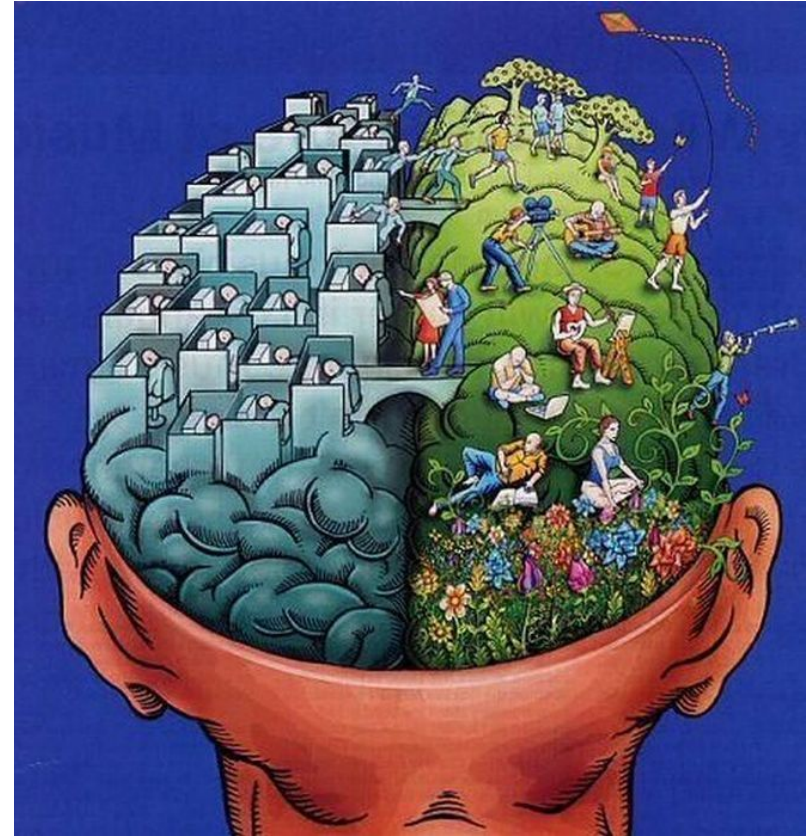


Mesterséges Intelligencia MI

Gépi tanulás elemei

Dobrowiecki Tadeusz
Eredics Péter, és mások

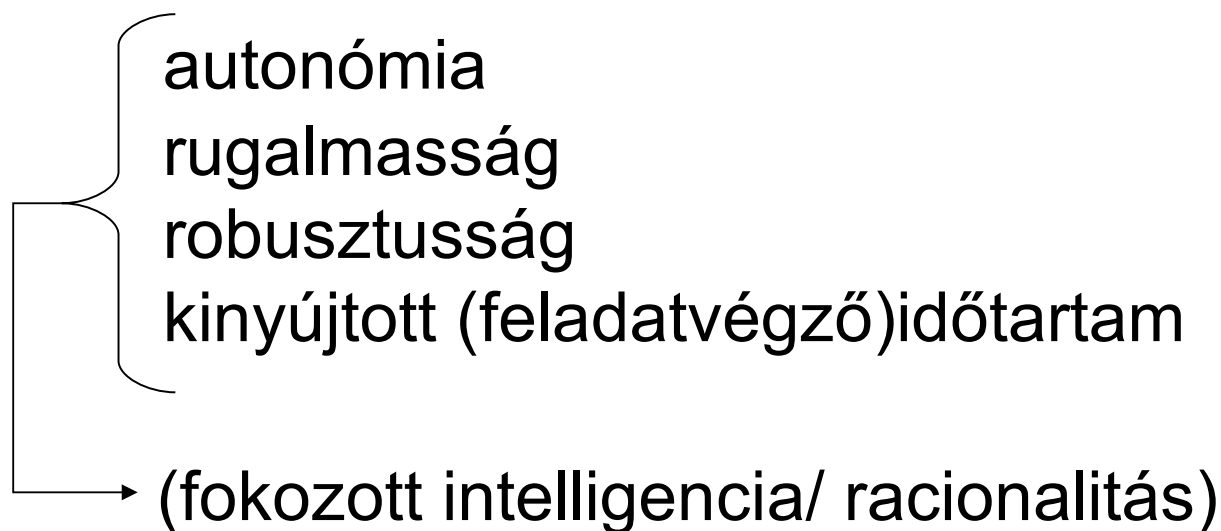


BME I.E. 437, 463-28-99

dobrowiecki@mit.bme.hu,

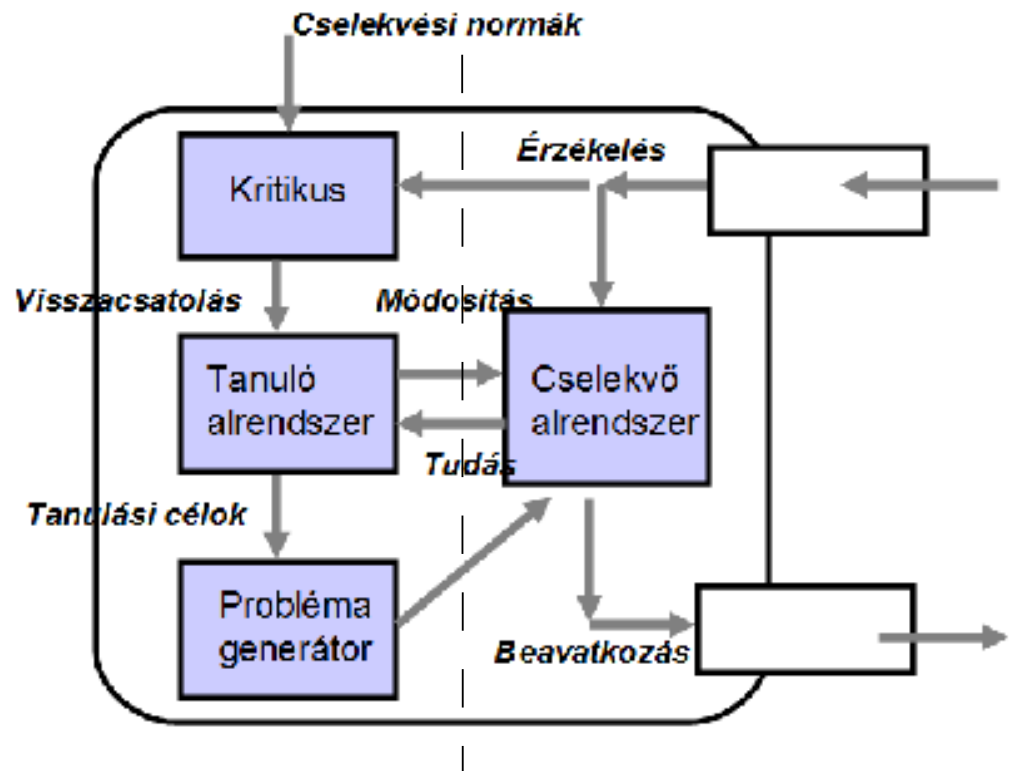
<http://www.mit.bme.hu/general/staff/tade>

Egy intelligens rendszer (gyorsan) tanul.
Mit várunk el ettől?



Tanuló ágens

- cselekvő alrendszer (+)
- tanuló alrendszer
- kritikus (+)
- problémagenerátor



Milyen feladatok körében?

- a rendszerek, folyamatok fizikai, kémiai, stb. modellje bonyolult vagy nem is ismert
- A feladatról nagy mennyiségű adat áll rendelkezésre
- A feladatok többsége emberi intelligenciával nagyon hatékonyan megoldható

Tanuló alrendszer tervezése:

A cselekvő alrendszer (eddiggi ágens) mely **részeit** kell javítani? (érzékelés? TB? következtetés? ...)

Milyen tudás **reprezentációt** használnak ezek az alrendszerek? (tábla? logika? valószínűség? ...)

Milyen a **priori információ** áll rendelkezésünkre?

Milyen **visszacsatolásra** van lehetőség? (mi a helyes eredmény?)

felügyelt tanulás egy komponensnek mind a bemenetét, mind a kimenetét észlelni tudjuk

megerősítéses tanulás az ágens az általa végrehajtott tevékenység csak bizonyos értékelését kapja meg = jutalom, büntetés, **megerősítés**

felügyelet nélküli tanulás semmilyen utalás sem áll rendelkezésünkre a helyes kimenetről (az észlelések közötti összefüggések tanulása)

A különböző feladatok közös magja

A cselekvő alrendszer minden komponense =
egy-egy függvény-kapcsolat (leképezés), l. pl.

$$cs_t = f(m_t, cs_{t-1}, TB_t)$$

**Minden tanulás felfogható úgy, mint egy függvény (leképezés)
valamilyen reprezentációjának a megtanulása.**

Induktív felügyelt tanulás:

tanulási példa: $(x, y = f(x))$ adatkár, ahol $f(x)$ ismeretlen

$h(x) = f(x)$, x – egy ismert példán

$h(x') \approx f(x')$, x' – a tanulás közben még nem látott esetben
(**általánosító képesség**)

