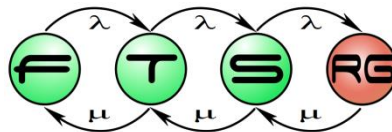


Mintavételezés, szűrés, kilógó esetek detektálása

Salánki Ágnes

salanki@mit.bme.hu

Budapest University of Technology and Economics
Fault Tolerant Systems Research Group



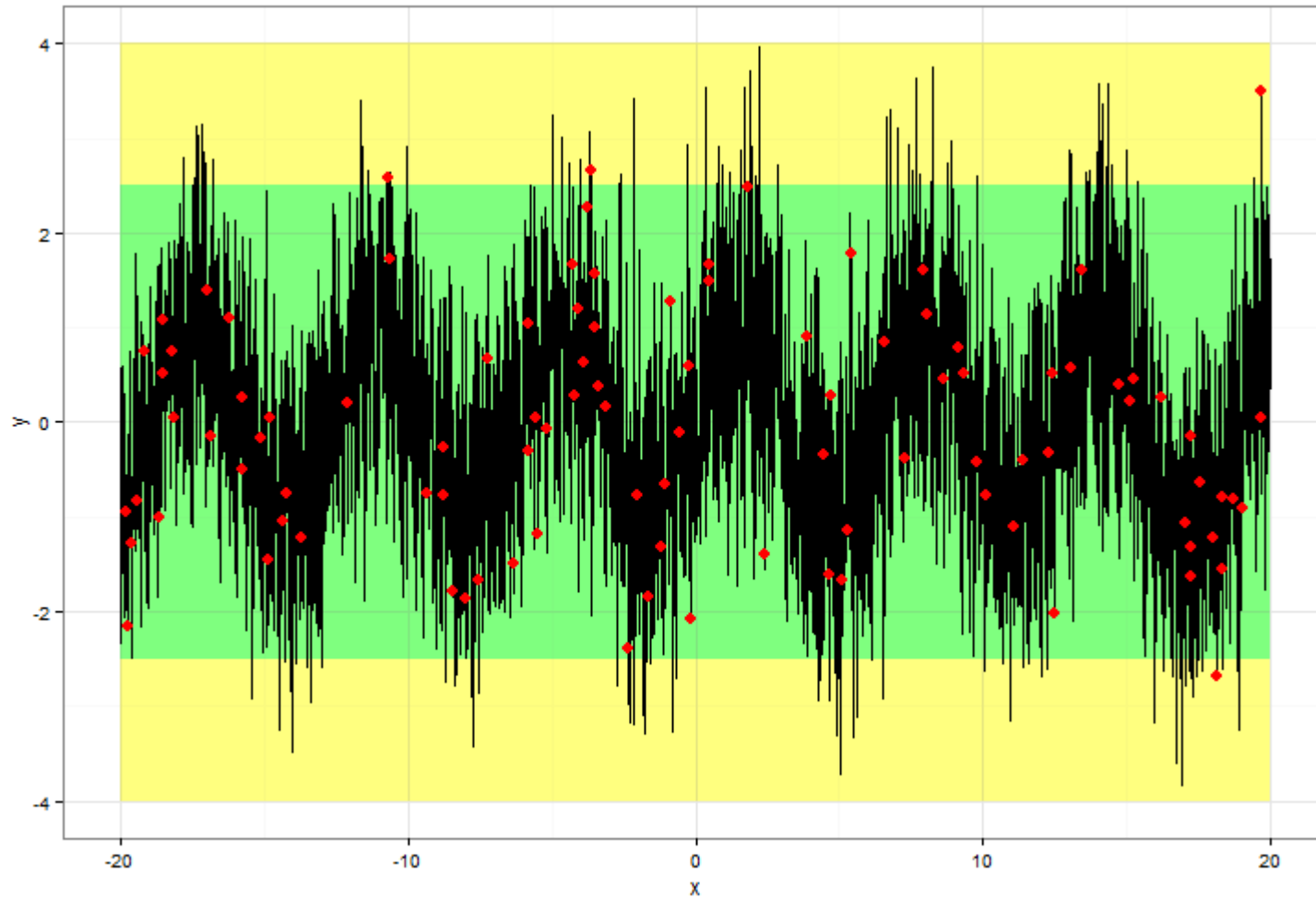
Alapfogalmak

- Az alapfeladat ugyanaz
- Az aspektus más

Alapfogalmak

■ Az alapfeladat ugyanaz

■ Az



OUTLIER DETEKTÁLÁS

Alapfeladat



Alapfeladat

Vannak-e
egyáltalán?



Alapfeladat

Vannak-e
egyáltalán?

Hogy néznek ki?



Alapfeladat

Vannak-e
egyáltalán?

Hogy néznek ki?

Hogyan
szeparálhatóak?

Miért?



Alapfeladat

Vannak-e
egyáltalán?

Vannak-e
egyáltalán?

Hogy néznek ki?

Szakterület
specifikus?

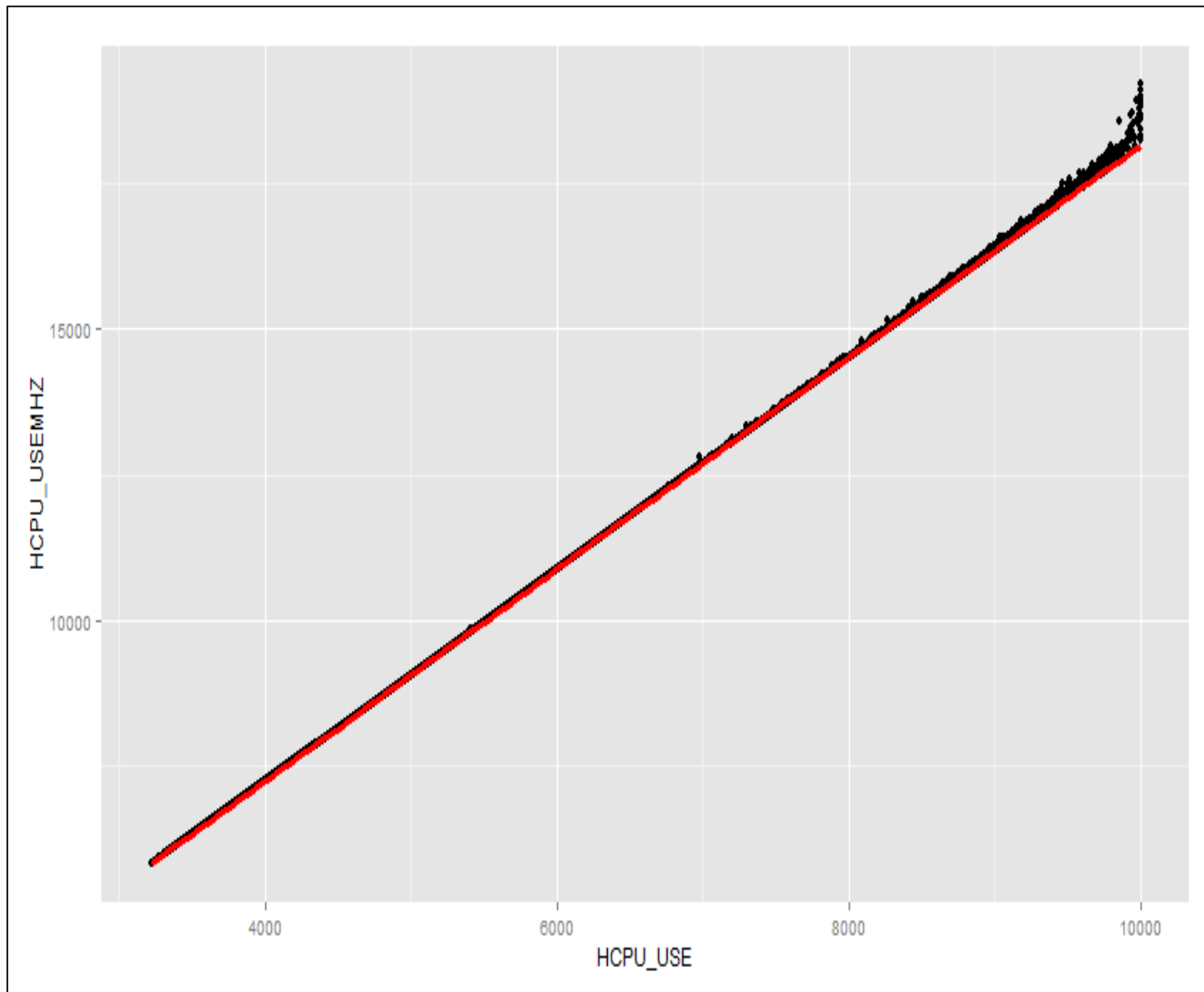
Hogyan
szeperálhatóak?

Nagy adat:
aggregálás?

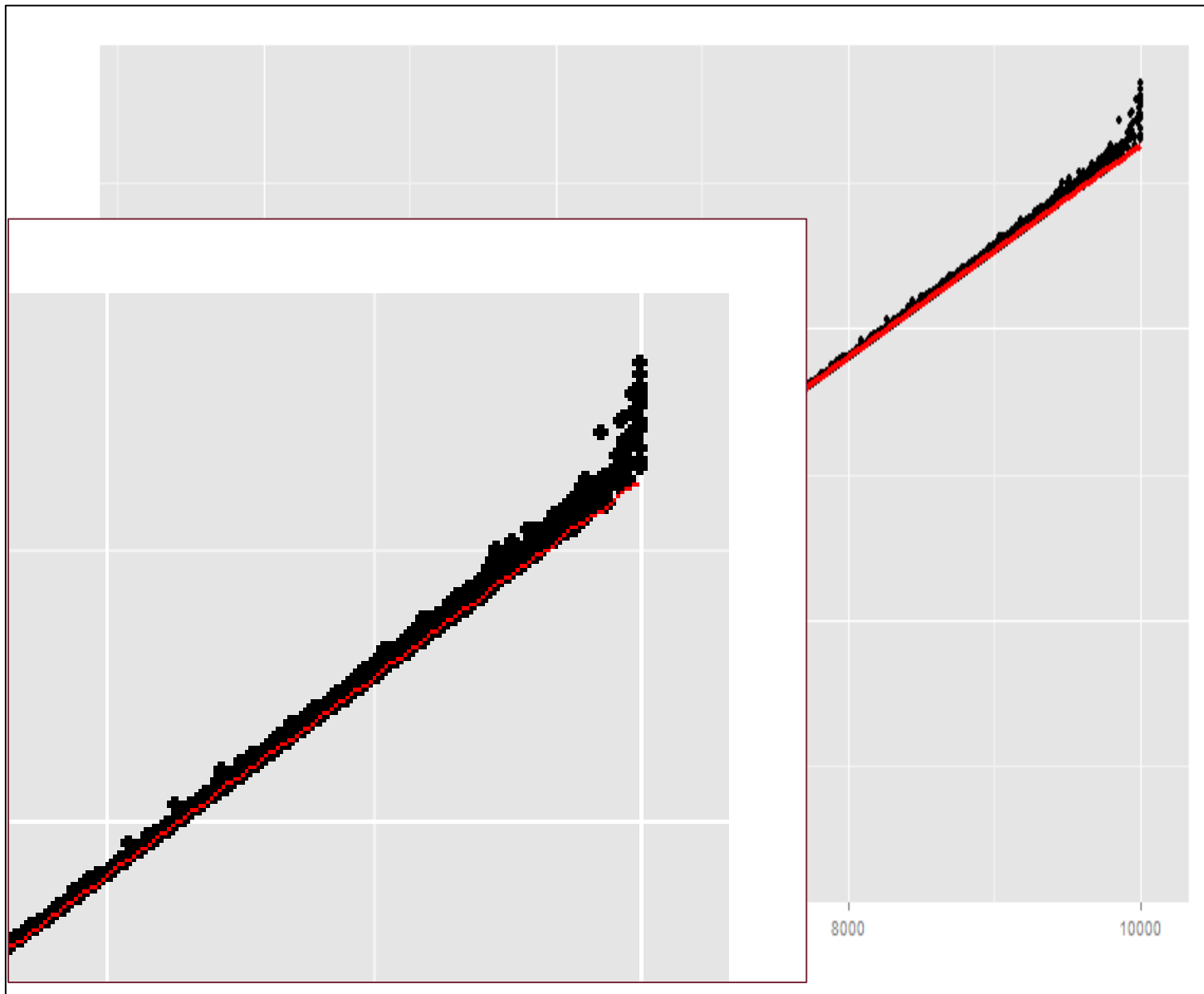
Miért?

Hatások?

