Adatelemzés

2017-2018 őszi félév

2018.03.19,26,27

1. Bevezetés
2. Osztályozás és klaszterezés feladata
3. Numpy és Ipython
4. Logisztikus regresszió

Elérhetőségek

Daróczy Bálint

daroczyb@ilab.sztaki.hu

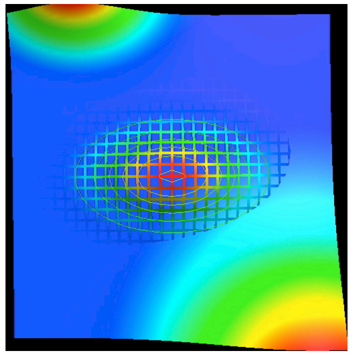
Személyesen: MTA SZTAKI, Lágymányosi u. 11

Tantárgy honlapja:

http://cs.bme.hu/~daroczyb/Adatelemzes2018/

MTA SZTAKI

DMS



Következő hetek

Március 19: bevezetés, osztályozás-klaszterezés, kiértékelés, kNN

Március 26-27: logisztikus regresszió, SVM, kernelek

Április 2: húsvét

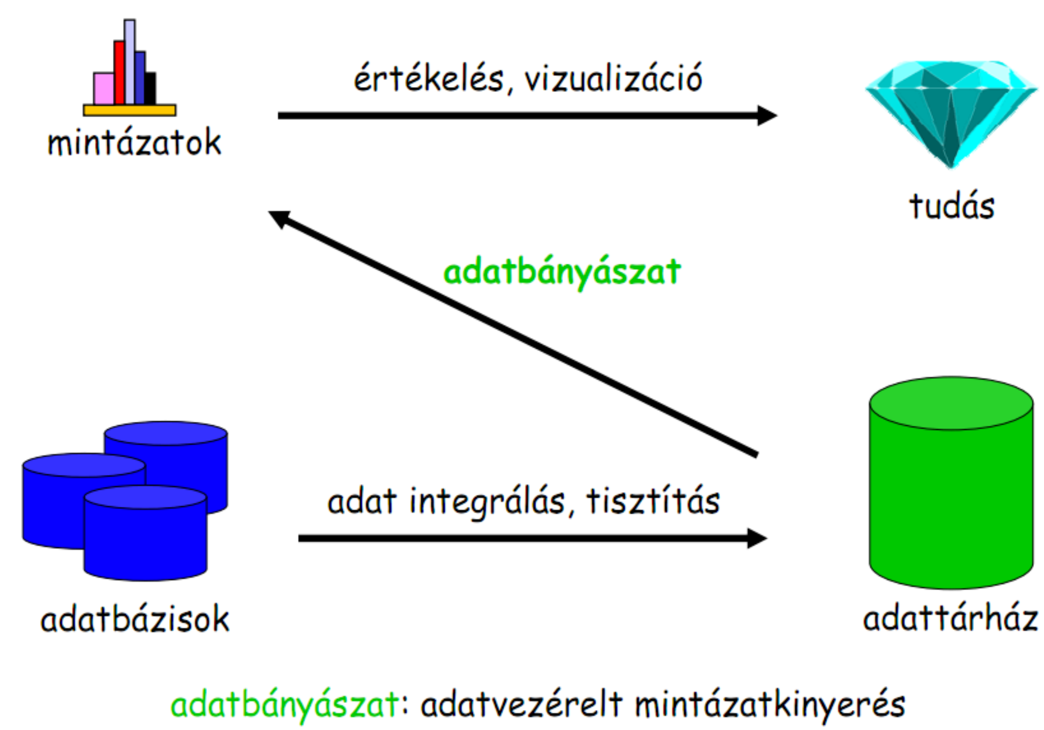
Április 9-10: MLP, CNN, RNN

Április 16: Bayes hálózatok, Boosting

Adatbányászat feladatai

Adatbányászat (Data Mining):

- nagy adathalmazokon
- nem triviális (implicit, eddig ismeretlen)információ
- Struktúrális minták kinyerése



Knowledge discovery in databases (KDD)

- ismeretszerzés adatbázisokból
- adat/mintázat vizsgálata
- összefüggések azonosítása
- stb.

Kép: Lukács András

Kicsit bővebben

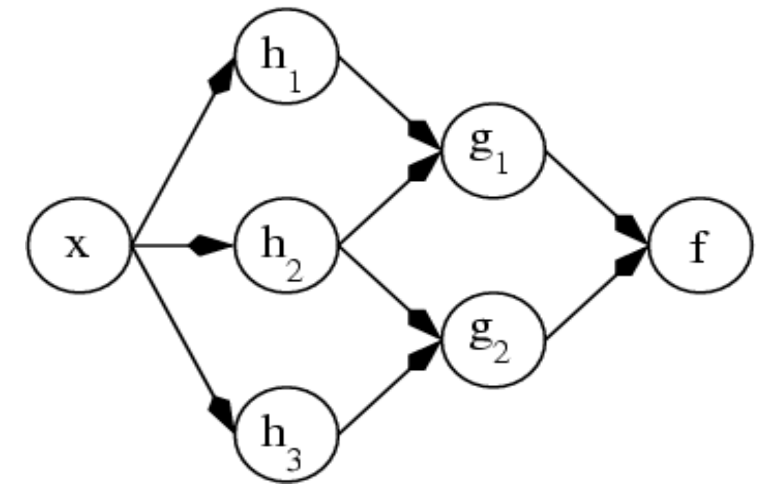
Jobbról balra......

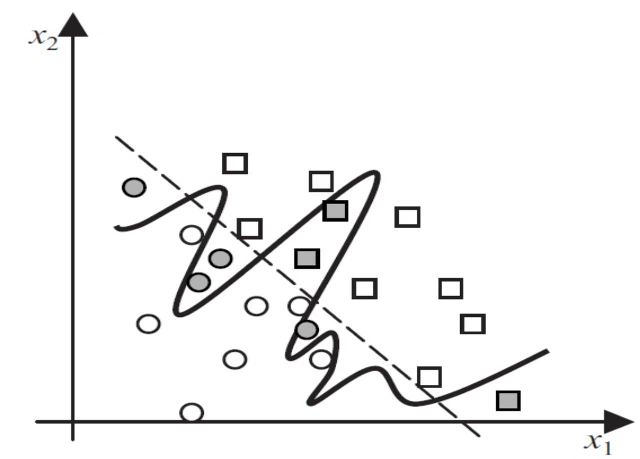
1. Keresés (IR)
   - szöveg, kép, videó, hang stb.
2. Ajánló rendszer (RecSys)
   - user-item ajánlás
   - link-prediction
3. Szociális hálózatok
   - de-anonymizálás
   - barát/termék ajánlás
   - reklám mérő
4. Klasszifikációs problémák
   - szöveg/kép/videó/hang annotálás
   - spam detekció
   - bioinformatika
5. Big Data
   - Entity Resolution
   - log elemzés

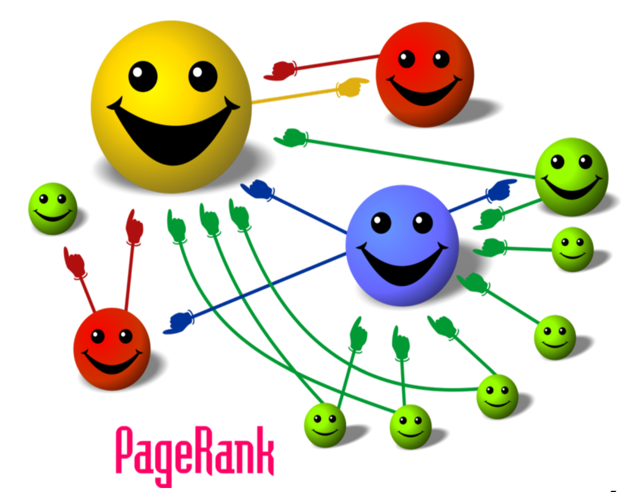
Kicsit bővebben

Balról jobbra......

1. Klasszifikációs problémák:
   - döntési fák, fa alapú algoritmusok
   - Bayes network
   - perceptron, logistic regression
   - Artificial Neural Networks
   - Deep Learning
   - Support Vector Machine
   - Kiértékelés
2. Klaszterezés:
   - K-means, GMM, X-means
   - DBSCAN, OPTICS, Single-link
3. Attribútum és feature szelekció
   - SVD (PCA), IG
4. Hálózati problémák:
   - HITS, PageRank, Influence
5. Ajánlórendszerek:
   - RBM, MF, SVD
6. Asszociációs szabályok







+ GPU-k és elosztott rendszerek (pl. OpenCL, CUDA, GraphLab)

Kép: wikipedia

Egyéb példák

Web és egyéb hálózat:

- keresés (szöveg,kép,videó,honlap,információ)
- ajánlás (reklám,cikk,kép…)
- log elemzés (telefonhálózat, ügyfélhálózat stb.)

Üzleti intelligencia:

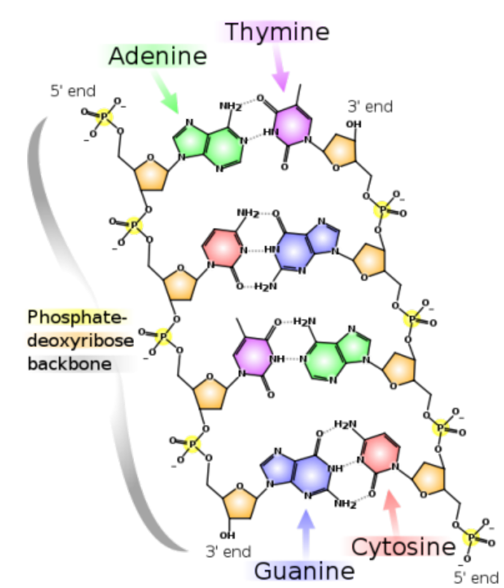
- döntés segítő rendszerek (tőzsde,biztosítás)
- összefüggés keresés (hasonló ügyfél, hasonló megoldás)

Multimédia:

- autómatikus annotáció (kép,videó,zene stb.)
- hangfelismerés és beszédfelismerés

Orvosi kutatás és diagnosztika:

- szűrés (rák, TBC stb)
- élettani folyamatok elemzése
- génkutatás (génszekvenciák keresése, azonosítása, modellezése)
- gyógyszerkutatás



Ref: Lukács András

Irodalom (ajánlott)

1. Tan, Steinbach, Kumar (TSK): Introduction to Data Mining
   Addison-Wesley, 2006, Cloth; 769 pp, ISBN-10: 0321321367, ISBN-13: 9780321321367
   http://www-users.cs.umn.edu/~kumar/dmbook/index.php

2. Bodon: Adatbányászati algoritmusok http://www.cs.bme.hu/~bodon/magyar/adatbanyaszat/tanulmany/index.html

3. Leskovic, Rajraman, Ullmann: Mining of Massive Datasets
   http://infolab.stanford.edu/~ullman/mmds.html

4. Devroye, Győrfi, Lugosi: A Probabilistic Theory of Pattern Recognition, 1996

5. Rojas: Neural Networks, Springer-Verlag, Berlin, 1996

6. Hopcroft, Kannan: Computer Science Theory for the Information Age
   http://www.cs.cmu.edu/~venkatg/teaching/CStheory-infoage/hopcroft-kannan-feb2012.pdf

\+ cikkek

Adatbányászati szoftverek

Weka - open source java

DATO, Flink stb.

RapidMiner

Clementine (SPSS)

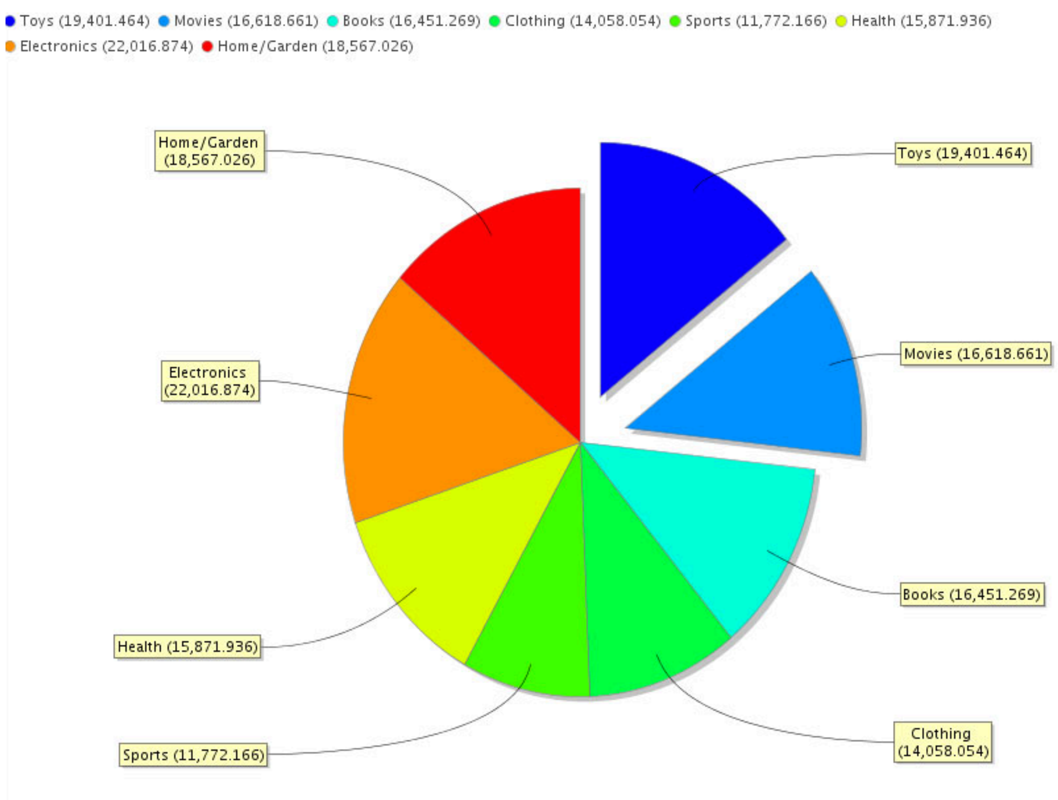
Darwin (Oracle)

SAS Enterprise Miner

IBM Intelligent Miner

+ Chainer, Tensorflow stb.

+ OpenSource libraries (R, Scikit learn Python)



Reprezentáció

- Adathalmaz objektumok halmaza, mely tartalmazza az attribútumokat is
- Reményeink szerint az attribútumok megfelelően reprezentálják az adott objektumot
- Attribútumok típusai lehetnek:
  - pl. bináris
    - nominális
    - numerikus

Attribútumok, “feature”, változók

Adat objektumok, “record”

Tid	**Refund**	**Marital Status**	**Taxable Income**	**Cheat**
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**

Reprezentáció

Egy adathalmaz lehet:

- rendezhető
    - időbeli vagy soros
    - térbeli
- ritka vagy sűrű

Pár kikötés:

- fix attribútumhalmaz
- előre meghatározott attribútumok

Attribútumok, “feature”, változók

Adat objektumok, „record”

Tid	**Refund**	**Marital Status**	**Taxable Income**	**Cheat**
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**

Gépi tanulás

Legyen adott egy T elemű ismert halmaz R^d-ben, $X=\{x_1,..,x_T\}$ és minden eleméhez egy y érték.