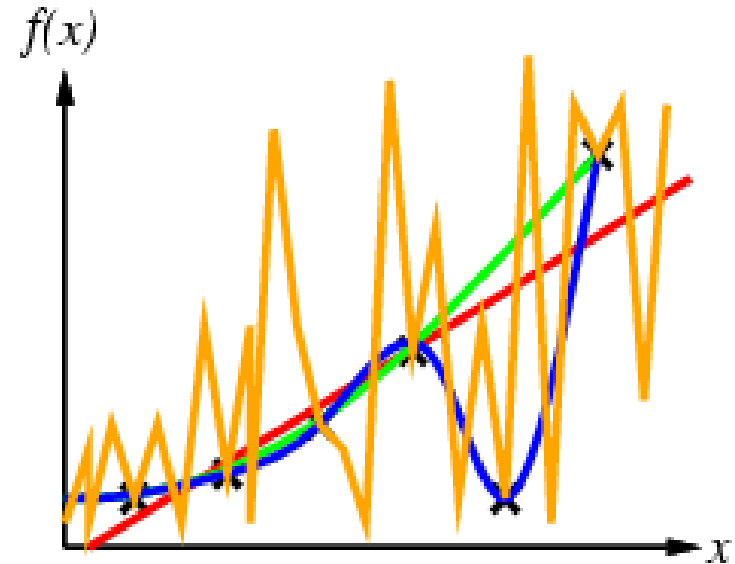
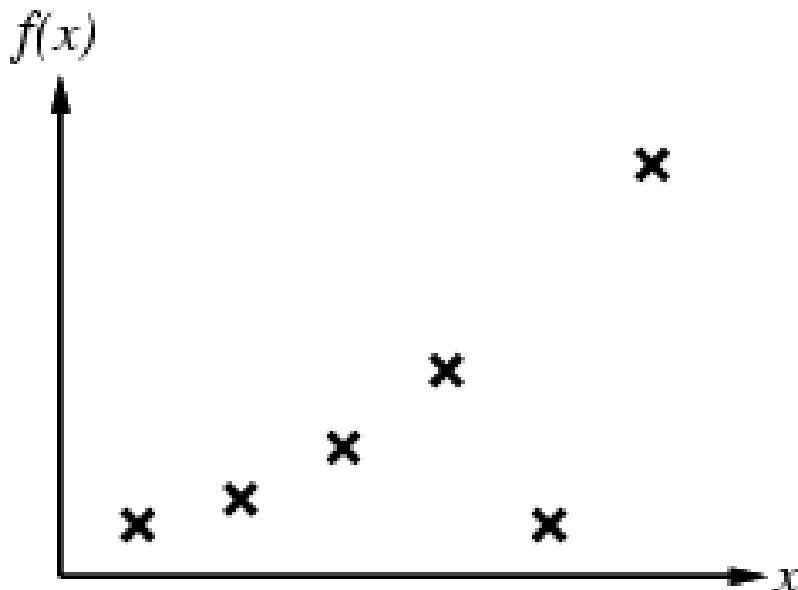
Tiszta induktív következtetés (indukció):

az f -re vonatkozó minták egy halmaza alapján (tanító halmaz),
adon meg egy olyan h függvényt (**hipotézist**), amely
tulajdonságaiban közelíti az f -et.

Elfogultság

A példának való megfelelésen túl (= **konzisztens hipotézisek**) előnyben részesítjük az egyik vagy a másik hipotézist.

Mivel mindig nagy számú lehetséges hipotézis létezik, az összes tanuló algoritmus mutat valamilyen elfogultságot. Miért?