Alapfeladat

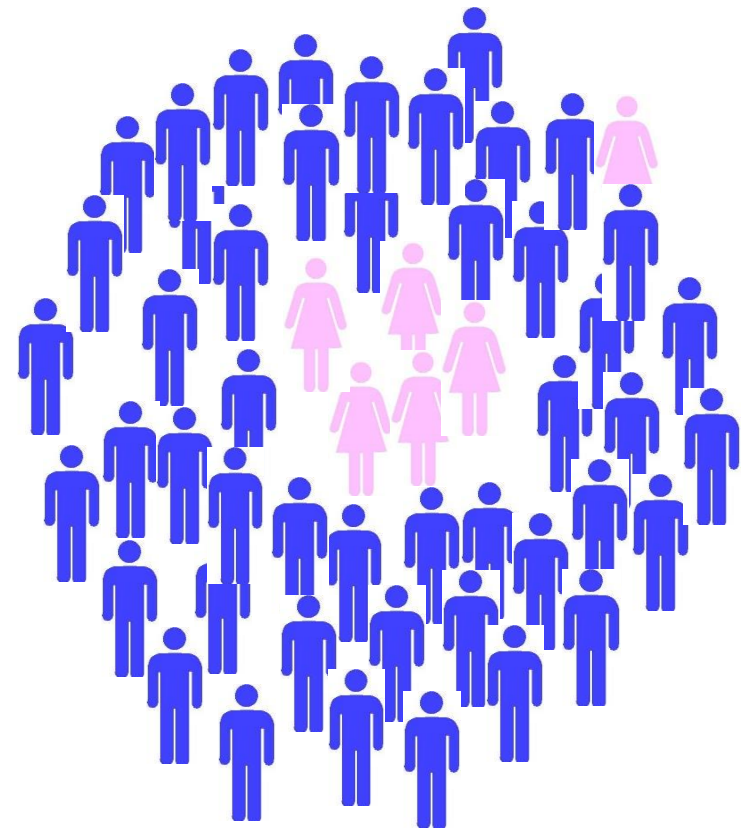


Alapfeladat

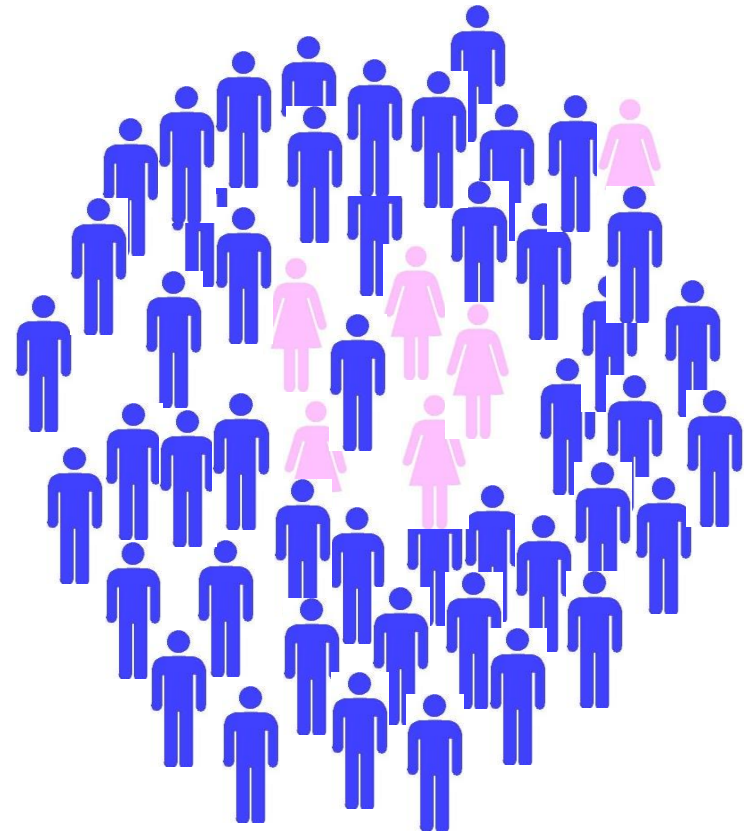


Megközelítések

- Távolság alapúak

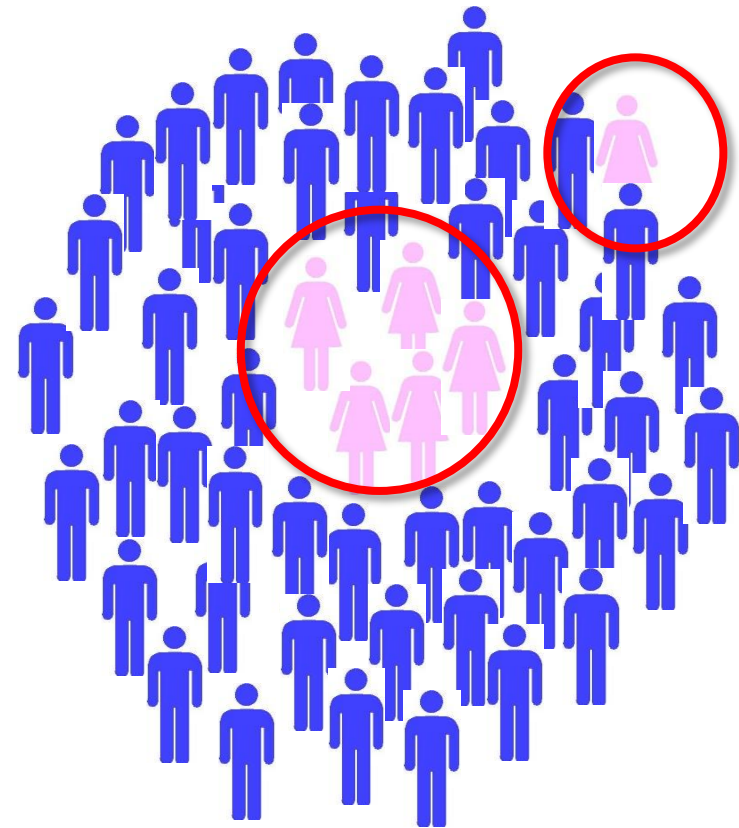


- Sűrűség alapúak

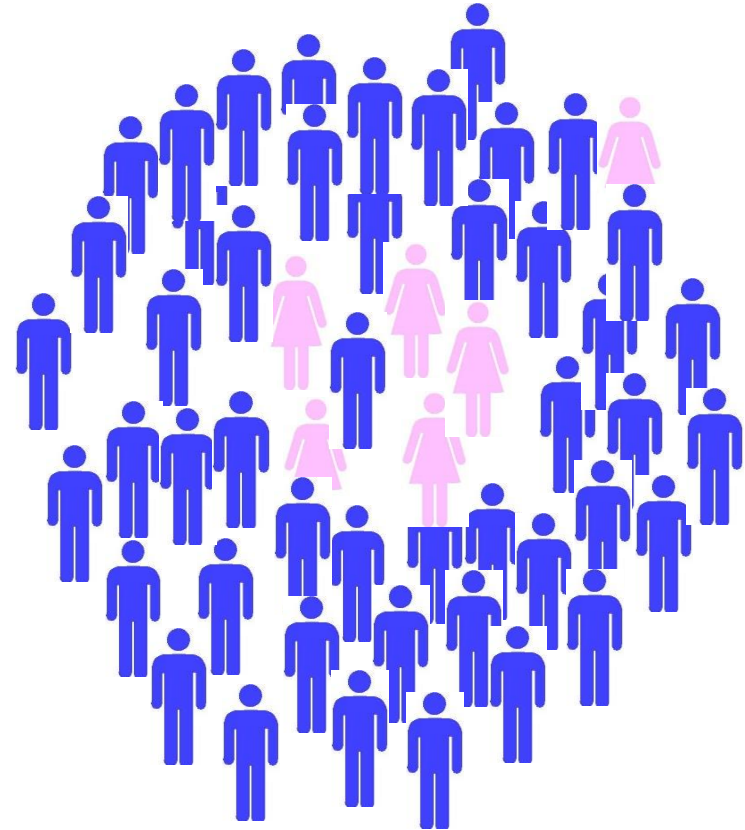


Megközelítések

- Távolság alapúak

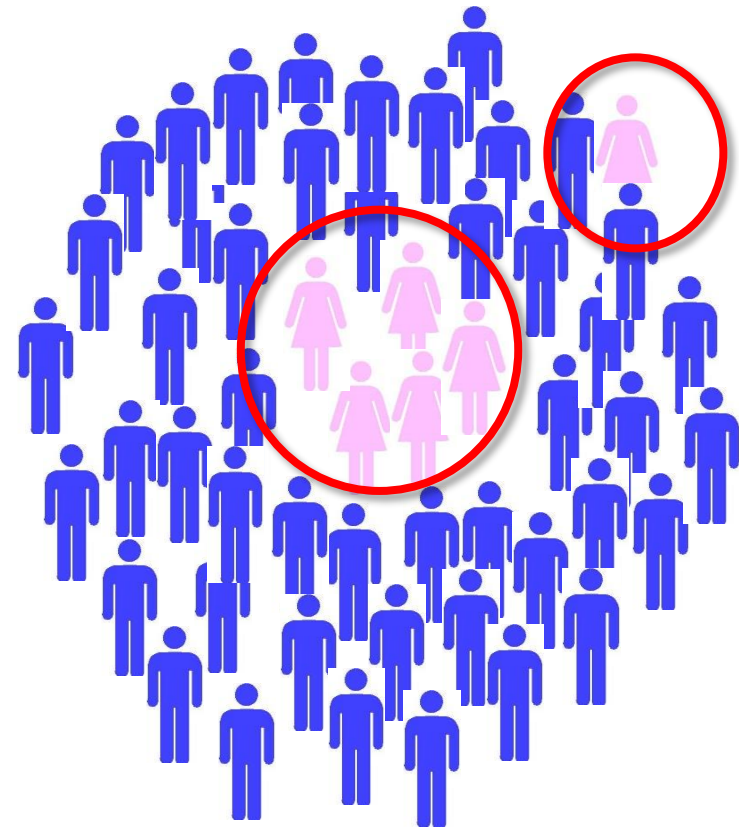


- Sűrűség alapúak

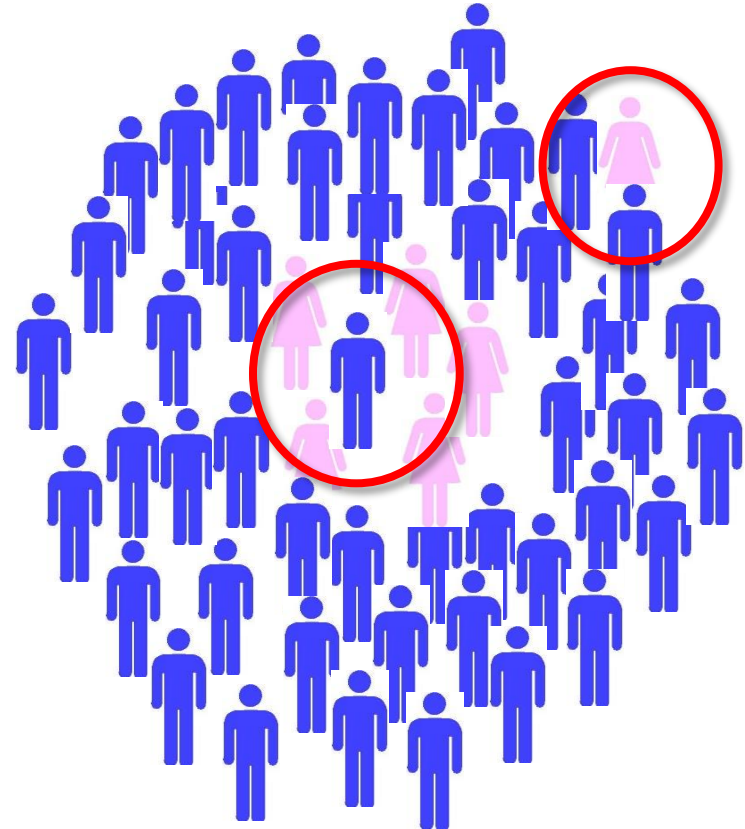


Megközelítések

- Távolság alapúak

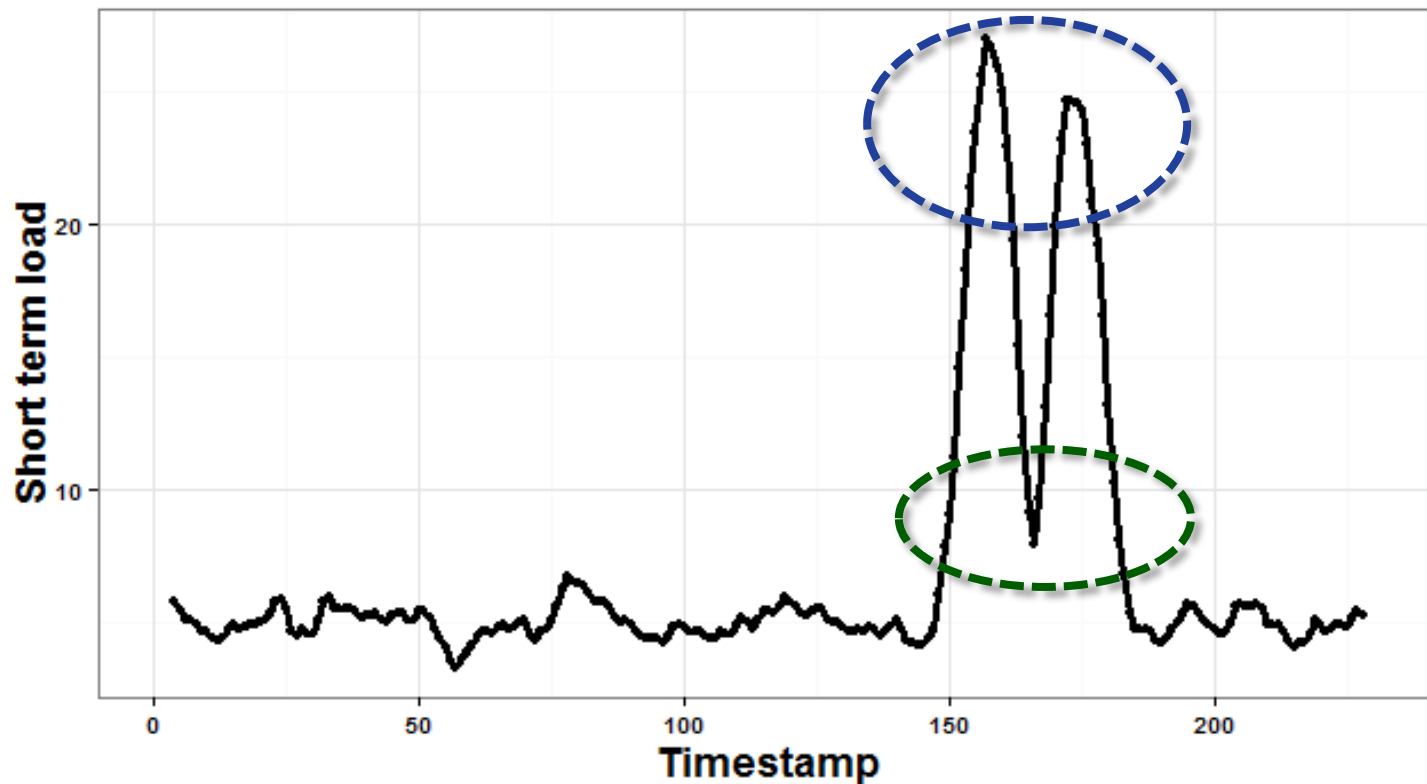


- Sűrűség alapúak



Megközelítések

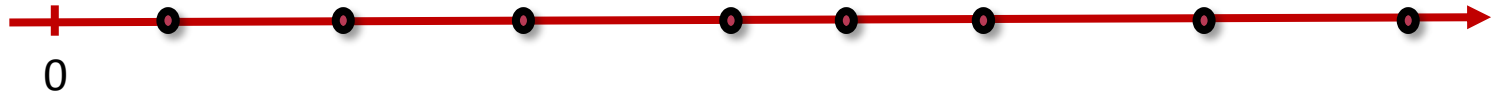
- Távolság alapúak
- Sűrűség alapúak



TÁVOLSÁG ALAPÚ TECHNIKÁK

Befoglaló burok

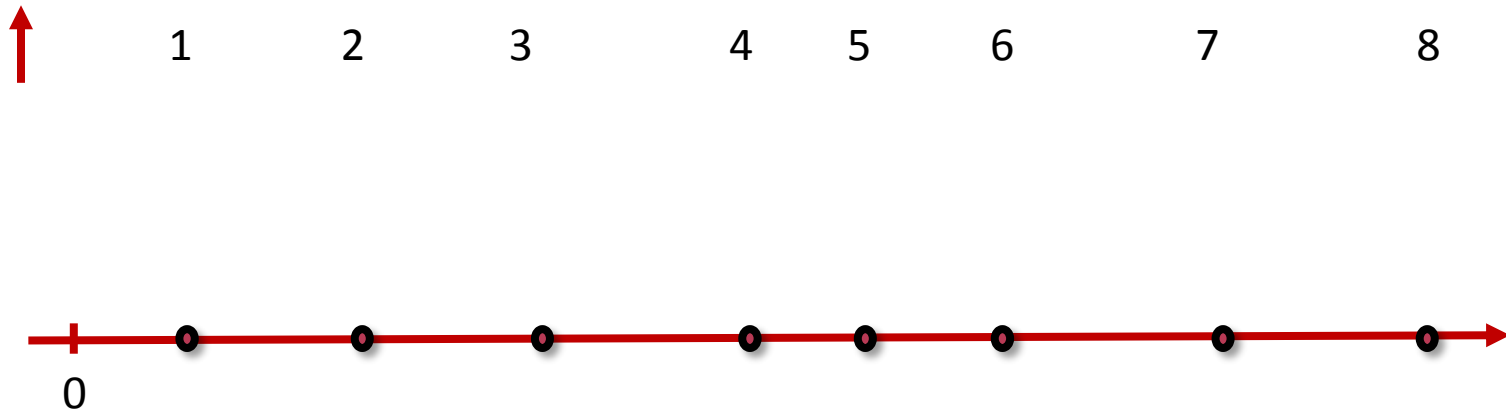
- Féltér-mélység: Tukey, 1974



$$hds(z): \min\{|x_i: x_i \leq z|, |x_j: x_j \geq z|\}$$

Befoglaló burok

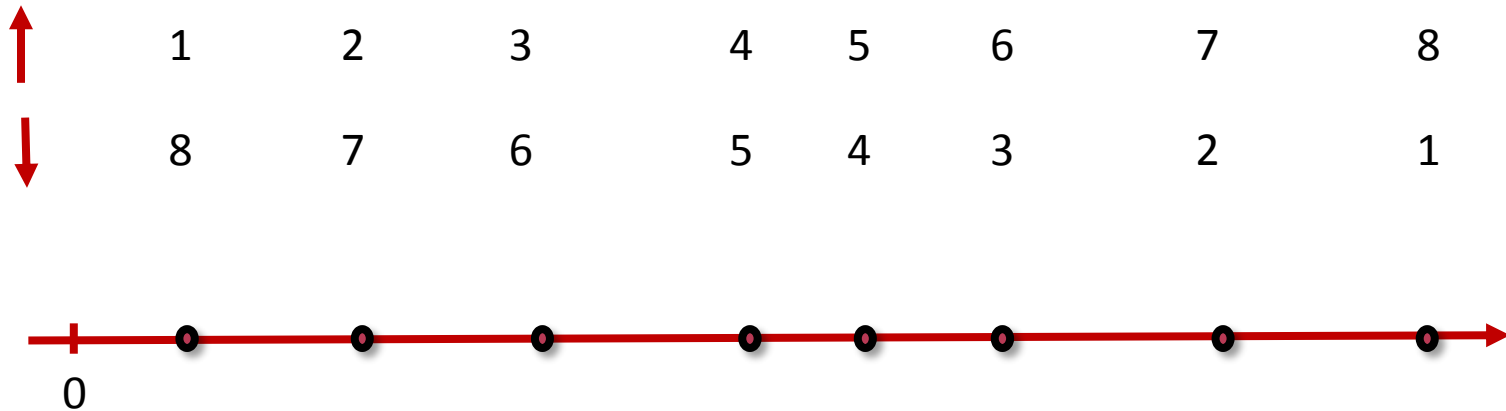
- Féltér-mélység: Tukey, 1974



$$hds(z): \min\{|x_i: x_i \leq z|, |x_j: x_j \geq z|\}$$

Befoglaló burok

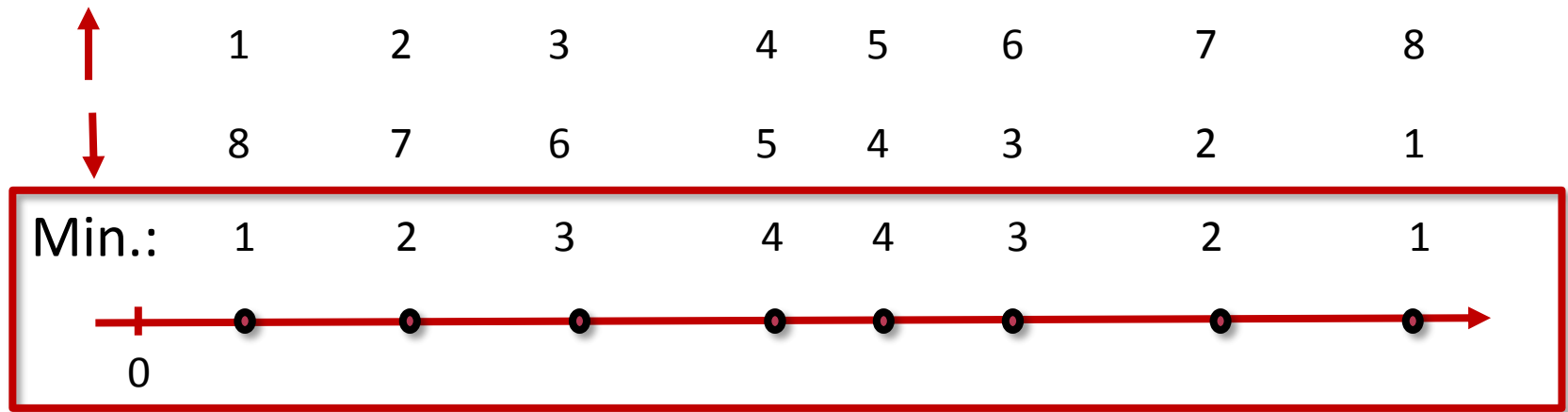
- Féltér-mélység: Tukey, 1974



$$hds(z): \min\{|x_i: x_i \leq z|, |x_j: x_j \geq z|\}$$

Befoglaló burok

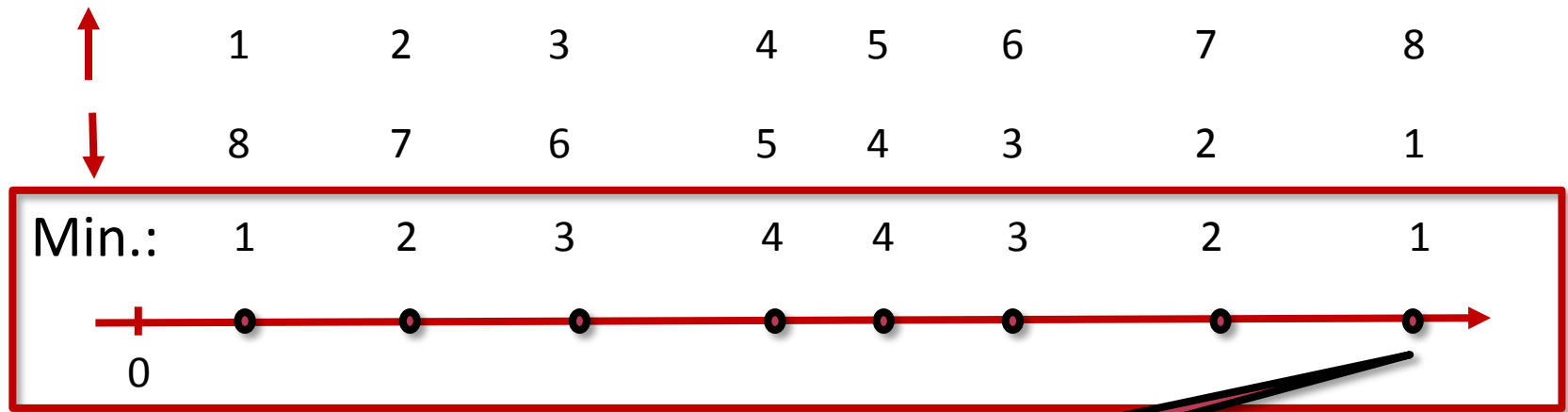
- Féltér-mélység: Tukey, 1974



$$hds(z): \min\{|x_i: x_i \leq z|, |x_j: x_j \geq z|\}$$

Befoglaló burok

- Féltér-mélység: Tukey, 1974



Extrém
pontok

$$): \min\{|x_i: x_i \leq z|, |x_j: x_j \geq z|\}$$

Befoglaló burok

- Féltér-mélység: Tukey, 1974



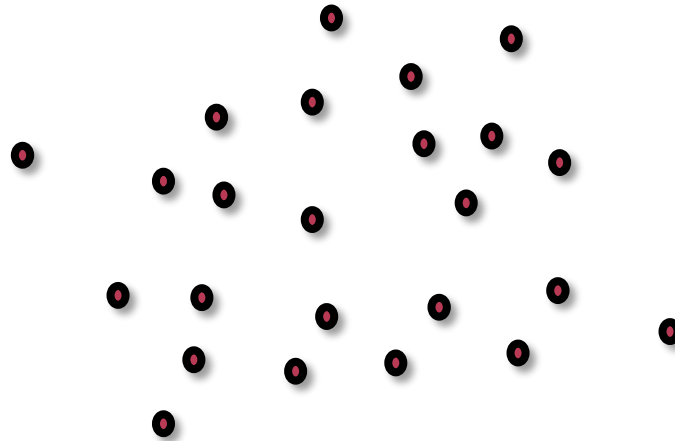
Extrém
pontok

$\min\{|x_i: x_i \leq z|\}$

Medián: majd a végén...

