Adatbányászati feladatgyűjtemény tehetséges hallgatók számára

Buza Krisztián

Számítástudományi és Információelméleti Tanszék

Budapesti Műszaki és Gazdaságtudományi Egyetem

Tartalomjegyék

Modellek kiértékelése

- Accuracy számítása
- Confusion Mátrix
- Precision, Recall, F-Measure számítása
- ROC görbe szerkesztése,
  görbe alatti terület számítása

Háttér (1/3)

Egy hitelintézet különböző adatokat rögzít az ügyfelekről, ezek közé tartozik, hogy egy-egy ügyfélnek volt-e késedelme a korábban folyósított hitel törlesztő részleteinek visszafizetése során.

A hitelintézet két különböző elemző céget bíz meg azzal, hogy egy-egy modellt fejlesszenek, amely előrejelzi, hogy egy-egy adott ügyfél mekkora valószínűséggel lesz késedelmes a hitel visszafizetése során.

Tehát összesen két előrejelző modell van, az egyiket az egyik cég fejleszti, a másikat a másik.

Háttér (2/3)

A hitelintézet a „software as service” elv szerint azzal arányosan fizet az egyik illetve másik elemző cégnek, hogy az adott cég modelljét hány alkalommal használta a hitelintézet előrejelzésre.

Ezért a hitelintézet abban érdekelt, hogy eldöntse, melyik modell teljesít jobban, és csak a jobban teljesítő modellt használja, illetve – különlegesen kockázatos esetekben – mind a két modell szerinti előrejelzést figyelembe veszi.

A modellek kiértékeléséhez adott egy teszt adatbázis, amely független a modellek készítésekor (tanításakor) használt adatbázistól (ugyanazt az ügyfelet nem tartalmazza mind a két adatbázis). A modellek készítésekor használt adatbázishoz hasonlóan, a teszt adatbázis esetén is tudjuk, hogy melyik ügyfél volt késedelmes és melyik nem (a teszt adatbázis is múltbeli eseményeket ír le).

Háttér (3/3)

- Pozitív osztály: késedelmes ügyfelek
- Negatív osztály: jó ügyfelek

- Tesztadatok:

Valóság	-	-	+	-	-	-	+	-	-	-	+	-
Model 1	0.2	0.1	0.49	0.26	0.3	0.12	0.31	0.2	0.1	0.32	0.4	0.2
Model 2	0.6	0.51	0.8	0.12	0.54	0.39	0.53	0.46	0.41	0.37	0.49	0.28

Accuracy

- **1. Feladat: accuracy számítás**
- Válasszuk küszöbszámnak 0.5-t, azaz:
  ha a modell jóslata szerint legalább 0.5 valószínűséggel tartozik egy példány a pozitív osztályba, akkor tekintsük úgy, hogy a modell az adott példányt a pozitív osztályba sorolta.
- Számoljuk ki mindkét modell accuracy-ját!
- Tesztadatok:

Valóság	-	-	+	-	-	-	+	-	-	-	+	-
Model 1	0.2	0.1	0.49	0.26	0.3	0.12	0.31	0.2	0.1	0.32	0.4	0.2
Model 2	0.6	0.51	0.8	0.12	0.54	0.39	0.53	0.46	0.41	0.37	0.49	0.28

Accuracy

- **Megoldás: accuracy számítás**
- accuracy = helyesen osztályozott példányok száma
  osztva az összes tesztpéldány számával
- Model 1: accuracy = 9 / 12 = 0.75
- Model 2: accuracy = 8 / 12 = 0.66

Valóság	-	-	+	-	-	-	+	-	-	-	+	-
Model 1	0.2	0.1	0.49	0.26	0.3	0.12	0.31	0.2	0.1	0.32	0.4	0.2
Model 2	0.6	0.51	0.8	0.12	0.54	0.39	0.53	0.46	0.41	0.37	0.49	0.28

Valóság	-	-	+	-	-	-	+	-	-	-	+	-
Model 1	-	-	-	-	-	-	-	-	-	-	-	-
Model 2	+	+	+	-	+	-	+	-	-	-	-	-

Accuracy

2. feladat

- Tényleg jobb-e az első modell a másodiknál?
- Miért (mikor) lehet félrevezető az accuracy?

Accuracy

Megoldás

- Tényleg jobb-e az első modell a másodiknál?
  - Lásd következő feladatok
- Miért (mikor) lehet félrevezető az accuracy?
  - Ha kiegyensúlyozatlan az osztályok eloszlása
  - Minden hibát egyformának tekint
  - (pozitív ügyfél negatív osztályba sorolása illetve negatív ügyfél pozitív osztályba sorolása)
  - Nem független a választott küszöbszámtól

Confusion mátrix

- **3. Feladat: confusion mátrix számítás**
- Válasszuk küszöbszámnak 0.5-t, azaz:
  ha a modell jóslata szerint legalább 0.5 valószínűséggel tartozik egy példány a pozitív osztályba, akkor tekintsük úgy, hogy a modell az adott példányt a pozitív osztályba sorolta.
- Adjuk meg mindkét modell confusion mátrix-át!
  (true positiv-ok, true negative-ok, false positive-ok, false negative-ok számát)
- Tesztadatok:

Valóság	-	-	+	-	-	-	+	-	-	-	+	-
Model 1	0.2	0.1	0.49	0.26	0.3	0.12	0.31	0.2	0.1	0.32	0.4	0.2
Model 2	0.6	0.51	0.8	0.12	0.54	0.39	0.53	0.46	0.41	0.37	0.49	0.28

Confusion mátrix

- **Megoldás: confusion mátrix számítás**

Valóság	-	-	+	-	-	-	+	-	-	-	+	-
Model 1	0.2	0.1	0.49	0.26	0.3	0.12	0.31	0.2	0.1	0.32	0.4	0.2
Model 2	0.6	0.51	0.8	0.12	0.54	0.39	0.53	0.46	0.41	0.37	0.49	0.28

↓

Valóság	-	-	+	-	-	-	+	-	-	-	+	-
Model 1	-	-	-	-	-	-	-	-	-	-	-	-
Model 2	+	+	+	-	+	-	+	-	-	-	-	-

Model 1		osztályozó szerint	
		+	-
Valóság-ban	+	0	3
	-	0	9

TP: 0, TN: 9, FP: 0, FN: 3

Model 2		osztályozó szerint	
		+	-
Valóság-ban	+	2	1
	-	3	6

TP: 2, TN: 6, FP: 3, FN: 1

Súlyozott accuracy

- **4. Feladat:**

  Az előbbi confusion mátrixok alapján (az előbbi küszöbszám mellett) számítsuk a modellek súlyozott accuracy-ját. Egy pozitív ügyfél negatív csoportba történő besorolásának súlya legyen kétszer akkora, mint az összes többi eset súlya.

Model 1		osztályozó szerint	
		+	-
Valóság-ban	+	0	3
	-	0	9

Model 2		osztályozó szerint	
		+	-
Valóság-ban	+	2	1
	-	3	6

Súlyozott accuracy

- **Megoldás:**

  Az előbbi confusion mátrixok alapján (az előbbi küszöbszám mellett) számítsuk a modellek súlyozott accuracy-ját. Egy pozitív ügyfél negatív csoportba történő besorolásának súlya legyen kétszer akkora, mint az összes többi eset súlya.

Model 1		osztályozó szerint	
		+	-
Valóság-ban	+	0	3
	-	0	9

Model 2		osztályozó szerint	
		+	-
Valóság-ban	+	2	1
	-	3	6

(0*w + 9*w) / (0*w + 3*2*w + 0*w + 9*w) = 9 / 15

(2*w + 6*w) / (2*w + 1*2*w + 3*w + 6*w) = 8 / 13

Precision, Recall, F-Measure

- **5. Feladat:**

  Az előbbi confusion mátrixok alapján (az előbbi küszöbszám mellett) számítsuk a modell precision-ját, recall-ját, és F-measure-jét!

Model 1		osztályozó szerint	
		+	-
Valóság-ban	+	0	3
	-	0	9

Model 2		osztályozó szerint	
		+	-
Valóság-ban	+	2	1
	-	3	6

Precision, Recall, F-Measure

- **Megoldás:**

  Az előbbi confusion mátrixok alapján (az előbbi küszöbszám mellett) számítsuk a modell precision-ját, recall-ját, és F-measure-jét!

Model 1		osztályozó szerint	
		+	-
Valóság-ban	+	0	3
	-	0	9

P: 0 / 0 = ? (Java-ban: NaN)

R: 0 / 3 = 0

F: 2*0*(0/0) / (0 + (0/0)) = ? (NaN)

Model 2		osztályozó szerint	
		+	-
Valóság-ban	+	2	1
	-	3	6

P: 2 / 5 = 0.4

R: 2 / 3 = 0.67

F: 2*0.4*0.67 / (0.4+0.67) = 0.5

AUC (Area Under ROC curve)

- **6. feladat**

  Szerkesszük meg az előbbi modellek ROC görbéjét és számoljuk ki a görbe alatti területet!