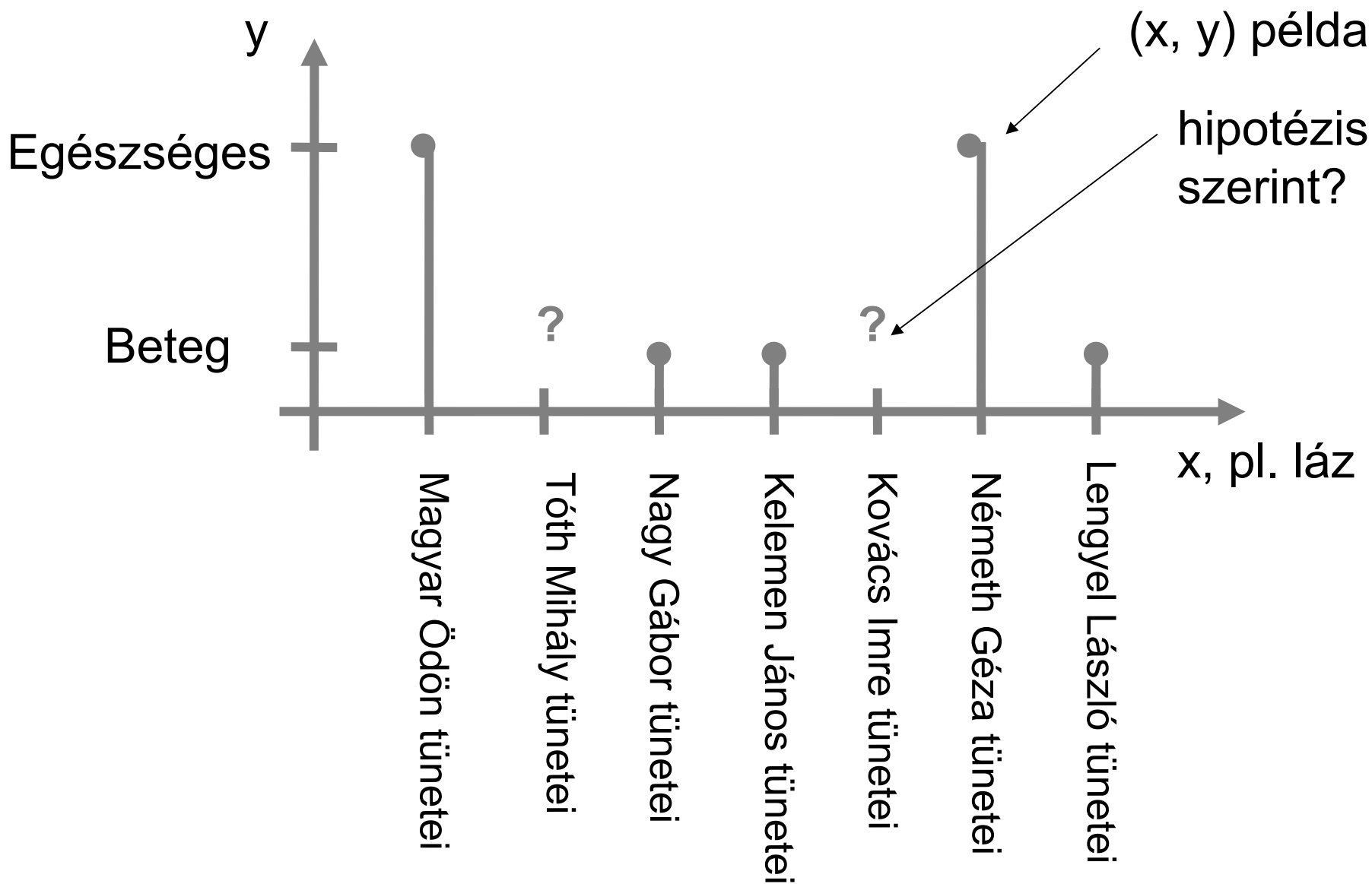


A legfontosabb döntés a tanuló ágens tervezője számára:
a kívánt függvény (és a hipotézisek) **reprezentációjának**
a megválasztása (az elfogultsággal együtt, ez a tanuló
algoritmus jellegét határozza meg, eldöntheti, hogy **a**
probléma egyáltalán **kezelhető-e**).

Kompromisszum: (cél: legyen tömör és jól általánosító)
a **kifejezőképesség** és a **hatékonyság** között.

Kifejezőképesség: a kívánt függvény jól (egyáltalán)
ábrázolható-e.

Hatékonyság: a tanulási probléma kezelhető lesz-e egy adott
reprezentációs nyelv választás esetén.
(tanulási algoritmikus tár-, idő-komplexitása)
(tanulás: valamilyen keresés hipotézisek terében)



$h(\text{Kovács Imre}) = \text{beteg} / \text{egészséges?}$

... ..

Hogyan lehet megtanulni egy döntést?

megtanulni = azaz eltárolni a leírását (algoritmusát), hogy később ne kelljen újra és újra kiszámítani.

(1) a leírás legyen logikai, mert így az ágens TB-ba jól elfér.

(2) tanulunk példákból („függvény tanulás”)

→ a példákat is kell valahogy „logikailag” kifejezni.

(2a) döntési helyzet jellemzése = a világhoz kötött attribútumok.

(2b) egy konkrét döntési helyzet = az attribútumok konkrét értékei.

(2c) logikai eszköz = ítélet konstansak, predikátumok, ...

egy konkrét döntési helyzetben egyes attribútumértékek előfordulnak (igazak), mások nem (hamisak)

(2d) döntési helyzetek megítélése: jó-e egy döntés

- ez is lehet egy ítélet konstans, vagy predikátum.

- predikátumok alapján egy predikátum kiszámítása!

Példa: döntés, hogy egy adott étteremben várjunk-e asztalra (jegyzet).

Cél: megtanulni a *VárniFog* **cél predikátum** definícióját

A probléma milyen tulajdonsága vagy **attribútuma** ismert?