Keressük az olyan f(x) függvényeket melyek jól közelítik y-t.

Sok a kérdés:

Hogyan határozzuk meg a közelítés minőségét?
Milyen függvényosztályok között keresünk?
Hogyan találunk egy adott függvényosztályon belül megfelelő példányt?
Komplexitás?
Generalizáció?
stb.

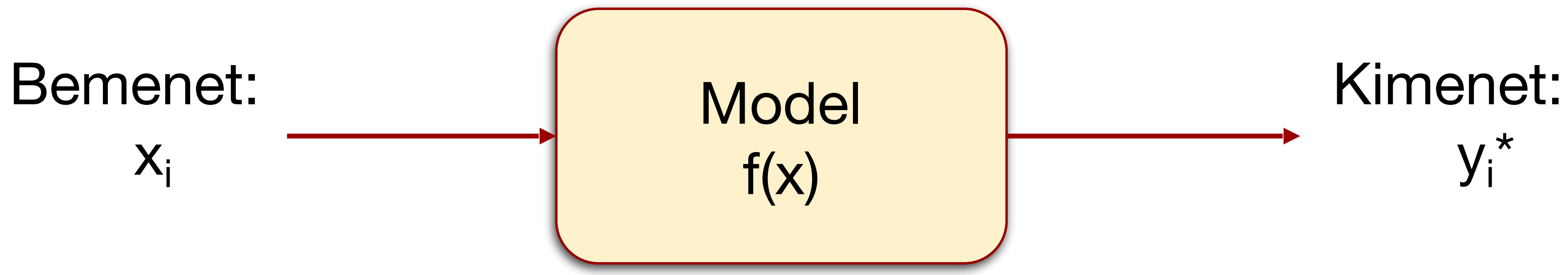
Osztályozás vs. generatív modellek

Gyakorlatban, ha a célváltozónk bináris vagy egy véges halmaza eleme akkor osztályozásról beszélünk.

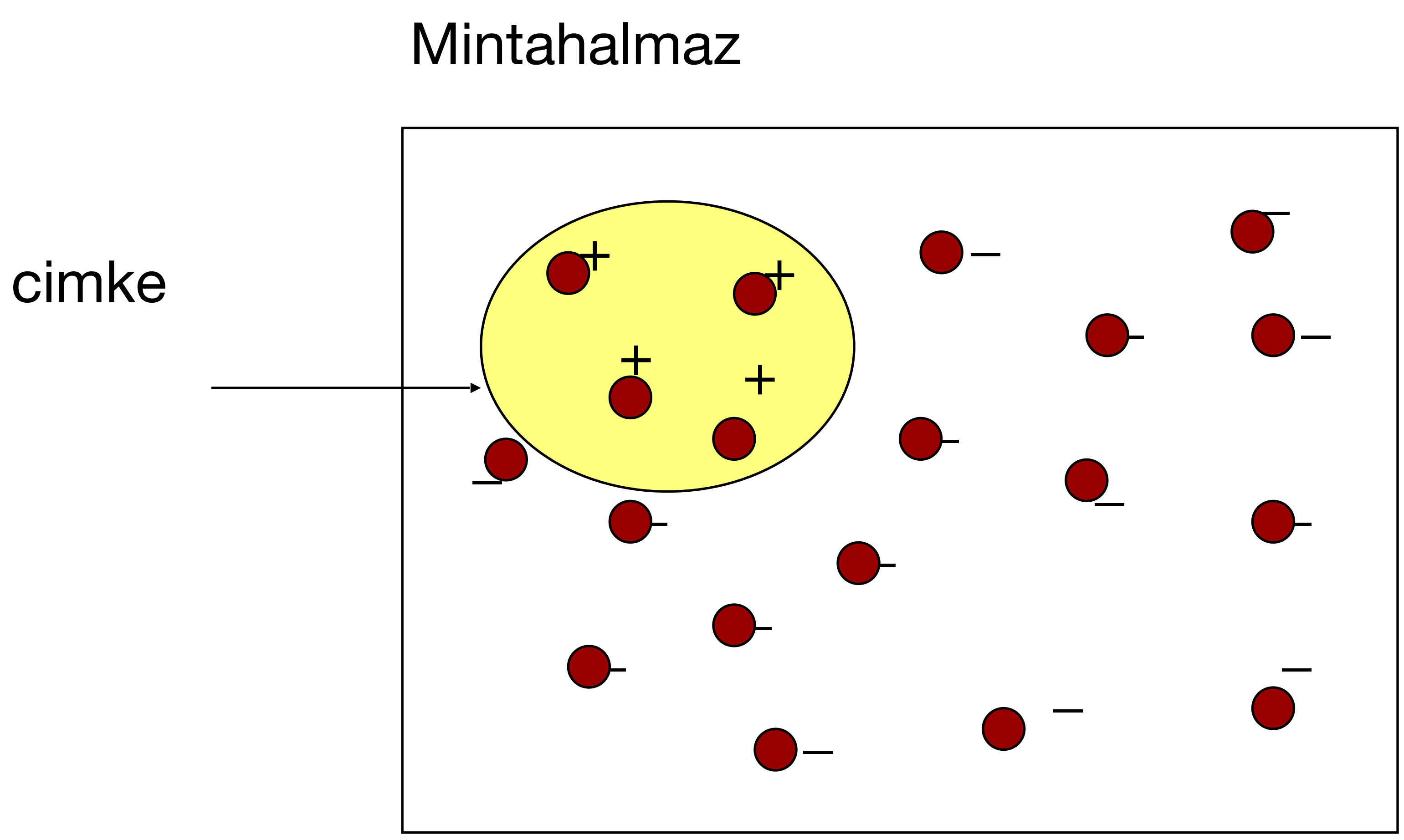
Nominális -> Bináris?

Megj.: A tanulás fontos része a visszajelzés.

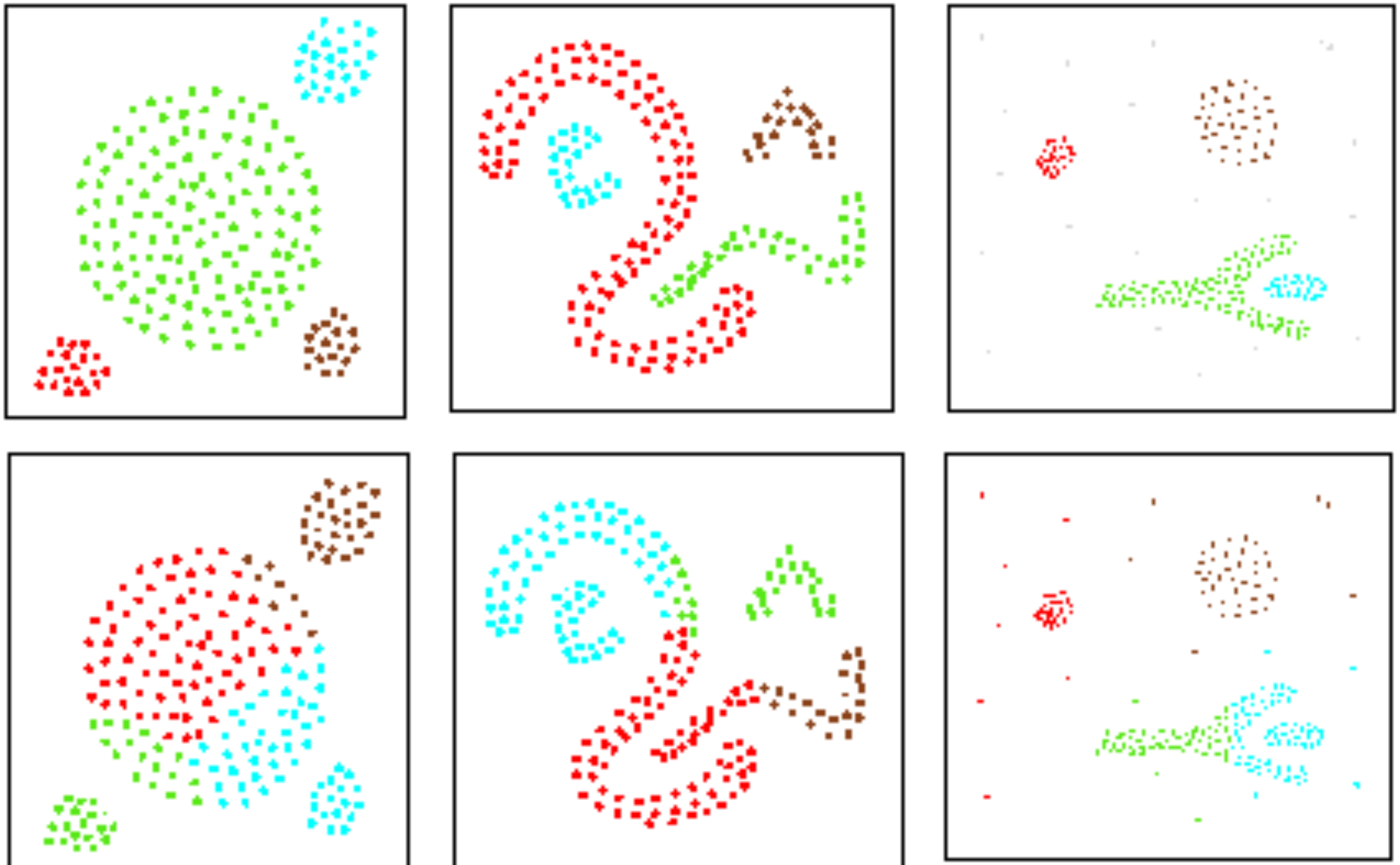
Generatív modellek?



Osztályozás



Klaszterezés



Ábra: TSK

K-Means

Feltesszük, hogy az adatpontjaink egy valós vektortérben helyezkednek el.

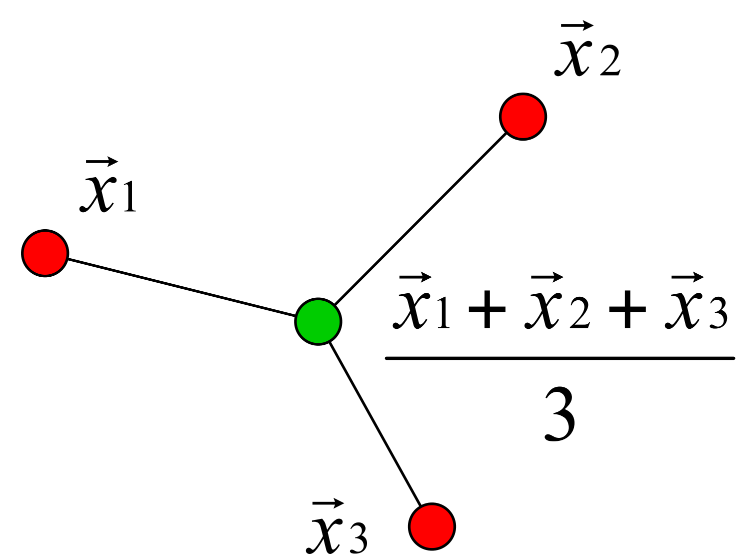
A klasztereket a középpontjuk határozza meg.

K-means (D; k)

    Adott egy C_1, C_2,... , C_k centroidok tetszőleges k elemű kezdeti halmaza
    Amíg változik a centroidok halmaza:
        minden d mintát soroljuk a hozzá legközelebbi centroid klaszterébe
        Módosítsuk a meglévő centroidokat
        C_i legyen C_i-be tartozó elemek középértéke

A kezdőpontok meghatározása lehet:

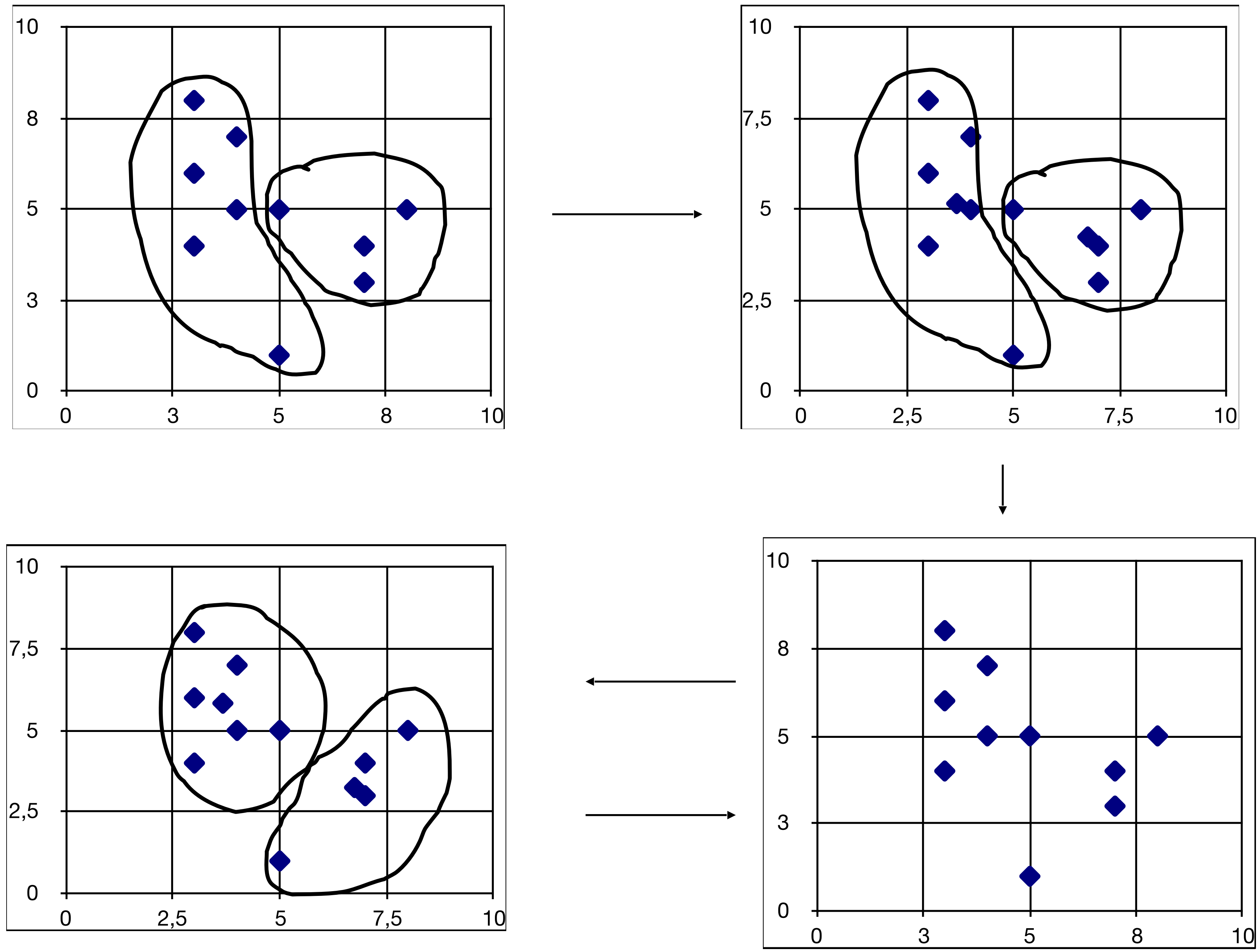
a) véletlen pontokból
b) véletlen választott tanulópontokból