Valóság	-	-	+	-	-	-	+	-	-	-	+	-
Model 1	0.2	0.1	0.49	0.26	0.3	0.12	0.31	0.2	0.1	0.32	0.4	0.2
Model 2	0.6	0.51	0.8	0.12	0.54	0.39	0.53	0.46	0.41	0.37	0.49	0.28

AUC (Area Under ROC curve)

- **Megoldás**
- 1. lépés: rendezzük a példányokat a modell kimenete szerint

Valóság	-	-	+	-	-	-	+	-	-	-	+	-
Model 1	0.2	0.1	0.49	0.26	0.3	0.12	0.31	0.2	0.1	0.32	0.4	0.2
Model 2	0.6	0.51	0.8	0.12	0.54	0.39	0.53	0.46	0.41	0.37	0.49	0.28

↓

Valóság	-	-	-	-	-	-	-	-	+	-	+	+
Model 1	0.1	0.1	0.12	0.2	0.2	0.2	0.26	0.3	0.31	0.32	0.4	0.49

Valóság	-	-	-	-	-	-	+	-	+	-	-	+
Model 2	0.12	0.28	0.37	0.39	0.41	0.46	0.49	0.51	0.53	0.54	0.6	0.8

AUC (Area Under ROC curve)

- 2. lépés: meghatározzuk TP-k és FP-k arányát minden értelmes küszöbszámra

Valóság	-	-	-	-	-	-	-	-	+	-	+	+	
Model 1	0.1	0.1	0.12	0.2	0.2	0.2	0.26	0.3	0.31	0.32	0.4	0.49	
TP	3		3	3			3	3	3	2	2	1	0
FP	9		7	6			3	2	1	1	0	0	0
TN	0		2	3			6	7	8	8	9	9	9
FN	0		0	0			0	0	0	1	1	2	3
TPR	1		1	1			1	1	1	2/3	2/3	1/3	0
FPR	1		7/9	6/9			3/9	2/9	1/9	1/9	0	0	0

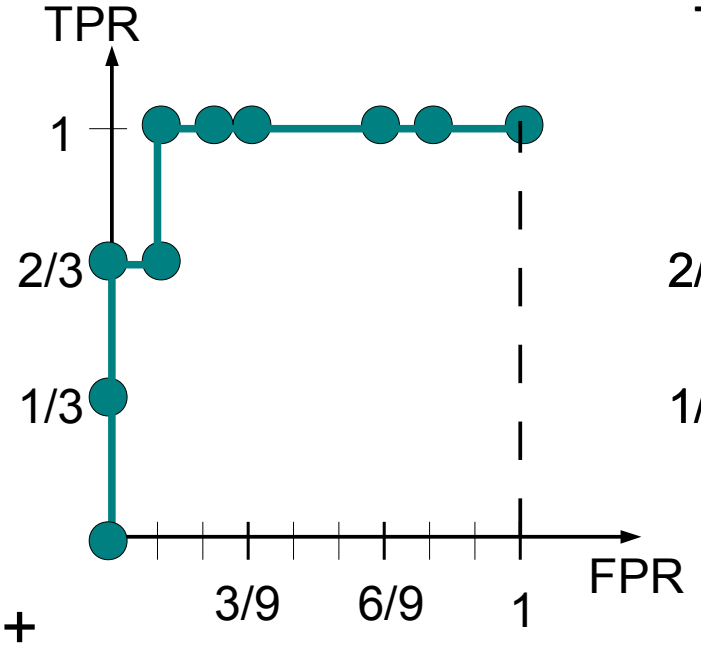
AUC (Area Under ROC curve)

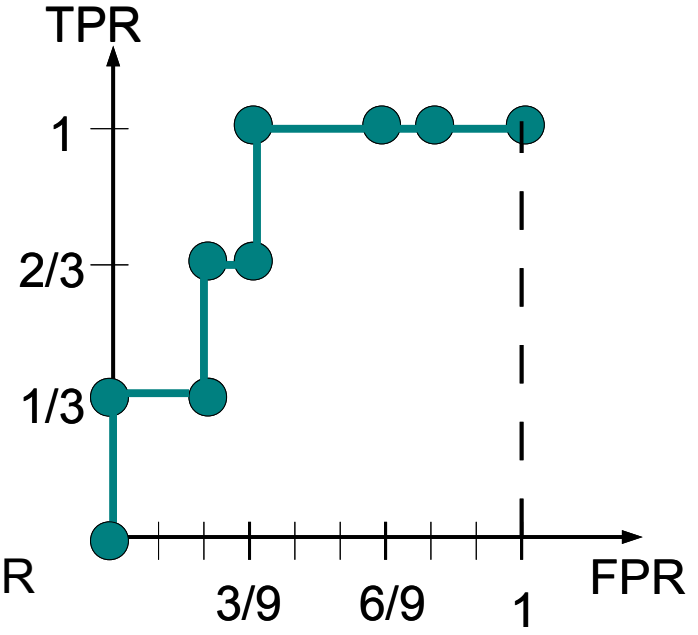
- 2. lépés: meghatározzuk TP-k és FP-k arányát minden értelmes küszöbszámra

Valóság	-	-	-	-	-	-	+	-	+	-	-	+	
Model 2	0.12	0.28	0.37	0.39	0.41	0.46	0.49	0.51	0.53	0.54	0.6	0.8	
TP	3	3	3	3	3	3	3	2	2	1	1	1	0
FP	9	8	7	6	5	4	3	3	2	2	1	0	0
TN	0	1	2	3	4	5	6	6	7	7	8	9	9
FN	0	0	0	0	0	0	0	1	1	2	2	2	3
TPR	1	1	1	1	1	1	1	2/3	2/3	1/3	1/3	1/3	0
FPR	1	8/9	7/9	6/9	5/9	4/9	3/9	3/9	2/9	2/9	1/9	0	0

AUC (Area Under ROC curve)

- 3. lépés: ábrázoljuk TP-k és FP-k arányát





Görbe alatti terület:
(1/9)*(2/3) + (8/9)*1 =
= **26/27**
(1/3)*(2/9) + (2/3)*(1/9) +
+ 1*(6/9) = **22/27**

...	...	...	...	...	...	...	...	...	...	...	...	...	...
TPR	1		1	1			1	1	1	2/3	2/3	1/3	0
FPR	1		7/9	6/9			3/9	2/9	1/9	1/9	0	0	0

...	...	...	...	...	...	...	...	...	...	...	...	...	...
TPR	1	1	1	1	1	1	1	2/3	2/3	1/3	1/3	1/3	0
FPR	1	8/9	7/9	6/9	5/9	4/9	3/9	3/9	2/9	2/9	1/9	0	0

ROC

- **7. feladat**

  a) Hogy néz ki a „tökéletes” modell ROC görbéje? Mennyi a görbe alatti terület?

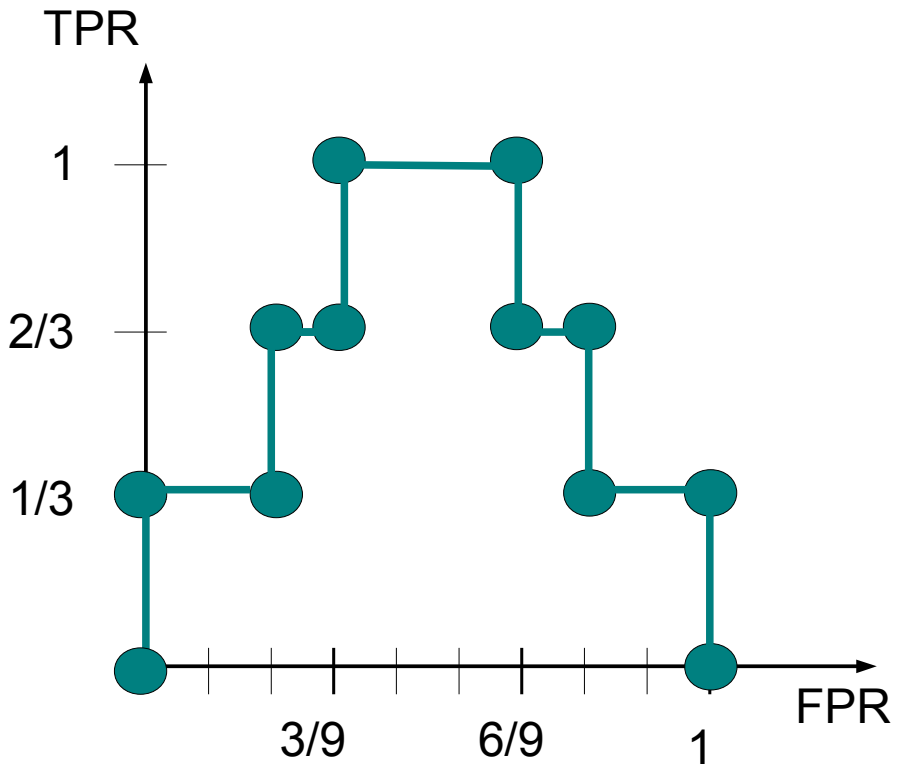
  b) Hogy néz ki egy véletlenszerű előrejelzést adó modell ROC görbéje? Mennyi a görbe alatti területe?

