az eredmények a *2. táblázatban* láthatók. Az egyes részindexeket, valamint a teljes indexet is párba állítottam a két klaszteres ill. a három klaszteres megoldások esetében. A két utolsó oszlop összehasonlításából látható, hogy az indexek nagyságában kb. 3.5-4 egység távolság ($3.5 < d < 4$) esetén váltás történik. Innentől kezdve tehát a három klasztert tartalmazó megoldást fogadjuk el a másikkal szemben. Vagyis, ha a két klaszter szórása 1-1 egység, akkor középpontjuk kb. 4 egység távolságra kell, hogy legyen, hogy két különböző klaszterként értékelje őket az index. Vagyis nem szükséges teljesen átfedésmentesnek lenniük („jól szeparált"), bizonyos átfedés esetén is felismerhető a kettő különbözősége.

[8]Itt még nem került sor annak vizsgálatára, hogy az index milyen értéke mellett különböztethető meg két klaszter. Erre később kerül sor.

Távolság d	$Dens\_bw^{**}$ $nc=2$	$Dens\_bw^{**}$ $nc=3$	$Scat$ $nc=2$	$Scat$ $nc=3$	S_Dbw^{**} $nc=2$	S_Dbw^{**} $nc=3$
0.0	0.0053	0.3281	0.0592	0.0776	0.0644	0.4057
0.5	0.0000	0.3076	0.0593	0.0790	0.0593	0.3866
1.0	0.0000	0.2266	0.0608	0.0770	0.0608	0.3036
1.5	0.0093	0.2336	0.0671	0.0792	0.0764	0.3128
2.0	0.0156	0.1911	0.0715	0.0782	0.0872	0.2693
2.5	0.0147	0.1774	0.0779	0.0792	0.0926	0.2566
3.0	0.0294	0.1188	0.0871	0.0776	0.1165	0.1964
3.5	0.0777	0.1004	0.0927	0.0744	*0.1704*	*0.1748*
4.0	0.0437	0.0408	0.1046	0.0723	*0.1483*	*0.1131*
4.5	0.0463	0.0383	0.1140	0.0725	0.1603	0.1108
5.0	0.0756	0.0146	0.1248	0.0693	0.2004	0.0838
5.5	0.1067	0.0099	0.1330	0.0660	0.2397	0.0759
6.0	0.0895	0.0045	0.1444	0.0618	0.2338	0.0662
6.5	0.0806	0.0036	0.1519	0.0600	0.2325	0.0637
7.0	0.1190	0.0056	0.1613	0.0569	0.2803	0.0625

nc : klaszterek száma

2. táblázat. A részindexek és a teljes index értékei a távolság függvényében 2 és 3 klaszter képzése esetén. *Forrás:* saját számítás.

A 2. táblázat alapján vizsgálhatjuk a két részindexet is, melyek összegeként áll elő az előbb vizsgált index. A *Scat* részindex méri a klasztereken belüli szórás értékét. Látható, hogy a két klaszteres számításnál növekszik az értéke, ha növeljük a C_1 és a C_2 klaszterek távolságát (ezt a két klasztert ugyanis egynek tekintjük ekkor). A három klaszteres változat esetében ez a részindex egyre csökken. Magyarázata: míg a három klaszter szórása különkülön változatlan, addig az összes megfigyelési egység által alkotott „nagy" klaszter szórása növekszik. A 6. egyenlet értelmében a hányadosuk csökken.

Ugyancsak a 2. táblázat alapján vizsgáljuk meg a másik, a $Dens_{bw}^{**}$ részindexet. A három klaszteres változat eredményeit (3. oszlop) figyelve megállapítható a csökkenő tendencia. Oka: a két távolodó klaszter között egyre kevesebb megfigyelési egység található, ezért a részindex számlálója (ld. 1. egyenlet) csökken, míg nevezője változatlan marad. A két klaszteres változat (2. oszlop) esetében, mivel C_1 és a C_2 klaszter alkot egy klasztert, a két klaszter távolodásakor a részindex nevezője csökken, vagyis a tört értéke növekszik.

A két részindex értéke 3 klaszter figyelembevételével csökken (tehát összegük is csökken), 2 klaszter esetében pedig növekszik (tehát összegük is). Ezen hatások eredményeként egy bizonyos távolságban a két index (utolsó két oszlop) nagyságának viszonya megfordul. Innentől a három klaszteres megoldást választjuk a két klaszteres megoldás helyett.