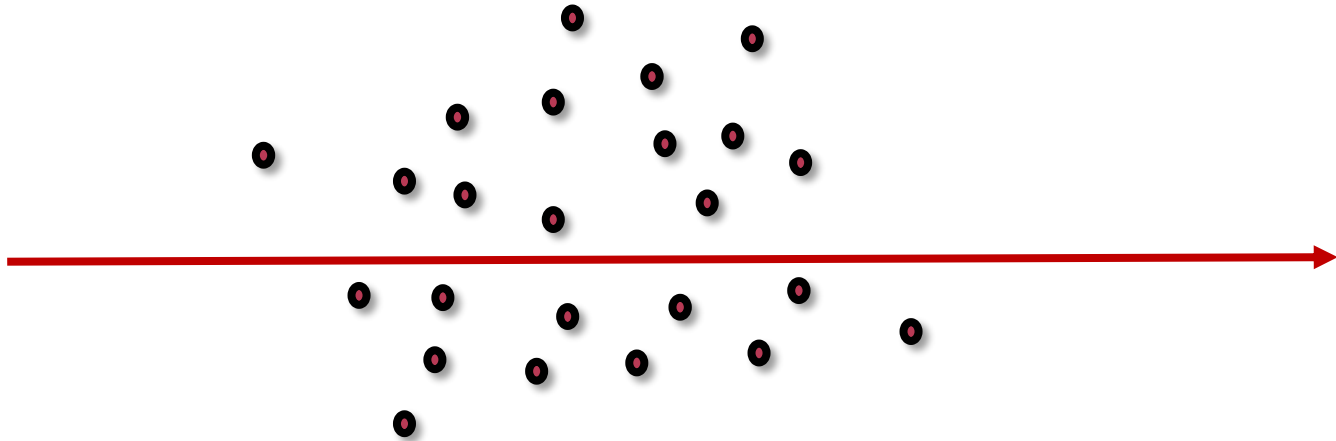
Befoglaló burok

- Féltér-mélység: Tukey, 1974



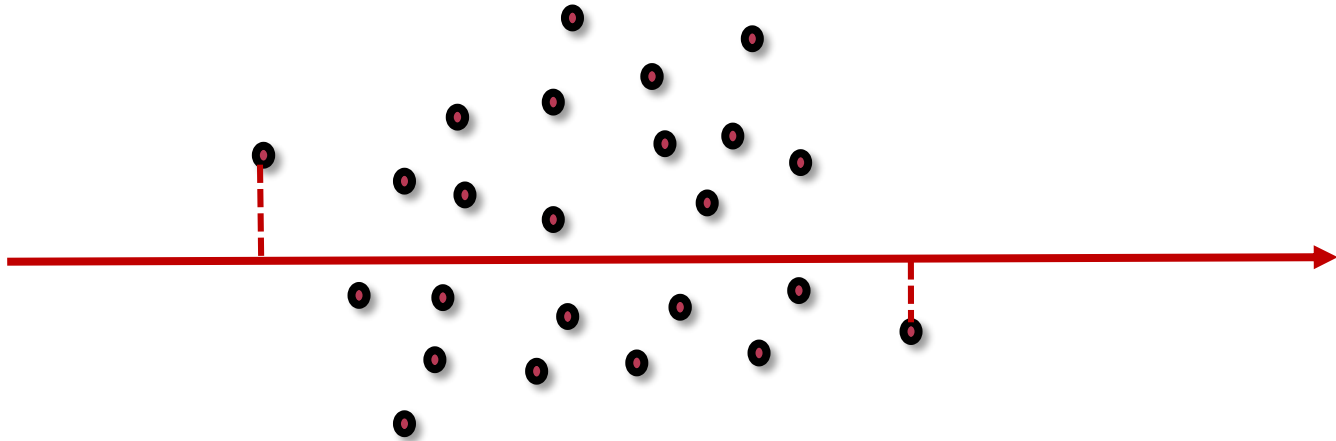
Befoglaló burok

- Féltér-mélység: Tukey, 1974



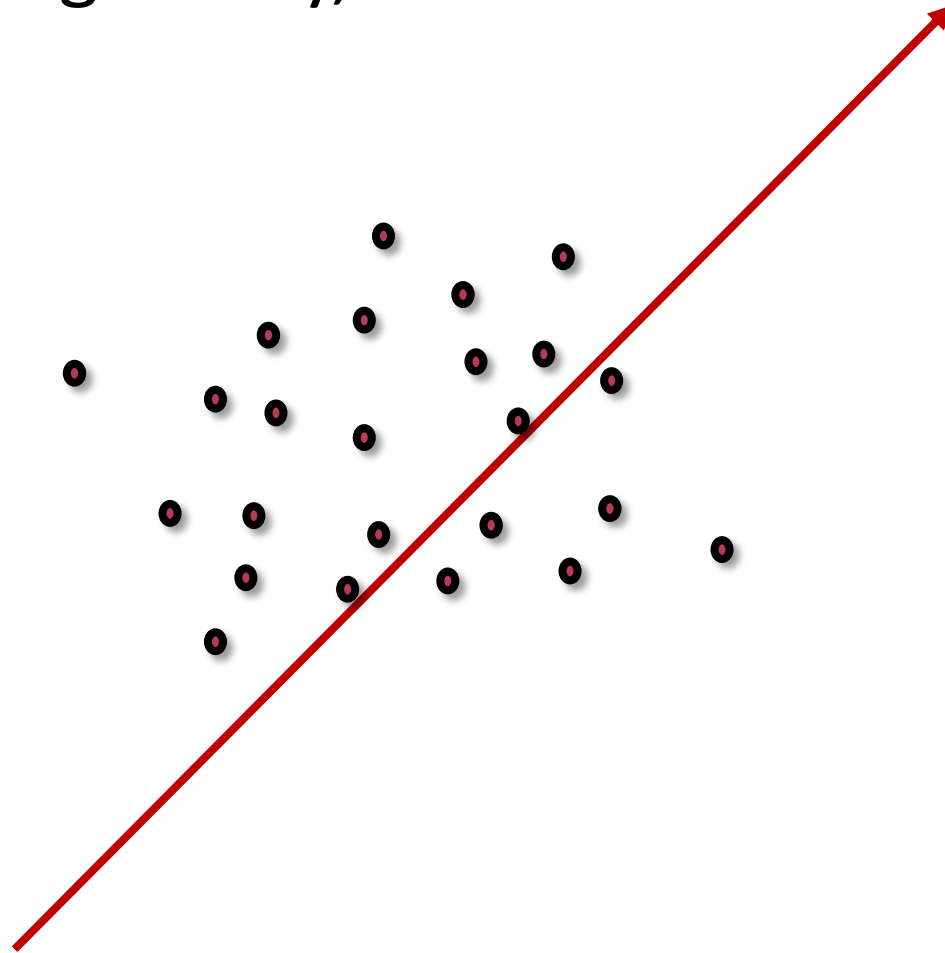
Befoglaló burok

- Féltér-mélység: Tukey, 1974



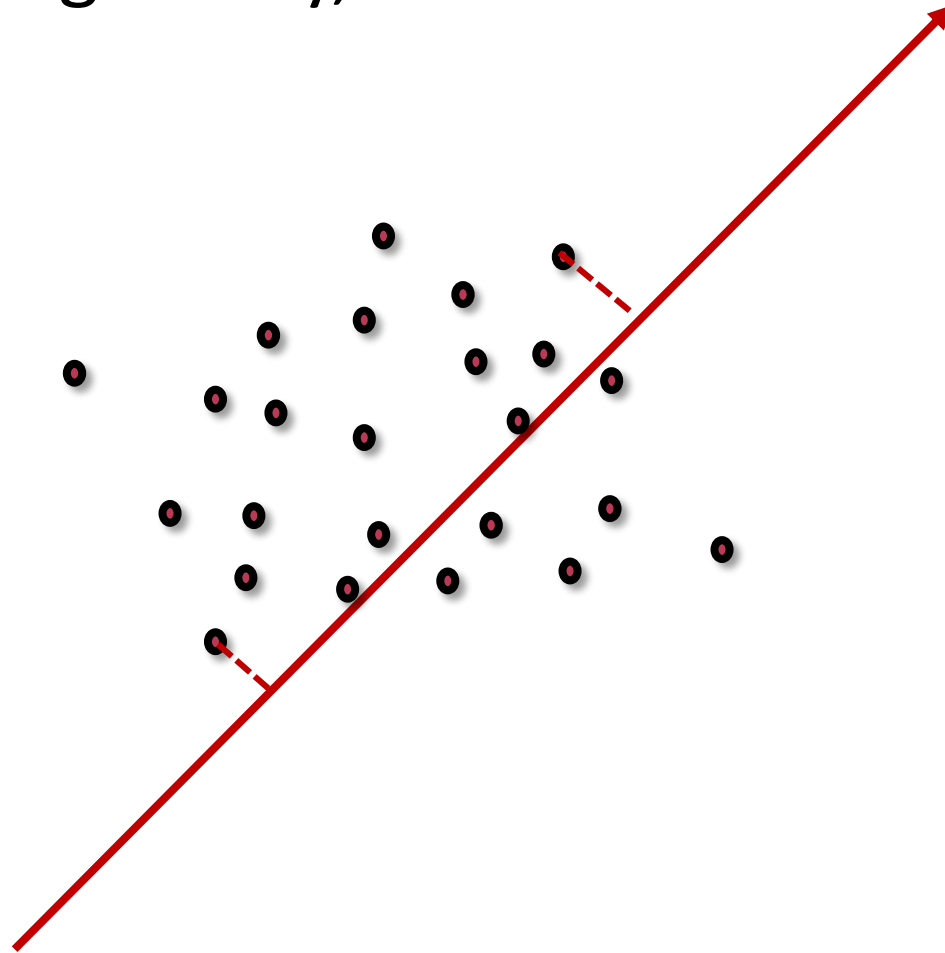
Befoglaló burok

- Féltér-mélység: Tukey, 1974



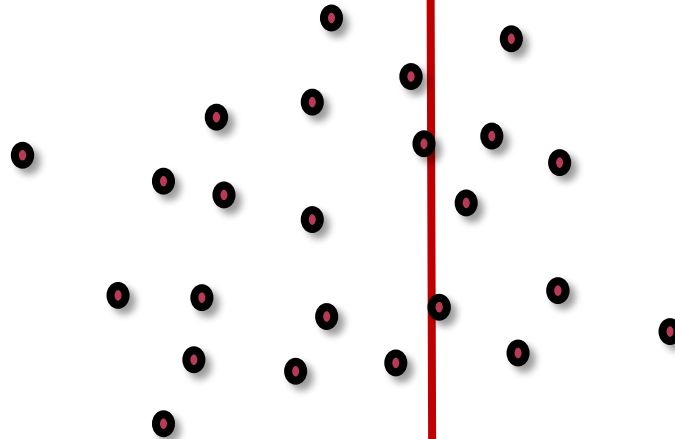
Befoglaló burok

- Féltér-mélység: Tukey, 1974



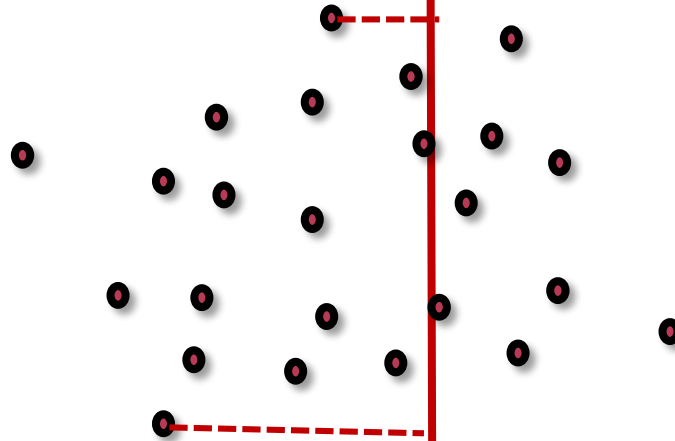
Befoglaló burok

- Féltér-mélység: Tukey, 1974



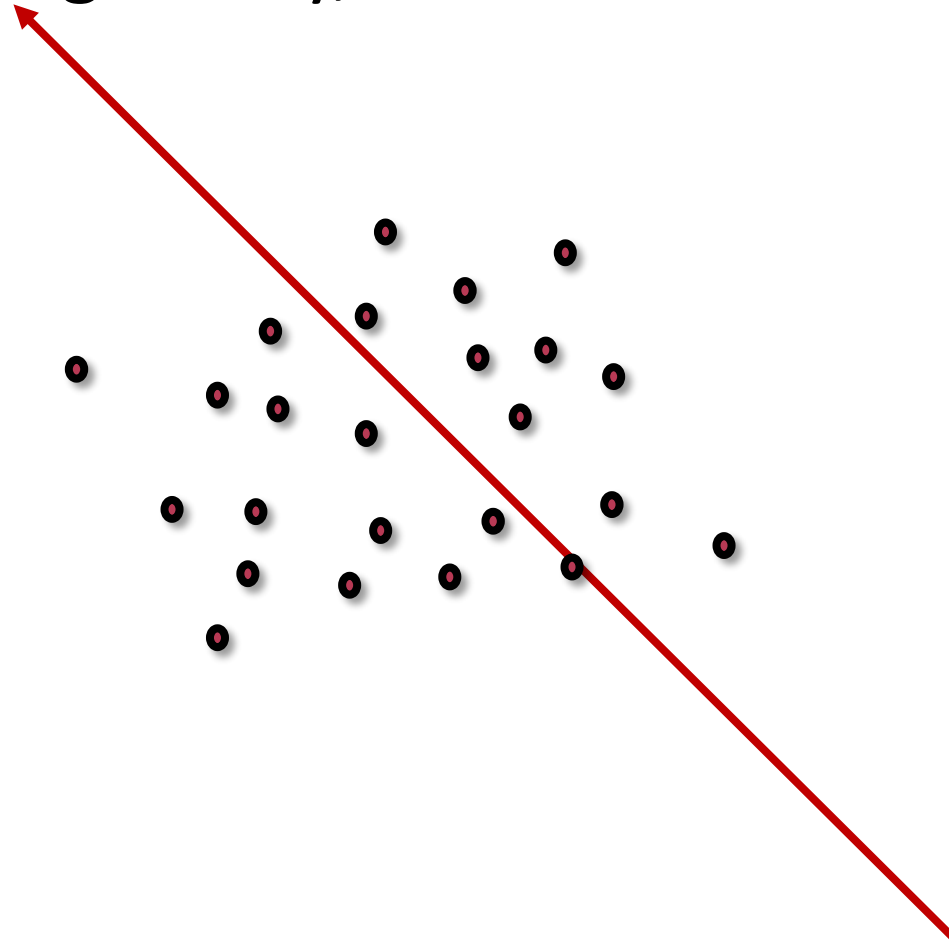
Befoglaló burok

- Féltér-mélység: Tukey, 1974



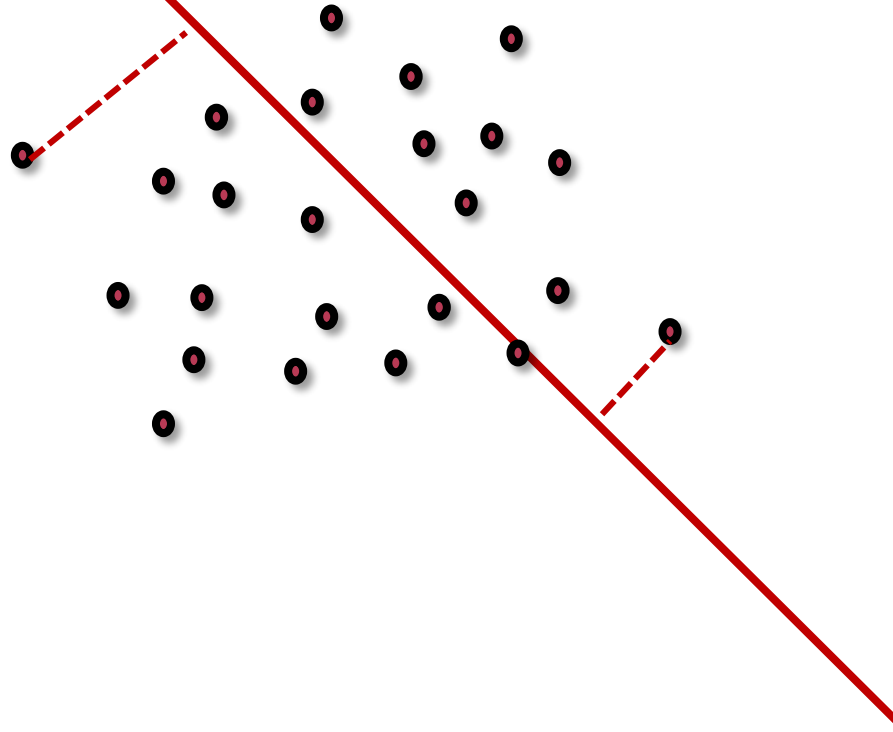
Befoglaló burok

- Féltér-mélység: Tukey, 1974



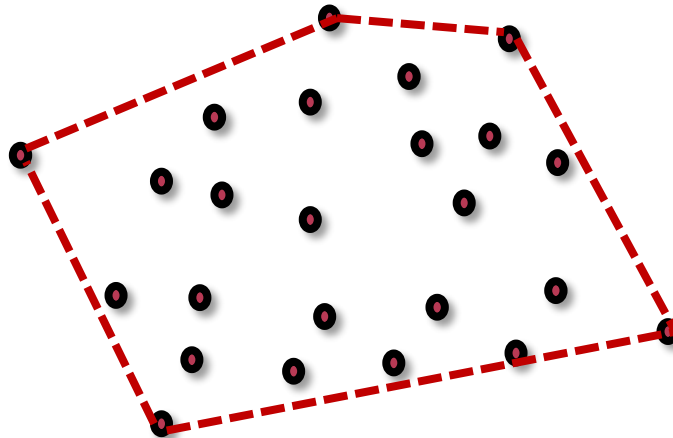
Befoglaló burok

- Féltér-mélység: Tukey, 1974



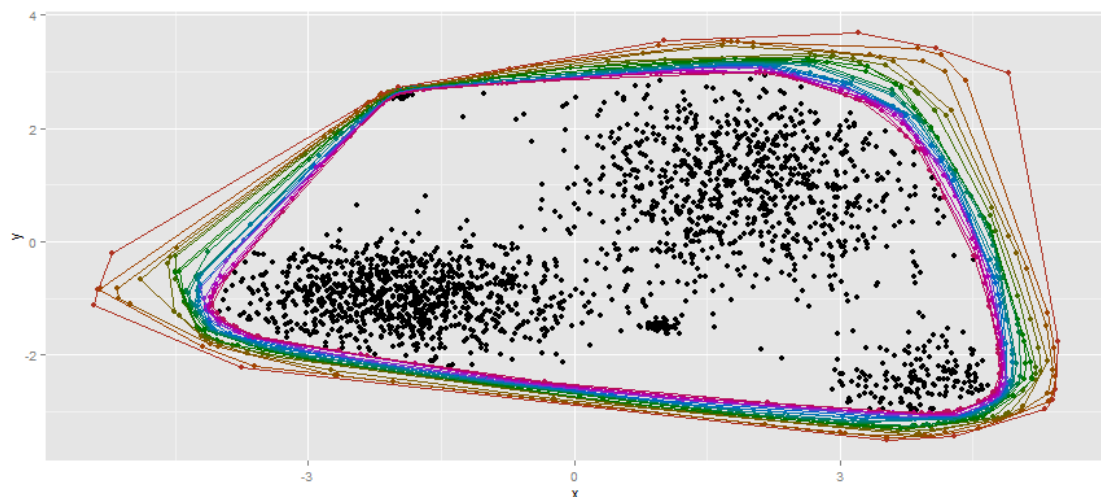
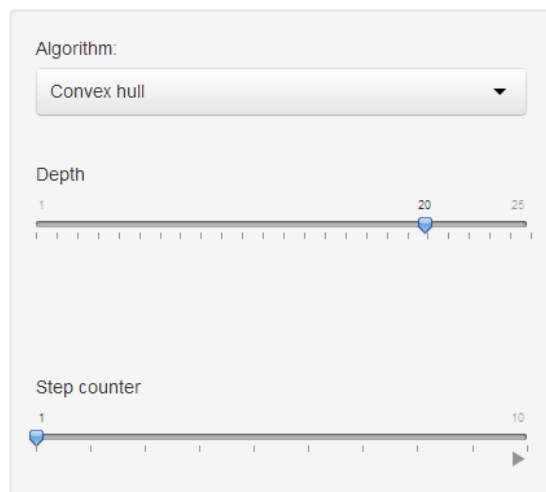
Befoglaló burok

- Féltér-mélység: Tukey, 1974



- Csomag: *depth*
- Hasznos függvények: *depth*, *isodepth*
- Paraméterek: *u* pont, *dpth* mélység

Rare event detection

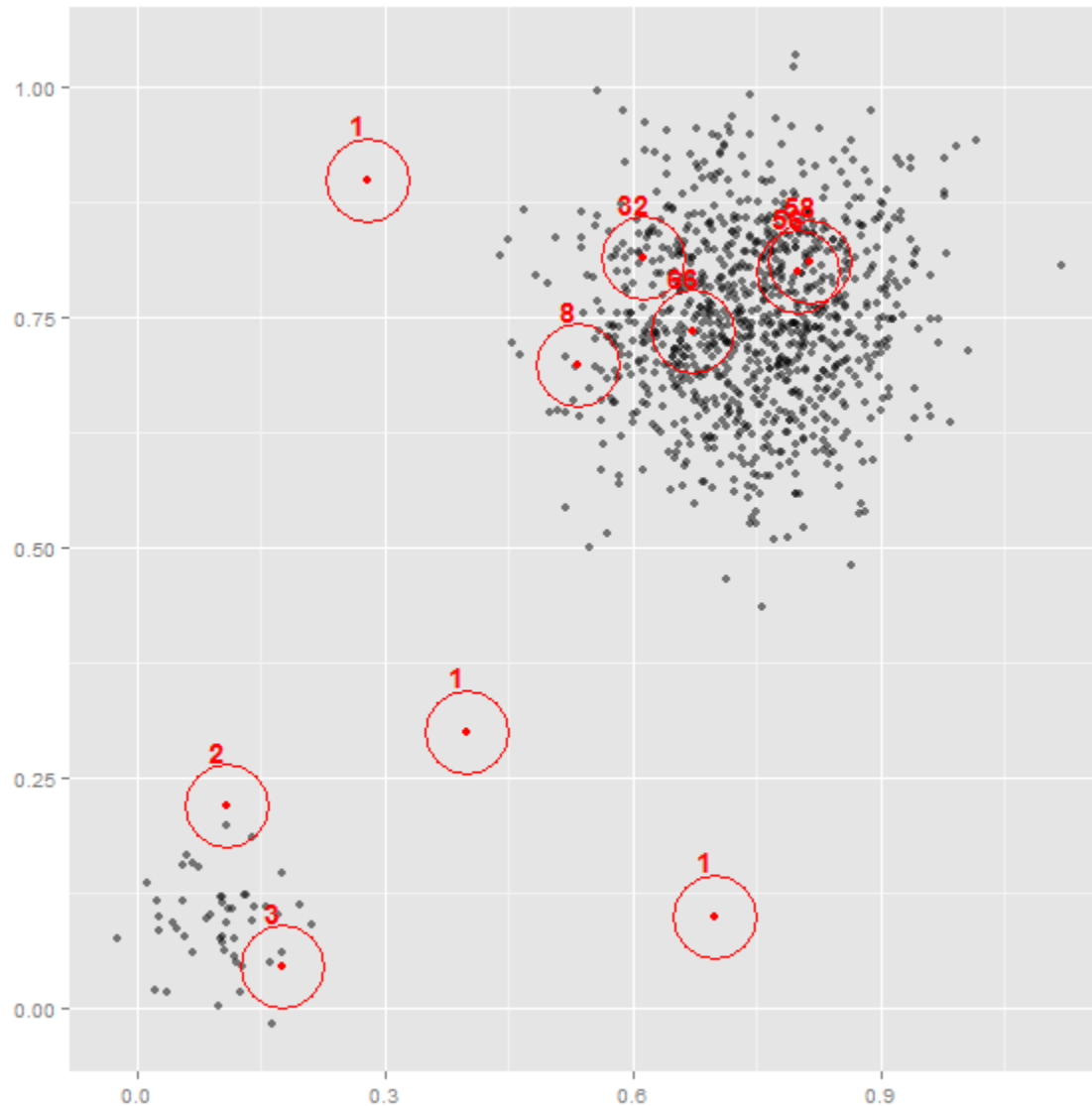


```
[1] "Depth:"  
[1] 20  
[1] "Ratio of chosen elements: "  
[1] 0.4095455
```

- Distance Based
- Outlier: szomszédok száma alacsony
- Paraméterek
 - r sugarú hipergömb
 - Szomszédok elvárt π aránya

DB

- Distance
- Outlier
- Parameter
 - r sug
 - Szomszéd



- Csomag: *fields*
- Függvény: *fields.rdist.near*
- Paraméterek: *delta* sugár

Rare event detection

Algorithm:

DB

Radius

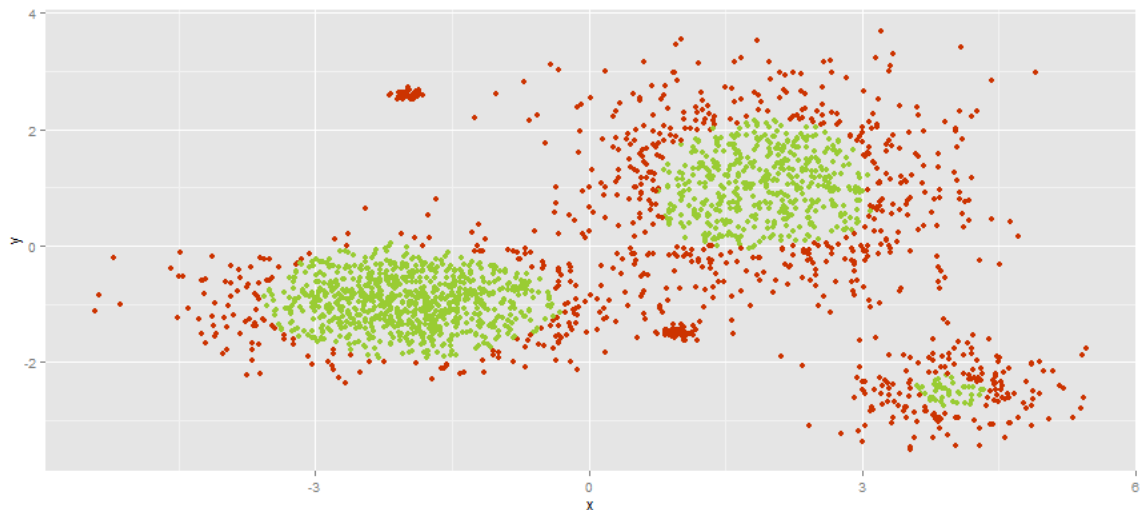
0 0.7 2

Ratio

0 0.05 0.25

Step counter

1 10

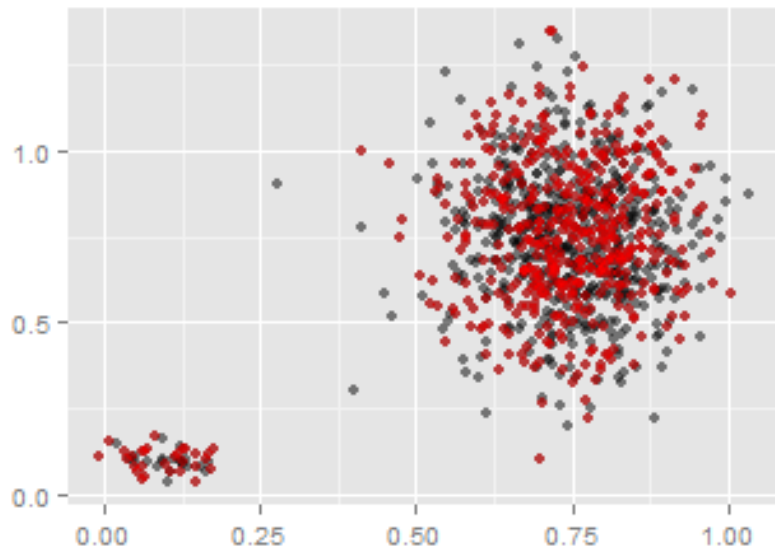


```
[1] "Ratio of chosen elements: "  
[1] 0.4145455
```

MCD

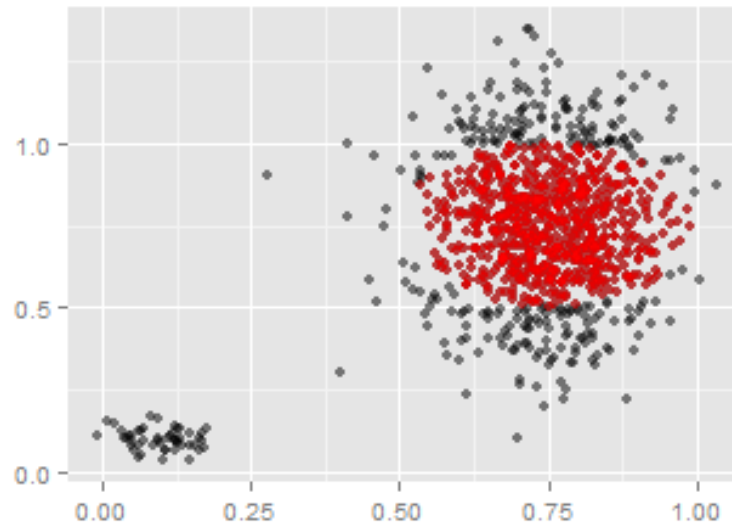
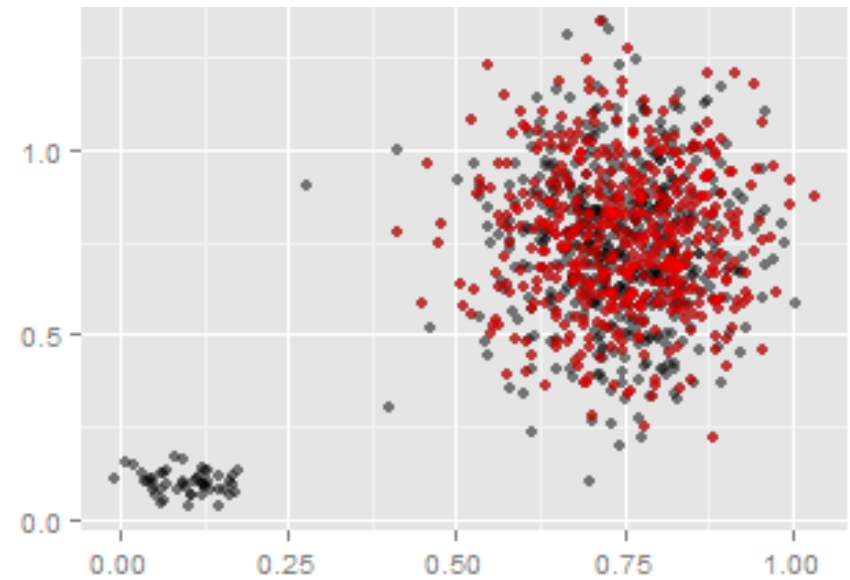
- Minimum Covariance Determinant
- Alapötlet
 - Keressük meg a legkompaktabb részhalmazt!