1. *Alternatíva*: van-e a környéken más megfelelő étterem. (I/N)
2. *Bár*: van-e az étteremben kényelmes bár, ahol várhatunk. (I/N)
3. *Pén/Szom*: igaz pénteken- és szombatonként. (I/N)
4. *Éhes*: éhesek vagyunk-e. (I/N)
5. *Kuncsaft*: hányan vannak már benn (*Senki*, *Néhány* és *Tele*).
6. *Ár*: mennyire drága (*Olcsó*, *Közepes*, *Drága*)
7. *Eső*: esik-e kint (I/N)
8. *Foglalás*: foglalni kell-e előzetesen az asztalt (I/N)
9. *Típus*: az étterem típusa (*Francia*, *Olasz*, *Thai* v. *Burger*)
10. *BecsültVár*: a pincér által bemondott várakozási idő
(0-10 perc, 10-30, 30-60, >60 perc)

Az eddigi kiruccanásaink tapasztalata

Pl.	Attribútumok										Cél
	Altern	Bár	Pént	Éhes	Kuncs	Ár	Eső	Fogl	Típus	Becs	VárniFog
X_1	Igen	Nem	Nem	Igen	Néhány	Drága	Nem	Igen	Francia	0–10	Igen
X_2	Igen	Nem	Nem	Igen	Tele	Olcsó	Nem	Nem	Thai	30–60	Nem
X_3	Nem	Igen	Nem	Nem	Néhány	Olcsó	Nem	Nem	Burger	0–10	Igen
X_4	Igen	Nem	Igen	Igen	Tele	Olcsó	Nem	Nem	Thai	10–30	Igen
X_5	Igen	Nem	Igen	Nem	Tele	Drága	Nem	Igen	Francia	>60	Nem
X_6	Nem	Igen	Nem	Igen	Néhány	Közep	Igen	Igen	Olasz	0–10	Igen
X_7	Nem	Igen	Nem	Nem	Senki	Olcsó	Igen	Nem	Burger	0–10	Nem
X_8	Nem	Nem	Nem	Igen	Néhány	Közep	Igen	Igen	Thai	0–10	Igen
X_9	Nem	Igen	Igen	Nem	Tele	Olcsó	Igen	Nem	Burger	>60	Nem
X_{10}	Igen	Igen	Igen	Igen	Tele	Drága	Nem	Igen	Olasz	10–30	Nem
X_{11}	Nem	Nem	Nem	Nem	Senki	Olcsó	Nem	Nem	Thai	0–10	Nem
X_{12}	Igen	Igen	Igen	Igen	Tele	Olcsó	Nem	Nem	Burger	30–60	Igen

Mi benne a viselkedésünk mintája? Van-e egyáltalán egy konzisztens viselkedési mintánk? (9216 lehetséges példa)

Döntési fák tanulása

döntési fa = **egy logikai függvény**

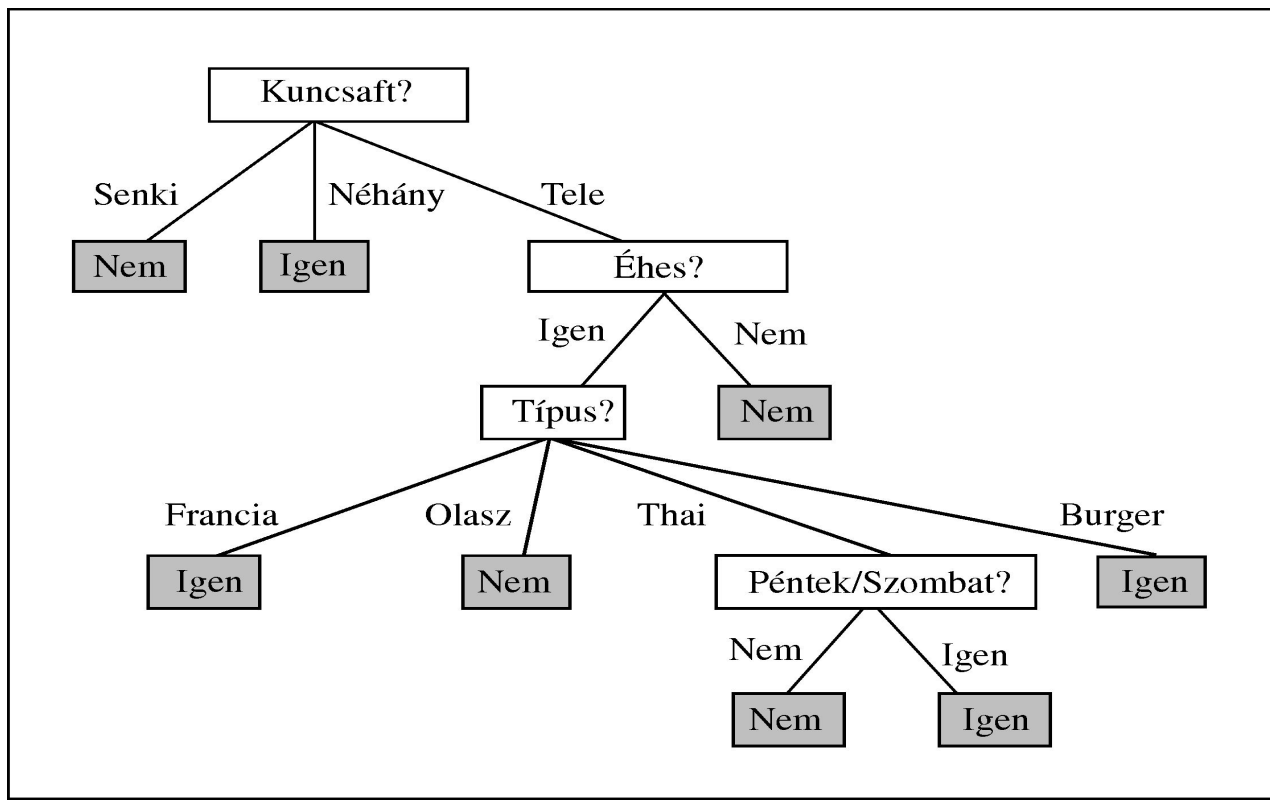
be: egy tulajdonsághalmazzal leírt objektum vagy szituáció