A szimulációt többféleképpen is elvégeztem. Először a klaszterek minden számítás (d érték) esetén ugyanazok voltak, és csak az egyik klaszter (C_2) elemeinek első változóját növeltem a megadott d értékkel („A" változat). A második esetben minden egyes távolság esetén új klasztereket állítottam elő a megfelelő pataméterek alapján („B" változat). Mindkét esetben különböző szórás-beállítások mellett is elvégeztem a szimulációt (σ_{1x}-et és σ_{2x}-et változtattam, a többi értékét konstansnak vettem), amint a *3. táblázatban* látható.

Kísérlet	σ_{1x}	σ_{2x}	Szimulációk száma az adott távolságeredményekkel														
			3.5	4	4.5	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10	10.5	11
A	1	1	2	8														
A	1	2			4	6												
A	1	3				2	5	2	1									
A	2	2						1	2	6	1							
A	2	3								2	2	3	3					
A	3	3												1	2	3	3	1
B	1	1	3	7														
B	1	2			2	7	1											
B	1	3					1	7	2									
B	2	2						1	3	4	2							
B	2	3									3	6	1					
B	3	3											1	2	3	3	1	

σ : szórás

3. táblázat. A szimulációk száma a három klaszter felismeréséhez szükséges középpontok közötti távolság legkisebb értéke szerint, különböző szórású klaszterek esetén. *Forrás:* saját számítás.

A szórások növekedése miatt a klaszterközéppontok távolságának is nagyobb tartományt kellett megadni, ez 0–11 egységig terjedt. A két index értékei ismét a fent leírtak szerint változtak (a két klaszteres változat esetében növekedett, a három klaszteres változat esetében csökkent az index értéke d növekedése esetén), természetesen a szórások értékének változása miatt másmás távolság esetén következett be a váltás.

Minden egyes paraméterbeállítás mellett 10-10 futtatást végeztem, és vizsgáltam egyrészt az index növekedését ill. csökkenését a távolság függvényében, másrészt azt a távolságot kerestem, ahol a kétklaszteres eredmény helyett a háromklaszteres eredmény kerül elfogadásra. A 3. táblázat adatai azt mutatják, hogy 10 kísérlet esetén melyik távolság esetén ismerte föl az index a három klaszter jelenlétét.

A táblázat adataiból megállapítható, hogy a három klaszter felismerésének nem feltétele, hogy a klaszterek teljesen szeparáltak legyenek. Az is látható azonban, hogy a szórások növekedése esetén a bizonytalanság is egyre növekszik, tehát a felismerési távolság szórása is nagyobb.[9]

A vizsgálatban használt C_3 klaszter szerepe annyi volt, hogy a C_1 és C_2 összevonása esetén is legyen két klaszterünk, amelyre az index számolható. Ezért ezt a C_1-től és C_2-től szeparáltan helyeztem el, a cél ugyanis a C_1 és C_2 közötti átfedés vizsgálata volt.

5 Az S_Dbw_{new} és S_Dbw^{**} index összehasonlítása

Az összehasonlításhoz szintetikusan előállított adatbázisokat használok, melyeken klaszterező eljárásokat futtatok le különböző paraméterbeállítások mellett, és a kapott klasztereken tesztelem a két indexet. Ezt az eljárást követték mindhárom cikkben, amelyek ennek az indexnek kidolgozásával foglalkoztak. Halkidi és Vazirgiannis [8] valamint Tong és Tan [17] elemzésében, többek

[9] A vizsgálatok során a klaszterek elemszáma nem változott.

között, az ún. DBSCAN [1] algoritmust alkalmazták. Ez a módszer a sűrűségek vizsgálatán alapszik, és nagyon hatékony nem konvex, de jól szeparált klaszterek elkülönítésére. Ezen vizsgálat fókuszában azonban a konvex és nem feltétlenül teljesen elkülönülő csoportok felismerése áll, ezért helyette az ún. MCLUST [3, 4] algoritmust választottam. Ez egy modell alapú klaszterezési eljárás, amelynek lényege, hogy az adott adatbázis alapján meghatározza annak az eloszlásnak a paramétereit, amelyből legnagyobb valószínűséggel keletkezhetett egy ilyen adatbázis, és ezen elméleti modell alapján, valószínűségeket számolva, sorolja be a megfigyelési egységeket klaszterekbe. Mindhárom cikkben alkalmazzák a K-means agglomeratív klaszterezési eljárást. Ennek ismertetésére nem térek ki, hiszen az egyik legszélesebb körben alkalmazott algoritmus. Ez lesz a másik klaszterező eljárás, amit alkalmazni fogok.