MCD

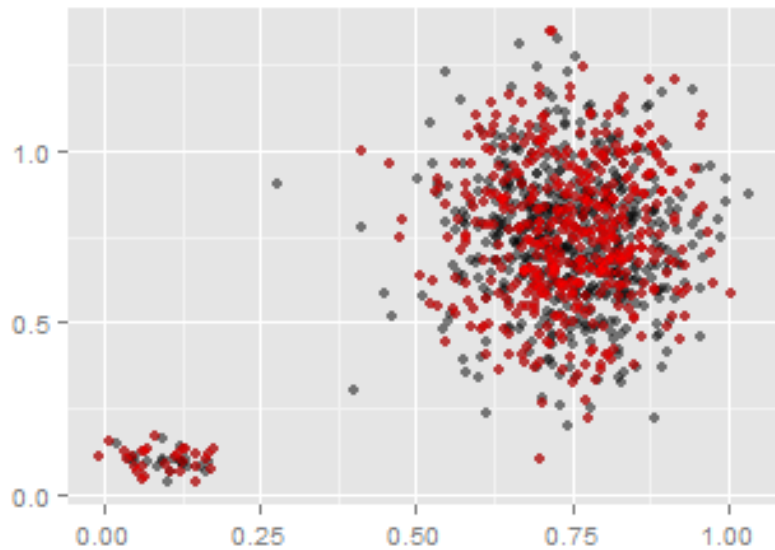


De

ipá



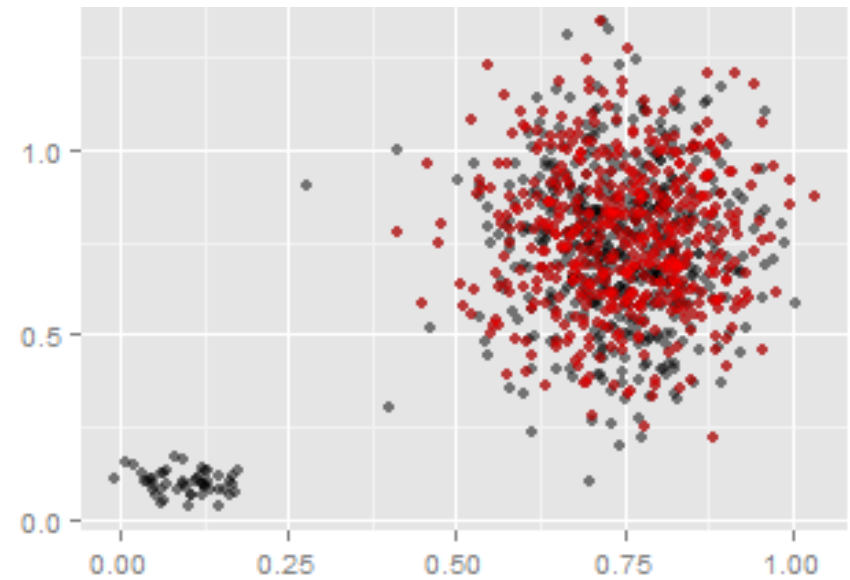
MCD



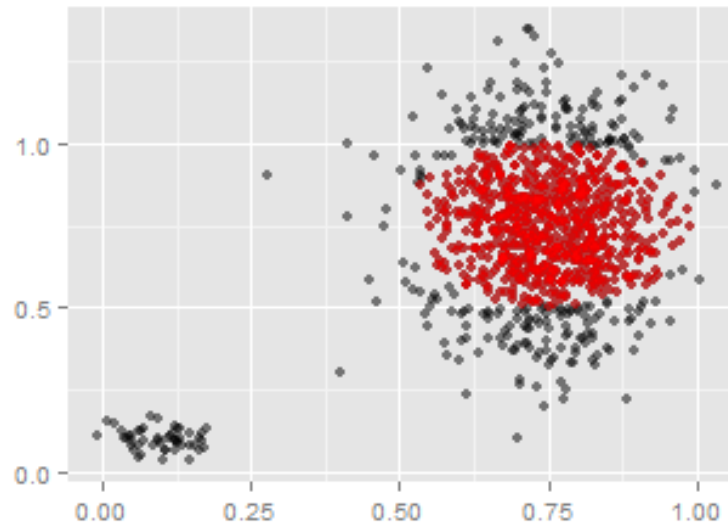
0.0014

De

ipá

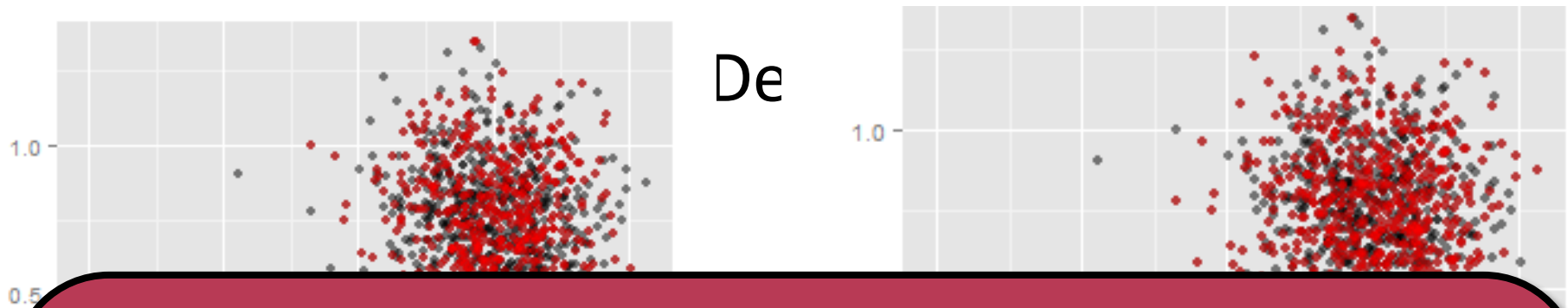


0.00041



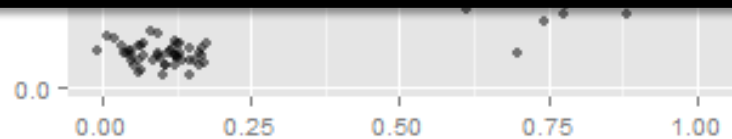
0.00011

MCD



Kimerítő keresés?

```
choose(n = 1000, k = 900)  
[1] 6.385051e+139
```



FAST-MCD

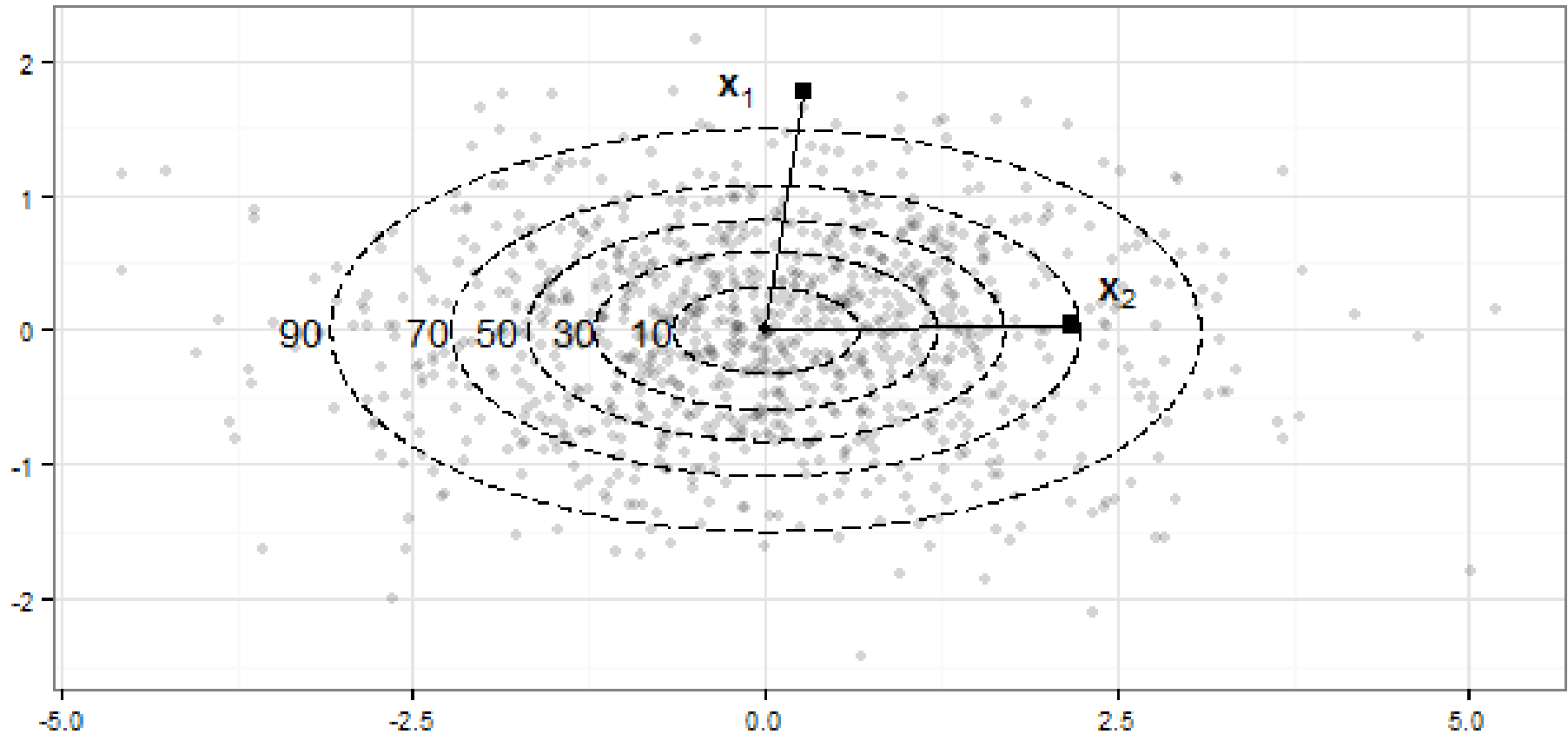
- Közelítő algoritmus
- Véletlenszerűen választott kezdőhalmaz
- Iteratív
- Legközelebbi pontok kiválasztása
 - Mahalanobis távolság alapján

Mahalanobis távolság

- $D(x, M) = \sqrt{(x - \vartheta)^T S^{-1} (x - \vartheta)}$
 - S – kovarianciamátrix
 - ϑ – súlypont

Mahalanobis távolság

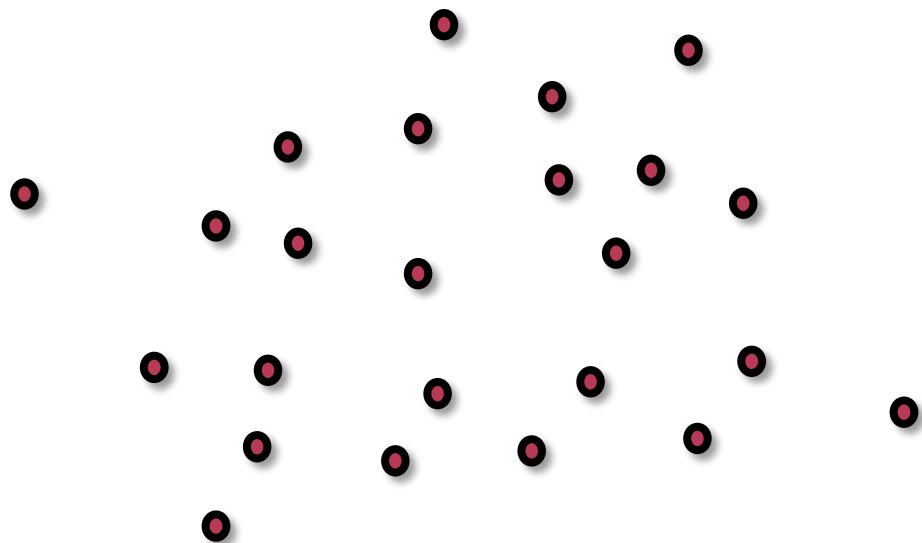
- $$D(x, M) = \sqrt{(x - \vartheta)^T S^{-1} (x - \vartheta)}$$



FAST-MCD

- Közelítő algoritmus
- Véletlenszerűen választott kezdőhalmaz

- Iteratív

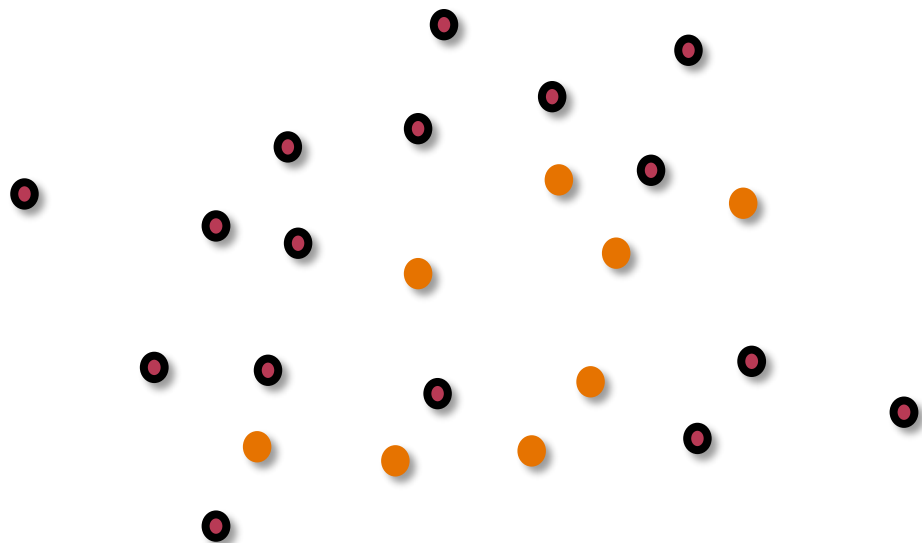


- Legközelebbi pontok kiválasztása
 - Mahalanobis távolság alapján
 - Legközelebbi $x\%$

FAST-MCD

- Közelítő algoritmus
- Véletlenszerűen választott kezdőhalmaz

- Iteratív



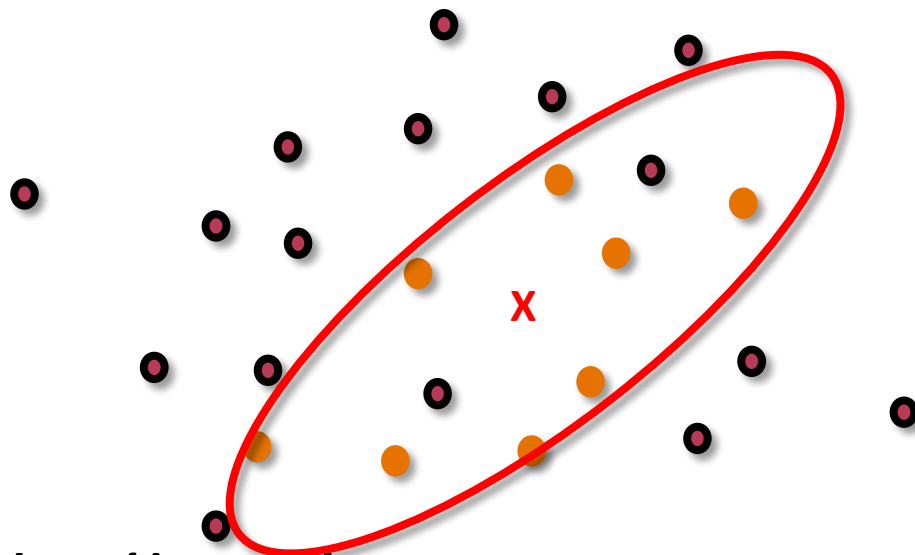
- Legközelebbi pontok kiválasztása
 - Mahalanobis távolság alapján
 - Legközelebbi $x\%$

FAST-MCD

- Közelítő algoritmus
- Véletlenszerűen választott kezdőhalmaz

- Iteratív

- Legközelebbi pontok kiválasztása
 - Mahalanobis távolság alapján
 - Legközelebbi $x\%$

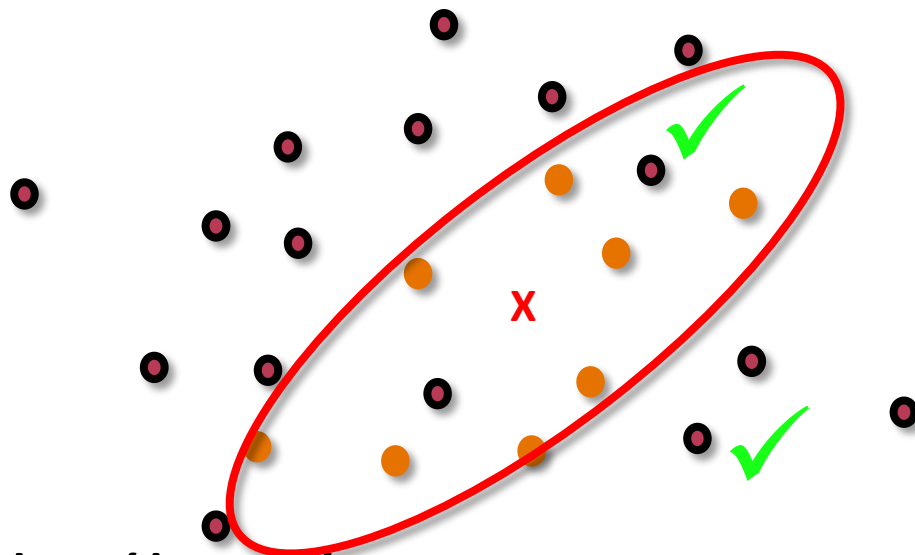


FAST-MCD

- Közelítő algoritmus
- Véletlenszerűen választott kezdőhalmaz

- Iteratív

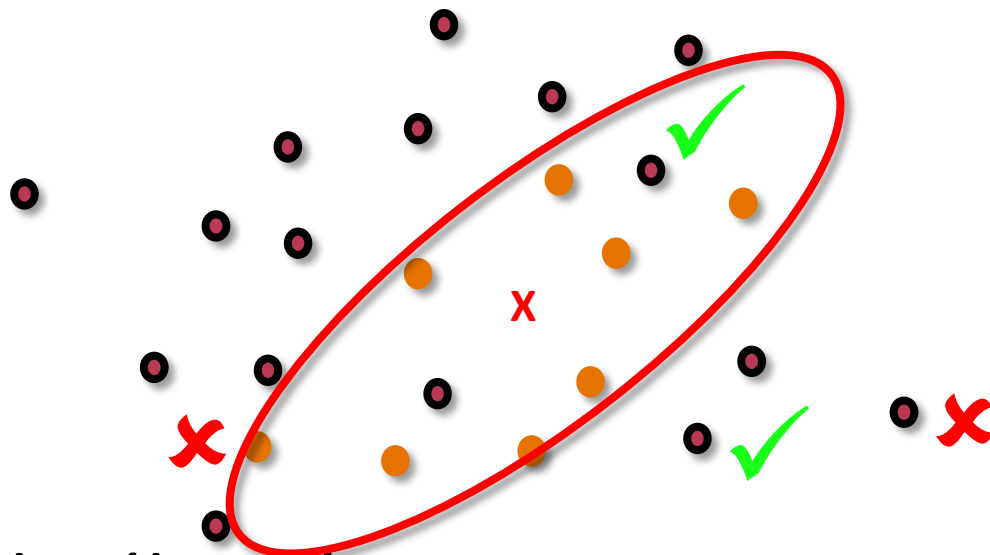
- Legközelebbi pontok kiválasztása
 - Mahalanobis távolság alapján
 - Legközelebbi $x\%$



FAST-MCD

- Közelítő algoritmus
- Véletlenszerűen választott kezdőhalmaz

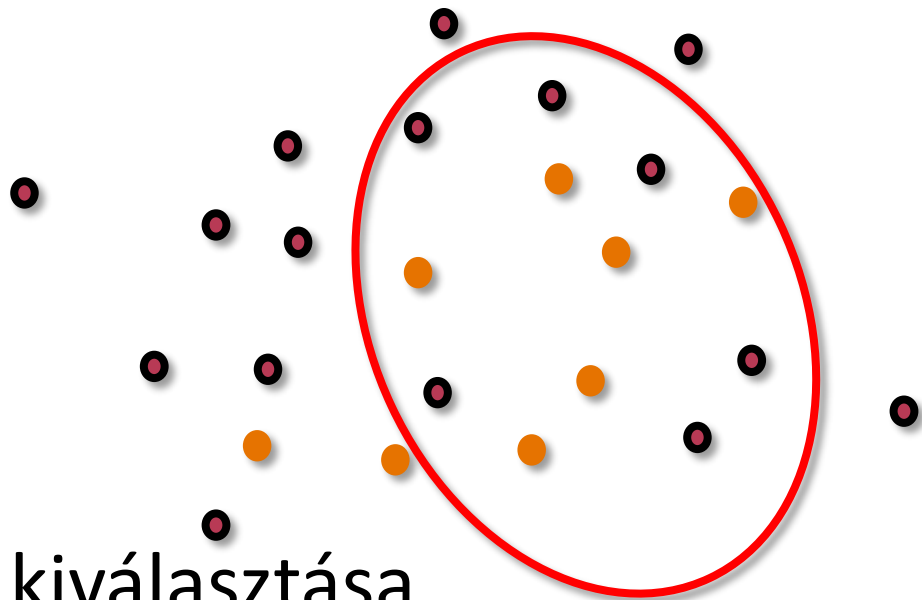
- Iteratív



- Legközelebbi pontok kiválasztása
 - Mahalanobis távolság alapján
 - Legközelebbi $x\%$

FAST-MCD

- Közelítő algoritmus
- Véletlenszerűen választott kezdőhalmaz
- Iteratív
- Legközelebbi pontok kiválasztása
 - Mahalanobis távolság alapján
 - Legközelebbi $x\%$



BACON

- Blocked Adaptive Computationally Efficient Outlier Nominators
- Kiinduló halmaz félig felügyelt módban is!
- Új halmaz: küszöbérték alapján

- Csomag: *robustX*
- Függvény: *mvBACON*
- Paraméterek
 - *init.sel* kezdőhalmaz
 - „manual” – *man.sel* kezdőhalmaz
 - „Mahalanobis”, „dUniMedian” – *m* kezdőhalmaz mérete

■ Csomag: *robustX*

Rare event detection

Algorithm:

BACON

Type:

Manual selection

alpha

0.01

0.95

Initial set:

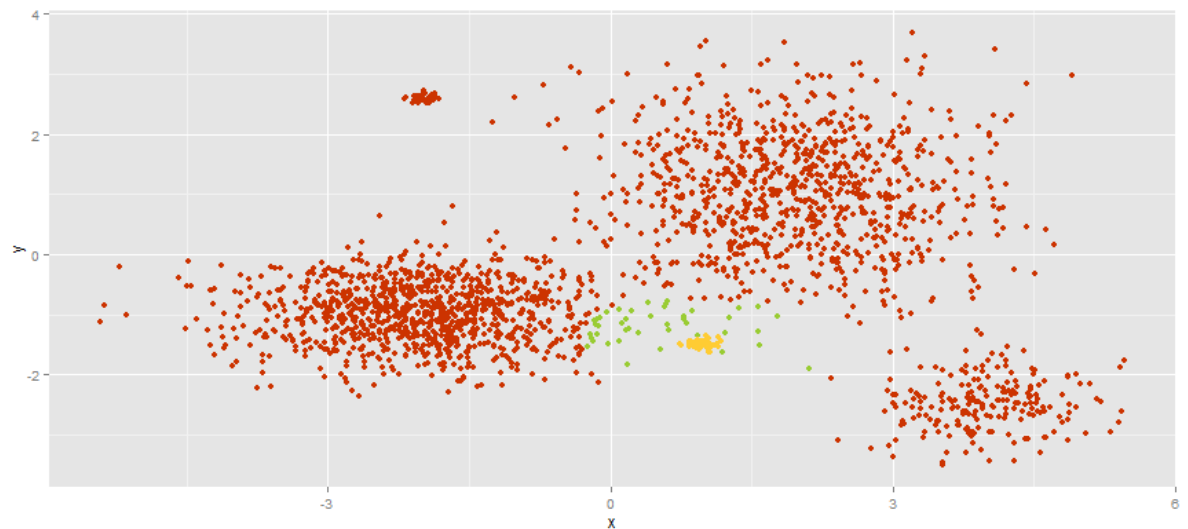
Set 5

Step counter

1

2

10

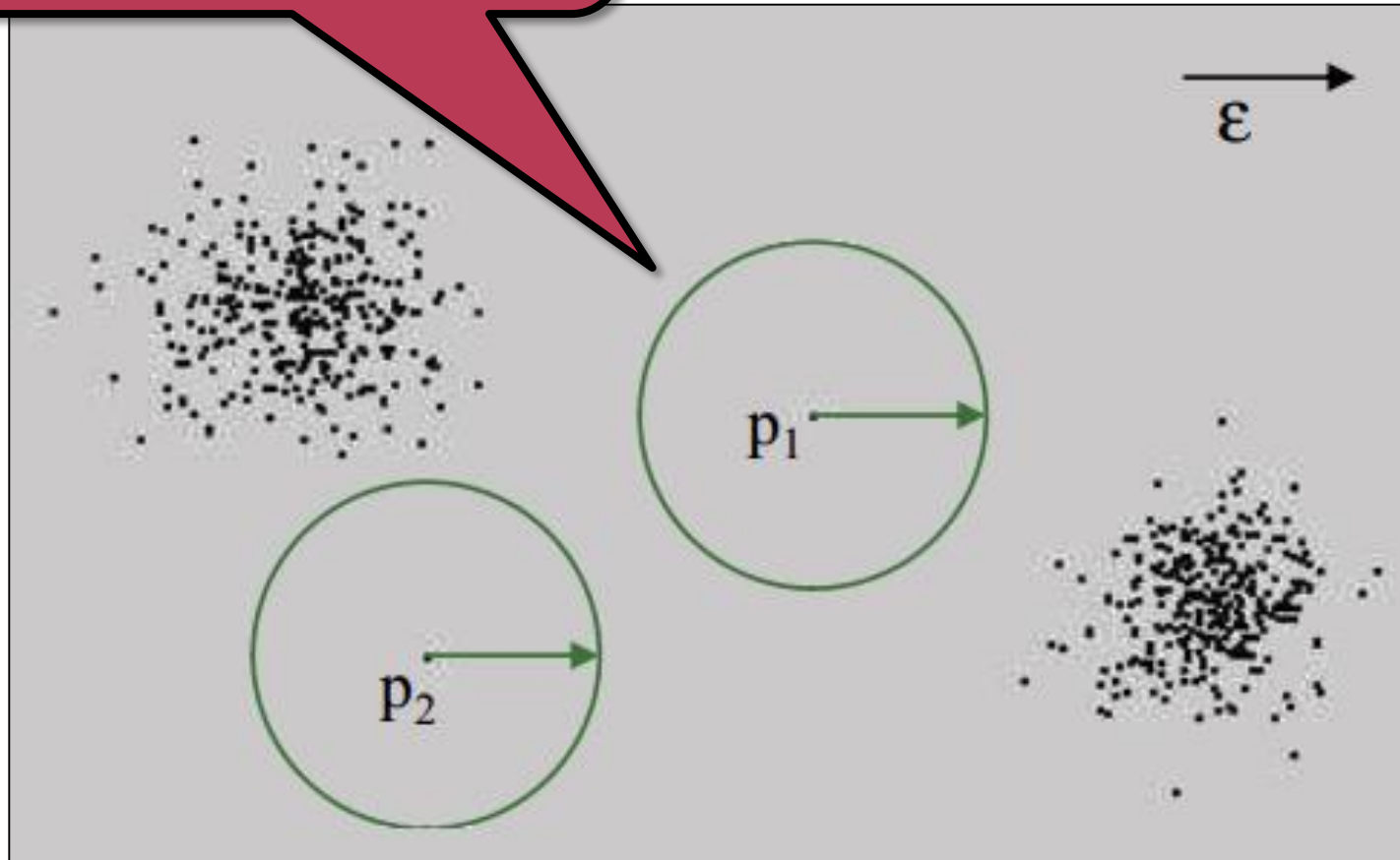


SŰRŰSÉG ALAPÚ TECHNIKÁK

DB alapötlete

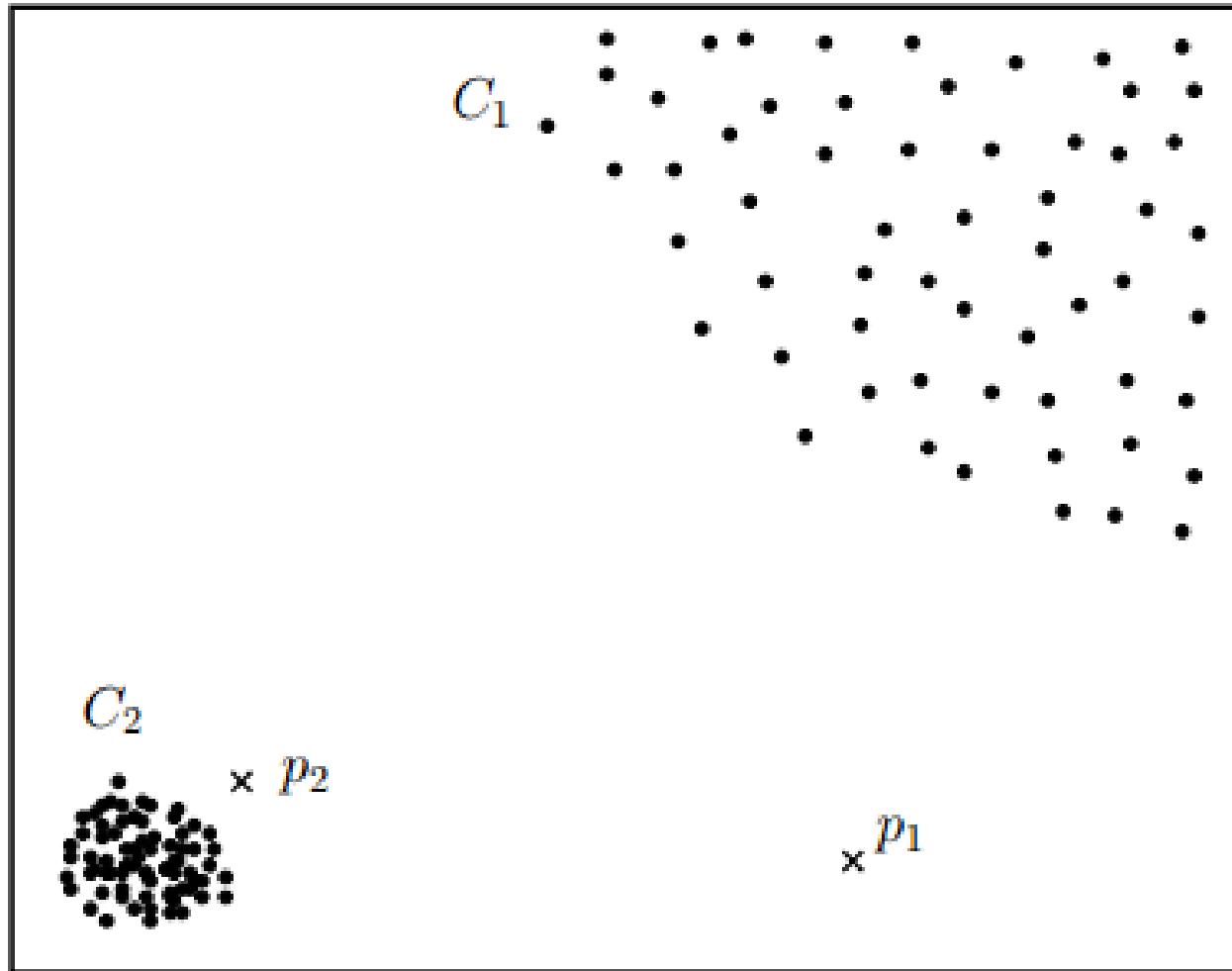
Hiába vagyunk a
középpontban, ha
nincsenek szomszédaink ☹️

Distance-based
approach



LOF motiváció: mikor jó a DB?

p_2 sem,
vagy C_1 is?