Az algoritmust megállítjuk, ha:

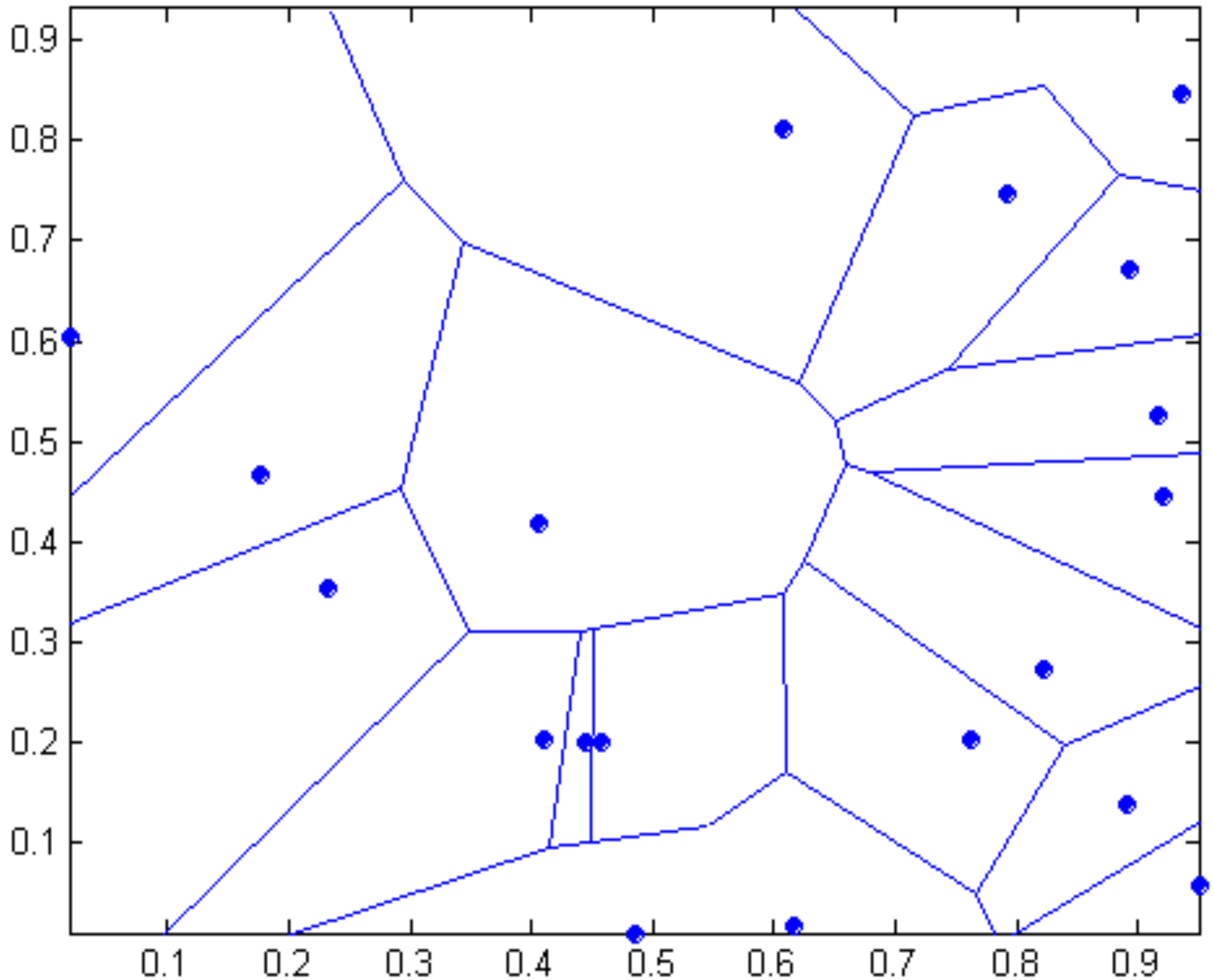
a) a klaszterezés nem változik
b) a küszöbhibánál kisebb az approx. hiba
c) elértük a maximális iterációk számát

K-means



Ábra: TSK

K-means



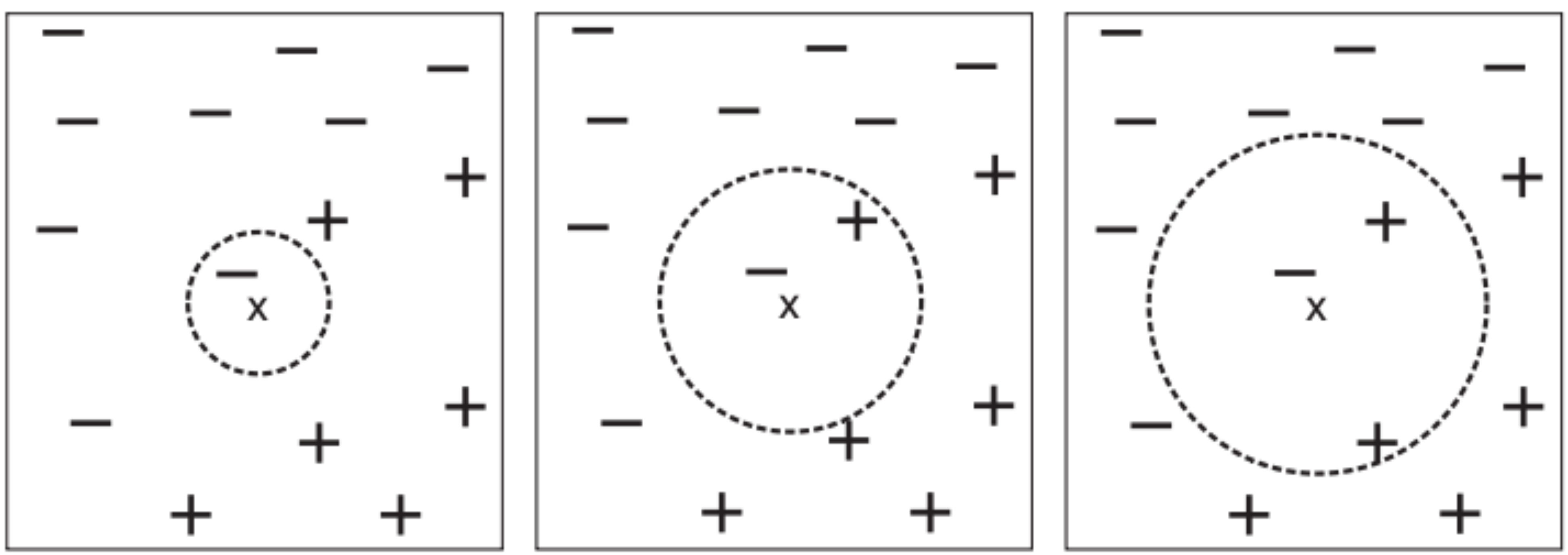
K- nearest neighbor (K-NN)

Alapelv:

“Úgy megy mint egy kacsa, úgy úszik mint egy kacsa, úgy eszik mint egy kacsa, tehát kacsa!”

1. Minden tesztadathoz keressük meg a K legközelebbi elemet.
2. A legtöbbet reprezentált osztályt jósoljuk az adott tesztaadatnak

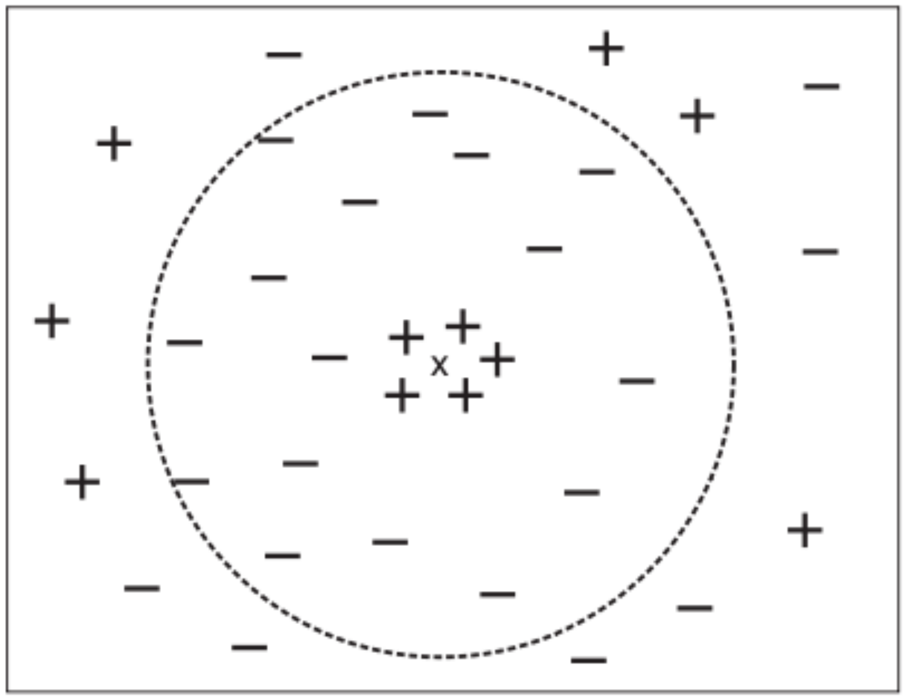
Példa:



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

Ábra: TSK

K- nearest neighbor (K-NN)



Mi a gond?

- mohó (eager) algoritmus: kész modell készítése amint a tanuló adat létezik → többé nincs szükség a tanulóhalmazra
- lusta (lazy) algoritmus: kész jóslatot a tanulóhalmaz segítségével készít → szükséges a tanulóhalmaz a jósláshoz

A K-NN lusta algoritmus :

Méret?
Bonyolultság?
Skálázhatóság?

Ábra: TSK

K- nearest neighbor (K-NN)

Távolság mértékek (pl.)

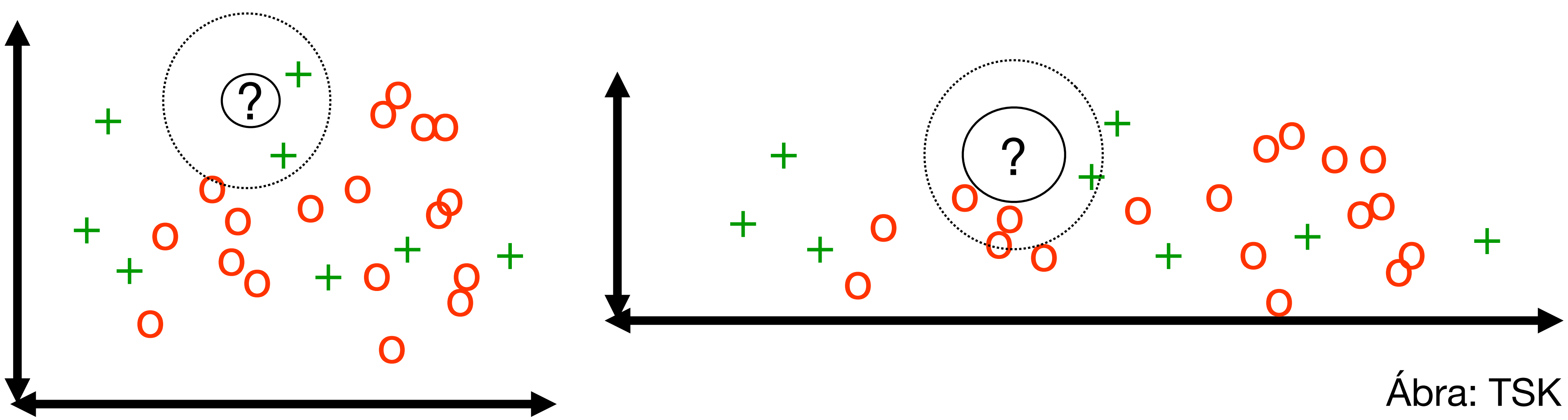
- Minkowski távolság

- Mahalanobis $D_M(x) = \sqrt{(x-\mu)^T S^{-1} (x-\mu)}.$

- Cosine, Jaccard, Kullback-Leibler, Jensen-Shannon stb.