ki: egy igen/nem „döntés”

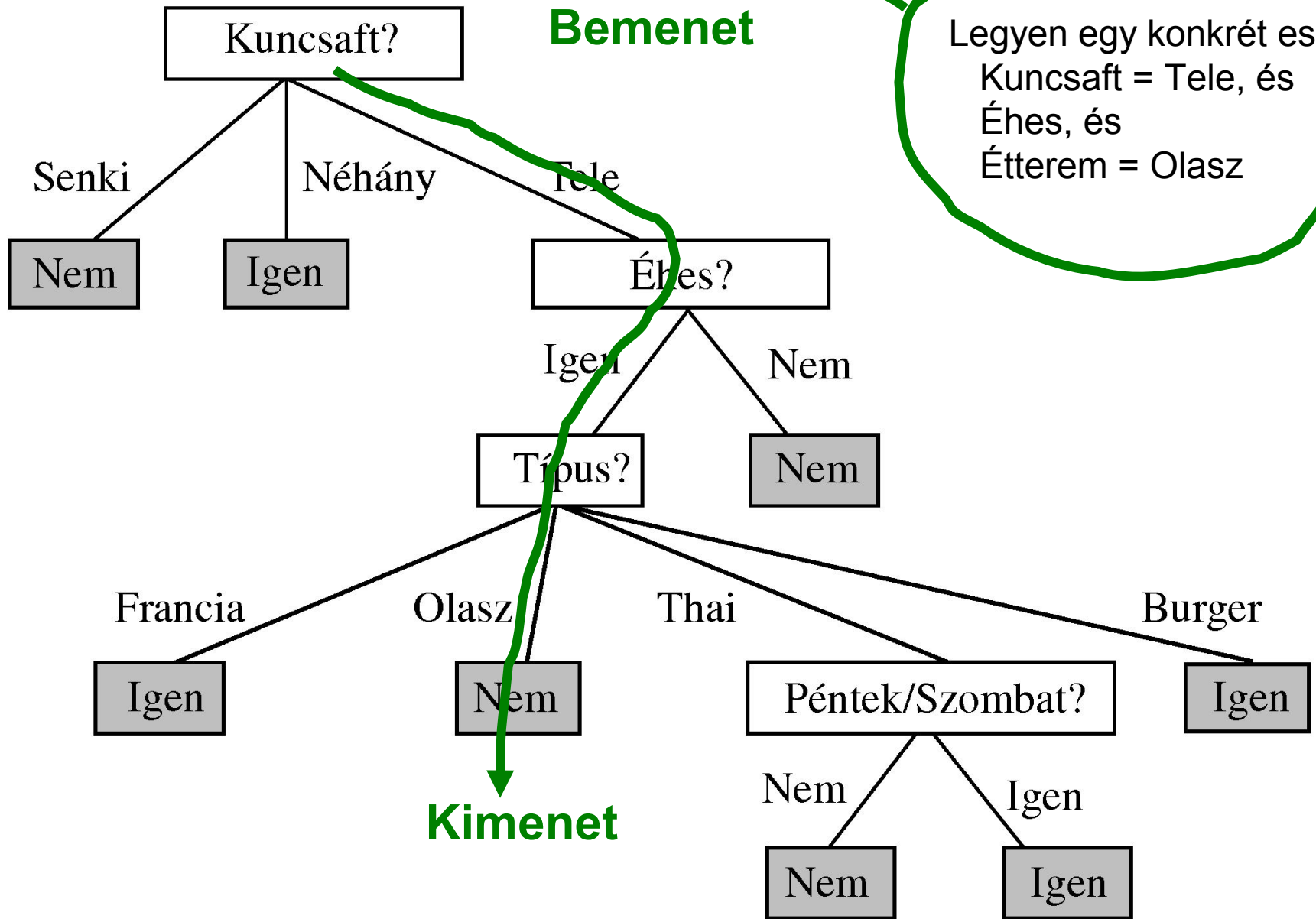
belső csomópont = valamelyik tulajdonság tesztje