- Local Outlier Factor

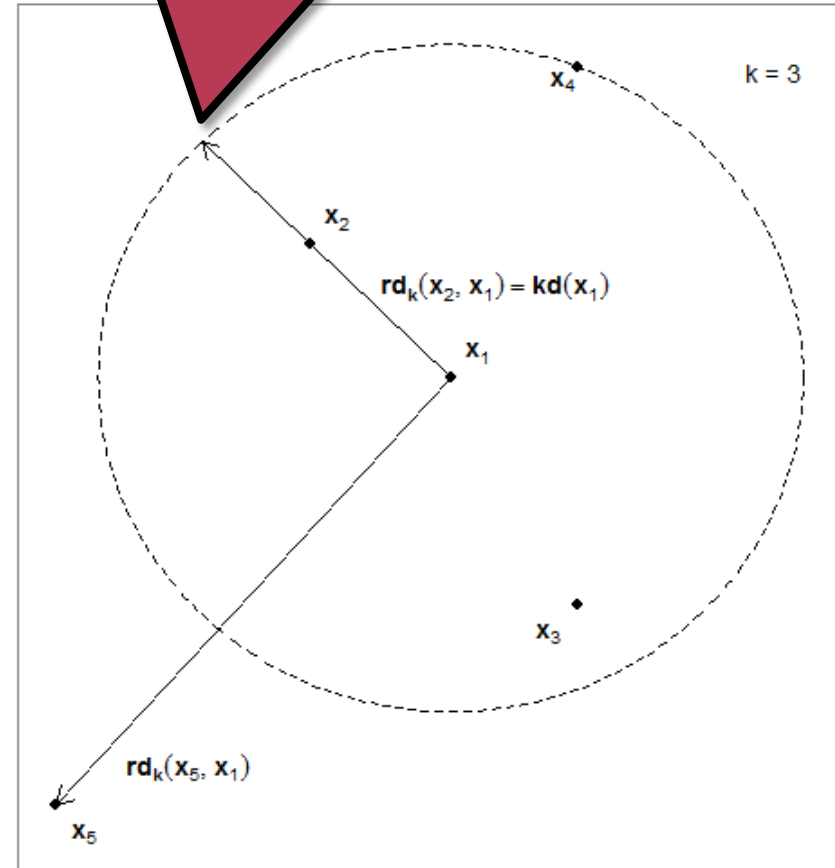
rd: reachability distance

- Alapötlet: csak a szomszédaival hasonlítsuk össze
 - lokális sűrűség
- Outlier kritérium
 - a lokális sűrűség jóval kisebb, mint a szomszédaimnak átlagosan

LOF

- Local Outlier Factor
- Alapötlet: csak a szomszédok
○ lokális sűrűség
- Outlier kritérium
○ a lokális sűrűség jóval kisebb

rd: reachability distance



- Local Outlier Factor

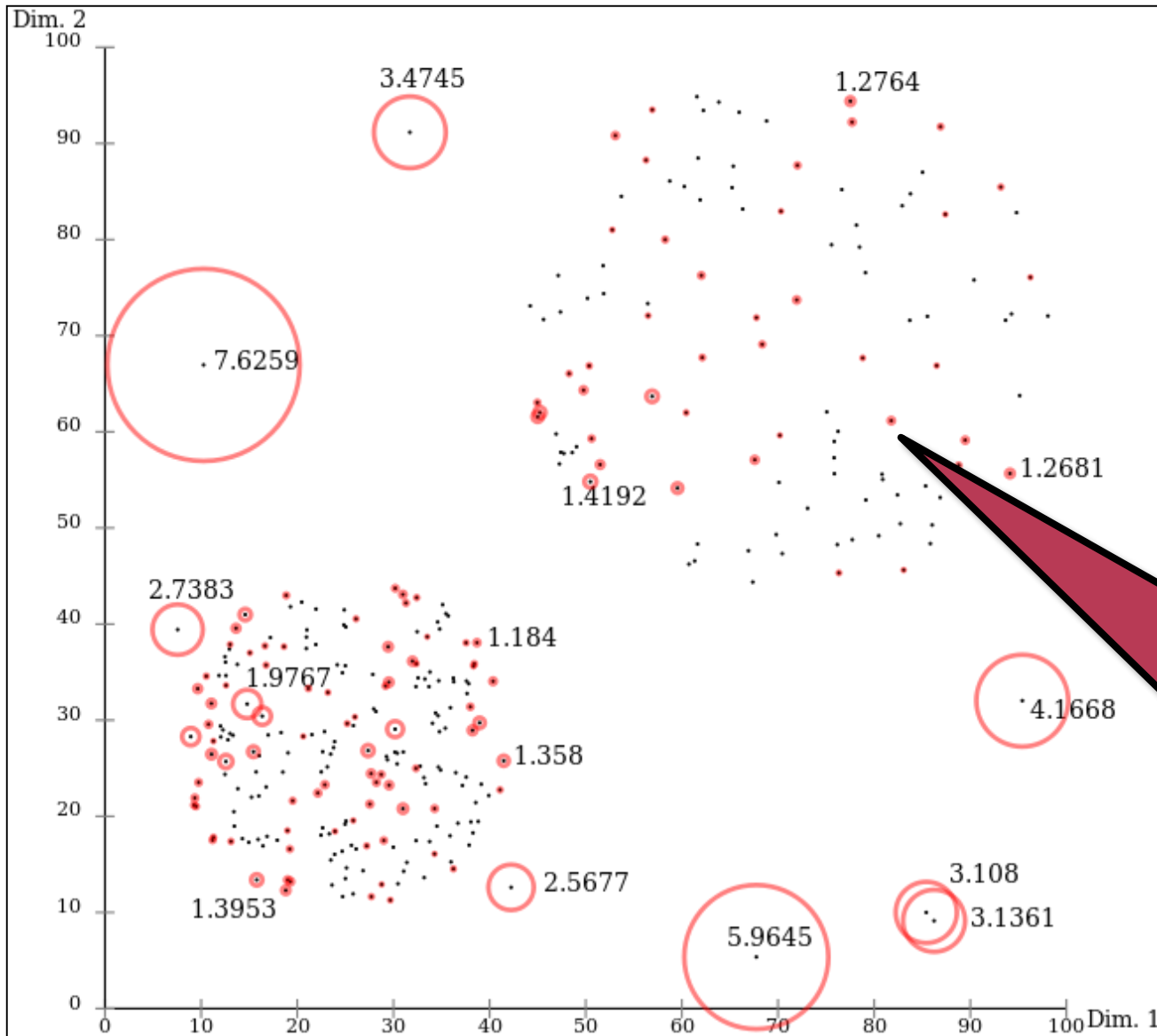
rd: reachability distance

- Alapötlet: csak a szomszédaival hasonlítsuk össze
 - lokális sűrűség
- Outlier kritérium
 - a lokális sűrűség jóval kisebb, mint a szomszédaimnak átlagosan

LOF

Local outlier factor

Ha a
szomszédaim
is
magányosak,
nincs nagy
gond 😊



LOF: DMwR::lofactor

DEMO LOF

- Csomag: *DMwR* (Data Mining with R)
- Függvény: *lofactor*
- Paraméterek: k szomszédság mérete

Rare event detection

Algorithm:

LOF

k

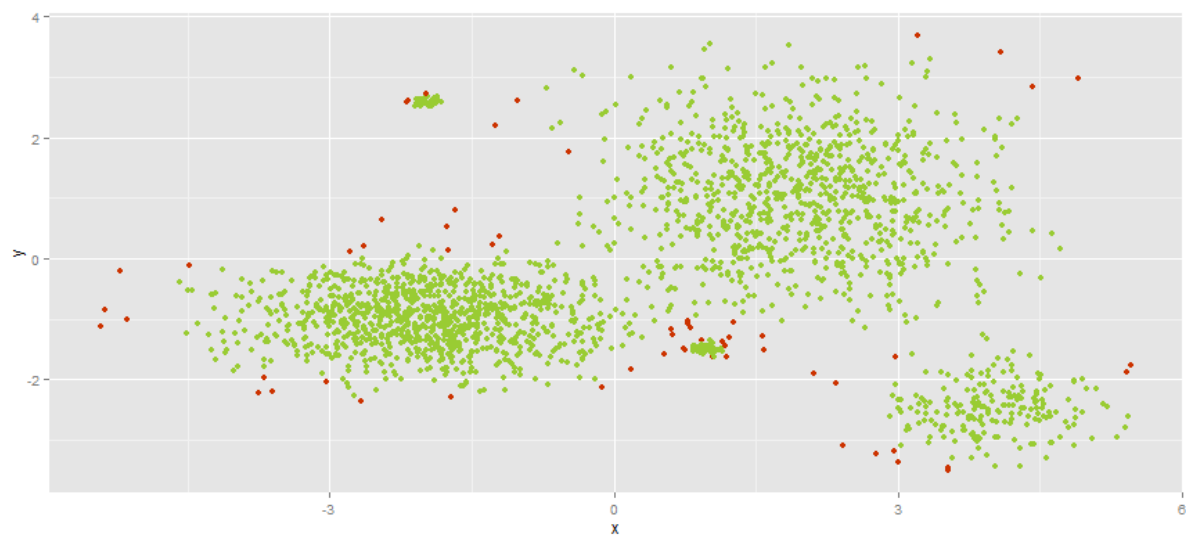
1 15 30

threshold

1 1.5 3

Step counter

1 10



OUTLIEREK ADATFOLYAMOKBAN

Adatfolyamok

Egyszer streamenként:
„Lokális maximum?”

Globális kérdések:
„Minden új maximumot
jelezzünk”

... 1, 5, 2, 7, 0, 9, 3

... a, r, v, t, y, h, b

... 0, 0, 1, 0, 1, 1, 0
time
→

Streams Entering

Ad-Hoc
Queries

Standing
Queries

Processor

Output

Buffer, megengedett
számítási memória
igény korlátos

Limited
Working
Storage

Archival
Storage

1. több forrásból,
2. ismeretlen sebességgel

Ábra és a számértékes példák forrása: [1]

Outlierek idősorokban

- IT Monitorozás
- Tőzsdei elemzések
- Banki csalásfelderítés

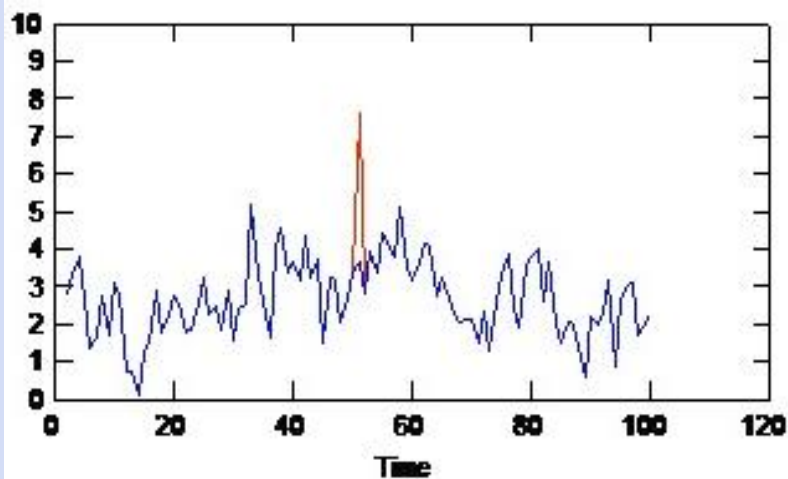
- Mindkét adattípus számít
 - Szenzorok: nagyrészt numerikus
 - *CPU_nice*: 0.12, 0.13, 0.12, 0.13, ...
 - Naplózás: nagyrészt kategorikus
 - *VM_operations*: Start, Stop, Start, Snapshot, Snapshot, ...

Hatások szerinti osztályozás

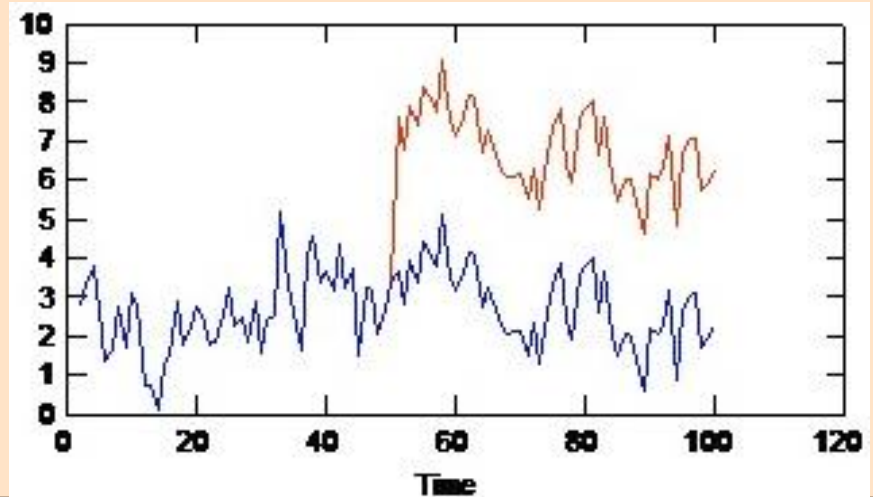
- Additive outlier
 - A rákövetkező elemekre teljesen hatástalan
- Level Shift Outlier
 - Permanens hatás
- Innovational Outlier
 - Kezdeti hatás + lecsengés, az ismétlések számával ez erősödhet
- Transient Change Outlier
 - ~Innovational outlier, de exponenciálisan lecseng a hatás, később visszatér normálra

Basic types

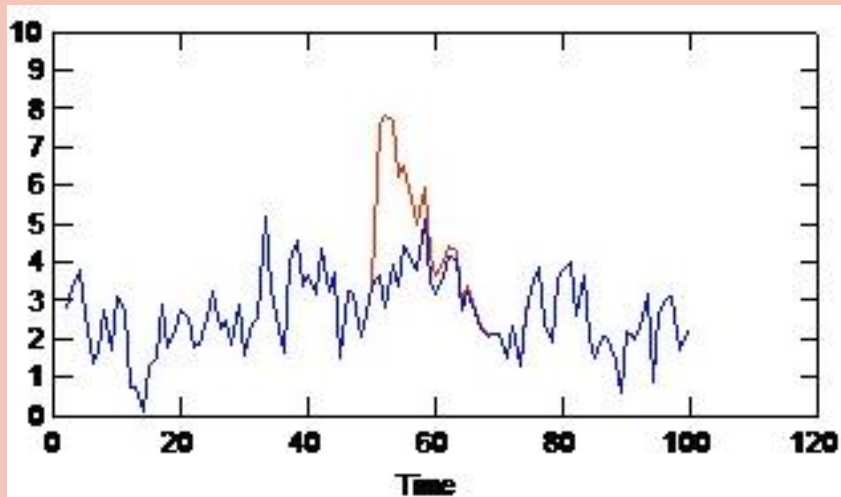
Additive



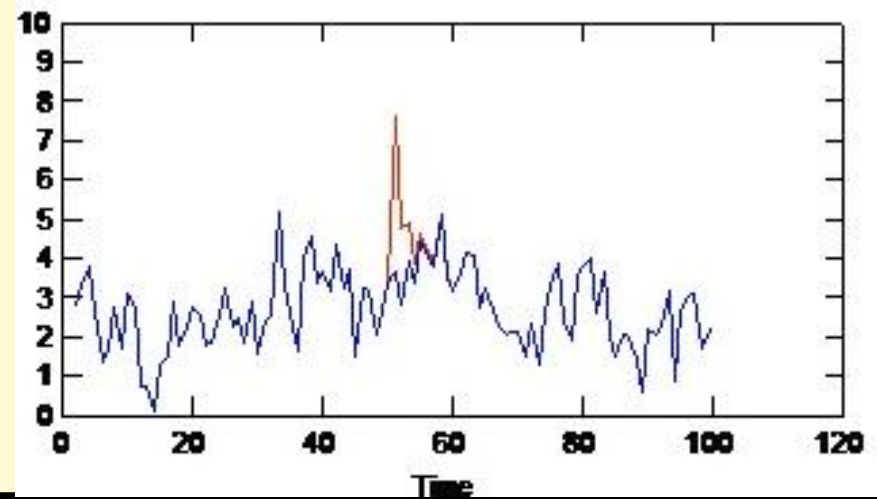
Level Shift



Innovational



Transient change



Outlierek szekvenciák *között*

- „Az aggregált adatokon látjuk, hogy baj van. Pontosan a rendszer melyik komponense hibás?”
- Feltételezések
 - Az idősorok hossza azonos
 - Keressük a legkiugróbbat

Outlierek szekvenciák *között*

- Ötletek
 - Képezzük le egy értékre az idősort
 - variancia
 - az első pillanat, amikor az érték elért egy küszöböt
 - Elemek egy hasonlósági mátrixba
- Innentől már akármelyik klasszikus klaszterezési módszer működik
- Távolságfüggvény a szomszédossághoz?

Idősorok távolságfüggvényei

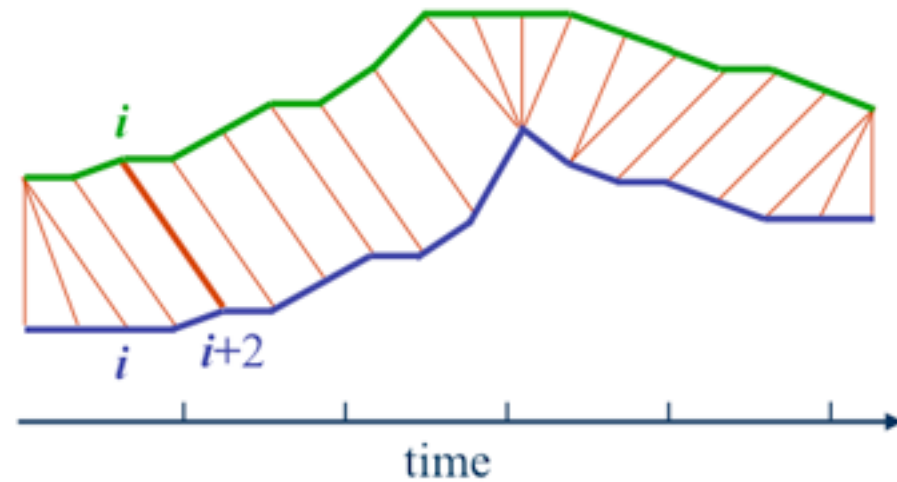
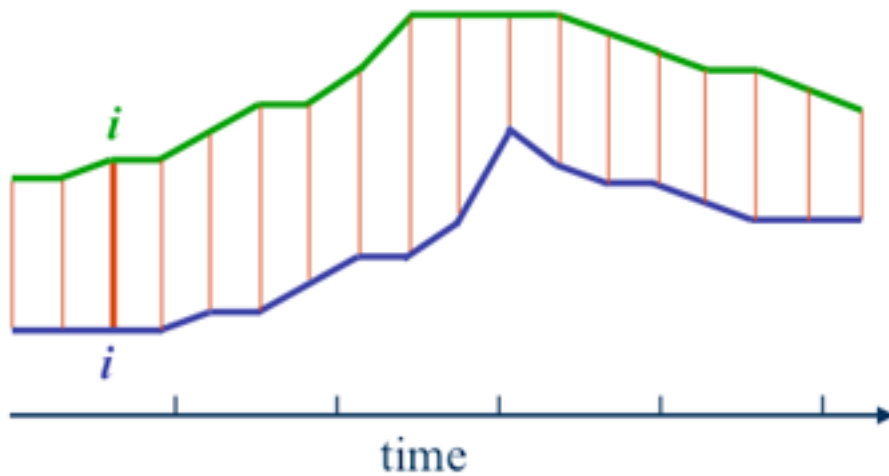
- Euklideszi távolság
 - X tengely menti eltolás (offset)?
- Dynamic time warping
 - eleve kiugró értékek alapján hasonlítunk
- Length of common subsequence

Dinamikus idővetemítés

- Az idősorok pontjait nem indexenként hasonlítjuk össze
 - Motiváció pl. hangfelismerésnél

Dinamikus idővetemítés

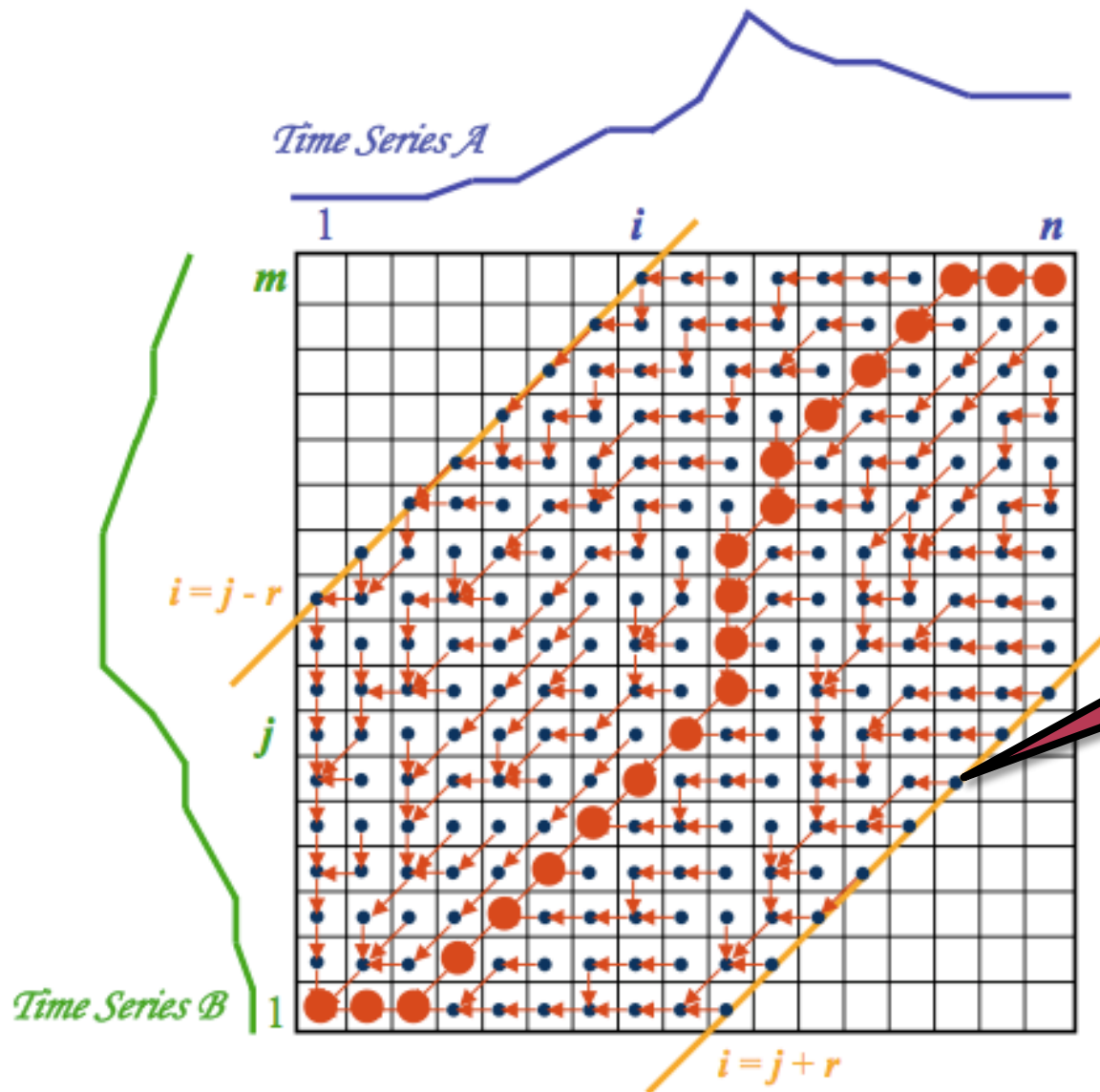
- Az idősorok pontjait nem indexenként hasonlítjuk össze
 - Motiváció pl. hangfelismerésnél