Problémák:

- skála
- autómatikus távolság választás
- normalizáció



Ábra: TSK

Kiértékelés

Confusion matrix:

Alapigazság/ predikció	p	n	Total
p	True Positive (TP)	False Negative (FN)	TP+FN
n	False Positive (FP)	True Negative (TN)	FP+TN
Total	TP+FN	FP+TN	

Kiértékelés

Accuracy: a helyesen klasszifikálás valószínűsége

TP+TN/(TP+FP+TN+FN)

Precision (p): egy relevans dokumentum helyes klasszifikálásának valószínűsége

TP/(TP+FP)

Recall (r): annak a valószínüsége, hogy egy releváns dokumentumot helyesen klasszifikálunk

TP/(TP+FN)

F-measure: a precision és a recall harmónikus közepe (2*p*r/(p+r))

Kiértékelés

False-Positive Rate (FPR) = FP/(FP+TN)

True-Positive Rate (TPR) = TP/(TP+FN)

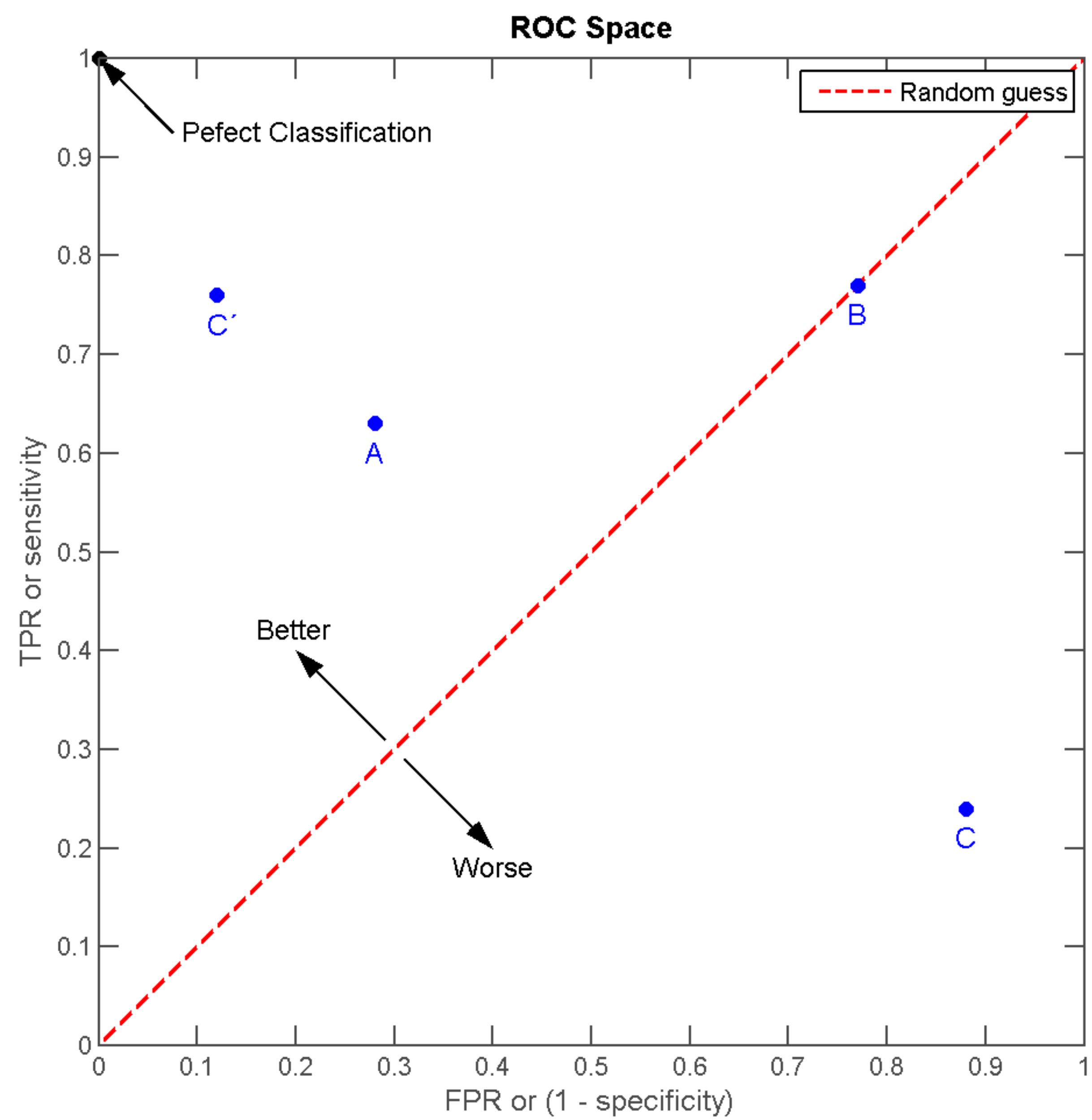
ROC: Receiver Operating Characteristic

MAP: Mean Average Precision

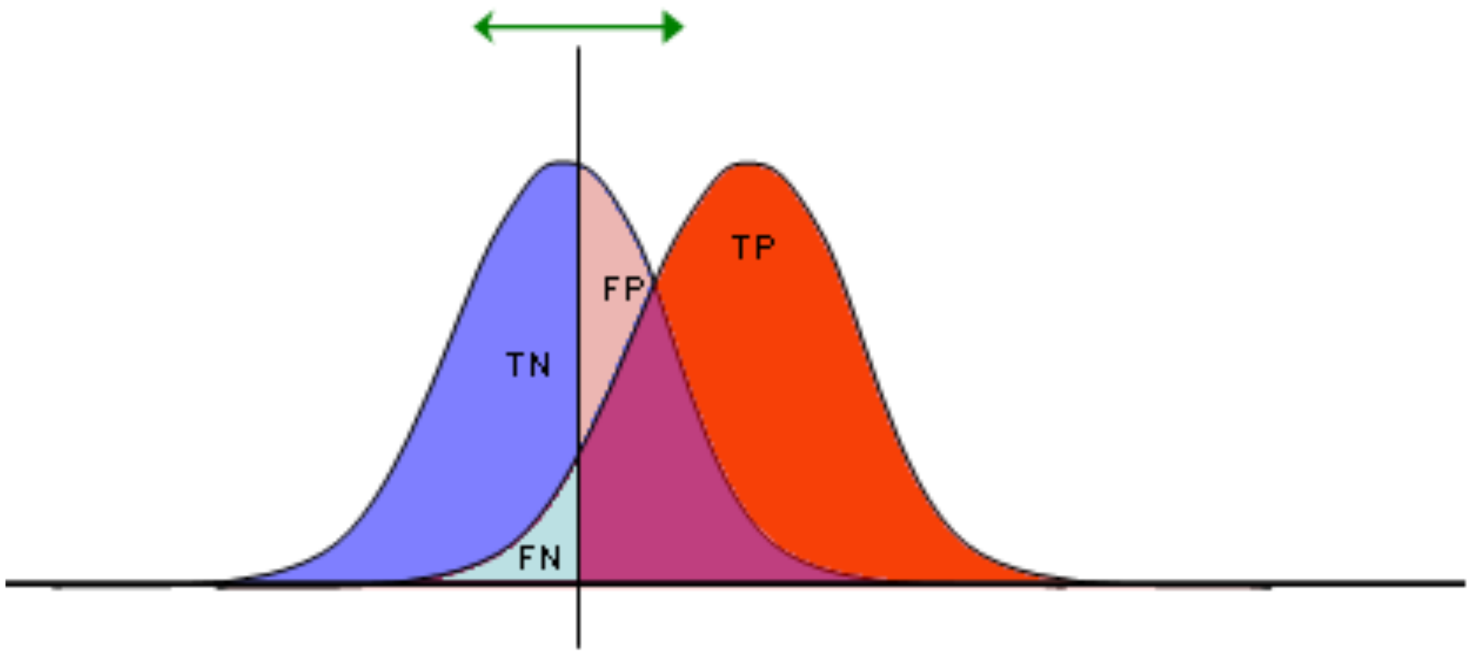
$$\mathrm{AveP} = \frac{\sum_{r=1}^{N}(P(r) \times \mathrm{rel}(r))}{\text{number of relevant documents}}$$

nDCG: normalized Discriminative Cummulative Gain

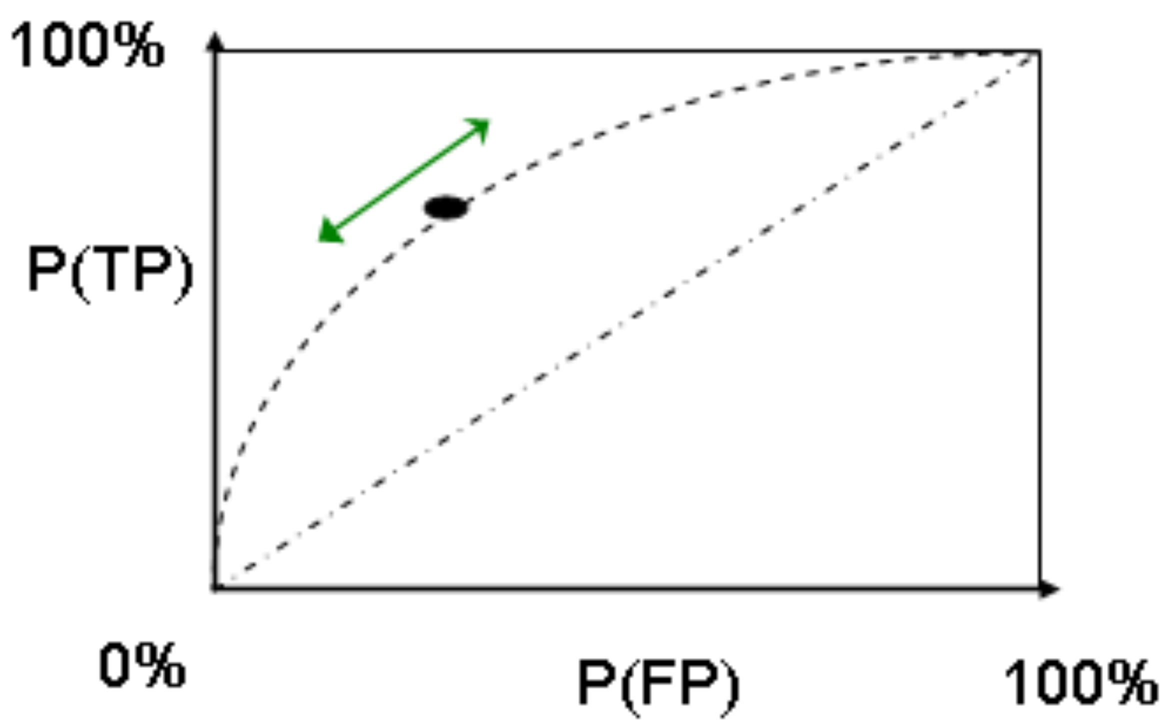
$$\mathrm{DCG_p} = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2(i)}$$



Kiértékelés



TP	FP
FN	TN
1	1



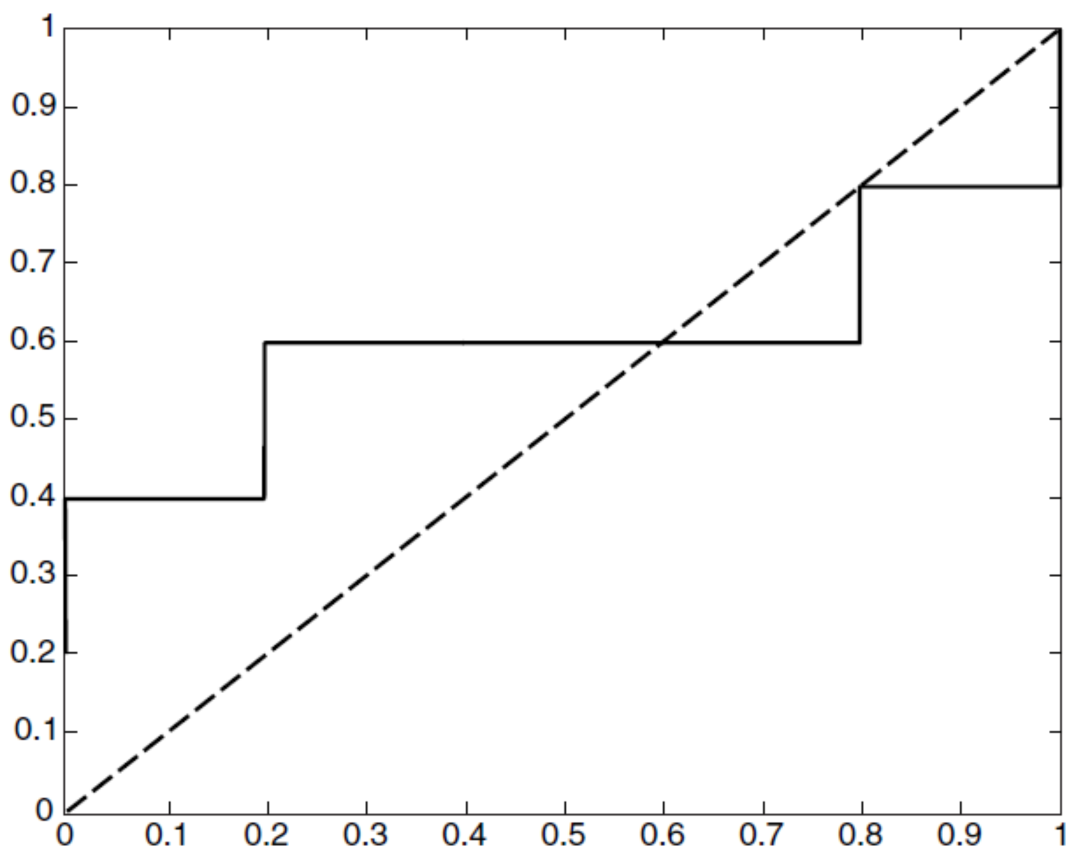
ROC: Receiver Operating Characteristic

- csak bináris osztályozásnál használható (i.e. egy osztályra)
- Area Under Curve: AUC annak a valószínűsége, hogy véletlen pozitív elemeket előrébb sorol mint véletlen negatívakat
- mivel az jóslatok sorrendjéből számítjuk, a klasszifikálónak nem csak bináris jóslatokat kell visszaadnia
- előnye, hogy nem függ a vágási ponttól
- Milyen szempontból lehetne továbbfejleszteni?

Class	+	−	+	−	−	−	+	−	+	+	
	0.25	0.43	0.53	0.76	0.85	0.85	0.85	0.87	0.93	0.95	1.00
TP	5	4	4	3	3	3	3	2	2	1	0
FP	5	5	4	4	3	2	1	1	0	0	0
TN	0	0	1	1	2	3	4	4	5	5	5
FN	0	1	1	2	2	2	2	3	3	4	5
TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0

ROC: Receiver Operating Characteristic

Class	+	−	+	−	−	−	+	−	+	+	
	0.25	0.43	0.53	0.76	0.85	0.85	0.85	0.87	0.93	0.95	1.00
TP	5	4	4	3	3	3	3	2	2	1	0
FP	5	5	4	4	3	2	1	1	0	0	0
TN	0	0	1	1	2	3	4	4	5	5	5
FN	0	1	1	2	2	2	2	3	3	4	5
TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0



AUC=?

+	+	-	+	-	-	+	+	-	+	
0.16	0.32	0.42	0.44	0.45	0.51	0.78	0.87	0.91	0.93	Score
										TP
										FN
										TN
										FP
										TPR
										FPR

Holnapi gyakorlaton:

Mennyi az AUC?
(a nagyobb score-al rendelkező elemek előrébb kerülnek a találati listán)

+	+	-	-	-	+	+	-	+	+	
0.43	0.56	0.62	0.78	0.79	0.86	0.89	0.89	0.91	0.96	Score
										TP
										FN
										TN
										FP
										TPR
										FPR

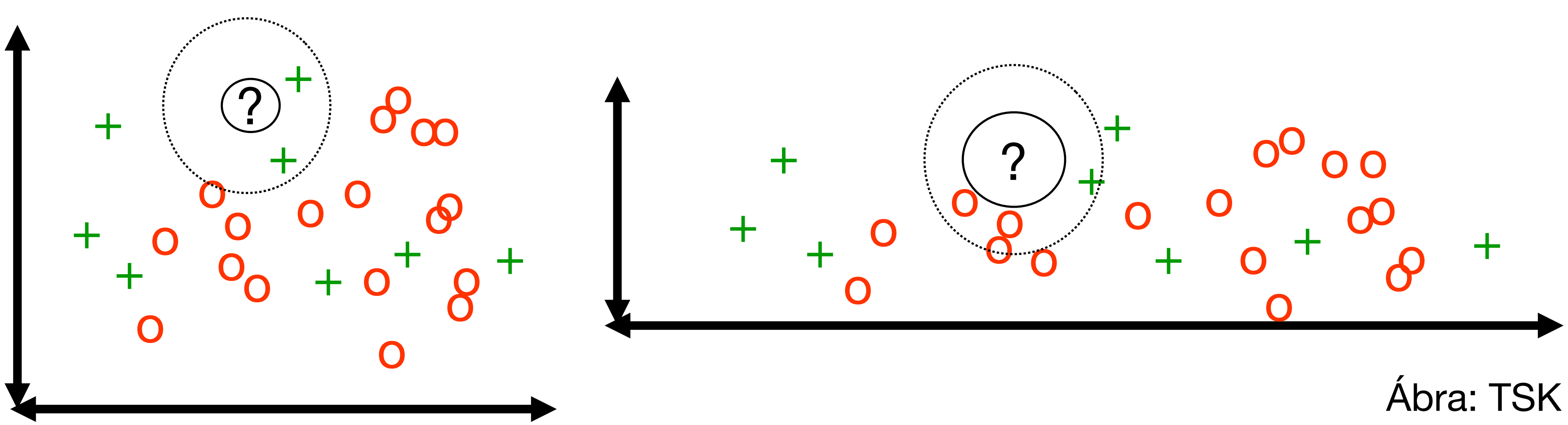
Távolság ism.