  - (Véletlenszerű előrejelzést adó modell:
    a modell által előrejelzett osztálycímke független
    a példányok valóságos osztálycímkéjétől)

  c) Mi a ROC alatti terület előnye accuracy-val szemben?

ROC

- **8. Feladat:** Lehet-e a ROC görbe ilyen?

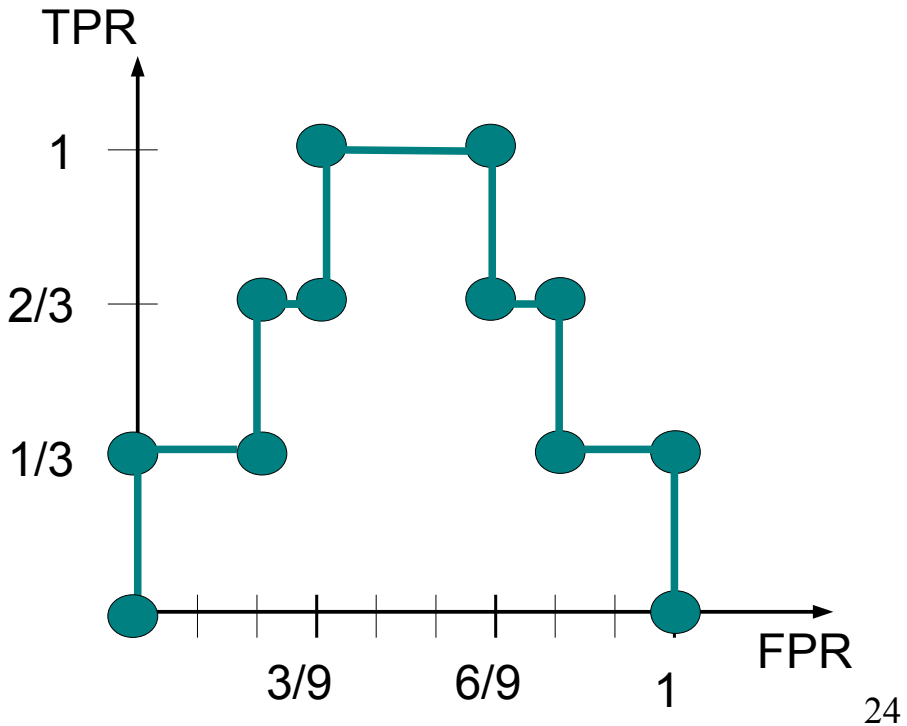


ROC

- **Megoldás**:

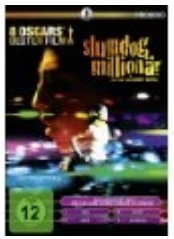
  Nem: a TP-k aránya nem csökkenhet

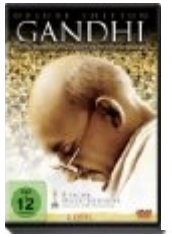
  az FP-k arányának növekedése mellett



Mátrix faktorizáció

Háttér







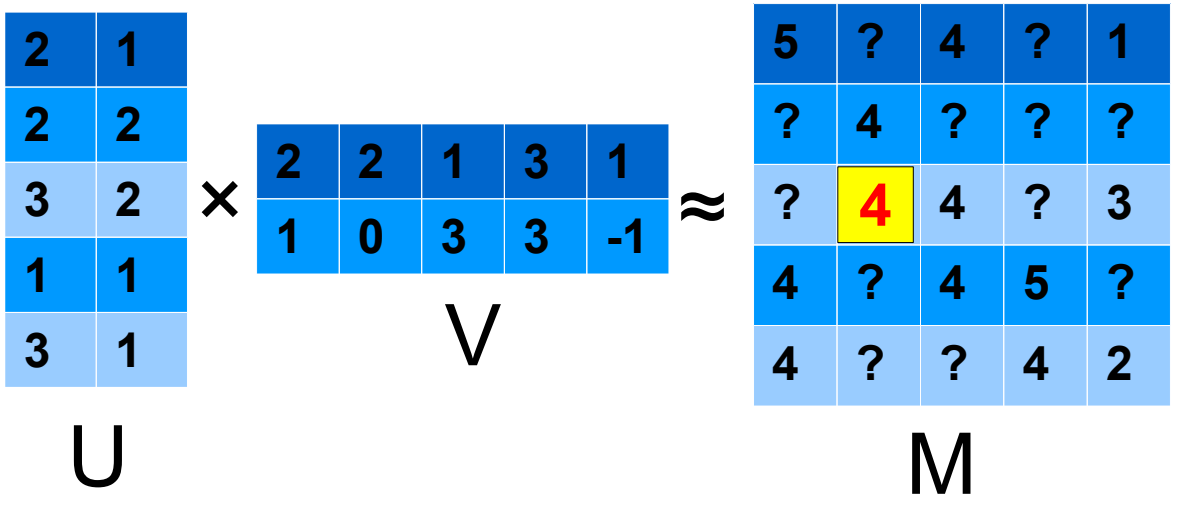

5	?	4	?	...
?	4	?	?	...
?	5	4	?	...
4	?	4	5	...
...	...	...	...	...

Online kereskedelmi rendszerben nyilván tartjuk, hogy melyik felhasználó milyen terméket (filmet) vásárolt és arról milyen visszajelzést adott (mennyire volt elégedett a termékkel). Ezek alapján kívánunk személyre szabott reklámokat küldeni a felhasználóknak.

Mátrix faktorizáció algoritmusa

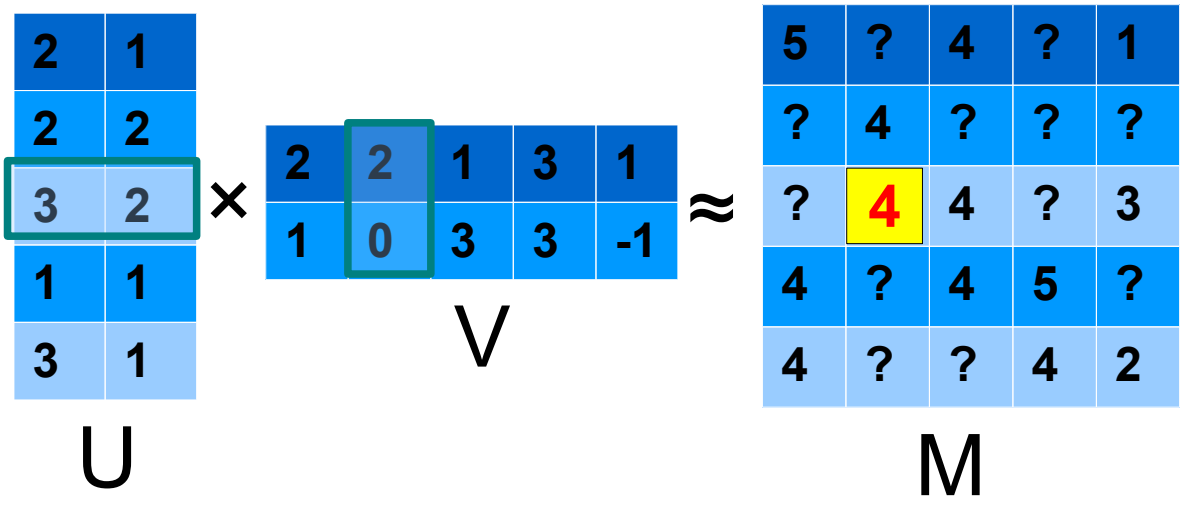
9. Feladat:

Hajtsa végre a tanult, ritka mátrixokat faktorizáló iteratív algoritmus egy javítási lépést az ábrában jelölt elemmel! Adott: ε = 0.05



Mátrix faktorizáció algoritmusa

Megoldás:



(3x2) + (2x0) = 6

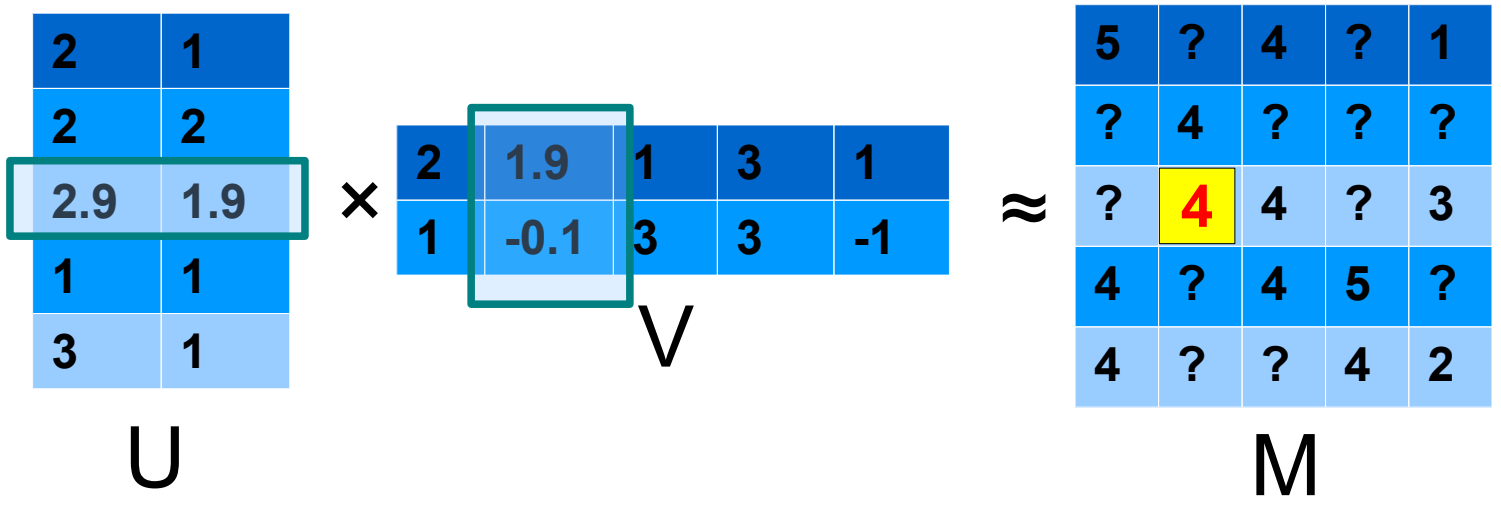
nagyobb, mint a választott elem → csökkentjük a U és V aktuális sorában ill. oszlopában lévő számokat

6 – 4 = 2, ε = 0.05 → 0.05 * 2 = 0.1-gyel csökkentjük a két jelölt vektor elemeit

- **Megjegyzés:** valós algoritmusok ennél valamivel összetettebb frissítési lépést használnak, lásd pl. Yehuda Koren, Robert Bell, Chris Volinsky: *Matrix factorization techniques for Recommender Systems*, IEEE Computer, Aug. 2009

Mátrix faktorizáció algoritmusa

Megoldás:



(3x2) + (2x0) = 6

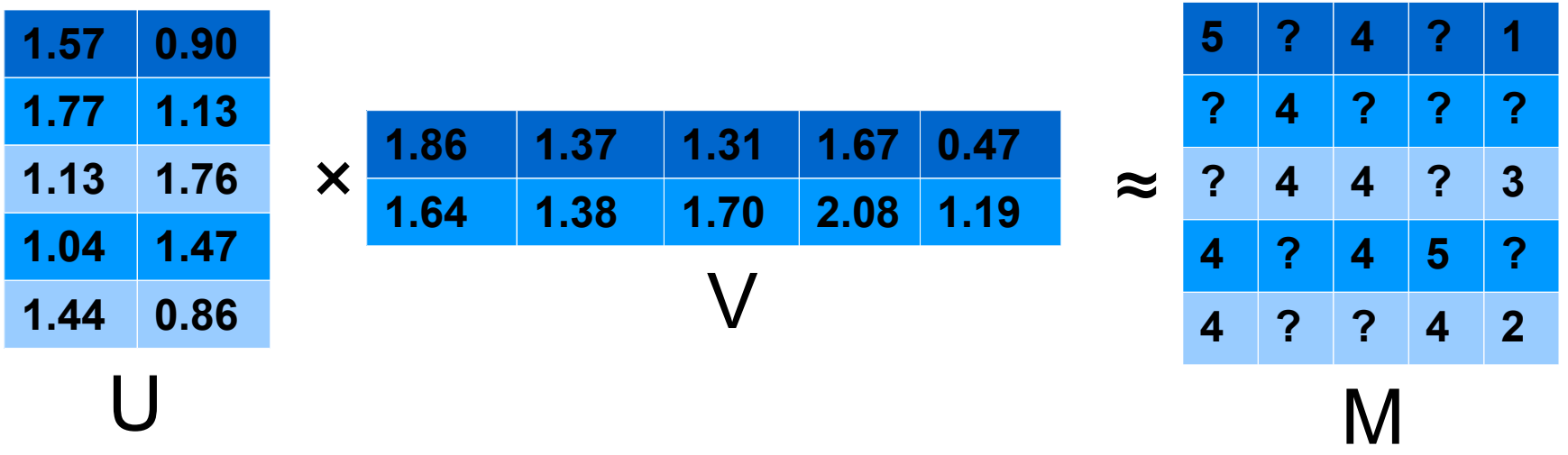
nagyobb, mint a választott elem → csökkentjük a U és V aktuális sorában ill. oszlopában lévő számokat

6 – 4 = 2, ε = 0.05 → 0.05 * 2 = 0.1-gyel csökkentjük a két jelölt vektor elemeit

- **Megjegyzés:** valós algoritmusok ennél valamivel összetettebb frissítési lépést használnak, lásd pl. Yehuda Koren, Robert Bell, Chris Volinsky: *Matrix factorization techniques for Recommender Systems*, IEEE Computer, Aug. 2009

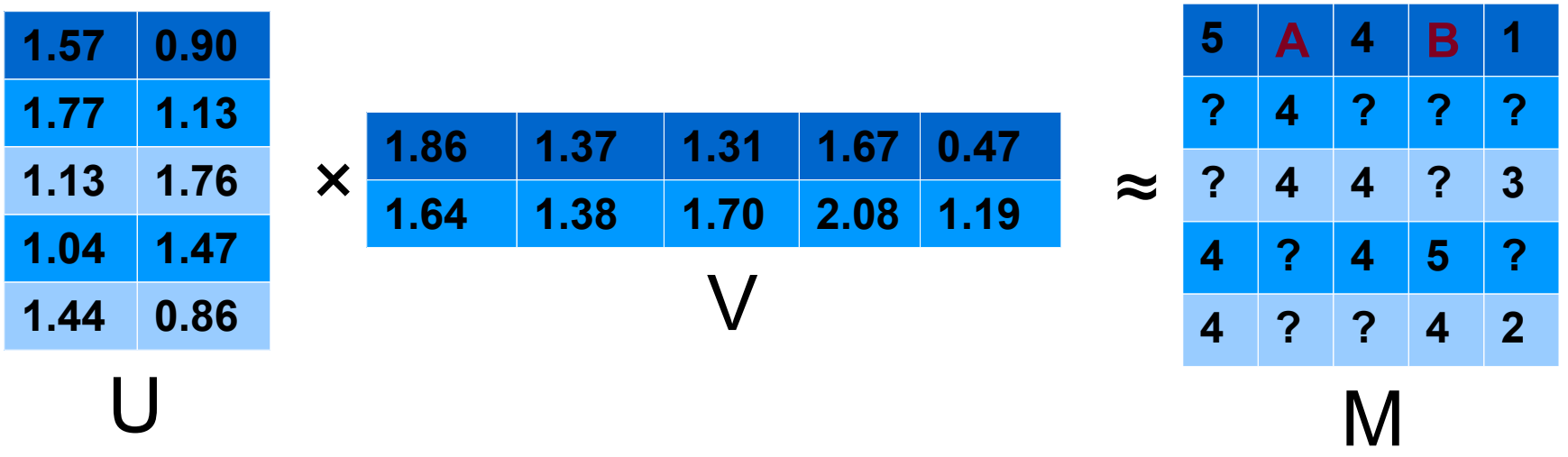
Mátrix faktorizáció algoritmusa

10. Feladat:
Kellőképpen sok iteráció után az algoritmus az alábbi *U* és *V* mátrixokat találja meg. Ezen *U* és *V* mátrixok alapján becsüljük meg *M* első sorbeli hiányzó értékeit!



Mátrix faktorizáció algoritmusa

Megoldás:
M hiányzó elemeinek becslése az *U* és *V* mátrix szorzataként adódik.



A = (1.57*1.37)+(0.9*1.38) = 3.39

B = (1.57*1.67)+(0.9*2.08) = 4.49

Mátrix faktorizáció algoritmusa

- **11. feladat:**

  a) Milyen probléma adódhat ha ε-t túl kicsinek vagy túl nagynak választjuk?

  b) Hogyan kerülhetjük el a túltanulást (overfitting-et) ?

Mátrix faktorizáció algoritmusa

- **Megoldás**

  a) Milyen probléma adódhat ha ε-t túl kicsinek vagy túl nagynak választjuk?

  Ha ε túl kicsi → túl keveset csökkentünk →
  lassan találjuk meg a legjobb közelítést

  Ha ε túl nagy → túl sokat csökkentünk →
  túl nagy léptekben haladunk, lehet, hogy nem jutunk az optimumhoz, hanem „átlépünk” felette

Mátrix faktorizáció algoritmusa

- **Megoldás:**

  b) Hogyan kerülhetjük el a túltanulást (overfitting-et) ?