Dinamikus idővetemítés számítása

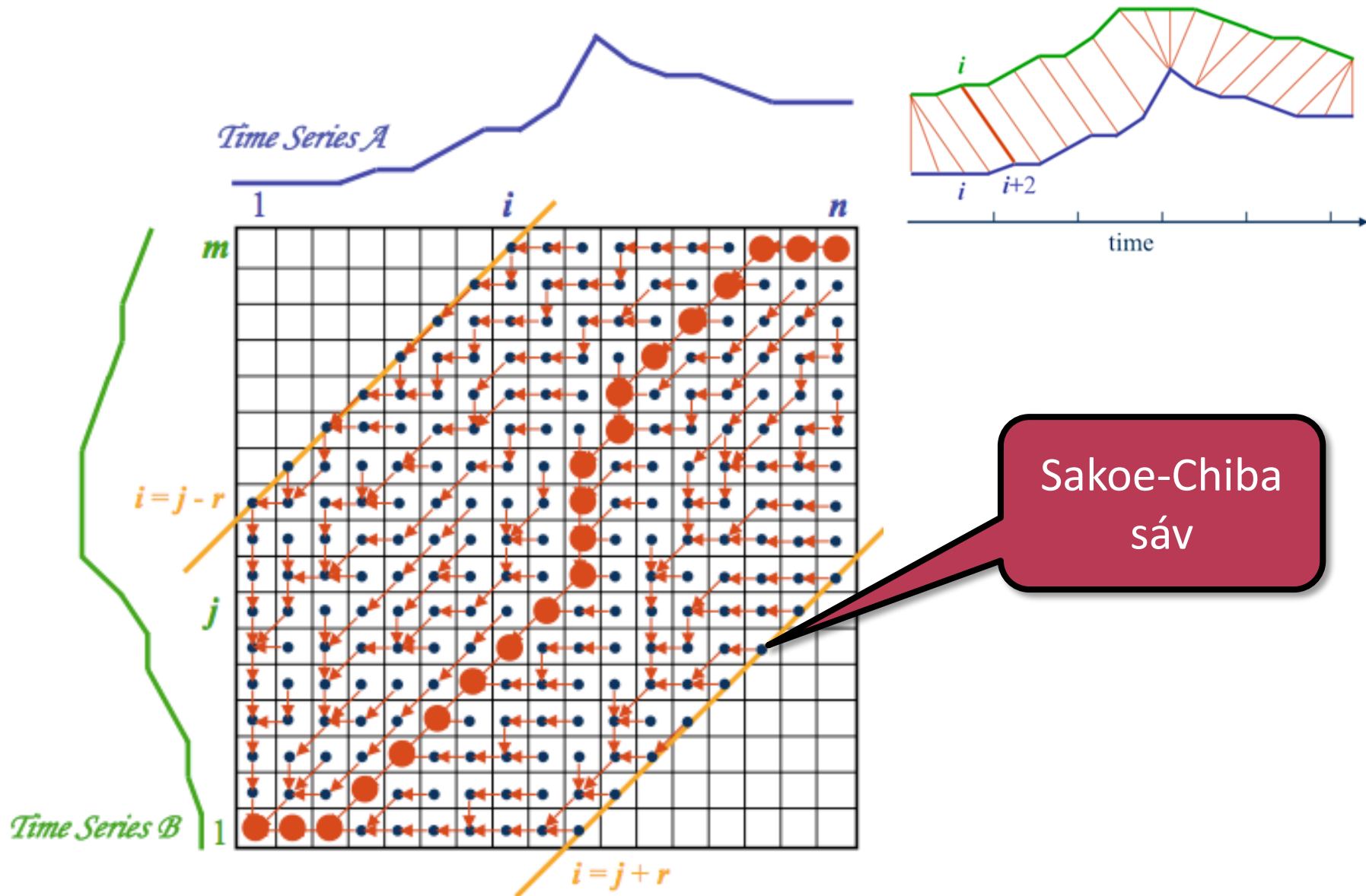
1. $n \times m$ -es D mátrixban rögzítjük a sorok egymástól való távolságát
2. Kell: $p = [p_1, p_2, \dots, p_k]$ útvonal a $D[1,1]$ és $D[n,m]$ között
3. Cél: minimális költség
4. Szabályok:
 1. Minden lépésben előre haladunk (nem távolodhatunk, tehát $[i,j] \rightarrow [\tilde{i}, \tilde{j}]$ esetén $\tilde{i} \geq i, \tilde{j} \geq j$)
 2. Az út folytonos, mindig csak szomszédos cellákra léphetünk

Dinamikus idővetemítés



Sakoe-Chiba
sáv

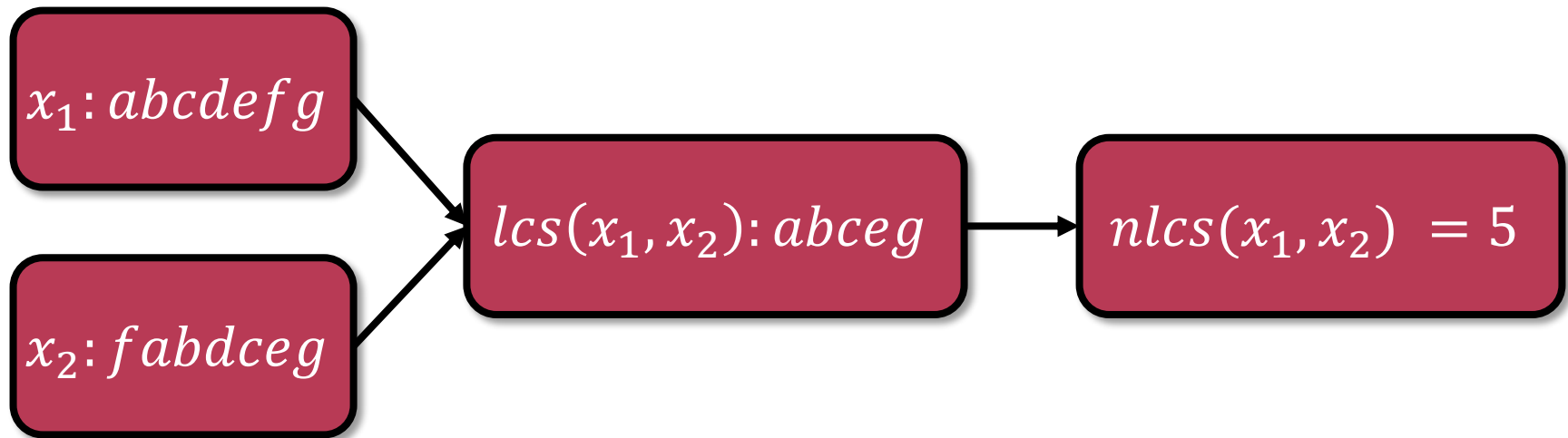
Dinamikus idővetemítés



Sakoe-Chiba
sáv

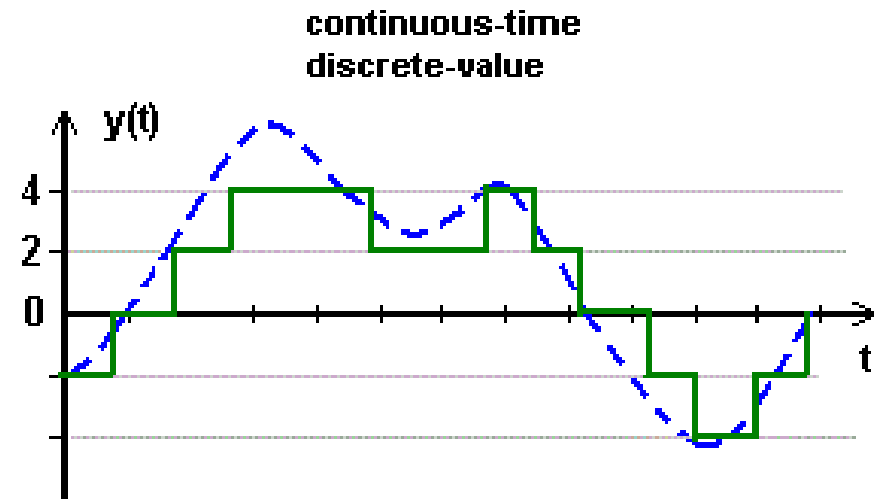
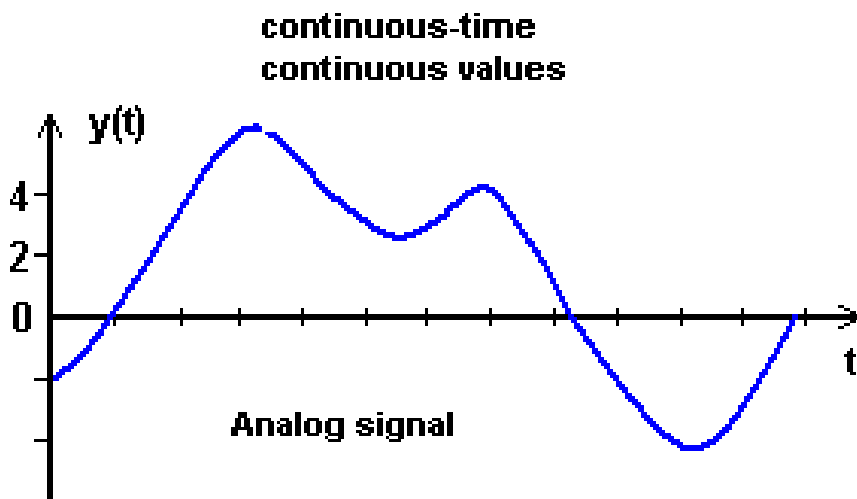
Longest common subsequence

- Nem a pontos időpont számít
- Csak a sorrend



Longest common subsequence

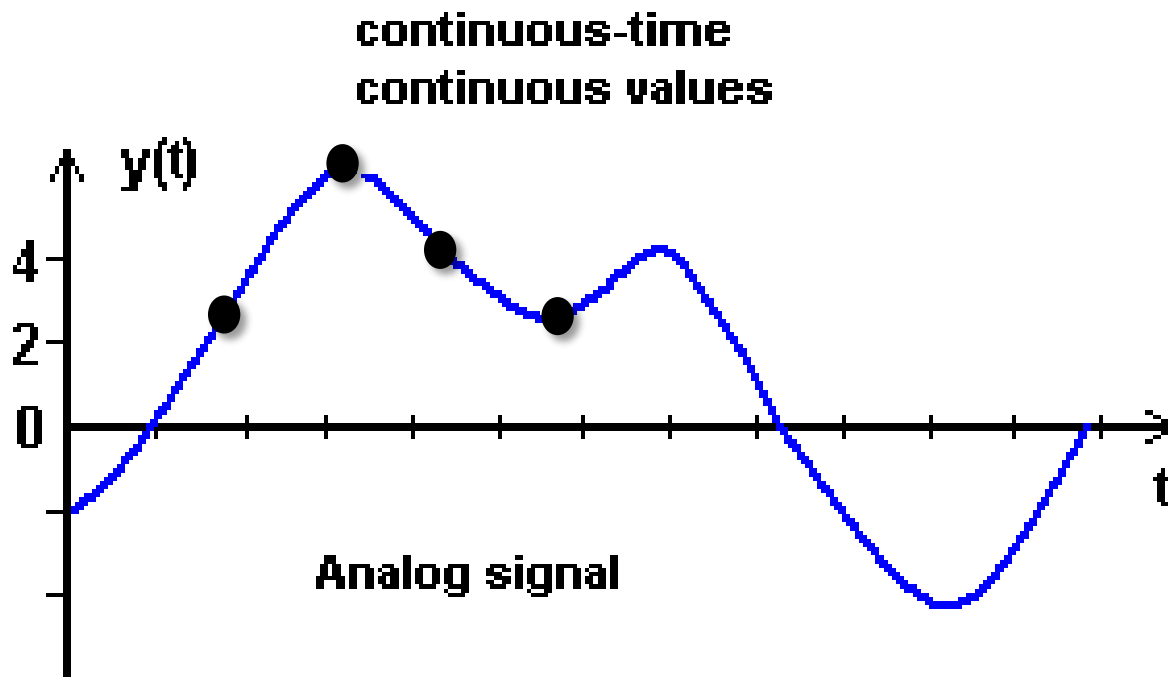
- Nem a pontos időpont számít
- Csak a sorrend
- Általánosítás folytonos értékekre



Outlierek szekvenciákban

- A legkiugróbb pont megtalálása

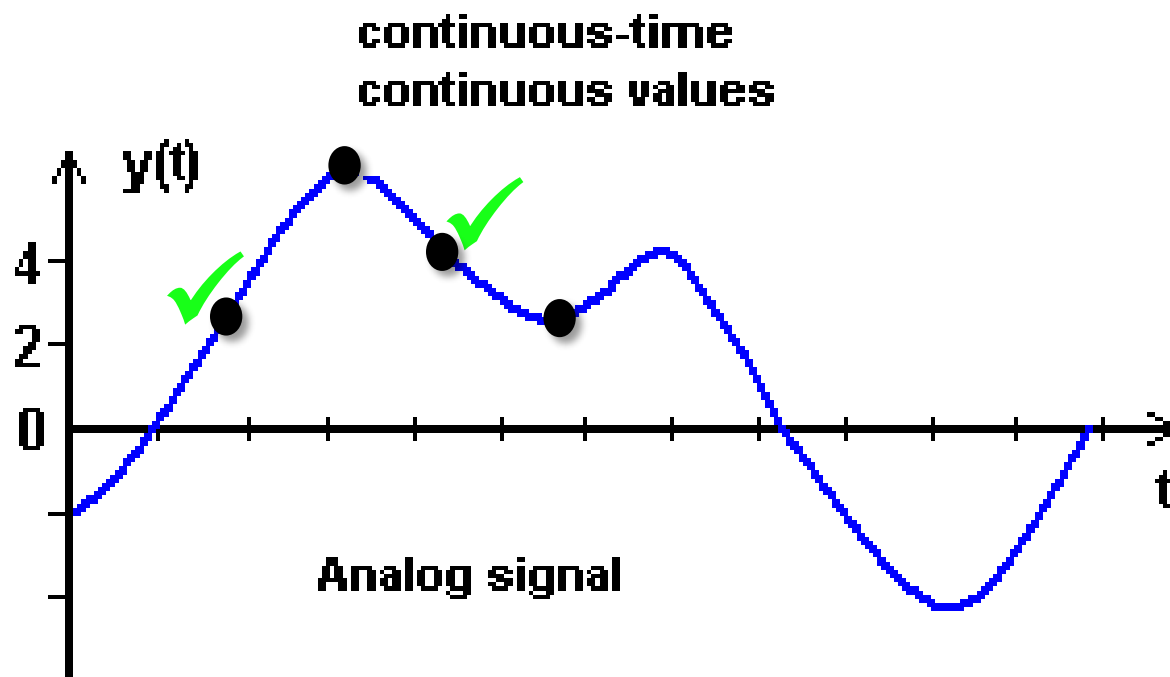
- $abs(T[k] - mean(T[k - l], ..., T[k + l])) : \max$



Outlierek szekvenciákban

- A legkiugróbb pont megtalálása

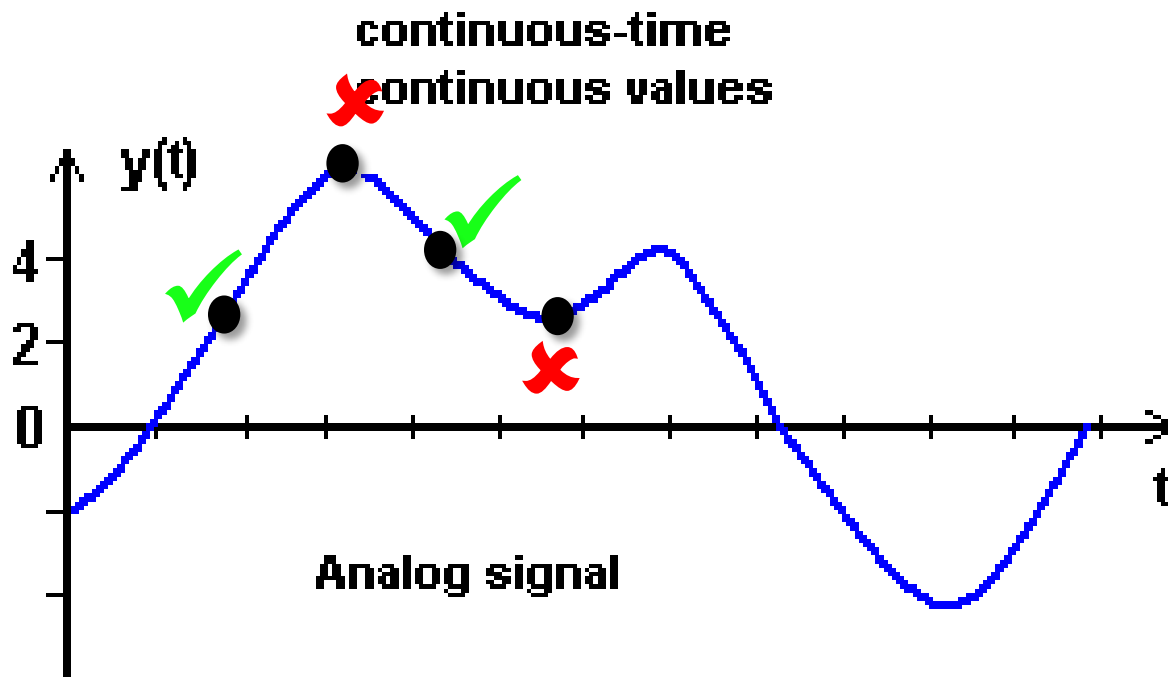
- $abs(T[k] - mean(T[k - l], ..., T[k + l])) : \max$



Outlierek szekvenciákban

- A legkiugróbb pont megtalálása

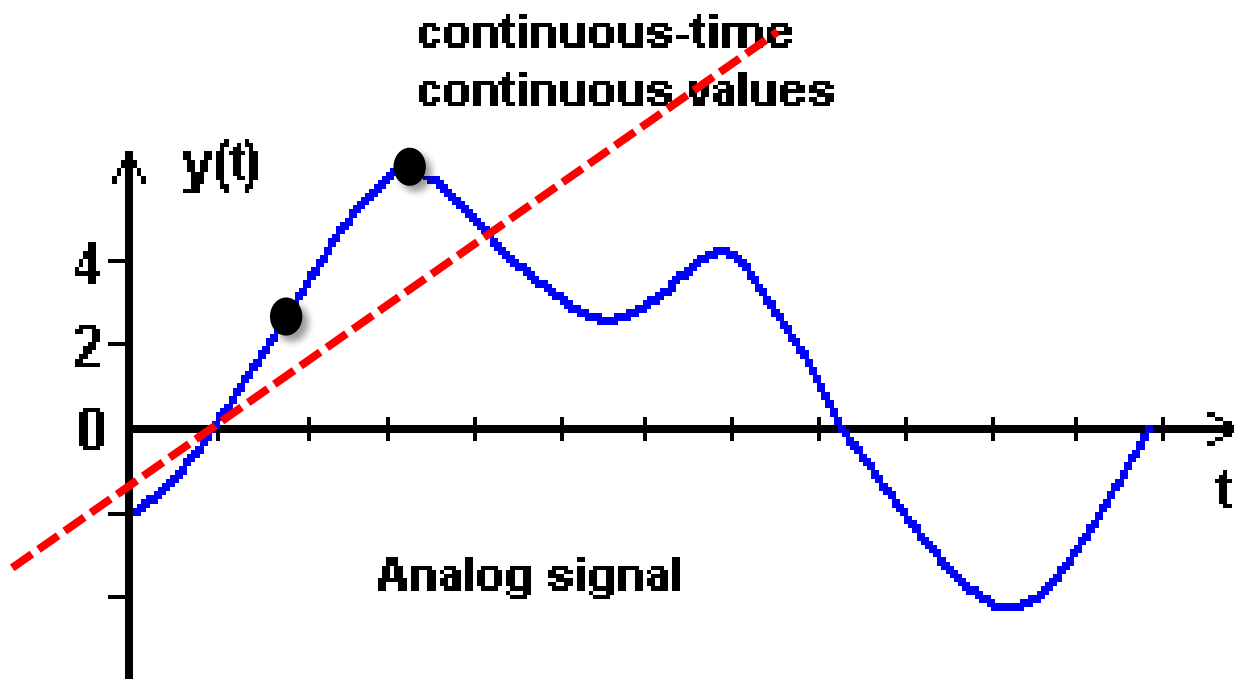
○ $abs(T[k] - mean(T[k - l], ..., T[k + l])) : \max$



Outlierek *szekvenciákban*

- A legkiugróbb pont megtalálása

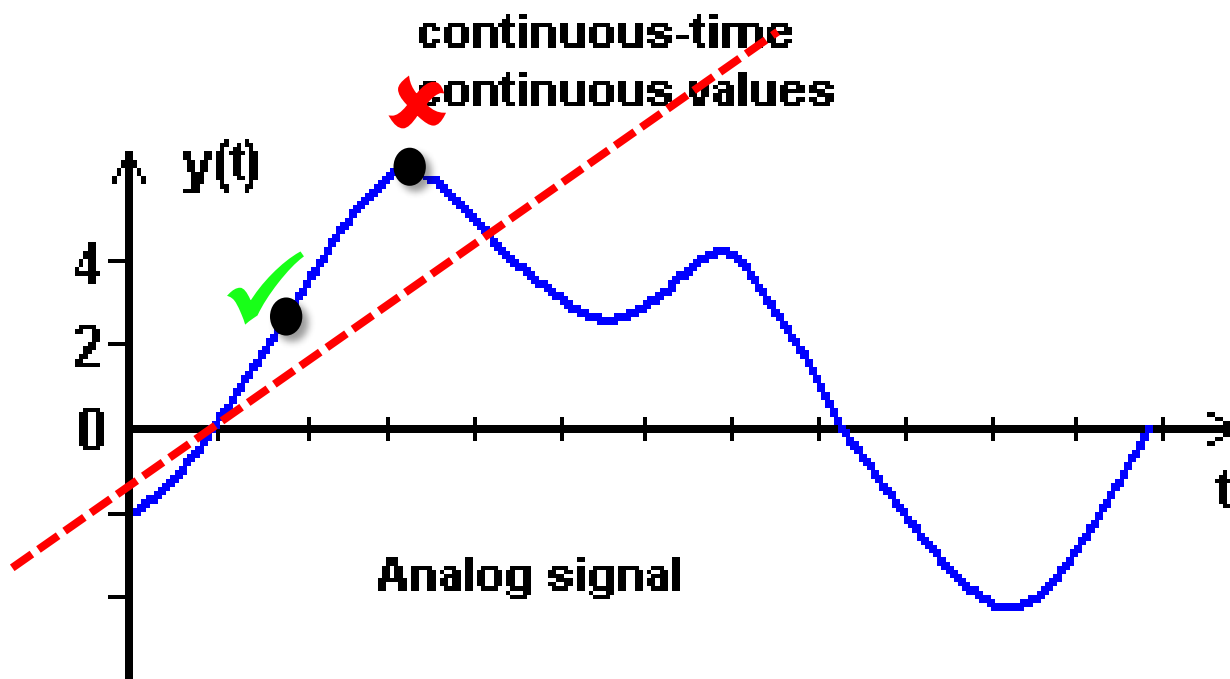
- $abs(T[k] - mean(T[k - l], ..., T[k + l]))$: max
- Square Error regresszióból: min



Outlierek szekvenciákban

- A legkiugróbb pont megtalálása

- $abs(T[k] - mean(T[k - l], ..., T[k + l]))$: max
- Square Error regresszióból: min