Távolság mértékek (pl.)

- Minkowski távolság
- Mahalanobis $D_M(x) = \sqrt{(x-\mu)^T S^{-1} (x-\mu)}.$
- Cosine, Jaccard, Kullback-Leibler, Jensen-Shannon stb.

Problémák:

- skála
- autómatikus távolság választás
- normalizáció



Ábra: TSK

Holnapi gyakorlat

NN_dataset.zip

image_histograms.txt és sample_histogram.txt:

Adottak az images könyvtár képein számolt 3x8 dimenziós RGB hisztogramok

Rendezzük sorrendbe távolság szerint a sample_histogram.txt fájl-ban található kép hisztogramjától számított L2 távolság alapján az image_histograms.txt képeit!

Mi lehet a hiba? Mi történik ha L2 normalizáljuk a hisztogrammokat?

≠

Naïve-Bayes

Bayes-i alapok:

$$p(h \mid d) = \frac{P(d \mid h)P(h)}{P(d)}$$

Priori: P(h)
Posteriori: P(h|d)
Evidence: P(d)
Total probability: P(h)=P(h,d)+P(h,/d)=P(h|d)P(d)+P(h|/d)P(/d)

Feladat: Az épületben megforduló emberek tizede tanár a többi diák. 10-ből 8 diák saját laptoppal rendelkezik míg minden 10 tanárból csak 5-nek van laptopja, mennyi a valószínűsége, hogy egy ember aki bejön az ajtón:

a) tanár
b) diák
c) van laptopja
d) van laptopja és diák
e) van laptopja és tanár

Naïve-Bayes

Szeretnénk attribútumok alapján meghatározni a posteriori valószínüségeket:

P(class=1| A1,A2, ...,An) és P(class=0| A1,A2,...,An)

Amelyik nagyobb arra döntünk. P(A1,A2,...,An | class=1) P(class=1)/P(A1,A2,...,An)

P(A1,A2,...,An | class=1)=πP(Ai|class=1)

konstans!

Saját háza van	Családi állapot	kereset	Van hitele?
van	egyedülálló	125e	nincs
nincs	házas	100e	nincs
nincs	egyedülálló	70e	nincs
van	házas	120e	nincs
nincs	elvált	95e	van
nincs	házas	60e	nincs
van	elvált	220e	nincs
nincs	egyedülálló	85e	van
nincs	házas	75e	nincs
nincs	egyedülálló	90e	van

Naïve-Bayes

Szeretnénk attribútumok alapján meghatározni a posteriori valószínűségeket:

P(class=1| A1,A2, ...,An) és P(class=0| A1,A2,...,An)

Amelyik nagyobb arra döntünk. P(A1,A2,...,An | class=1) P(class=1)/P(A1,A2,...,An)

P(A1,A2,...,An | class=1)=πP(Ai|class=1)

konstans!

Saját háza van	Családi állapot	kereset	Van hitele?
van	egyedülálló	125e	nincs
nincs	házas	100e	nincs
nincs	egyedülálló	70e	nincs
van	házas	120e	nincs
nincs	elvált	95e	van
nincs	házas	60e	nincs
van	elvált	220e	nincs
nincs	egyedülálló	85e	van
nincs	házas	75e	nincs
nincs	egyedülálló	90e	van

P(saját ház=nincs | hitel=nincs)=4/7
P(saját ház=van | hitel=nincs)=3/7
P(saját ház=nincs | hitel=van)=1
P(saját ház=van | hitel=van)=0
P(családi_áll=egyedülálló | hitel=nincs)=2/7
P(családi_áll=házas | hitel=nincs)=4/7
P(családi_áll=elvált | hitel=nincs)=1/7
P(családi_áll=egyedülálló | hitel=van)=2/3
P(családi_áll=házas | hitel=van)=0
P(családi_áll=elvált | hitel=van)=1/3

P(kereset=? | hitel=van)=?
P(kereset=? | hitel=nincs)=?

Naïve-Bayes

Folytonos változók esetében

- diszkretizáljuk az adatatot : 0-90e Ft-ig 91-125e stb
- modellezzük a problémát normál eloszlással!

$$P(X_i = x_i | Y = y_j) = \frac{1}{\sqrt{2\pi}\sigma_{ij}} \exp^{-\frac{(x_i - \mu_{ij})^2}{2\sigma_{ij}^2}}$$

Saját háza van	Családi állapot	kereset	Van hitele?
van	egyedülálló	125e	nincs
nincs	házas	100e	nincs
nincs	egyedülálló	70e	nincs
van	házas	120e	nincs
nincs	elvált	95e	van
nincs	házas	60e	nincs
van	elvált	220e	nincs
nincs	egyedülálló	85e	van
nincs	házas	75e	nincs
nincs	egyedülálló	90e	van

Kereset:

hitel=nincs:

Átlag: 110e
Szórás^2: 2975

hitel=van:

Átlag: 90e
Szórás^2: 25

Naïve-Bayes

Szeretnénk attribútumok alapján meghatározni a posteriori valószínüségeket:

P(class=1| A1,A2, ...,An) és P(class=0| A1,A2,...,An)

Amelyik nagyobb arra döntünk. P(A1,A2,...,An class=1)P(class=1)/P(A1,A2,...,An)

P(A1,A2,...,An | class=1)=πP(Ai|class=1)

konstans!

Saját ház	Családi állapot	kereset	Van hitele?
van	egyedülálló	125e	nincs
nincs	házas	100e	nincs
nincs	egyedülálló	70e	nincs
van	házas	120e	nincs
nincs	elvált	95e	van
nincs	házas	60e	nincs
van	elvált	220e	nincs
nincs	egyedülálló	85e	van
nincs	házas	75e	nincs
nincs	egyedülálló	90e	van

Van-e hitele a következő tulajdonságokkal rendelkező személynek:

- nincs saját háza
- házas
- 120e a keresete

P(hitel=van | saját ház=nincs, családi_áll=házas, kereset=120e)=?
P(hitel=van | saját ház=nincs, családi_áll=házas, kereset=120e)=?

Naïve-Bayes

A 0-a valószínűségek több esetben zajként jelentkeznek (pl. nincs rá elem az eredeti adathalmazban ergo nem is lehetséges)

M-estimate:

Legyen p egy előre meghatározott minimum valószínűség

Módosítsuk a feltételes valószínűségek kiszámítását:

$$P(x_i|y_j) = \frac{n_c + mp}{n + m}$$

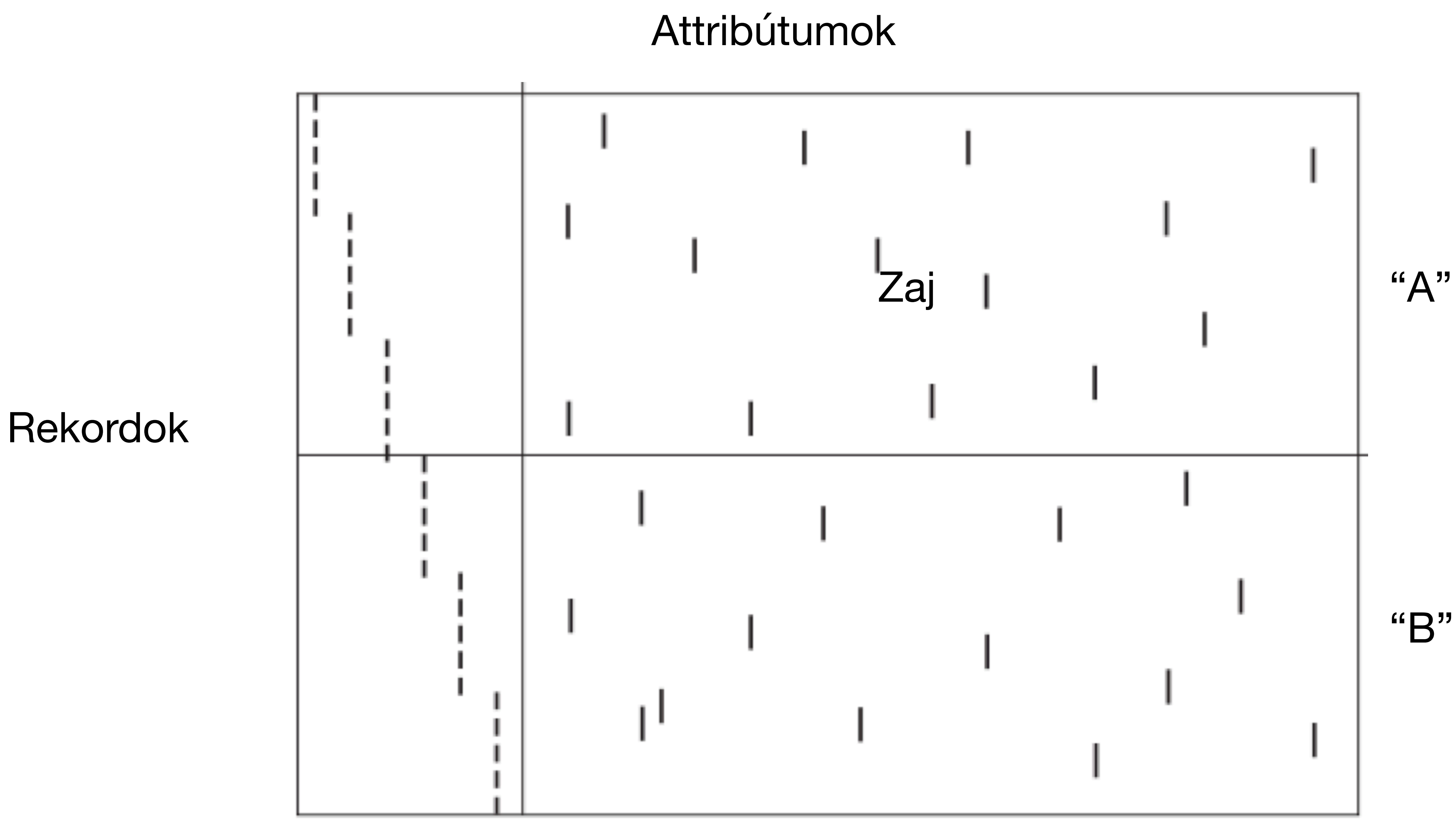
Ahol m egy előre meghatározott konstans, n_c azon x_i tulajdonsággal rendelkező tanulópontok száma melyek osztályváltozója y_i , n pedig az összes ide tartozó tanulópont száma.

Olyan esetekben is p lesz a posterior valószínűség, ha egyáltalán nincs olyan tanulópont melynek y_i az osztályváltozója.

M-estimate nagyban segíti a zajos, hiányos vagy egyszerűen speciális esetek korrekt kiértékelését anélkül, hogy azok szerepeltek volna az tanulóhalmazban.

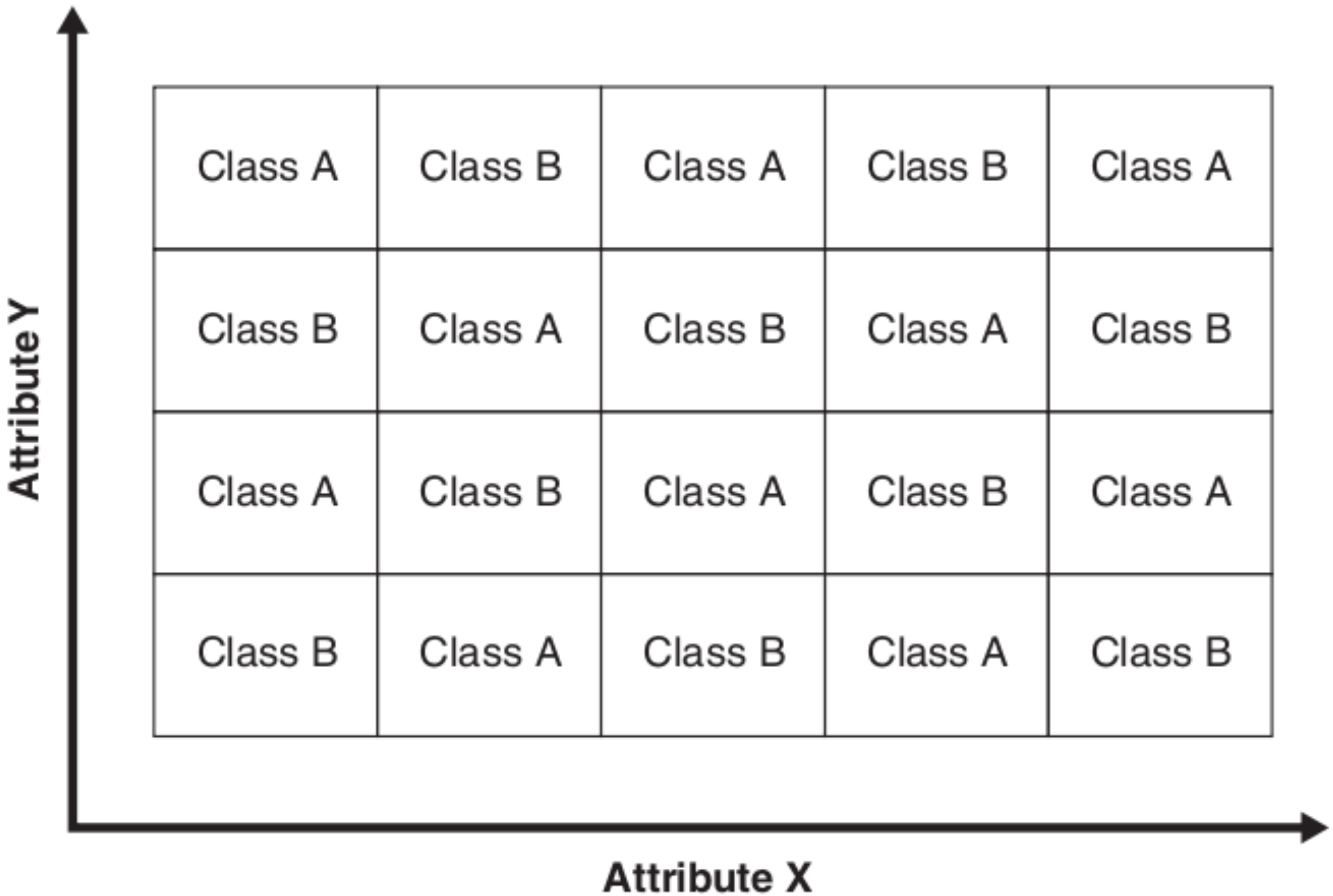
Zajos attribútumok

Hogyan viselkedik a kNN és egy NB a következő adaton?