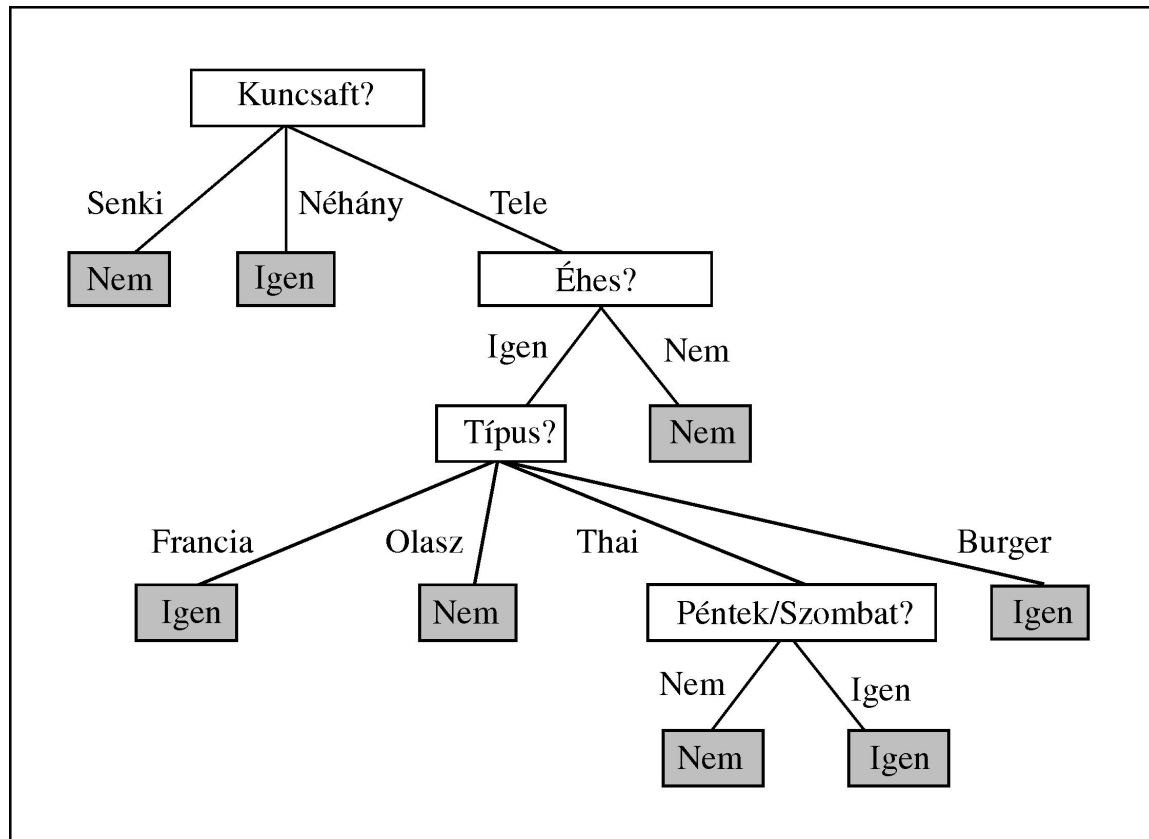
él = a teszt lehetséges értéke

levél = logikai érték, amelyet akkor kell kiadni, ha ezt a levelet elértük.



Bemenet





$VárniFog \leftarrow (Kuncsaft = Néhány) \vee$
 $((Kuncsaft = Tele) \wedge Éhes \wedge (Típus = Francia)) \vee$
 $((Kuncsaft = Tele) \wedge Éhes \wedge (Típus = Thai) \wedge P/Sz) \vee$
 $((Kuncsaft = Tele) \wedge Éhes \wedge (Típus = Burger))$

Döntési fák kifejezőképessége

teljes - minden (ítélet)logikai függvény felírható döntési faként.

(az igazságtábla minden sora = a fa egy bejárása, de így az igazságtábla mérete
= a fa mérete = exponenciális!)