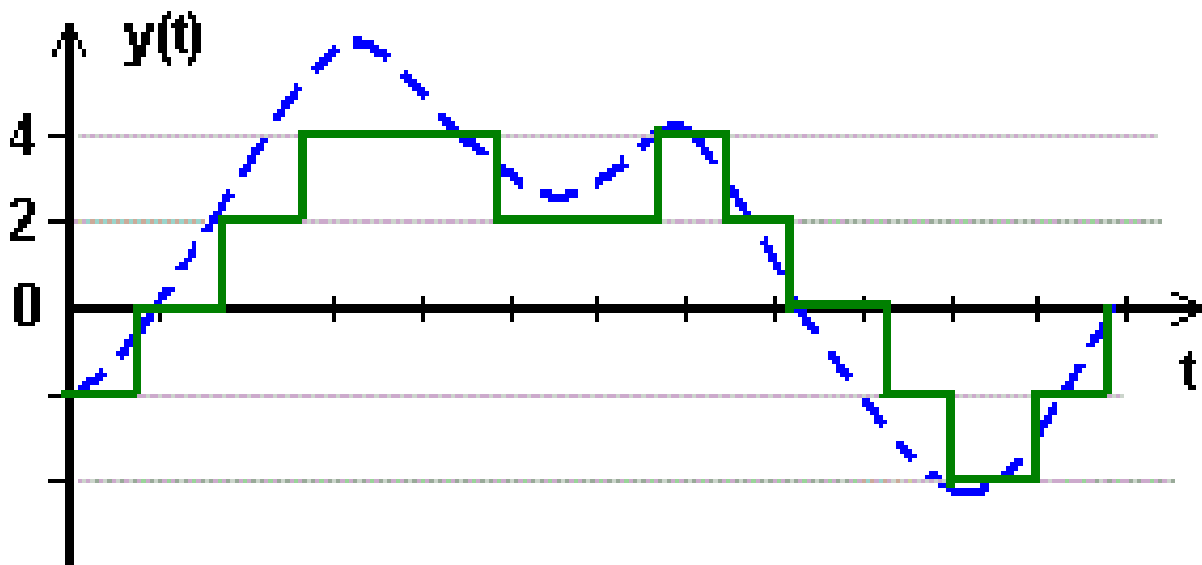


Outlierek *szekvenciákban*

- A legkiugróbb pont megtalálása

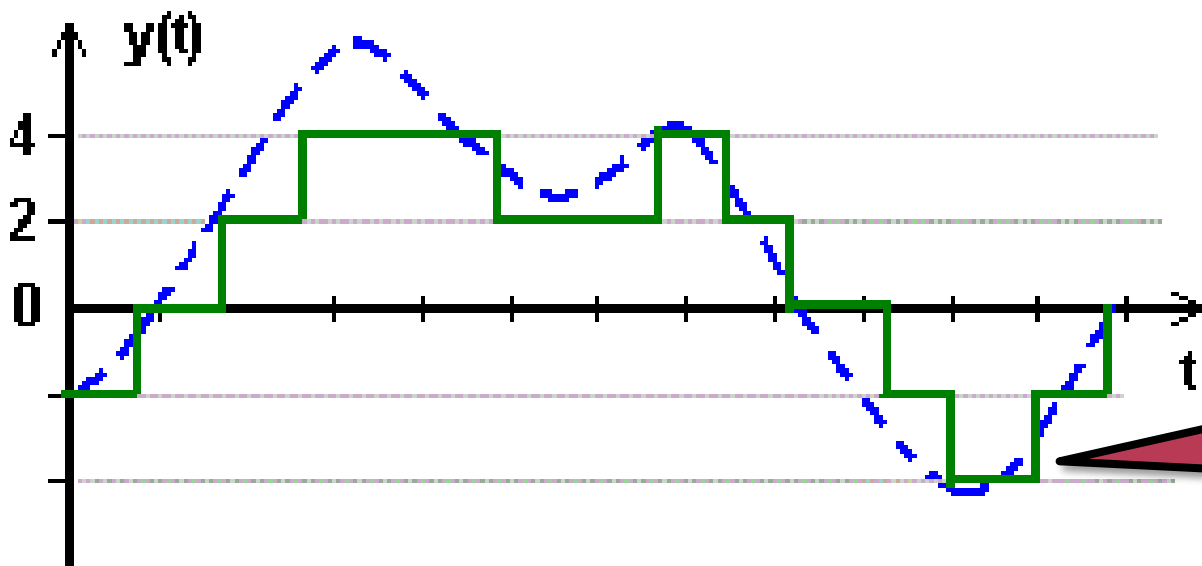
- $abs(T[k] - mean(T[k - l], ..., T[k + l]))$: max
- Square Error regresszióból: min
- A pont törlésével a „minimum description length” a lehető legjobban lecsökken.

Eredeti: 5
különböző érték



Outlierek szekvenciákban

- A legkiugróbb pont megtalálása
 - $abs(T[k] - mean(T[k - l], ..., T[k + l]))$: max
 - Square Error regresszióból: min
 - A pont törlésével a „minimum description length” a lehető legjobban lecsökken.



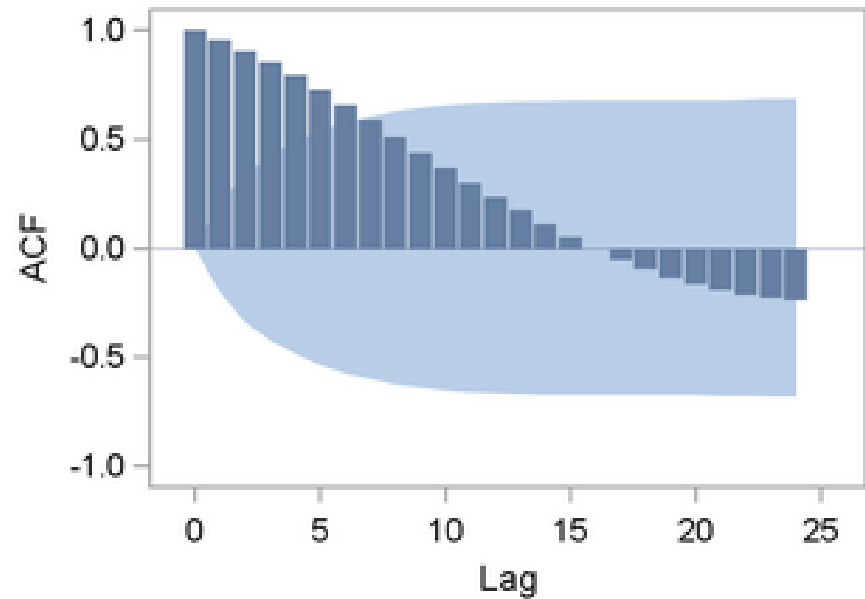
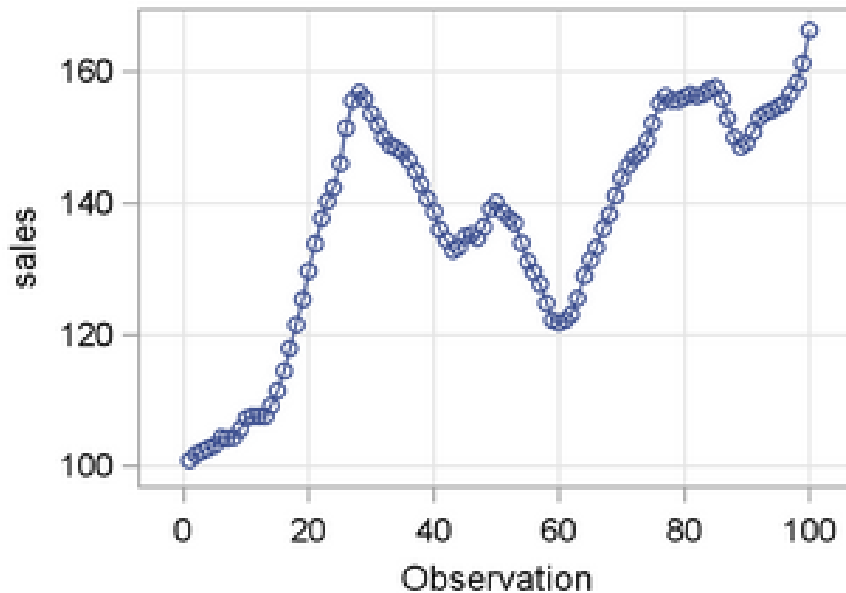
Eredeti: 5
különböző érték

-2 törlése után:
4 különböző
érték is elég

Autokorrelációs módszerek

- Autokorrelációs módszerek
 - Hol térünk el nagyon a prediktált értéktől?
 - Hol változik legjobban az autokorrelációs modell?

Trend and Correlation Analysis for sales



Egy kis kitérő: NNDB

- Felügyelt: feltételezzük, hogy létezik orákulum
- Milyen sorrendben kérdezzük meg tőle a pontokat, hogy a lehető leggyorsabban megtaláljuk a ritkákat?
- Pl.: domain expert leellenőrzi, amit mondunk neki, de minél kevesebbet kelljen manuálisan dolgozni
- Variációk egy témára
 - Mennyi információnk van?
 - Milyen adatunk van? (csak attribútumok? Kapcsolatok is?)

Kiindulási feltételek

- Simaság
 - A többségi osztály eloszlásfüggvénye megfelelően sima
- Kompaktság
 - A ritka osztályba tartozó elemek egymástól vett távolsága kisebb, mint a többségtől vett távolság
- Ami nem kell feltételül: szeparáltság
- Ha nincs: véletlen mintavételezés

Kiindulási feltételek

- Sima

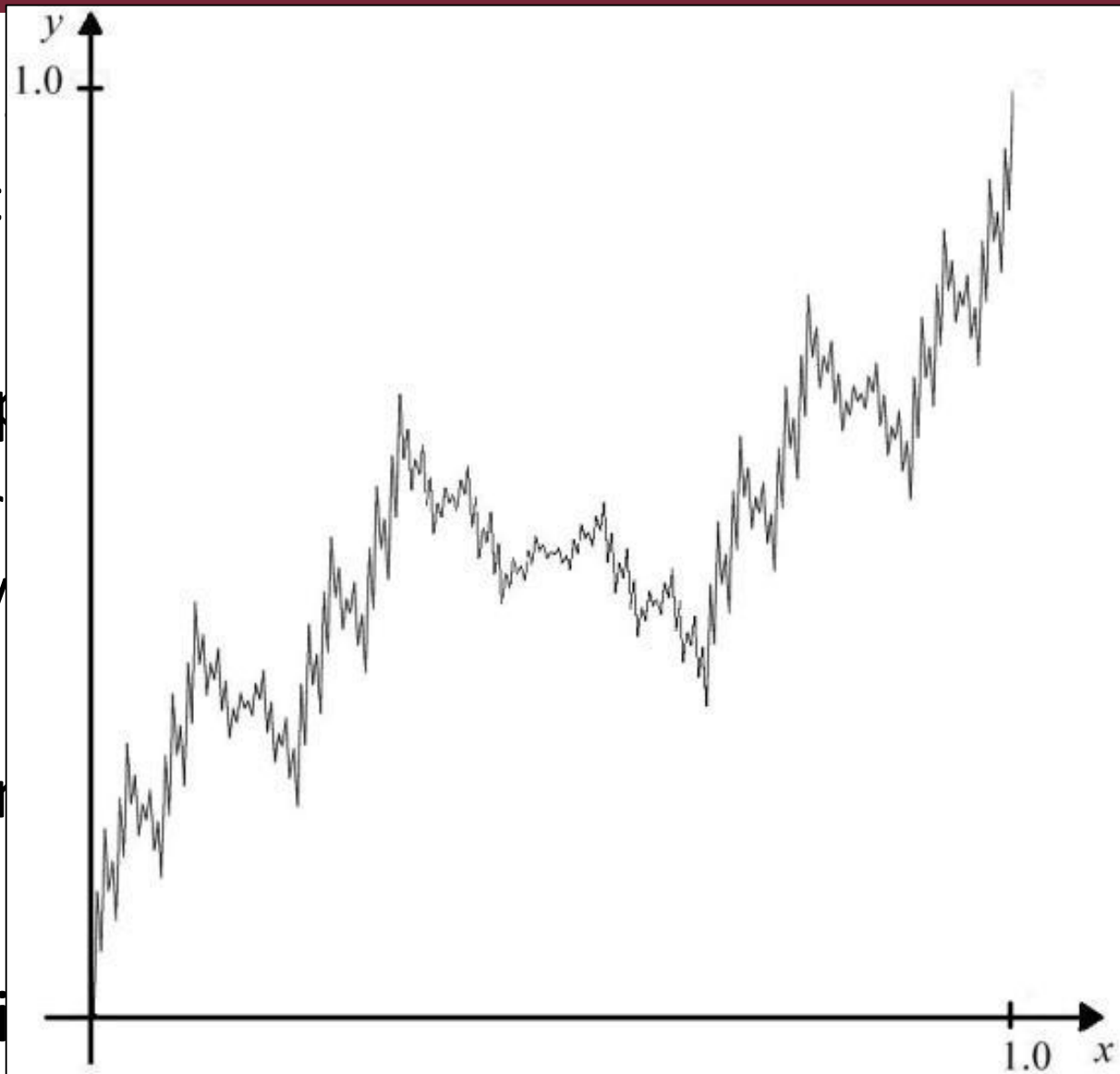
- A t

- Komp

- A r
 - táv

- Ami r

- Ha ni



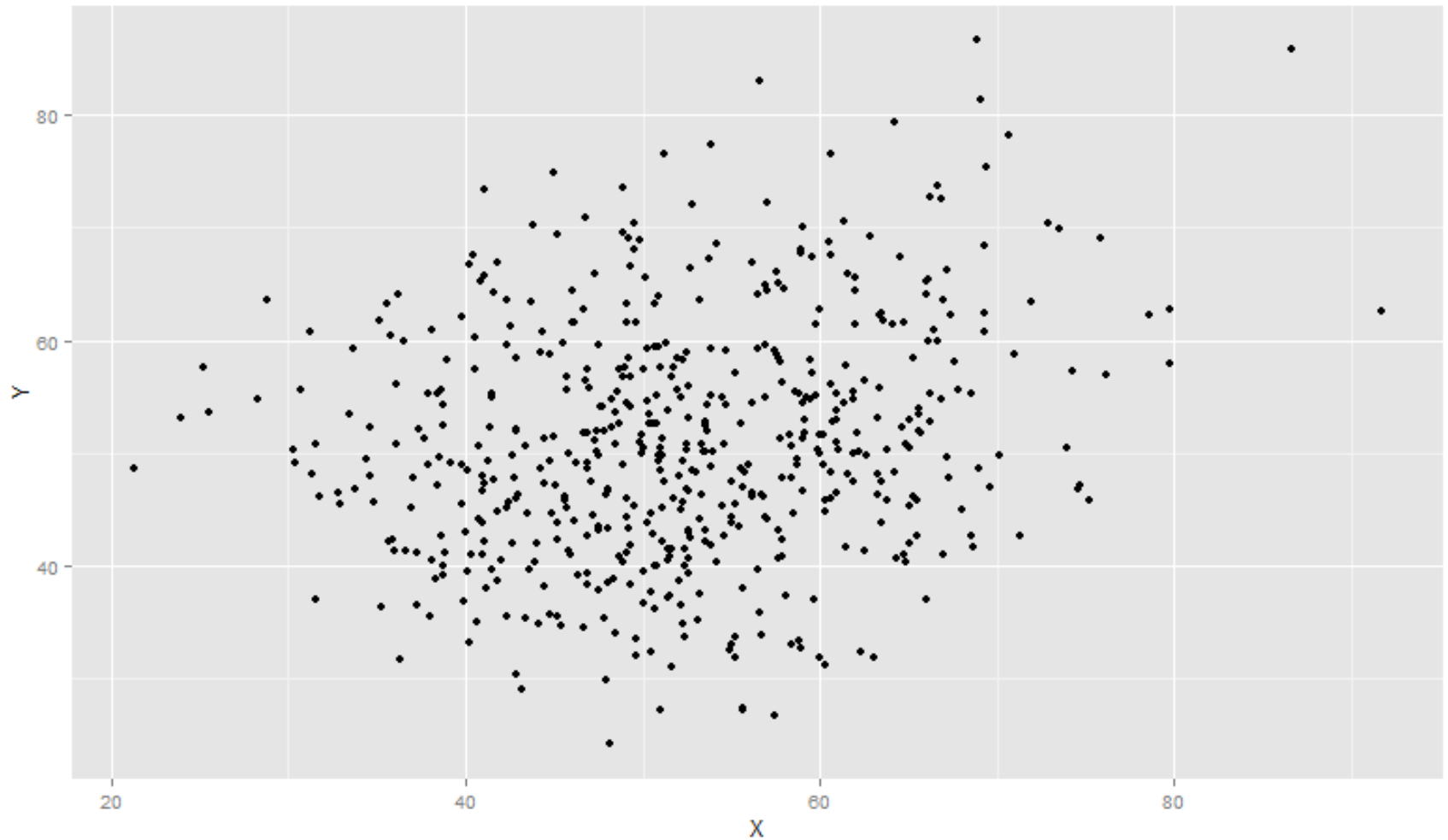
elően sima

vett
olság

Kiindulási feltételek

- Simaság
 - A többségi osztály eloszlásfüggvénye megfelelően sima
 - Matematikája kell?
- Kompaktság
 - A ritka osztályba tartozó elemek egymástól vett távolsága kisebb, mint a többségtől vett távolság
- Ami nem kell feltételül: szeparáltság
- Ha nincs: véletlen mintavételezés

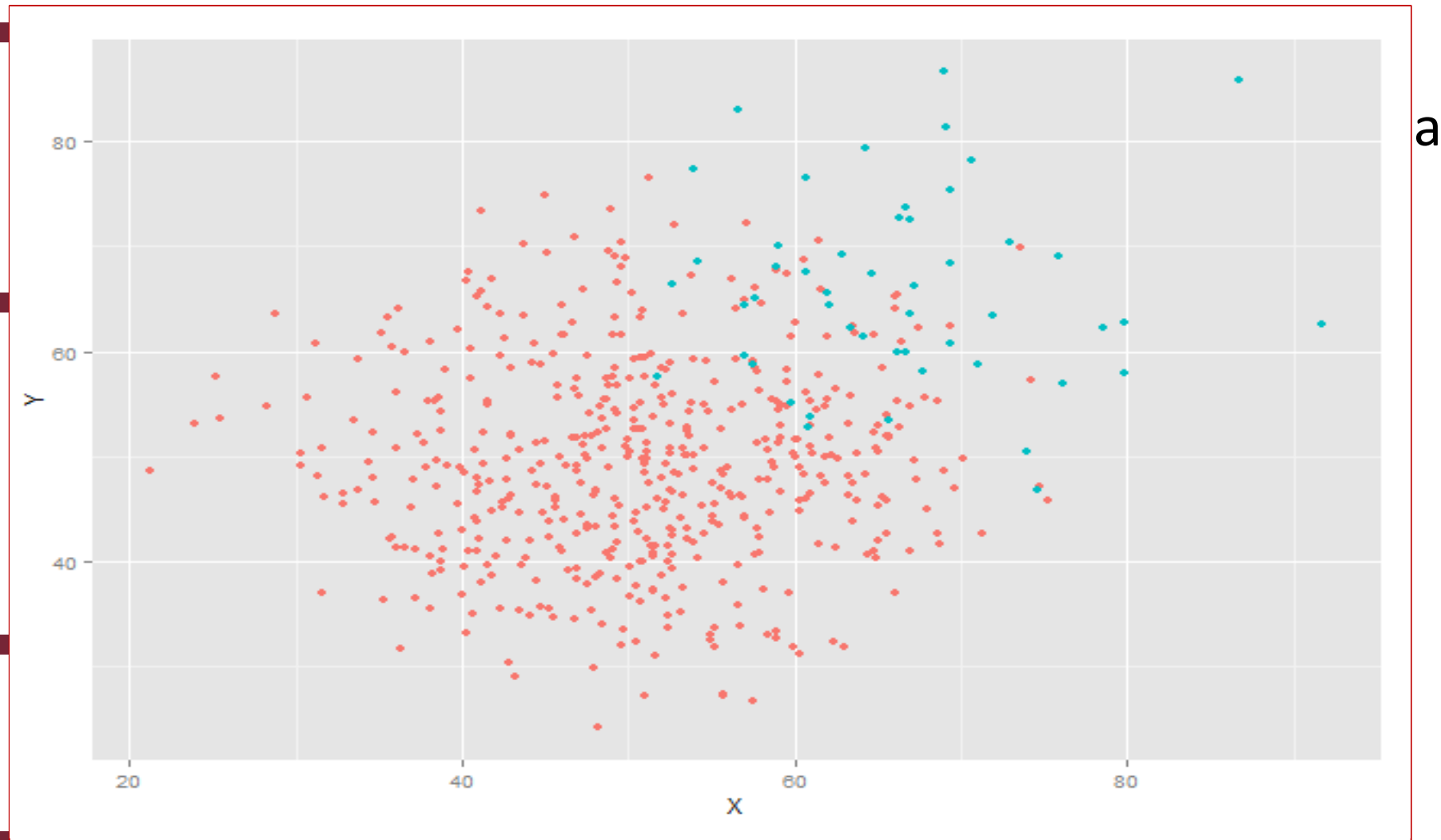
Kiindulási feltételek



a

Ha nincs. véletlen mintavétel

Kiindulási feltételek



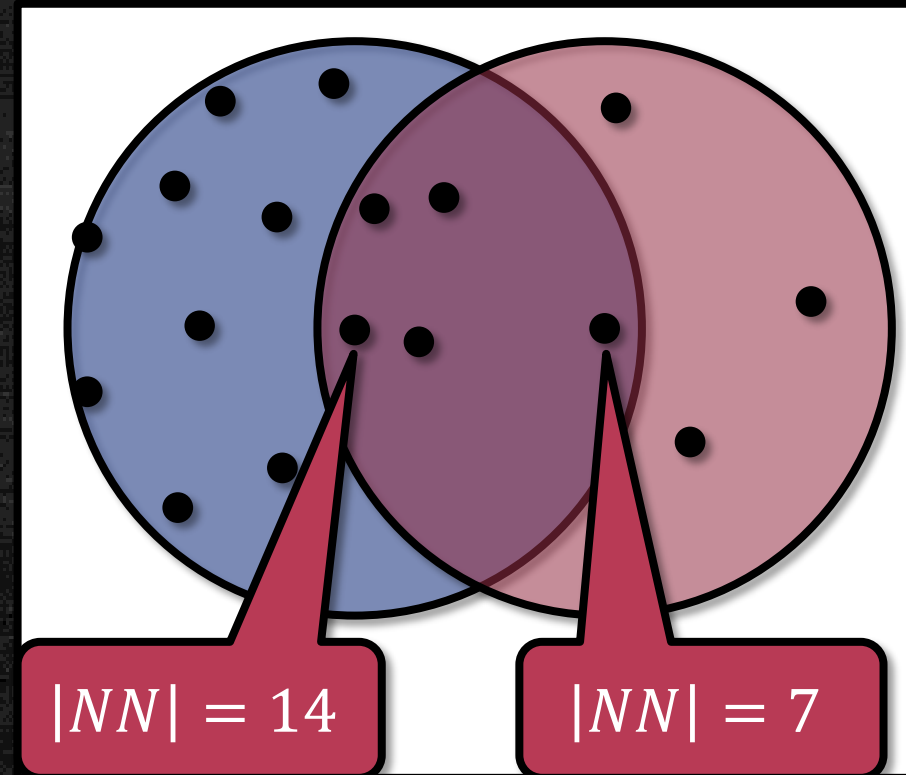
Ha nincs. véletlen mintavételzés

Kiindulási feltételek

- Simaság
 - A többségi osztály eloszlásfüggvénye megfelelően sima
 - Matematikája kell?
- Kompaktság
 - A ritka osztályba tartozó elemek egymástól vett távolsága kisebb, mint a többségtől vett távolság
- Ami nem kell feltételül: szeparáltság
- Ha nincs: véletlen mintavételezés

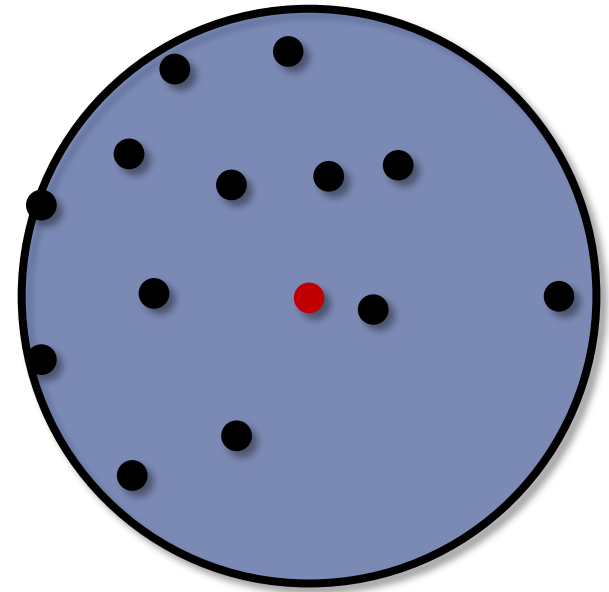
NNDB

1. $\forall i$ -re $NN = \{x \in S, d(i, x) \leq r_i\}$, r_i az adott ciklusban megengedhető maximális sugár
2. $\forall i$ -re $s_i = \max_{x \in NN(i)} |NN(i)| - |NN(x)|$
3. Sejtett ritka elem: i , amire s_i maximális.
4. Ha i ritka, vége.
5. Ha nem,
 $r_{i+1} = (i + 1) \times r_1$,
ugrás 1-re.