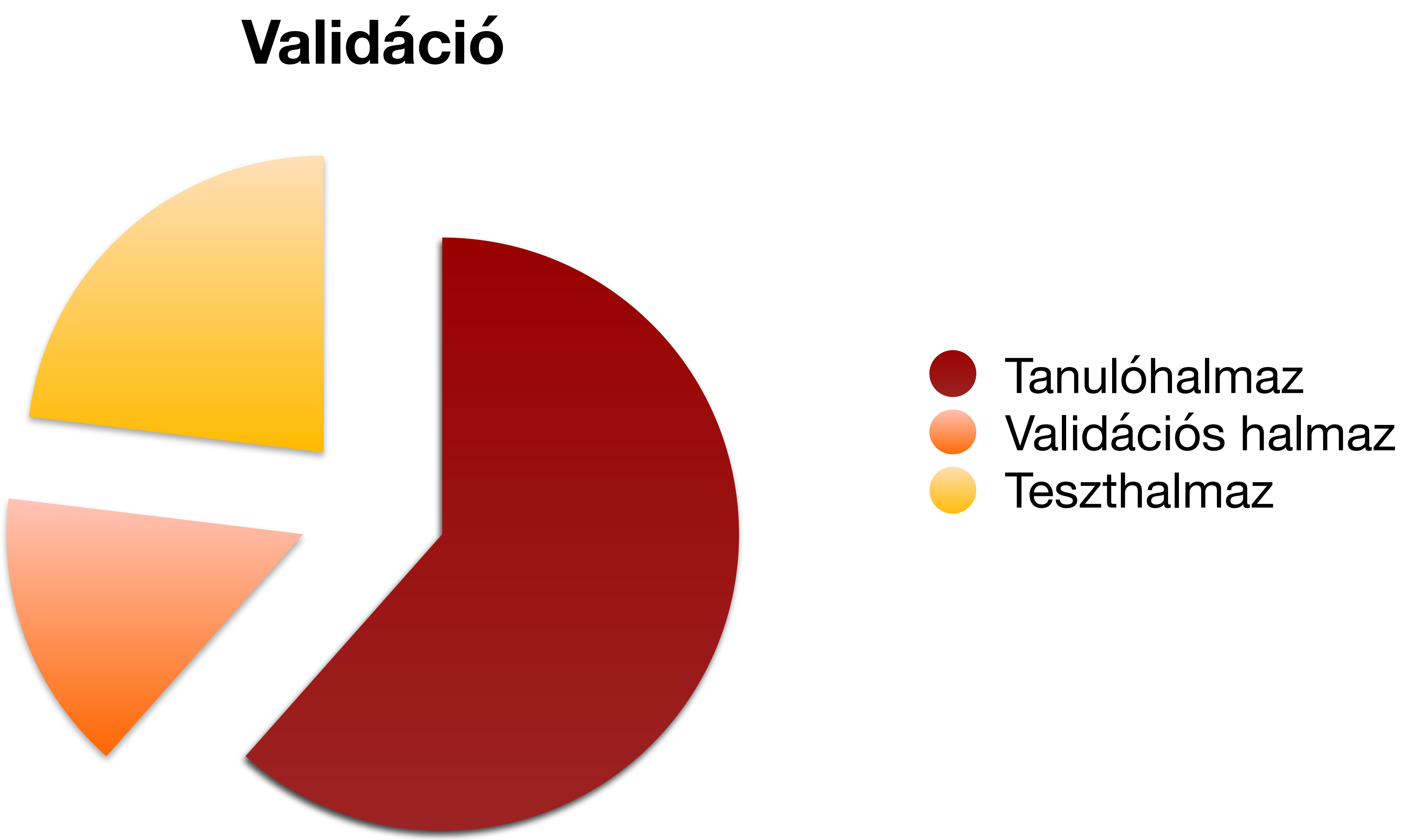


Órai feladat 1:

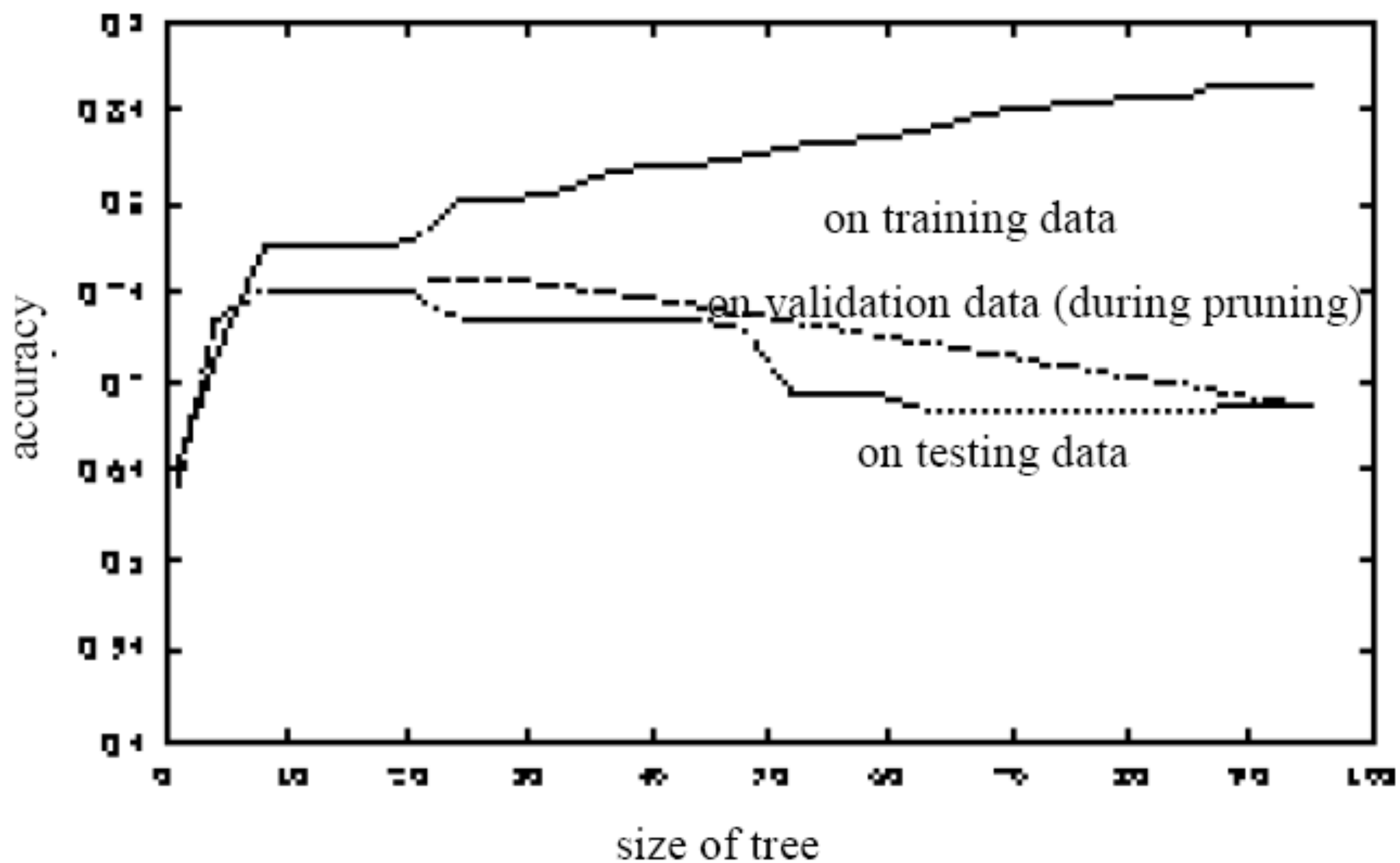
Melyik klasszifikátor működik jól/rosszul s miért?
(K-NN, Naïve-Bayes)



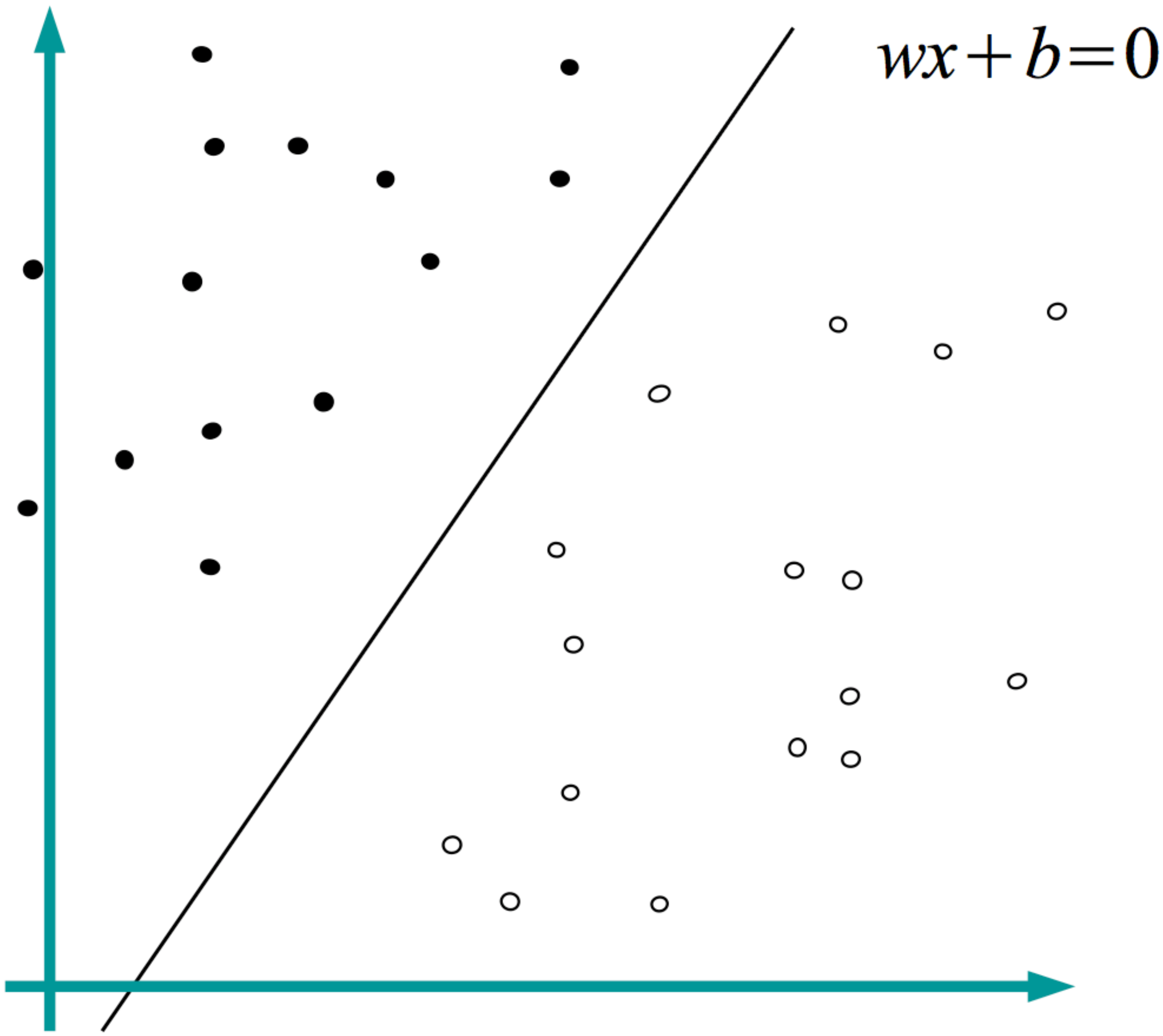
Kereszt validáció és hiba



Kereszt validáció és hiba



Lineáris szeparálás



Lineáris szeparálás

Legyen adott egy véges n elemű mintahalmaz $\mathbf{X}=\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ egy d-dimenziós térben illetve minden elemhez rendeljünk egy osztályváltozót ($\mathbf{y}_i$). Keressünk egy d-dimenziós **w** normál vektort melyre a következő egyenlőtlenségek igazak

$\mathbf{w}\cdot\mathbf{x}_i > b$ minden $\mathbf{x}_i$ elemre melynek a cimkéje +1

$\mathbf{w}\cdot\mathbf{x}_i < b$ minden $\mathbf{x}_i$ elemre melynek a cimkéje -1

Azon vektor-küszöb párosok (**w**, b), melyekre igazak a fenti egyenlőtlenségek **X,y** lineáris szeparátorai.

Lineáris szeparálás

Egészítsük ki a meglévő vektorainkat egy extra dimenzióval. A probléma átírható a következő formára

$$(\mathbf{w}'\cdot\mathbf{x}'_{\mathbf{i}})\ l_i > 0$$

ahol $1 \le i \le n$ és $\mathbf{x}'_{\mathbf{I}} = (\mathbf{x}_{\mathbf{i}}, 1)$ illetve $\mathbf{w}' = (\mathbf{w}, b)$.

Hogyan találjunk egy lineáris szeparáló hipersíkot?

Perceptron tanulás

Iteratív algoritmus, hogy találjunk egy lineáris szeparálót

Kezdőállapot:

Legyen $\mathbf{w} = y_1\mathbf{x_1}$ és $|\mathbf{x_i}|=1$ minden $\mathbf{x_i}$-re

Amíg létezik $\mathbf{a_i}$ melyre nem igaz az egyenlőtlenség azaz $(w \cdot a_i)l_i \leq 0$, módosítsuk a modellünket

$$\mathbf{w}^{\mathbf{t+1}} = \mathbf{w}^{\mathbf{t}} + y_i\mathbf{x_i}$$

Az algoritmus megáll ha létezik lineáris szeparáló!