  - Hold-out módszer:

    Az ismert elemek (tanító adatok) közül kiválasztunk néhányat, amelyeket „félre teszünk”, azaz amelyek sohasem lesznek a „kiválasztott” elemek a javítási lépések során. Egy-egy javítás végrehajtása után megnézzük nem romlott-e (sokat) a „félretett elemek” becslése.

  - Regularizáció

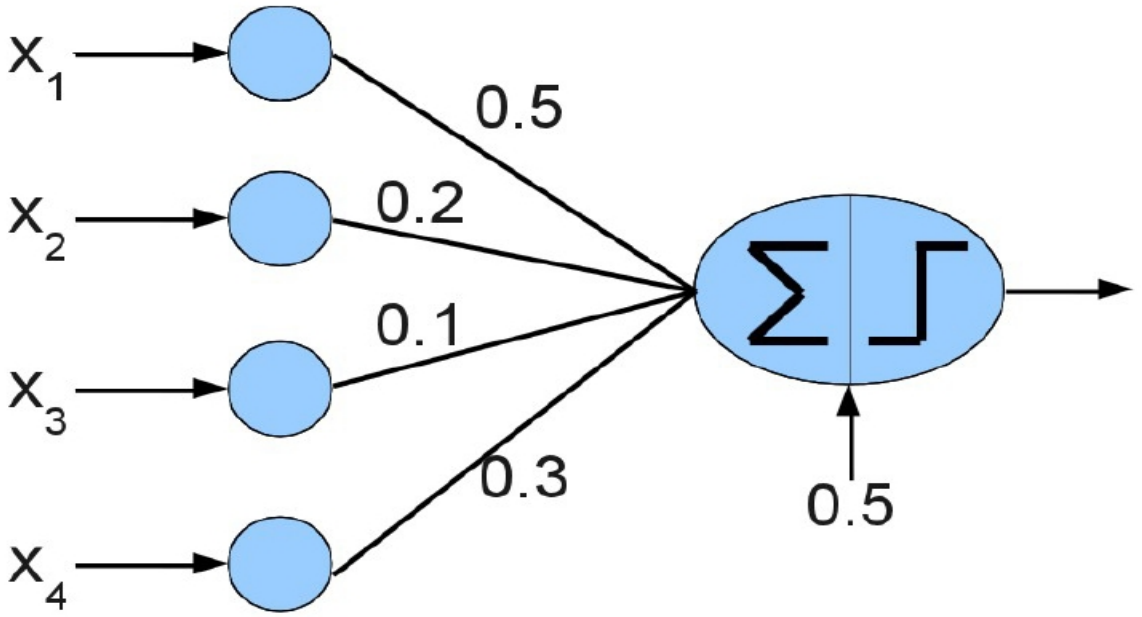
    Pl. U és V oszlopainak ill. sorainak számát viszonylag kicsire választjuk

Osztályozó algoritmusok

- Perceptron algoritmus
- Naive Bayes
- Döntési fák

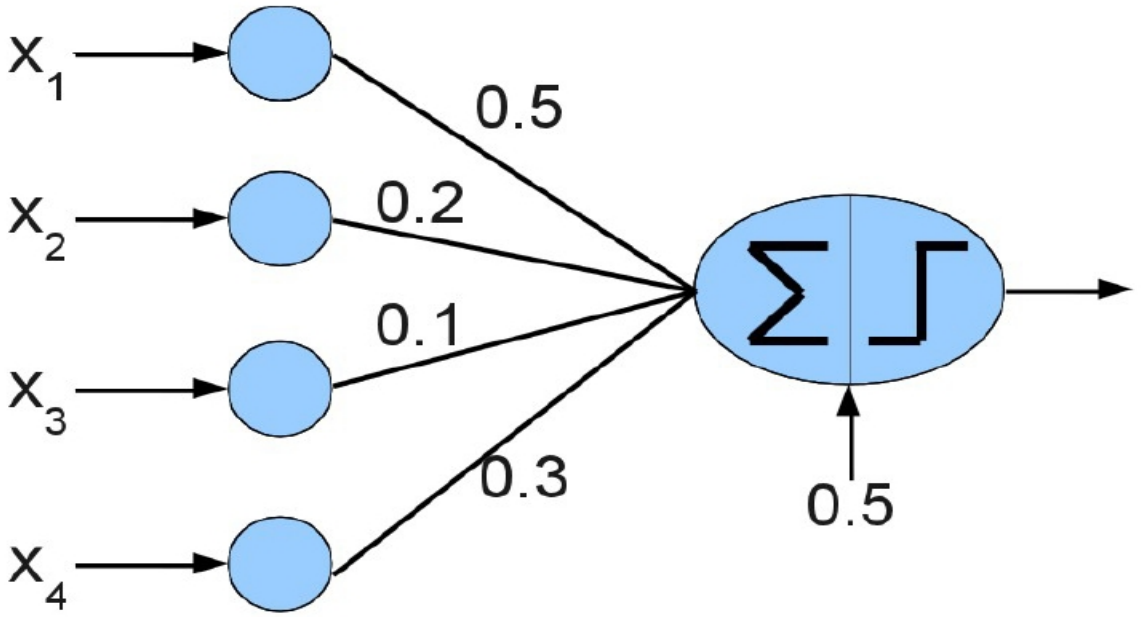
Perceptron algoritmus

- **12. Feladat:** Adott az alábbi perceptron. A két osztálycímkét +1-gyel és -1-gyel jelöljük.
  a) Hogyan osztályozza ez a perceptron az
  (x_1=1, x_2=-0.8, x_3=-0.3, x_4=1.5) példányt?



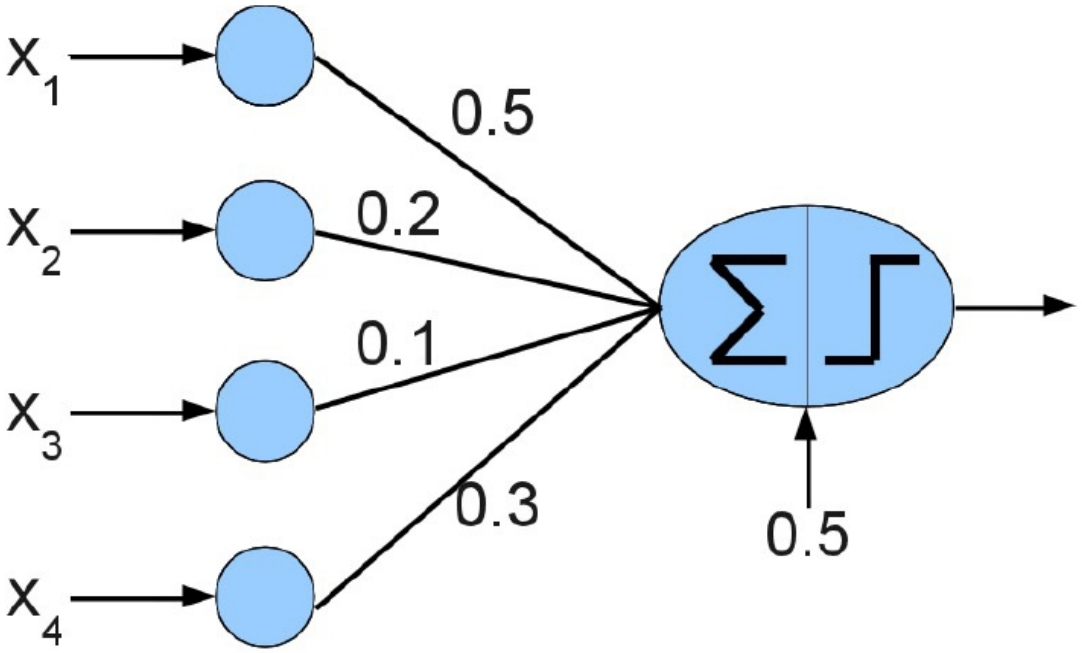
Perceptron algoritmus

- **Megoldás:**
  1*0.5+(-0.8)*0.2+(-0.3)*0.1+1.5*0.3 – 0.5 = 0.26, ami nagyobb 0-nál → +1 osztályba sorolja



Perceptron algoritmus

- **12. Feladat:** b) Az (x_1=1, x_2=-0.8, x_3=-0.3, x_4=1.5) valódi osztálycímkéje -1, és ezt a példányt a perceptron tanításához használjuk, λ = 0.1 . Hajtsa végre a tanítási algoritmus egy javító lépést! Változnak-e az élek súlyai, ha igen, hogyan?