7 db adatbázison tesztelem az indexeket, melyeket mintavétellel állítok elő adott paraméterű normál eloszlásokból. A minták konvex csoportokat tartalmaznak, melynek fölismerésében mindkét algoritmus jó eredményeket ért el. Ezen adatbázisok előállításának szempontjai a következők voltak:

- legyen kisebb és nagyobb elemszámú klasztereket is tartalmazó adatbázis,
- legyen sűrűbb és ritkább klasztereket is tartalmazó adatbázis,
- legyen jól szeparált és kevésbé jól szeparált klasztereket is tartalmazó adatbázis.

Az adatbázisok két változót tartalmaztak, hogy az eredmények kiértékelésekor legyen lehetőség annak szemléltetésére is, így lehetőséget teremtve annak jobb megértésére. Természetesen a továbbiakban semmi akadálya annak, hogy többváltozós adatbázisok esetén is teszteljük/alkalmazzuk az indexet, azonban ekkor a szemléltetés nehezebb, vagy nem megoldható.

A *4. táblázat* mutatja a létrehozott adatbázisok paramétereit (klaszterek középpontja, szórása, elemszáma). Az első négy esetben 4 klasztert állítottam elő, és az első esetben olyan távol helyeztem el őket, hogy teljesen szeparáltak legyenek. A további 3 esetben közelebb helyeztem őket ill. változtattam az elemszámukat (egyrészt úgy, hogy egyszerre mindegyik kevesebb elemet tartalmazzon, másrészt úgy, hogy különböző legyen az elemszámuk). Az 5. adatbázis egy háromklaszteres elrendezés, melyben K1 és K2 között átfedés van, míg K3 egy távolabb levő klaszter, sűrűségük pedig különböző. A 6. adatbázis tartalmaz ellipszis alakú klasztereket is, ezenfelül a K1 kivételével a többi átfedéseket is tartalmaz, azaz ezen klaszterek közötti elemszám nagyobb, mint az első négy klaszter esetében. Az utolsó adatbázis esetében a K3 elkülönül a többitől, a többi három pedig jobban átfedi egymást, mint az eddigi példákban generált klaszterek esetében.

Természetesen a szimulációval nem lehet minden lehetséges helyzetet ellenőrizni. Itt a cél annak vizsgálata volt, hogy az egymáshoz közelebb levő klaszterek esetében kimutatható különbség van-e a két index eredményei között.

Mindkét módszer (K-means, Mclust) esetében a klaszterek számát 2-től 7-ig változtattam. Ezután összehasonlítottam a besorolásokat a tényleges klaszterbesorolásokkal, és a legtöbb egyezést mutató eredményt választottam legjobbnak. Ezután azt vizsgáltam, hogy a két index melyik besorolást

fogja legjobbként értékelni. Mindegyik esetben 10-10 mintát generáltam (a megadott paraméterek mellett, ld. 4. táblázat), és a kapott eredményeket rendeztem az *5. táblázatba.*

	K1			K2			K3			K4		
	$\mathbf{v}_1$	σ_1	N_1	$\mathbf{v}_2$	σ_2	N_2	$\mathbf{v}_3$	σ_3	N_3	$\mathbf{v}_4$	σ_4	N_4
1	(0,0)	(1,1)	500	(7,0)	(1,1)	500	(0,-7)	(1,1)	500	(2,7)	(1,1)	500
2	(0,0)	(1,1)	500	(4,0)	(1,1)	500	(0,-7)	(1,1)	500	(2,5)	(1,1)	500
3	(0,0)	(1,1)	100	(4,0)	(1,1)	100	(0,-7)	(1,1)	100	(2,5)	(1,1)	100
4	(0,0)	(1,1)	500	(4,0)	(1,1)	100	(0,-7)	(1,1)	500	(2,5)	(1,1)	250
5	(2,2)	(1,1)	750	(6,0)	(2,2)	500	(2,-7)	(0.5,0.5)	500			
6	(-4,0)	(1,1)	500	(4,0)	(2,2)	1000	(0,-7)	(3,2)	500	(2,5)	(2,1)	500
7	(0,0)	(1,1)	500	(4,0)	(1,1)	500	(0,-7)	(1,1)	500	(2,2)	(1,1)	500