Lineáris regresszió

Tegyük fel újra, hogy a tanulóhalmaz attribútumait kiegészítettük megint egy bias változóval. (X a tanulóhalmaz (N elemű), Y az osztályváltozó) Ebben az esetben az előbbi líneáris kombináció :

$$Y = X^T w$$

Szeretnénk megtalálni a regressziós görbét, melynél a négyzetesen hiba minimális (lehet más hibamértéket is alkalmazni):

$$error_{square}(f) = E[(Y - f(X))^2]$$

Mivel a tanulóhalmaz véges, a hiba :

$$error(f) = \sum_1^N (y_i - x_i^T w_i)^2$$

Lineáris regresszió

Vagy:

$$error(f) = \sum_1^N (y_i - x_i^T w_i)^2 = (y - Xw)^T (y - Xw)$$

Mindig létezik minimuma és egyértelmű, így képezzük w szerinti deriváltját, s megkeressük hol 0-a:

$$-2X^T y + 2X^T Xw = 0 \qquad X^T (y - Xw) = 0$$

Melyből következik , hogy ha nem 0-a a determináns →

$$w = (X^T X)^{-1} X^T y$$

Logisztikus regresszió

Balról jobbra:

Épp ezért el kell döntenünk hogy mikor döntünk 1-es osztályra vagy 0-as osztályra (feltételezve, hogy eredetileg az osztályváltozóink értékkészlete {0,1}). Amennyiben a jóslás nagyobb 0.5 akkor (ebben az esetben közelebb van 1-hez vagy nagyobb egynél) legyen a jóslat értéke 1, ha pedig kisebb 0-a.

$$y = x^T w - 0.5$$

$$f(y) = y_{osztály} = \frac{1 + sgn(y)}{2}$$

Sajnos már nem alkalmazhatunk lineáris regressziót a w meghatározásához.

Logisztikus regresszió

Hogy meghatározhassuk w-t, keresnünk kell egy olyan f(y) függvényt melyre a következő állítások igazak:

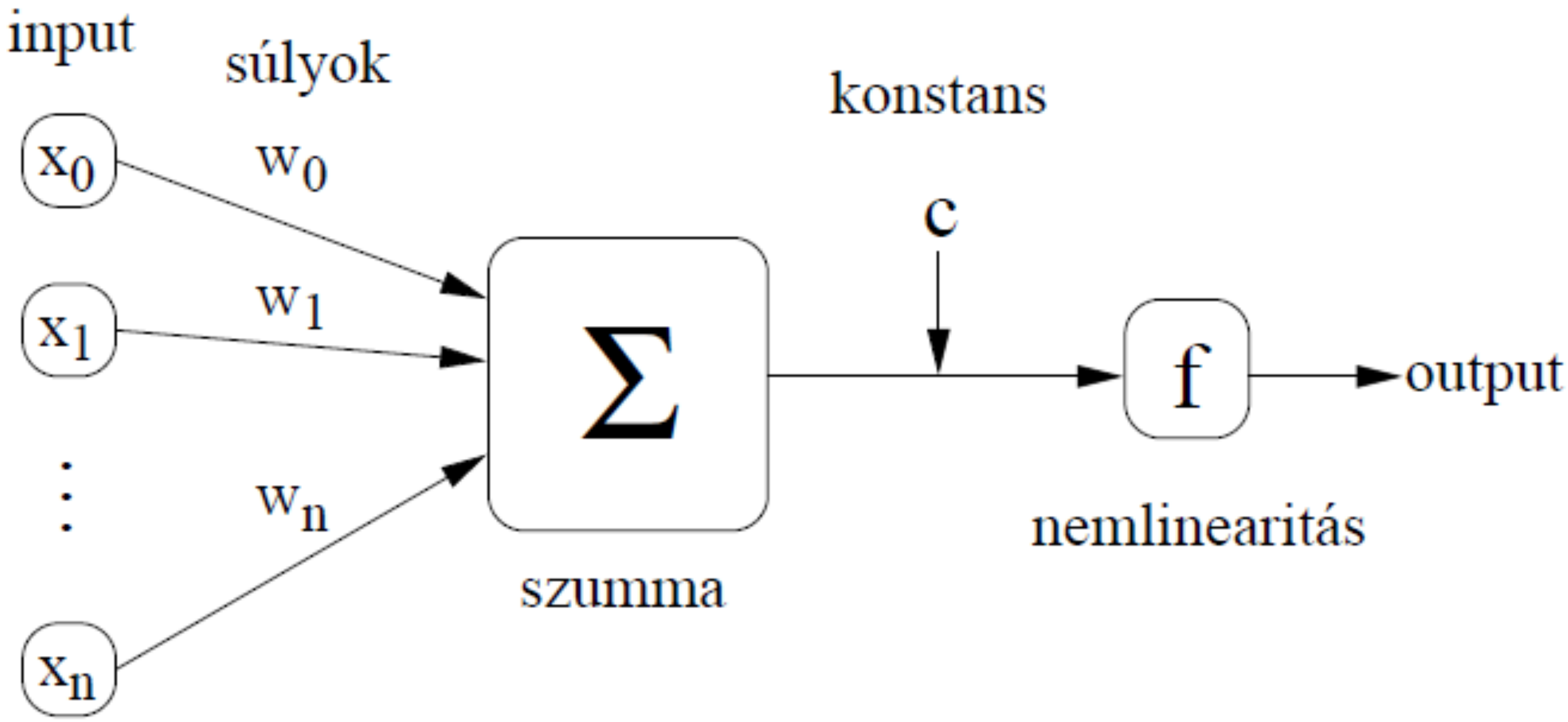
1. értéke 1-hez közelít, ha y értéke végtelenhez tart
2. értéke 0-hoz közelít, ha y értéke mínusz végtelenhez tart
3. f(0) = 0.5
4. szimmetrikus nullára nézve, tehát f(y) + f(-y) = 1 azaz 2f(0)
5. f(y) differenciálható minden pontban
6. ne legyenek lokális szélsőértékei (pl. monoton növekvő)

A szigmoid függvények megfelelnek a fenti követelményeknek

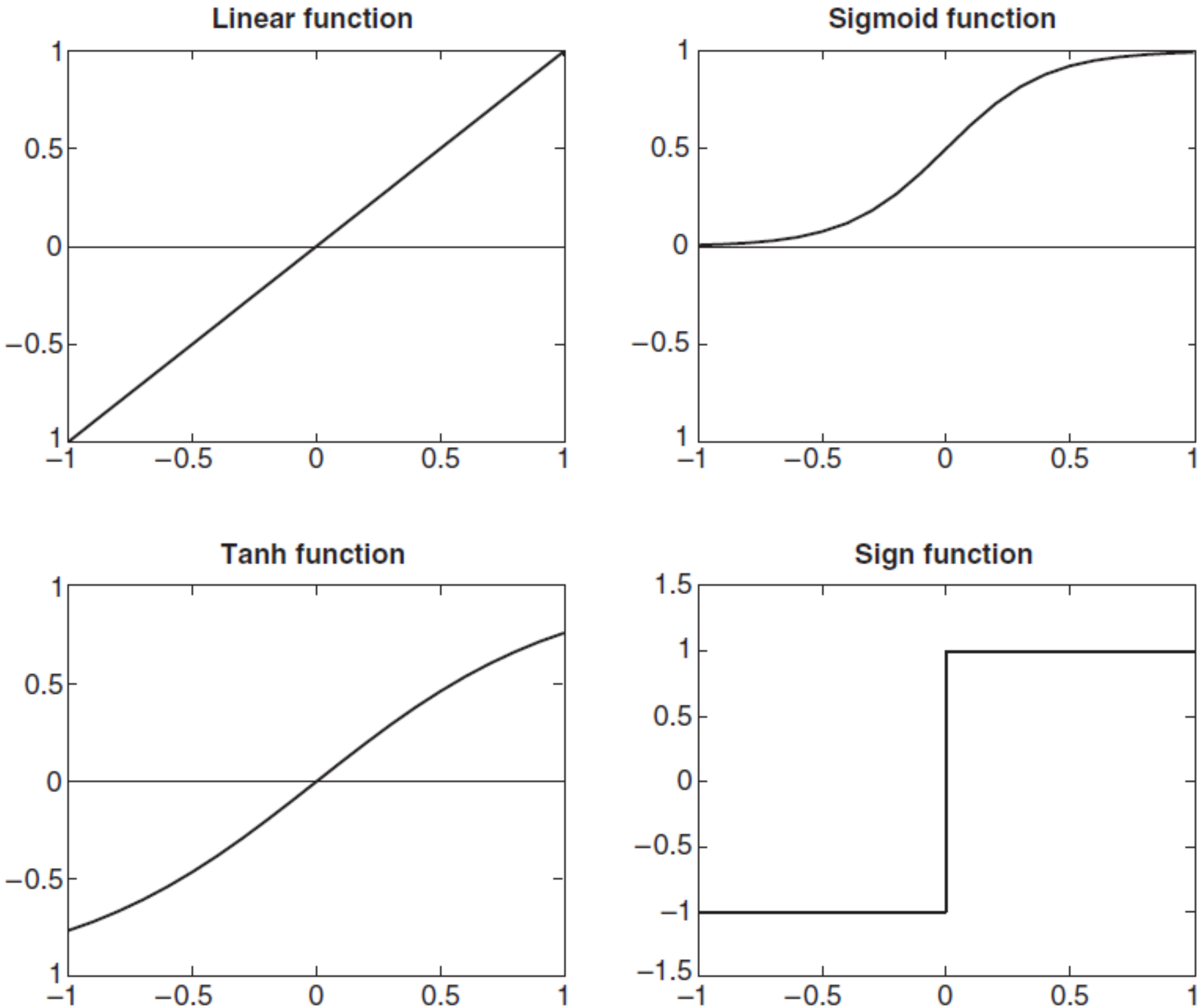
$$f(y) = 1/(1 + a^{(-y)})$$

amennyiben a > 1.

Logisztikus regresszió



Aktiválási függvények



Logisztikus regresszió

Optimalizálás lehet pl. gradiens módszerrel (részletesen később):

$$w_{opt}=\mathrm{argmax}_w \sum \ln(P(y_i|x_i,w))$$

Viszont felhasználhatjuk, hogy bináris osztályozás esetén $y_i=0$ vagy $y_i=1$:

$$L(w)=\sum y_i \ln(P(y_i=1\,|x_i,w)) + (1-y_i)\ln(P(y_i=0\,|x_i,w))$$

Kiindulunk $w^{(0)}$-ból, majd minden iterációban $w^{(l)}$-hez hozzáadjuk $dL(w)/dw$ (parciális deriváltak) λ szorosát (mely egy előre meghatározott konstans, a tanulási növekmény): (w_j-re)

$$\sum x_{ij}(y_i - P(y_i|x_{ij},w_j))$$

Logisztikus regresszió

Balról jobbra: Legyen p(x) az első osztályra döntés valószínűsége és tegyük fel, hogy:

$$\ln \frac{p(x)}{1-p(x)} \approx x^T\omega + \omega_0$$

Melyből:

$$p(x \mid \omega) = sigm(x \mid \omega) = \frac{1}{1+\mathrm{e}^{-(x^T\omega+\omega_0)}}$$

Amennyiben a log-likelihood-ra optimalizálunk (tanulóhalmaz $X=\{x_1,..,x_t\}$) és a mintáinkat függetlennek tekintjük:

$$\begin{aligned}
\frac{\partial \mathcal{L}(\omega \mid X)}{\partial \omega_i} &= \sum_{x_t \in X^{(+)}} \frac{\partial \ln p(x_t \mid \omega)}{\partial \omega_i} + \sum_{x_t \in X^{(-)}} \frac{\partial \ln(1-p(x_t \mid \omega))}{\partial \omega_i} \\
&= \sum_{x_t \in X^{(+)}} (1-p(x_t \mid \omega))x_{ti} - \sum_{x_t \in X^{(-)}} p(x_t \mid \omega)x_{ti} \\
&= \sum_{x_t \in \{X^{(-)},X^{(+)}\}} (y_t - p(x_t \mid \omega))x_{ti}
\end{aligned}$$

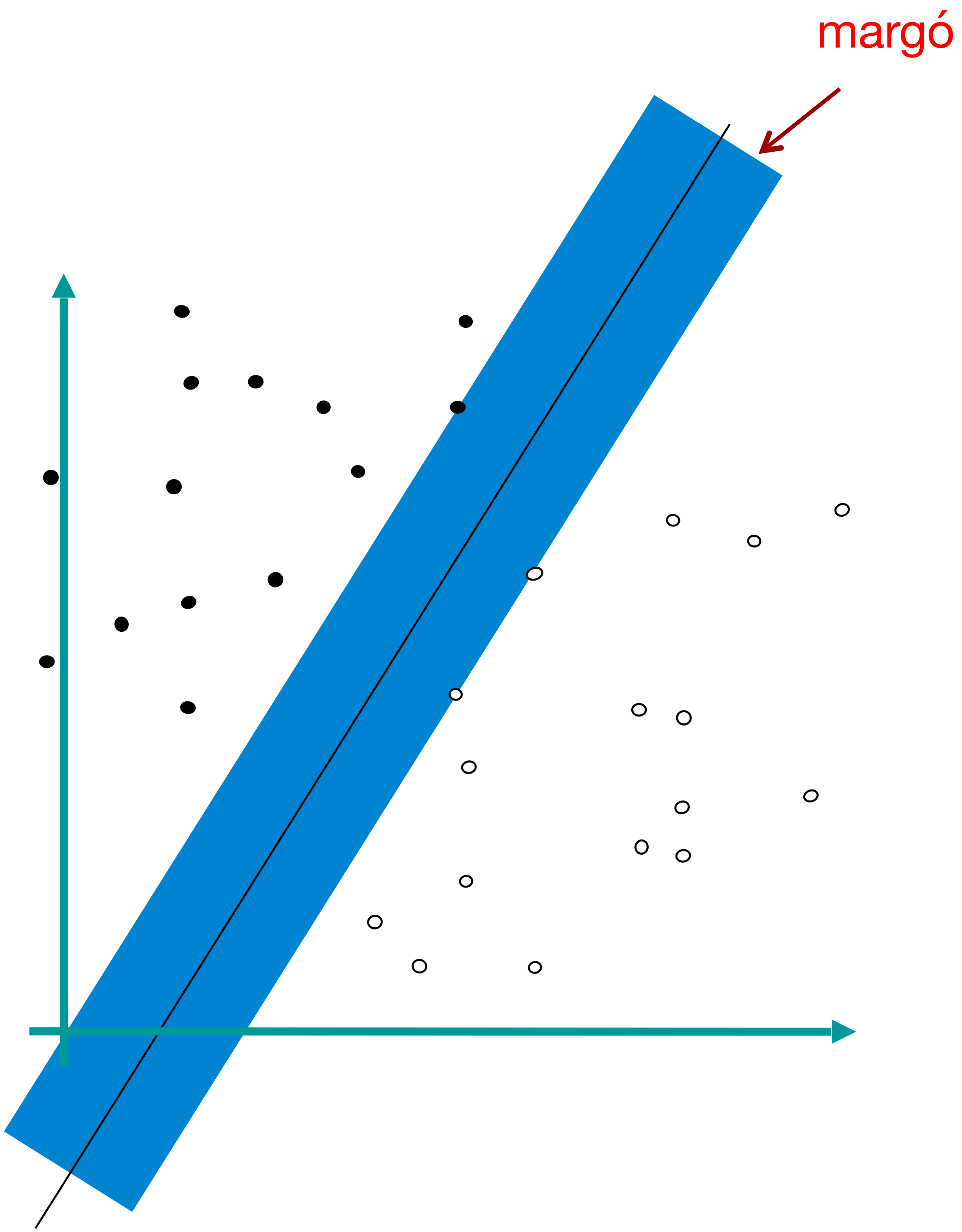
Maximális margó

Definíció: Legyen **w** (**X,y**) egy lineáris szeparátora. Ebben az esetben a margó a hipersík és bármely minta x_i távolsága

$$min_i \frac{w\, x_i}{|w|}$$

Amennyiben létezik egy ideális megoldás (ahol maximális a margó), a perceptron tanulás konvergál.