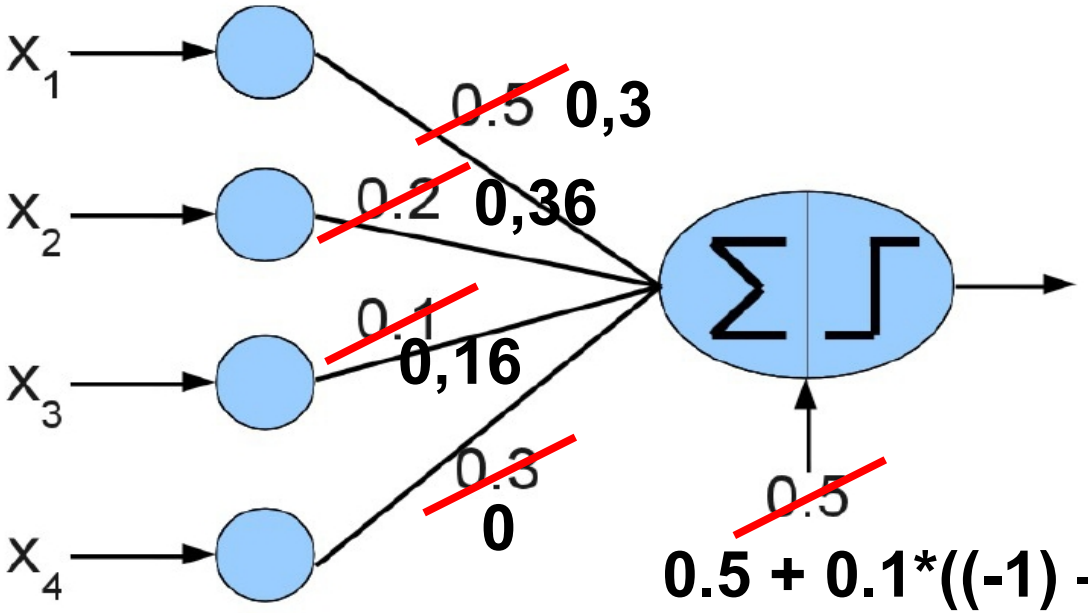


Perceptron algoritmus

- **Megoldás:**
  b) i-dik új súly = i-dik régi súly +
  + λ*(valódi osztálycímke – felismert osztálycímke)*x_i

  Tehát: $w_1^{új} = 0.5 + 0.1*((-1) – (+1))*1 = 0.3$

  $w_2^{új} = 0.2 + 0.1*((-1) – (+1))*(-0.8) = 0.36$



Naive Bayes

- **13. Feladat**

  Étteremláncunk TelePocak néven új akciót tervez indítani. Néhány embert megkérdeztünk arról, hogy érdekelné őket az akció. Ez alapján készítendő egy olyan osztályozó, amely eldönti, hogy kiket érdekelhet még az akció.

  Adott megkérdezett emberek adatait tartalmazó adatbázis (következő lap), készítsen egy Naive Bayes osztályozót és ezzel osztályozza a további potenciális ügyfeleket.

Naive Bayes

A megkérdezés eredménye

Sorsz.	Életkor	Testsúly	Sportol	Érdekli-e az akció
1	fiatal	alacsony	igen	igen
2	idős	közepes	nem	nem
3	középkorú	magas	nem	igen
4	idős	közepes	igen	nem
5	fiatal	magas	nem	igen
6	középkorú	alacsony	nem	nem
7	idős	alacsony	nem	nem
8	fiatal	közepes	nem	igen
9	középkorú	magas	igen	igen
10	idős	közepes	igen	nem

Naive Bayes

Osztályozandó potenciális ügyfelek

Sorsz.	Életkor	Testsúly	Sportol	Érdekli-e az akció
11	fiatal	közepes	igen	?
12	idős	alacsony	igen	?
13	középkorú	közepes	igen	?
14	középkorú	közepes	nem	?

Naive Bayes

- **Megoldás**
- Jelölés:
  Életkor → X, Testsúly → Y, sportol → Z, érdekli-e → C
- P(C | X, Y, Z) -t szeretnénk becsülni.
- Bayes-tétel szerint P(C | X, Y, Z) egyenlő

$$\frac{P(X, Y, Z \mid C) * P(C)}{P(X, Y, Z)}$$

- Elég a számlálót kiszámolnunk (lásd később)

Naive Bayes

- Naive Bayes feltételezi, hogy az attribútumok adott osztályváltozó („érdekli-e”) esetén feltételesen függetlenek:

  $P(X, Y, Z \mid C) = P(X \mid C) * P(Y \mid C) * P(Z \mid C)$

- Ezért tehát $P(X \mid C)$-t, $P(Y \mid C)$-t, $P(Z \mid C)$-t és $P(C)$-t kell becsülnünk a megadott tanító adatok alapján.

Naive Bayes

Sor.	X	Y	Z	C
1	fiatal	alacsony	igen	igen
2	idős	közepes	nem	nem
3	középkorú	magas	nem	igen
4	idős	közepes	igen	nem
5	fiatal	magas	nem	igen
6	középkorú	alacsony	nem	nem
7	idős	alacsony	nem	nem
8	fiatal	közepes	nem	igen
9	középkorú	magas	igen	igen
10	idős	közepes	igen	nem

P(X=fiatal | C=igen) = 3/5
P(X=idős | C=igen) = 0
P(X=középk. | C=igen) =2/5

P(X=fiatal | C=nem) = 0
P(X=idős | C=nem) = 4/5
P(X=középk. | C=nem) = 1/5

P(Y=alacsony | C=igen) = 1/5
P(Y=közepes | C=igen) = 1/5
P(Y=magas | C=igen) =3/5

P(Y=alacsony | C=nem) = 2/5
P(Y=közepes | C=nem) = 3/5
P(Y=magas | C=nem) = 0

Naive Bayes

- **Megoldás (folyt.)**

Sor.	X	Y	Z	C
1	fiatal	alacsony	igen	igen
2	idős	közepes	nem	nem
3	középkorú	magas	nem	igen
4	idős	közepes	igen	nem
5	fiatal	magas	nem	igen
6	középkorú	alacsony	nem	nem
7	idős	alacsony	nem	nem
8	Fiatal	közepes	nem	igen
9	középkorú	magas	igen	igen
10	idős	közepes	igen	nem

P(Z=igen | C=igen) = 2/5
P(Z=nem | C=igen) = 3/5

P(Z=igen | C=nem) = 2/5
P(Z=nem | C=nem) = 3/5

P(C=igen) = 5/10
P(C=nem) = 5/10

Naive Bayes

Sorsz.	X	Y	Z	C
11	fiatal	közepes	igen	?

- A becsült valószínűségek alapján kiszámoljuk, hogy melyik osztálynak mekkora a valószínűsége az osztályozandó példányok esetében

$$P(C=igen \mid X=fiatal, Y=közepes, Z=igen) =$$

$$= \frac{P(C=igen)*P(X=fiatal|C=igen)*P(Y=közepes|C=igen)*P(Z=igen|C=igen)}{P(X=fiatal, Y=közepes, Z=igen)} =$$

$$= \frac{5/10 * 3/5 * 1/5 * 2/5}{P(X=fiatal, Y=közepes, Z=igen)} = \frac{6 / 250}{P(X=fiatal, Y=közepes, Z=igen)}$$

Hasonlóképpen: $P(C=nem \mid X=fiatal, Y=közepes, Z=igen) =$

$$= \frac{5/10 * 0 * 3/5 * 2/5}{P(X=fiatal, Y=közepes, Z=igen)} = \frac{0}{P(X=fiatal, Y=közepes, Z=igen)}$$

Döntési fa

- **14. Feladat:**
  A TelePocak akció adatbázisa alapján építendő egy döntési fa, amely képes eldönteni, hogy mely ügyfeleket érdekli potenciálisan az akció. A döntési fa építésekor a GINI-index szerinti javítás alapján válassza ki a legjobb vágást, csak multiway split-eket vizsgáljon. A vágás entrópiáját (SplitINFO) ne vegye figyelembe.