K1, K2, K3, K4: klaszterazonosító
$\mathbf{v}_i$: az i-edik klaszter középpontja
σ_i: az i-edik klaszter elemeinek x és y irányú szórása
N_i: az i-edik klaszter elemszáma

4. táblázat. Az indexek összahasonlításához használt adatbázisok paraméterei. *Forrás:* saját összeállítás.

	Szimuláció	Klaszterek száma					
	sorszáma	KM	T-KM	S-KM	MC	T-MC	S-MC
	1	4	4	4	4	5	4
	2	4	4	4	4	4	4
	3	4	5	4	4	5	4
	4	3	6	5	4	4	4
1. adatbázis	5	4	4	4	4	6	4
	6	4	7	4	4	5	4
	7	4	4	4	4	5	4
	8	4	4	4	4	4	4
	9	4	4	4	4	5	4
	10	4	7	4	4	7	4
	1	4	4	4	4	6	4
	2	4	4	4	4	5	4
	3	4	7	4	4	7	4
	4	4	6	4	4	4	4
2. adatbázis	5	4	5	4	4	7	4
	6	4	5	4	4	5	4
	7	4	6	4	4	4	4
	8	4	6	4	4	6	4
	9	4	7	4	4	5	4
	10	4	6	4	4	6	7
	1	4	6	4	4	4	4
	2	3	6	2	4	7	4
	3	4	4	4	4	4	4
	4	5	5	5	4	7	6
3. adatbázis	5	4	6	4	4	5	4
	6	4	5	5	4	7	7
	7	4	4	4	4	4	4
	8	4	7	4	4	7	4
	9	4	4	4	4	7	3
	10	4	5	4	4	5	6

(a táblázat folytatódik)

5. táblázat. Az indexek összehasonlításának eredményei. *Forrás:* saját számítás.

	Szimuláció sorszáma	Klaszterek száma					
		KM	T-KM	S-KM	MC	T-MC	S-MC
	1	3	4	3	3	6	3
	2	4	5	4	4	5	4
	3	4	4	3	4	5	4
	4	4	5	4	4	4	4
4. adatbázis	5	3	6	3	3	6	6
	6	4	5	4	4	4	4
	7	3	7	3	3	7	6
	8	4	4	4	3	3	3
	9	4	5	4	4	6	4
	10	4	7	6	4	3	4
	1	3	4	3	4	5	2
	2	3	6	3	3	5	2
	3	3	3	3	4	2	2
	4	3	7	2	3	2	2
5. adatbázis	5	3	3	3	2	3	2
	6	3	5	3	3	2	2
	7	3	5	3	4	2	2
	8	3	4	3	4	4	2
	9	3	3	3	4	2	2
	10	3	5	3	3	5	2
	1	4	5	6	4	7	4
	2	4	7	6	4	7	5
	3	4	6	6	4	7	4
	4	4	7	7	4	7	7
6. adatbázis	5	4	7	7	4	6	7
	6	4	4	4	4	7	4
	7	4	6	7	4	6	7
	8	4	7	7	4	7	7
	9	4	5	5	4	6	7
	10	4	6	6	4	5	4
	1	4	6	2	4	6	2
	2	5	5	2	4	4	2
	3	3	5	2	4	5	2
	4	4	7	2	4	7	2
7. adatbázis	5	4	4	2	4	3	3
	6	4	6	2	4	6	2
	7	3	6	2	4	5	2
	8	4	6	2	4	7	2
	9	5	7	2	4	7	2
	10	4	4	2	4	3	2

KM, MC: Legjobb csoportosítás klaszterszáma (K-means, Mclust)
T-KM, T-MC: Tong index eredménye (K-means, Mclust)
S-KM, S-MC: saját index eredménye (K-means, Mclust)

5. táblázat. Az indexek összehasonlításának eredményei (folytatás).