Margó



Maximális margó (balról)

Legyen **δ** a **w** margója.

Módosítsuk az eredeti egyenlőtlenségünket a következő módon

$$y_i(\mathbf{v}^T \mathbf{x_i}) > 1$$

ahol

$$v = \frac{w}{\rho \left| w \right|}$$

Maximális margó helyett minimális|**v**|-ra optimalizálunk!

Maximális margó (balról)

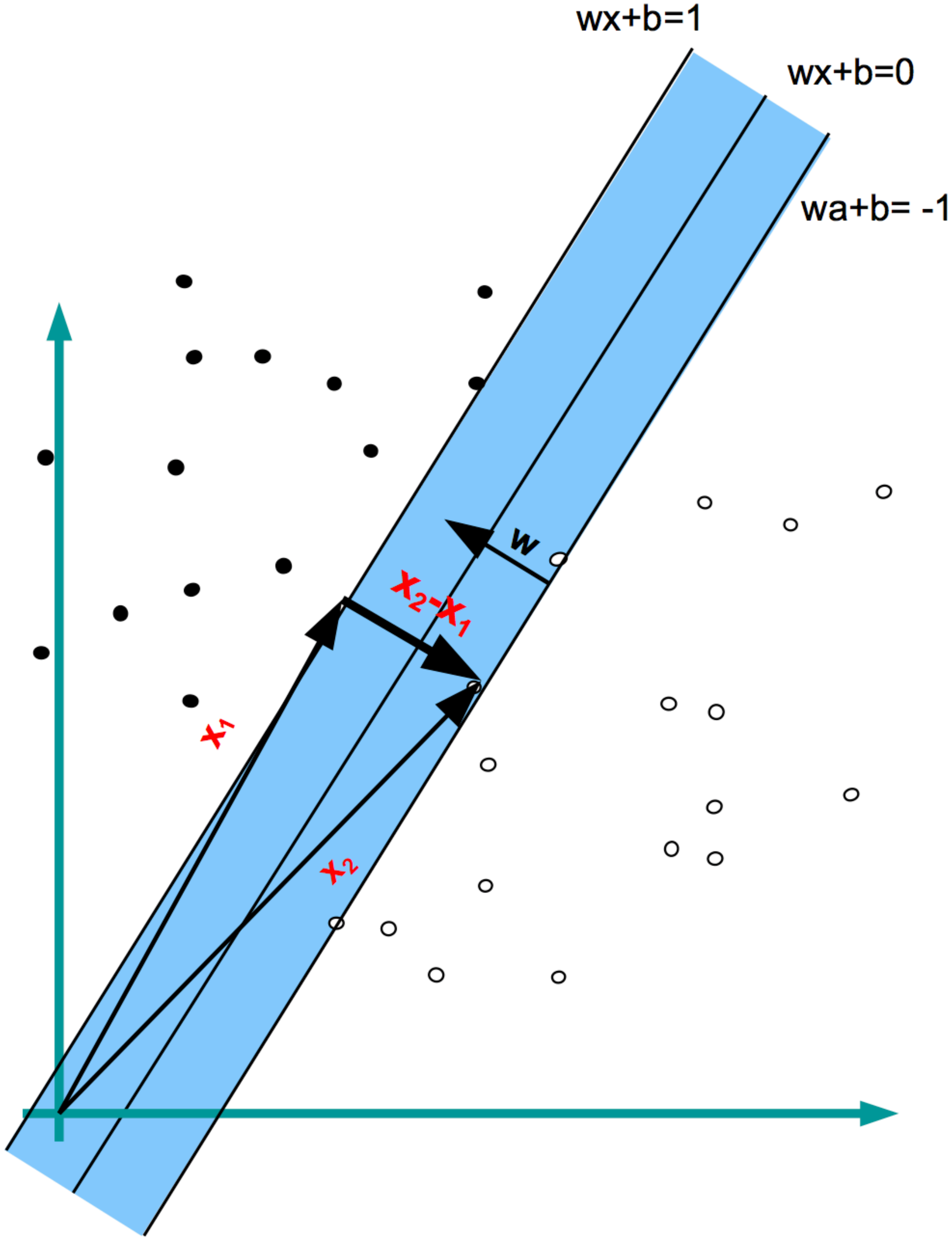
Az új optimalizálási problémánk

$$\min |\mathbf{v}| \text{ ahol } y_i(\mathbf{v}^T \mathbf{x_i}) > 1,\ \forall i.$$

Miután a fenti probléma optimuma megegyezik a következő probléma optimumával, a végső optimalizáció:

$$\min |\mathbf{v}|^2$$
$$\text{ahol } y_i(\mathbf{v}^T \mathbf{x_i}) \geq 1,\ \forall i.$$

Maximális margó



$$wx_1+b=1$$

$$wx_2+b=-1$$

$$x_2-x_1=-n\frac{w}{\|w\|} \longrightarrow x_2=x_1-n\frac{w}{\|w\|}$$

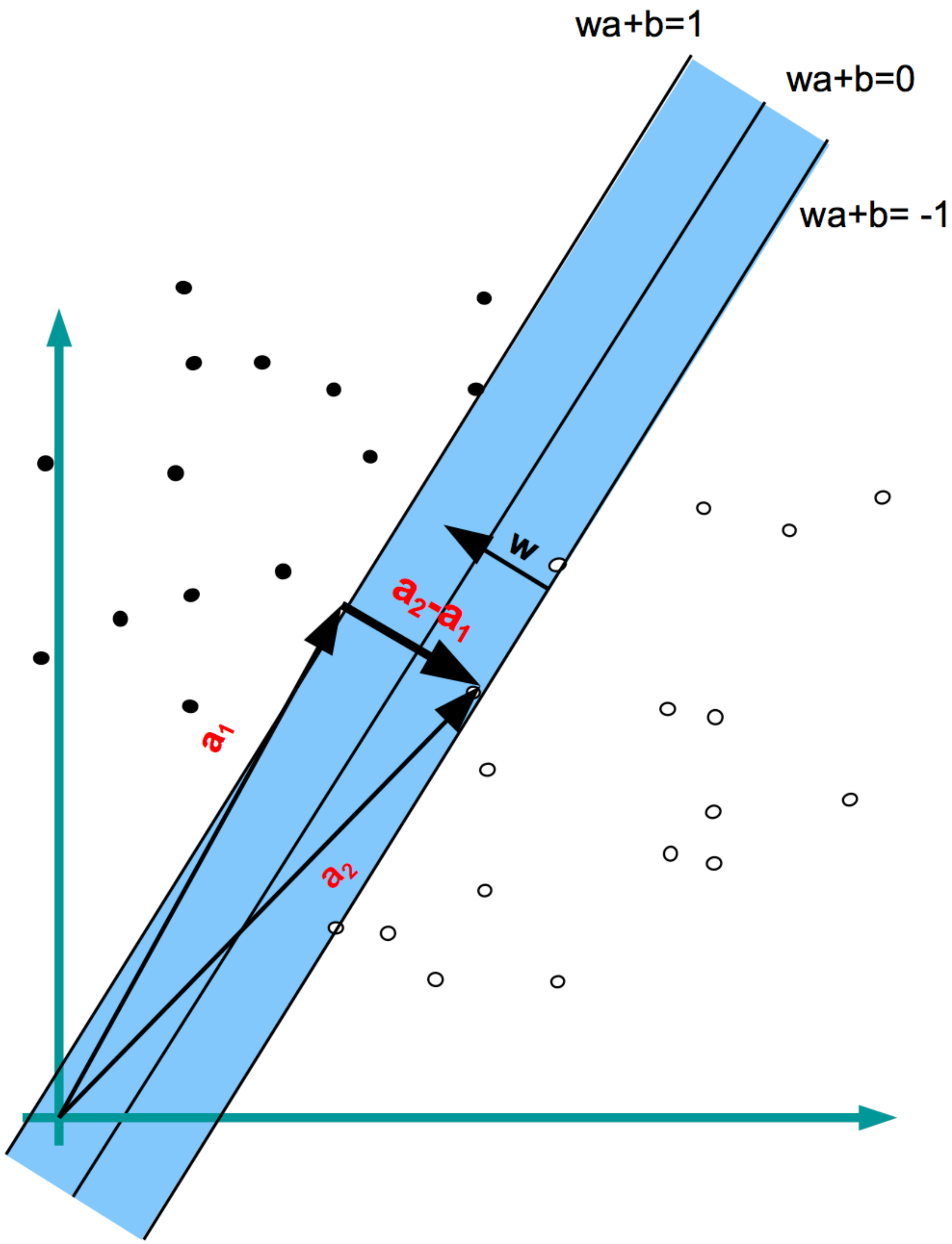
$$wx_2+b=\left(x_1-n\frac{w}{\|w\|}\right)w+b=-1$$

$$wx_1+b=1$$

$$wx_1+b-n\frac{w}{\|w\|}w=-1$$

$$1-n\frac{w}{\|w\|}w=-1 \longrightarrow \boxed{n=\frac{2}{\|w\|}}$$

Maximális margó



$$wa_1+b=1$$

$$wa_2+b=-1$$

$$a_2-a_1=-n\frac{w}{\|w\|} \longrightarrow a_2=a_1-n\frac{w}{\|w\|}$$

Maximize $n=\frac{2}{\|w\|}$

$\downarrow$

Minimize $\|w\|$

$$1-n\frac{w}{\|w\|}w=-1 \longrightarrow n=\frac{2}{\|w\|}$$

Lineáris szeparálhatóság

Lehetséges-e és ha igen, hogyan a következő függvényeket megadni lineáris szeparátorral?

- OR
- AND

Keressük meg gradiens módszerrel a következő függvény maximumát:

sin(x)

Lineáris szeparálhatóság

Lehetséges-e és ha igen, hogyan a következő függvényeket megadni lineáris szeparátorral?

- OR
- AND

Keressük meg gradiens módszerrel a következő függvény maximumát:

sin(x)

Majd pedig a következőét: x*sin(x)

Lineáris szeparálhatóság

Lehetséges-e és ha igen, hogyan a következő függvényeket megadni lineáris szeparátorral?

- OR
- AND

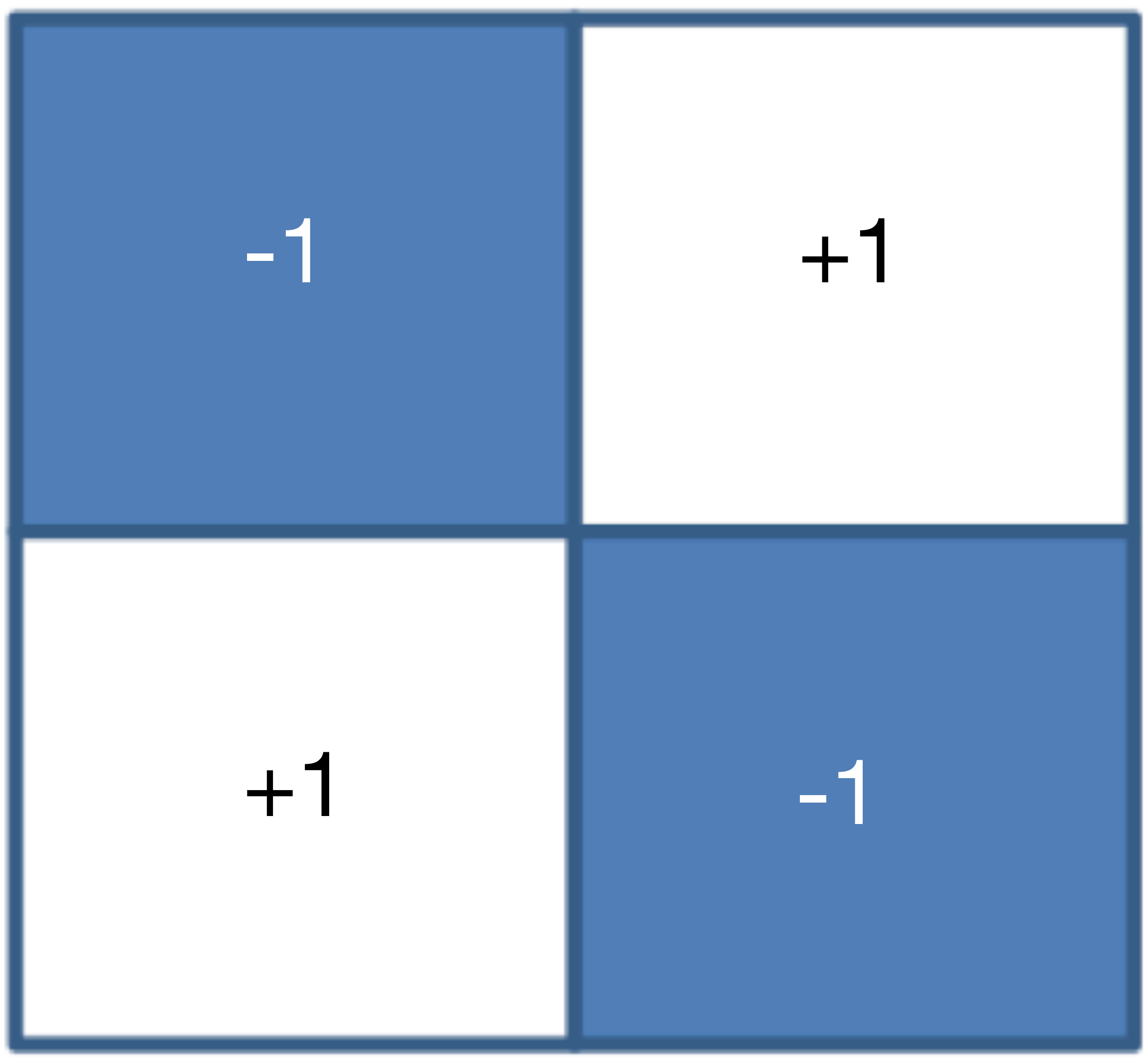
Keressük meg gradiens módszerrel a következő függvény maximumát:

sin(x)

Majd pedig a következőét: x*sin(x)
Ha $x>-10$ és $x<10$…

Lineáris szeparálhatóság

Lehetséges-e lineáris szeparálót találni a lenti adathalmazhoz?



Mit is jelent ez számunkra a lineáris osztályozók szempontjából?