Apriori információval – NNDB

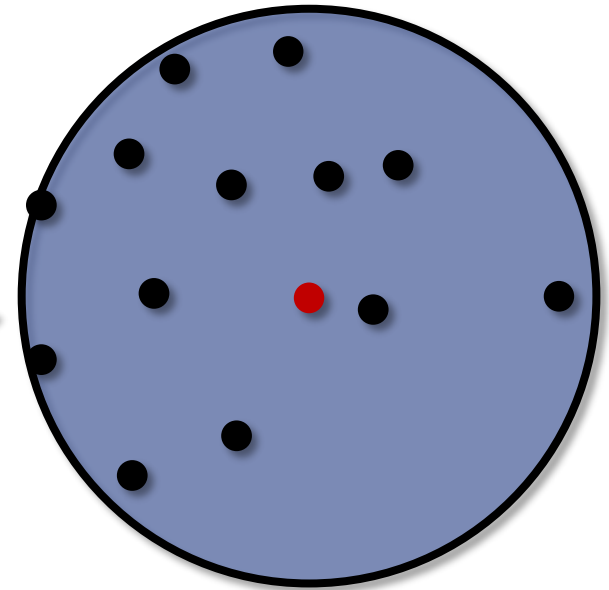
- $r_{i+1} = (i + 1) \times r_1$, na de mekkora legyen r_1 ?
- Ötlet: ha a ritkák aránya $\approx p_2$, akkor legyen $K = p_2 \times |S|$, számítsuk ki $\forall i$ -re a K . legközelebbi elem távolságát: n_i .
- Legyen $r_1 = \min_{i \in S} n_i$.



Apriori információval – NNDB

- $r_{i+1} = (i + 1) \times r_1$, na de mekkora legyen r_1 ?
- Ötlet: ha a ritkák aránya $\approx p_2$, akkor legyen $K = p_2 \times |S|$, számítsuk ki $\forall i$ -re a K . legközelebbi elem távolságát: n_i .
- Legyen $r_1 = \min_{i \in S} n_i$.

Ha a ritkák tényleg nagyon közel vannak egymáshoz, akkor beleférnek egy körbe



Implementációs kérdések

- $kNN(x_i)$, majd $NN(x_i, r')$ – milyen adatszerkezettel?
- - $ort(x)[k]$, $which(x \leq r')$
- Partíciós módszerek?
 - Pl. fák: k-d tree, VP-tree

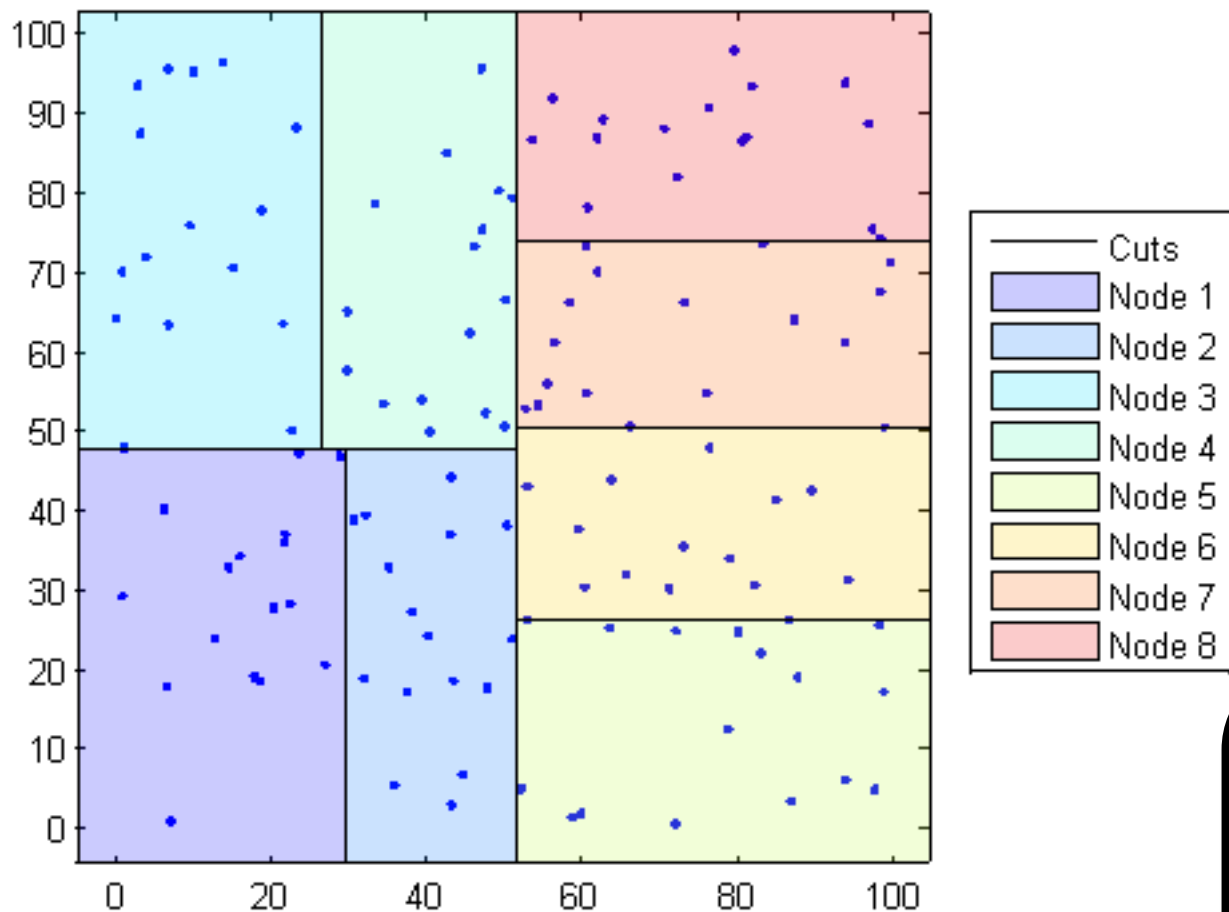
Implementációs kérdések

- $x \leq r'$
- $rr' \leq x \leq r'xxhcciiwwh$ $kNN(x_i)$, majd $NN(x_i, r')$
– milyen adatszerkezettel?
- Naiv
 - Távolságmátrixot tárolunk
 - $sort\ x\ k, [k], which(x \leq r')$
- Partíciós módszerek?
 - Pl. fák: k-d tree, VP-tree

Implementációs kérdések

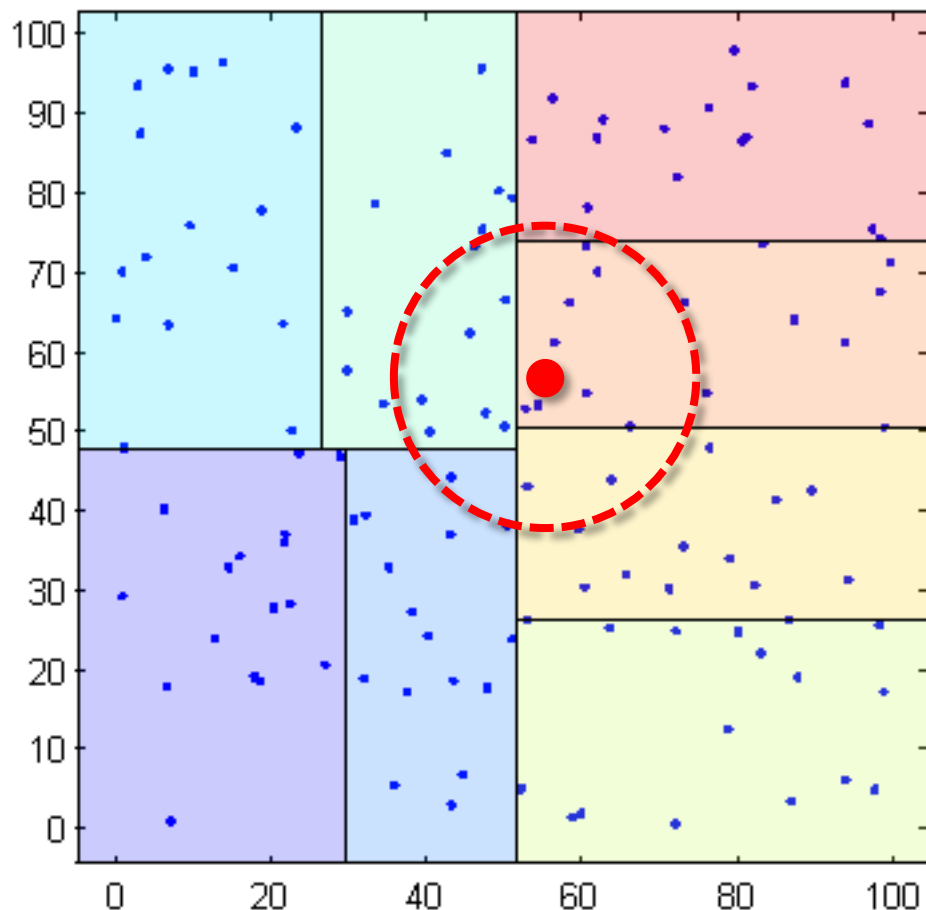
- $x \leq r'$
- $rr' \leq x \leq r'xxhcciiwwh$ $kNN(x_i)$, majd $NN(x_i, r')$
– milyen adatszerkezettel?
- Naiv
 - Távolságmátrixot tárolunk
 - $sort\ x\ k, [k], which(x \leq r')$
- Partíciós módszerek?
 - Pl. fák: k-d tree, VP-tree
 - Pl. fák: k-d tree, VP-tree

Implementációs kérdések



1. ÉPÍT
Hierarchikus
adatszerkezetben a
közeli ponthalmazok

Implementációs kérdések



≤ 9 (7. zóna)

$= 10$

2.1 $r' \leq 9$ (7. zóna)

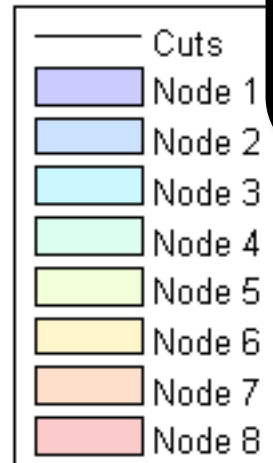
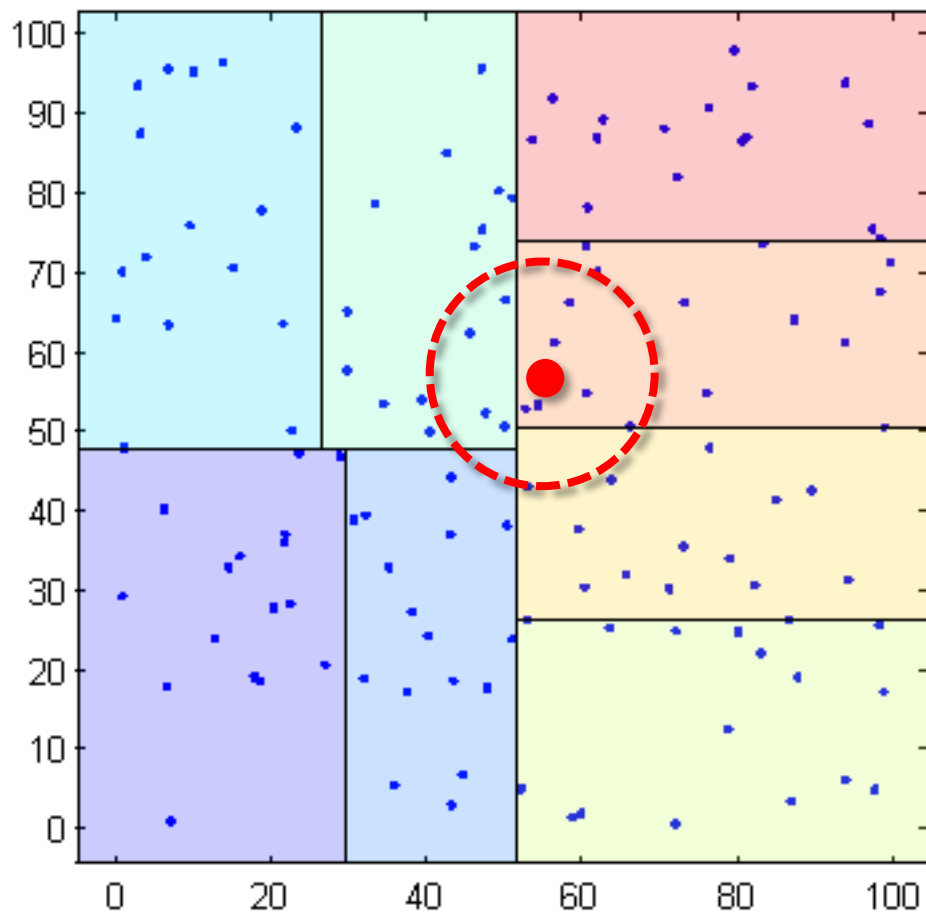
2.1 $r' \leq 9$ (7. zóna)

2.2 $r' = 7$ (5-8. zóna)

1. ÉPÍT

Hierarchikus
adatszerkezetben a
közeli ponthalmazok

Implementációs kérdések



≤ 9 (7. zóna)

$= 10$

2.2 $r' = 7$ (5-8. zóna)

2.1 $r' \leq 9$ (7. zóna)

2.2 $r' = 7$ (5-8. zóna)

1. ÉPÍT

Hierarchikus
adatszerkezetben a
közeli ponthalmazok

Implementációs kérdések



Nem kell mindent kiszámolni 😊
Többször kell kiszámolnunk ugyanazt 😞

≤ 9 (7. zóna)
 $= 10$
 $2.2 \ r' = 7$ (5-8. zóna)
 $2.1 \ r' \leq 9$ (7. zóna)

Metaregionális
adatszerkezetben a
közeli ponthalmazok

Map-Reduce?

- n elég nagy $\rightarrow$ muszáj bontani

Map-Reduce?

- n elég nagy $\rightarrow$ muszáj bontani

MAP

Csomópont milyen más csp-
ok kNN-jeit frissítheti?

REDUCE

Ha megvan minden jelölt:
tényleges távolságszámítás

Map-Reduce?

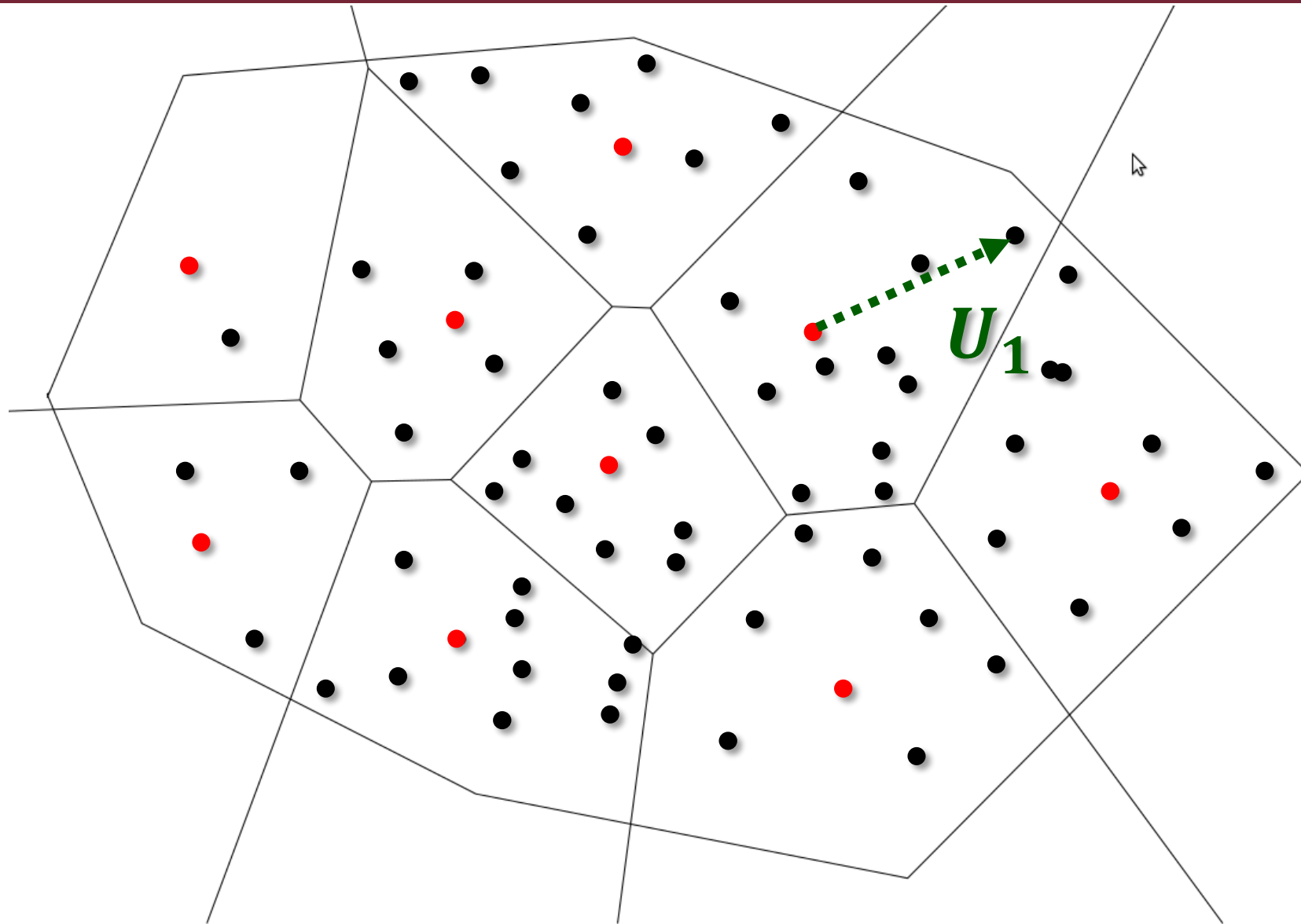
- n elég nagy $\rightarrow$ muszáj bontani

Mi van, ha már a felosztást is elosztottan akarom végezni?

REDUCE

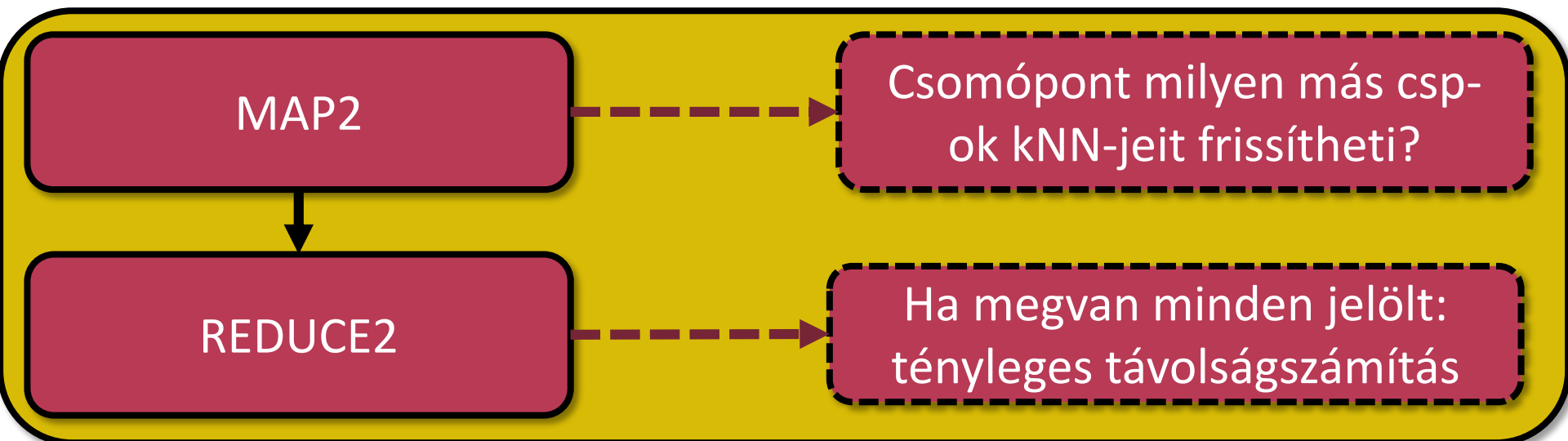
Ha megvan minden jelölt:
tényleges távolságszámítás

Egy kis család.. Voronoi cellák



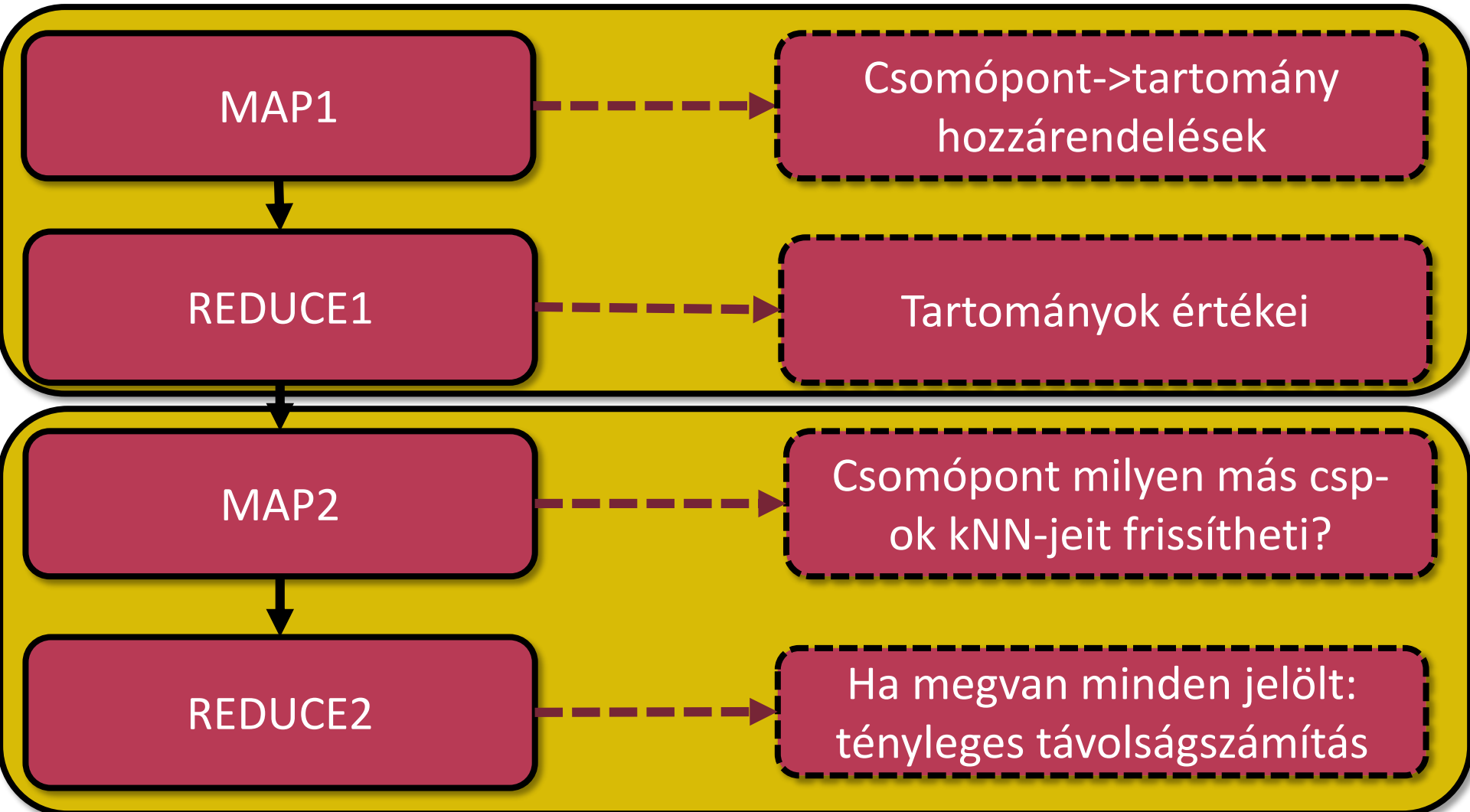
Amiért jó: MapReduce ☺

- n elég nagy $\rightarrow$ muszáj bontani



Amiért jó: MapReduce ☺

- n elég nagy $\rightarrow$ muszáj bontani



Hivatkozásjegyzék

- [1] Stream Processing, filtering: Mining of Massive Data Sets
 - Alapmű:
<http://infolab.stanford.edu/~ullman/mmds/book.pdf>
 - Coursera tárgy:
<https://www.coursera.org/course/mmds>
- [2] Outlier Detection
 - Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. ACM Computing Surveys (CSUR), 41(3):15, 2009