A kapott eredményeket olyan szempont szerint értékeltem, hogy az egyes indexek eltalálták-e az adott algoritmus által előállított megoldások közül a ténylegeshez legközelebb álló megoldást. Az 1. adatbázis tartalmazott jól szeparált klaszthereket, mindkét index ebben jó eredményt ért el.

A 2., 3. és 4. adatbázisok esetében az 1. adatbázis klaszterei közelebb kerültek egymáshoz, ill. az elemszámaik is változtak. Ezekben az esetekben megfigyelhető, hogy a lecsökkentett elemszám (3. adatbázis), valamint az egyenlőtlen elemszám esetén (4. adatbázis) a saját index teljesítménye is romlott. A Tong index viszont ezen klaszterelrendezések esetén már nem

tudott elfogadható eredményt adni. Az általam módosított index a legjobb csoportosításnak megfelelő klaszterszámokat többször találta el, mint a Tong index. A találatok különbsége jelentős.

Az 5. adatbázis esetében lényeges különbség van az egyes klaszterek sűrűsége között, továbbá a K3 klaszter elkülönül a másik kettőtől. Az eredmények tanulmányozásából az derül ki, hogy a K-means algoritmus esetében a háromklaszteres elrendezés bizonyult a legjobbnak mind a tíz szimuláció esetén, míg az Mclust algoritmus mindössze 4 esetben adott az eredetihez hasonló megoldást. Az indexeket vizsgálva, a K-means által előállított klaszterek esetében a saját index jobb eredményt ért el (a tíz szimuláció összesítéseként), mint a Tong féle. Ugyanakkor az Mclust által előállított klasztereken végzett szimulációk esetében a saját index mindig a kétklaszteres megoldást részesítette előnyben, és csak egyszer találta el a legjobb csoportosítást. Megfigyelhető még, hogy ezen adatbázis esetén az Mclust algoritmus által előállított klaszterek száma változékony volt, 2, 3 és 4 klaszteres megoldás is előállt.

A 6. adatbázis előállításakor a szórások változtatásával olyan klasztereket is képeztem, amelyek nem kör alakúak. Továbbá elemszámban és sűrűségben is van közöttük különbség. A négy klaszter nem teljesen szeparált egymástól. Mind a K-means, mind pedig az Mclust legjobb besorolása a négyklaszteres megoldás volt (az eredeti adatbázis is ennyi klasztert tartalmazott). Ennek ellenére mindkét index lényegében rossz besorolást határozott meg. A megoldások véletlenszerűnek tűnnek. Vagyis a módosított index alkalmazhatósága ezen adatbázis esetében már szintén megkérdőjelezhető.

A 7. adatbázis esetében a négy klaszterből három átfedi egymást, a negyedik különálló (K3). Ezen klaszterek felismerésében az Mclust algoritmus egyenletes teljesítményt nyújtott K-means-szel szemben. Az Tong-féle index ebben az esetben is sokféle eredményt adott, szinte véletlenszerűen, míg a saját index lényegében a kétklaszteres megoldást mutatta legjobbnak. Ennek az oka, hogy a három egymás mellett levő klaszter olyan közel került egymáshoz, hogy megkülönböztetésük a módosított index használatával nem lehetséges. Ugyanakkor a Tong-féle index eredményeinél jobban hasznosítható eredményt adott.

6 Összefoglalás

A vizsgálat alá vont index kritikai elemzése után az index módosítására került sor. A szimulációs kísérletek alátámasztották a 3. fejezetben összefoglalt kritikai megjegyzéseket, végeredményben, hogy a Tong-féle index csak jól szeparált klaszter-elrendezések esetén nyújt megfelelő segítséget a klaszterszám megválasztásához. Az 5. fejezetben végrehajtott szimulációk megmutatták az index továbbfejlesztett változatának alkalmazhatóságát olyan esetekben is, amikor a Tong-féle index eredményeiből használható információ nem származik. Mivel sűrűségkülönbségek alapján számoljuk az index értékét, ezért olyan klaszterelrendezések esetén, amikor a klaszterek annyira közel vannak, hogy a klaszterek között már nincs ritkább terület (ld. 6. adatbázis), az index

már nem alkalmazható, vagy egyáltalán nem tud bizonyos klasztereket megkülönböztetni, és egy másik, ugyanakkor jó megoldást eredményez (7. adatbázis).