Döntési fa

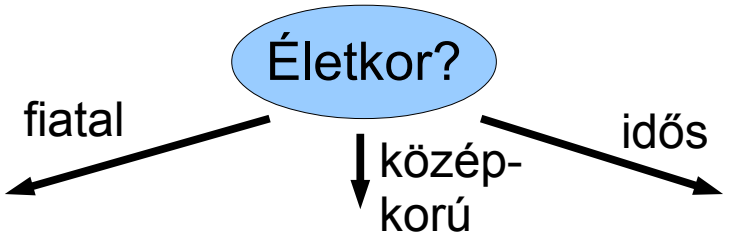
- **Megoldás**

Sorsz.	Életkor	Testsúly	Sportol	Érdekli-e az akció
1	fiatal	alacsony	igen	igen
2	idős	közepes	nem	nem
3	középkorú	magas	nem	igen
4	idős	közepes	igen	nem
5	fiatal	magas	nem	igen
6	középkorú	alacsony	nem	nem
7	idős	alacsony	nem	nem
8	fiatal	közepes	nem	igen
9	középkorú	magas	igen	igen
10	idős	közepes	igen	nem

Kezdeti GINI: $1- ((5/10)^2+(5/10)^2) = 0.50$

Döntési fa

Sorsz.	Életkor	...	.. .	Érdekli-e az akció
1	fiatal			igen
2	idős			nem
3	középkorú			igen
4	idős			nem
5	fiatal			igen
6	középkorú			nem
7	idős			nem
8	fiatal			igen
9	középkorú			igen
10	idős			nem



Kezdeti GINI: 0.48

Életkor szerinti vágás GINI-jének számítása:

GINI(fiatal) = $1-((3/3)^2+(0/3)^2) = 0$

GINI(köz.) = $1-((2/3)^2+(1/3)^2) = 4/9$

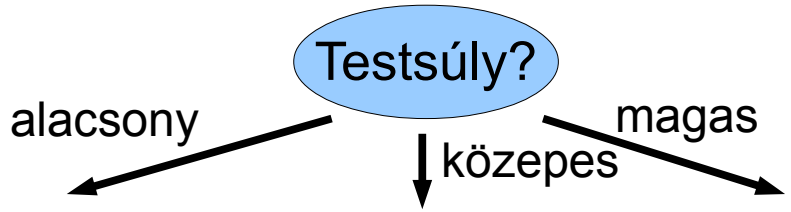
GINI(idős) = $1-((0/4)^2+(4/4)^2) = 0$

Életkor szerinti vágás GINI-je:
$0.3*0+0.3*(4/9)+0.4*0 = 0.13$

GINI nyereség életkor szerinti vágás esetén: $0.50 - 0.13 = 0.37$

Döntési fa

Sorsz.	...	Testsúly	...	Érdekli-e az akció
1		alacsony		igen
2		közepes		nem
3		magas		igen
4		közepes		nem
5		magas		igen
6		alacsony		nem
7		alacsony		nem
8		közepes		igen
9		magas		igen
10		közepes		nem



Kezdeti GINI: 0.48

Testsúly szerinti vágás GINI-jének számítása:

GINI(alacsony) =
$= 1-((1/3)^2+(2/3)^2) = 4/9$

GINI(közepes) =
$= 1-((1/4)^2+(3/4)^2) = 6/16$

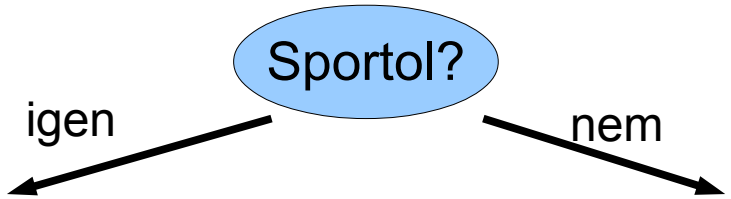
GINI(magas) =
$= 1-((3/3)^2+(0/3)^2) = 0$

Testsúly szerinti vágás GINI-je:
$0.3*(4/9)+0.4*(6/16)+0.3*0 = 0.28$

GINI nyereség testsúly szerinti vágás esetén: $0.50 - 0.28 = 0.22$

Döntési fa

Sorsz.	...	...	Sportol	Érdekli-e az akció
1			igen	igen
2			nem	nem
3			nem	igen
4			igen	nem
5			nem	igen
6			nem	nem
7			nem	nem
8			nem	igen
9			igen	igen
10			igen	nem



Kezdeti GINI: 0.48

Sportol szerinti vágás GINI-jének számítása:

GINI(sportol=igen) =
$= 1-((2/4)^2+(2/4)^2) = 0.5$

GINI(sportol=nem) =
$= 1-((3/6)^2+(3/6)^2) = 0.5$

Sportol szerinti vágás GINI-je:
0.4*0.5+0.6*0.5 = 0.5

GINI nyereség „sportol” szerinti vágás esetén: 0.50 – 0.50 = 0

Döntési fa

- A nyereség a vizsgált vágások estén:
  - **Életkor: 0.37 (legjobb vágás)**
  - Testsúly: 0.22
  - Sportol: 0

Döntési fa

fiatal

Sorsz.	Életkor	…	…	Érdekli-e az akció
1	fiatal			igen
5	fiatal			igen
8	fiatal			igen

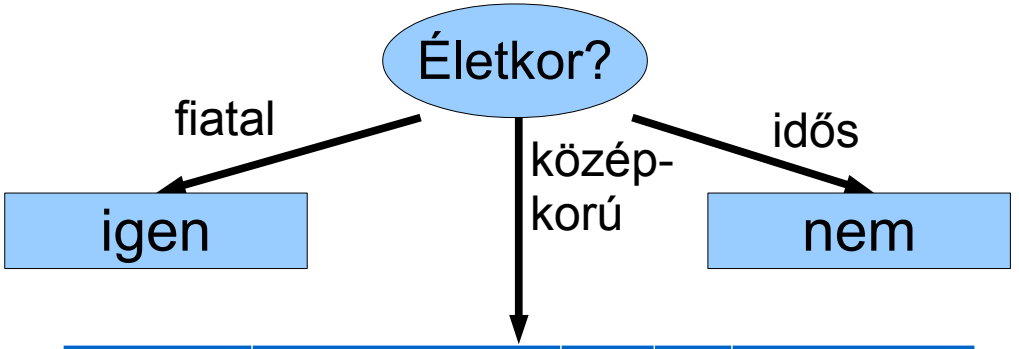
idős

Sorsz.	Életkor	…	…	Érdekli-e az akció
2	idős			nem
4	idős			nem
7	idős			nem
10	idős			nem

középkorú

Sorsz.	Életkor	…	.. .	Érdekli-e az akció
3	középkorú			igen
6	középkorú			nem
9	középkorú			igen

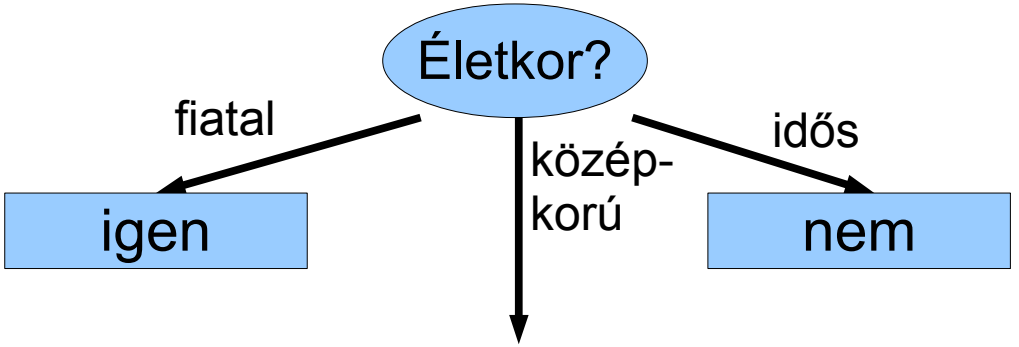
Döntési fa



Sorsz.	Életkor	…	.. .	Érdekli-e az akció
3	középkorú			igen
6	középkorú			nem
9	középkorú			igen

- A két szélső ágon minden példány ugyanazon osztályba tartozik → osztály nevével címkézett levél
- A középső ágon tovább vágunk
  (ha nincs más leállási feltétel, ami teljesül)

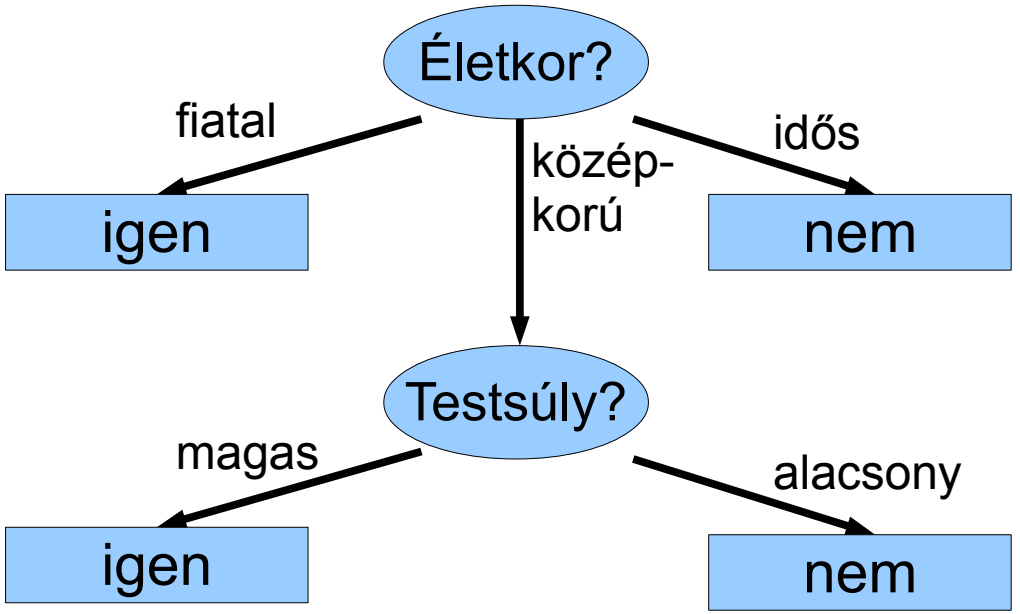
Döntési fa



Sorsz.	Életkor	Testsúly	Sportol	Érdekli-e az akció
3	középkorú	magas	nem	igen
6	középkorú	alacsony	nem	nem
9	középkorú	magas	igen	igen