A két index vizsgálata azt mutatja, hogy a módosítás eredményeként előállt index alkalmazhatósága szélesebb körű, ugyanakkor a korlátait is megmutatta a szimulációs kísérlet. Egy adott adatbázis esetén, ahol a csoportok meglétét, számát keressük, az adatbázisból vett mintákon való tesztelések eredményének egyezősége vagy változékonysága mutatja az index alkalmazhatóságát.

Irodalom

1. Ester M., Kriegel H. P., Sander J., Xu. X. (1996): A density based algorithm for discovering clusters in large spatial databases with noise, in: *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining*, Portland, OR, pp. 226–231.
2. Everitt B. S., Landau S., Leese M., Stahl D. (2011): *Cluster Analysis*, Wiley, 5th ed. 346 p.
3. Fraley C., Raftery A. E. (2002): Model-Based Clustering, Discriminant Analysis and Density Estimation, *Journal of the American Statistical Association*, vol. 97, pp. 611–631.
4. Fraley C., Raftery A. E. (2006): MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering, Technical Report No. 504. Department of Statistics. University of Washington, pp. 1–56.
5. Füstös L., Kovács E. (1989): *A számítógépes adatelemzés statisztikai módszerei*, Tankönyvkiadó, Budapest, 380 p.
6. Füstös L., Kovács E., Meszéna G., Simonné N. M. (2004): *Alakfelismerés*, Új Mandátum Könyvkiadó, 644 p.
7. Hajdu O. (2003): *Többváltozós statisztikai számítások*, Központi Statisztikai Hivatal, 457 p.
8. Halkidi M., Vazirgiannis M. (2001): Clustering validity assessment: finding the optimal partitioning of a data set, in: ICDM 2001, *Proceedings IEEE International Conference on Data Mining*, IEEE, pp. 187–194.
9. Kaufman L., Rousseeuw P. J. (2005): *Finding groups in data: an introduction to cluster analysis*, Wiley, Hoboken, N.J.
10. Kim Y., Lee S. (2003): A Clustering Validity Assessment Index, in: K. Y. Whang, J. Jeon, K. Shim, J. Srivastava (eds.) *Advances in Knowledge Discovery and Data Mining*, vol. 2637 of Lecture Notes in Computer Science, Springer Berlin, Heidelberg, pp. 562–562.
11. Kovács E., Füstös L., Meszéna G. (2007): *Alakfelismerés: Sokváltozós statisztikai módszerek*, Új Mandátum Könyvkiadó, 660 p.
12. Legány C., Juhász S., Babos A. (2006): Cluster validity measurement techniques, in: *Proceedings of the 5th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases*, World Scientific and Engineering Academy and Society (WSEAS), Stevens Point, Wisconsin, pp. 388–393.
13. Liu Y., Li Z., Xiong H., Gao X., Wu J. (2010): Understanding of Internal Clustering Validation Measures, in: *Proceedings of the 2010 IEEE International*

Conference on Data Mining, ICDM '10, IEEE Computer Society, Washington, DC, USA, pp. 911– 916.

14. Simon J. (2006): A klaszterelemzés alkalmazási lehetőségei a marketingkutatásban, *Statisztikai Szemle*, vol. 7, pp. 627–650.
15. Sneath P. H. (2005): Numerical Taxonomy, in: D. J. Brenner, N. R. Krieg, J. T. Staley, G. M. Garrity (eds.) *Bergey's Manual of Systematic Bacteriology*, Springer US, Boston, MA, pp. 39–42.
16. Theodoridis S., Koutroumbas K. (2003): *Pattern recognition*, Academic Press, 2nd ed. 689 p.
17. Tong J., Tan H. (2009): Clustering validity based on the improved S-Dbw* index, *Journal of Electronics (China)*, vol. 26, pp. 258–264.

A POSSIBLE SOLUTION OF THE DETERMINATION OF NUMBER OF CLUSTERS

This paper deals with the problem that in the case of cluster analysis we can get many solutions. Which is the best approximation of the – hypothetically existing – groups in the database among these solutions? There are processes trying to answer this question. On one hand this paper is a critical analysis of such a process, and on the other hand tries to develop that. The essence of the method is creating an index which contains the number of elements around the cluster centers and around a dividing point between the cluster centers. By the help of this index the classification accuracy can be characterized. In the case of popular algorithms, such as K-means – where the desired number of clusters has to be given in advance – this (modified) index provides assistance for the decision.