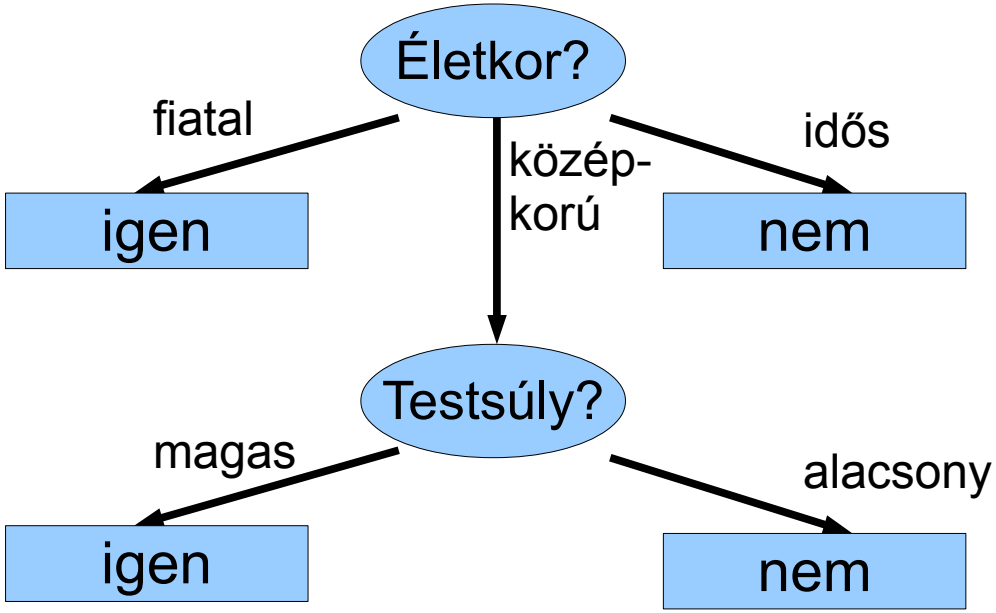
- Vágjunk a testsúly szerint! (Teljesen tiszta vágást eredményez → GINI-növekedést is maximalizálja)

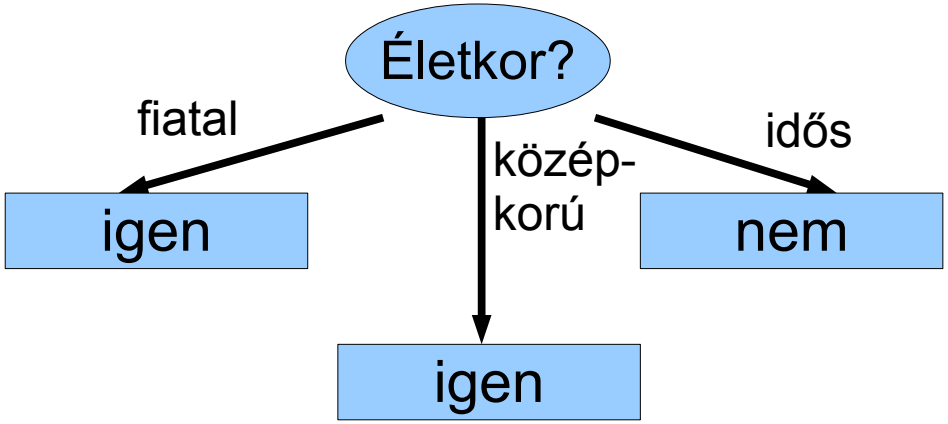
Döntési fa


Életkor?
fiatal
középkorú
idős
igen
Testsúly?
nem
magas
alacsony
igen
nem

Döntési fa - MDL

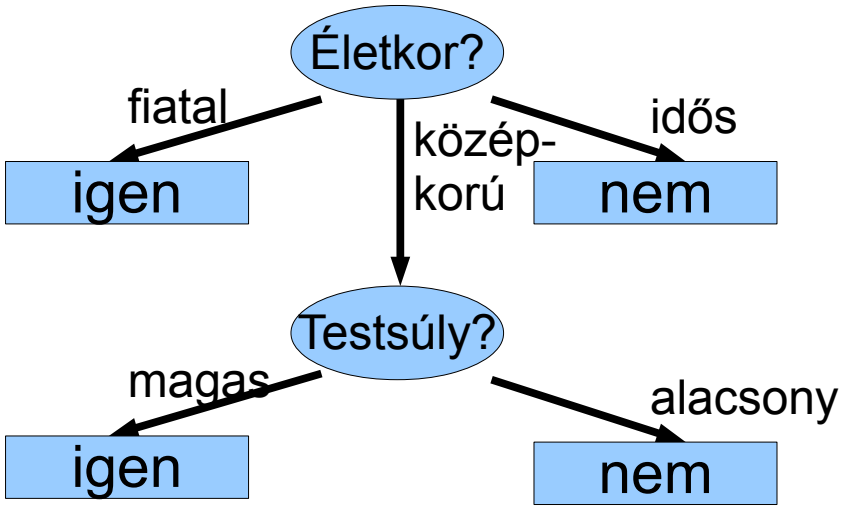
- **15. Feladat:** Az előbbi adatbázishoz egy másik döntési fát is építettünk. A második döntési fa építésekor kikötöttük, hogy minden ágra legalább 3 példány jusson (lásd alább). Minimum Description Length (MDL) elv alapján a két fa közül várhatóan melyik fa fogja jobban osztályozni az ismeretlen osztályba tartozó példányokat? Tételezzük fel, hogy 1-1 osztályozási hiba description length-je 2 egység, a döntési fa minden nem-levél csomópontjának description length-je 3 egység, minden levél csomópont description length-je 1 egység.

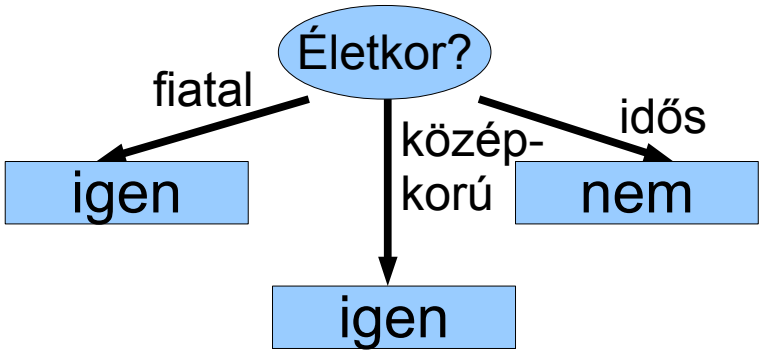




Döntési fa - MDL

- **Megoldás:** A description length-ek:





2*3	3	(nem-levél csomópontok)
+4*1	+3*1	(levél csomópontok)
+0*2	+2*1	(osztályozási hibák)
= 10	**= 8**	

- MDL szerint a második (jobb oldali) fát preferáljuk, mert annak description length-e kisebb.

Döntési fa

- **16. Feladat:**
  a) Milyen esetekben fontos a vágás entrópiájának (SplitINFO) figyelembe vétele?

  b) Mit jelent a ReducedErrorPruning? Mikor használjuk?

  c) Miért szokás kikötni, hogy csak olyan vágást engedünk, amely során minden ágra legalább *N* darab példány kerül? (Az *N* egy felhasználó által állítható paraméter.)

  d) Használhatunk-e a GINI helyett más mértéket annak eldöntésére, hogy mi a legjobb vágás?

További feladatok

Waveletek

- **17. feladat:**
  Számolja ki az alábbi jel Haar-Wavelet transzformáltját!

4	**8**	**12**	**20**	**18**	**16**	**15**	**13**	**12**	**8**	**6**	**4**	**3**	**2**	**1**	**0**

Waveletek

- **Megoldás:**
  (Megjegyzés: az egyszerűség kedvéért a számítás során a skálázástól eltekintettünk.)

4	8	12	20	18	16	15	13	12	8	6	4	3	2	1	0
6	16	17	14	10	5	2.5	0.5	*-4*	*-8*	*2*	*2*	*4*	*2*	*1*	*1*
11	15.5	7.5	1.5	*-10*	*3*	*5*	*2*	*-4*	*-8*	*2*	*2*	*4*	*2*	*1*	*1*
13.25	4.5	*-4.5*	*6*	*-10*	*3*	*5*	*2*	*-4*	*-8*	*2*	*2*	*4*	*2*	*1*	*1*
8.875	*8.75*	*-4.5*	*6*	*-10*	*3*	*5*	*2*	*-4*	*-8*	*2*	*2*	*4*	*2*	*1*	*1*