Döntési fák kialakítása példák alapján

példa: (attribútumok értékei, cél predikátum értéke)

példa besorolása: a cél predikátum értéke

pozitív/negatív példa: a cél predikátum értéke igaz / hamis

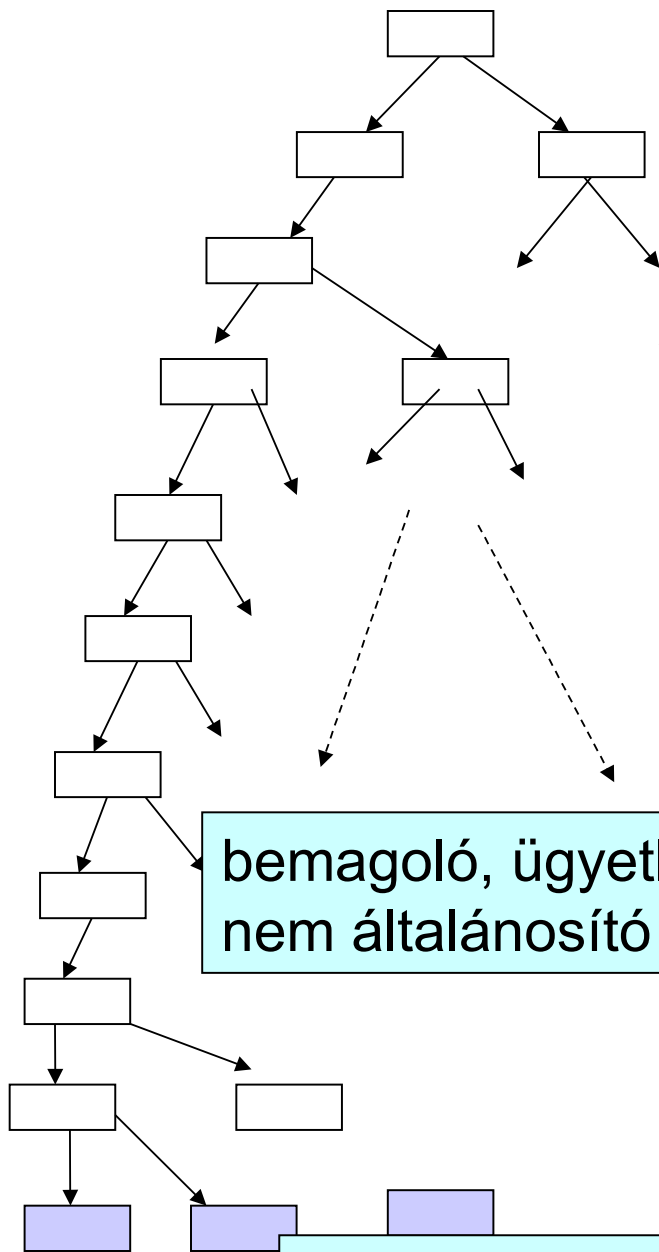
tanító halmaz: a teljes példa halmaz

triviális fa, könnyű - mindegyik példához egy önálló bejárési út

a levél a példa besorolását adja.

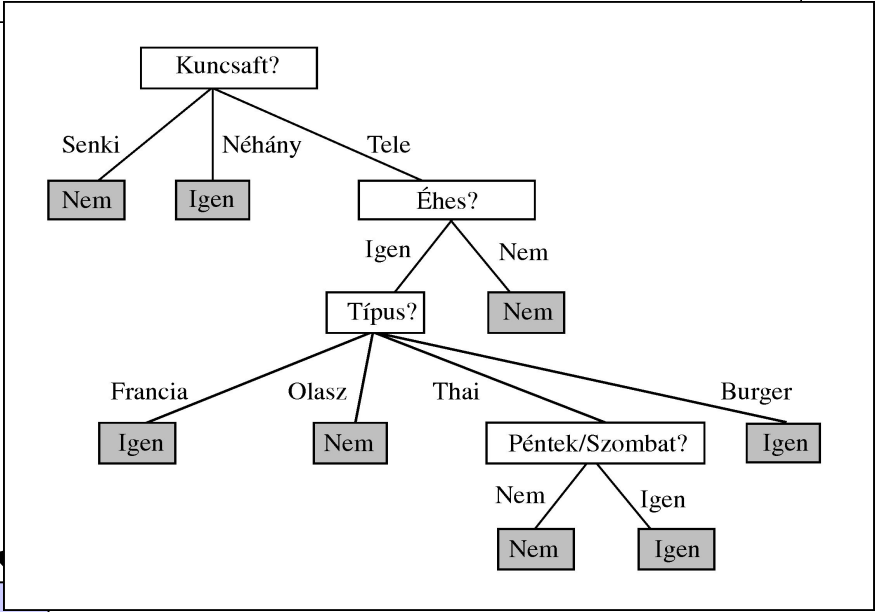
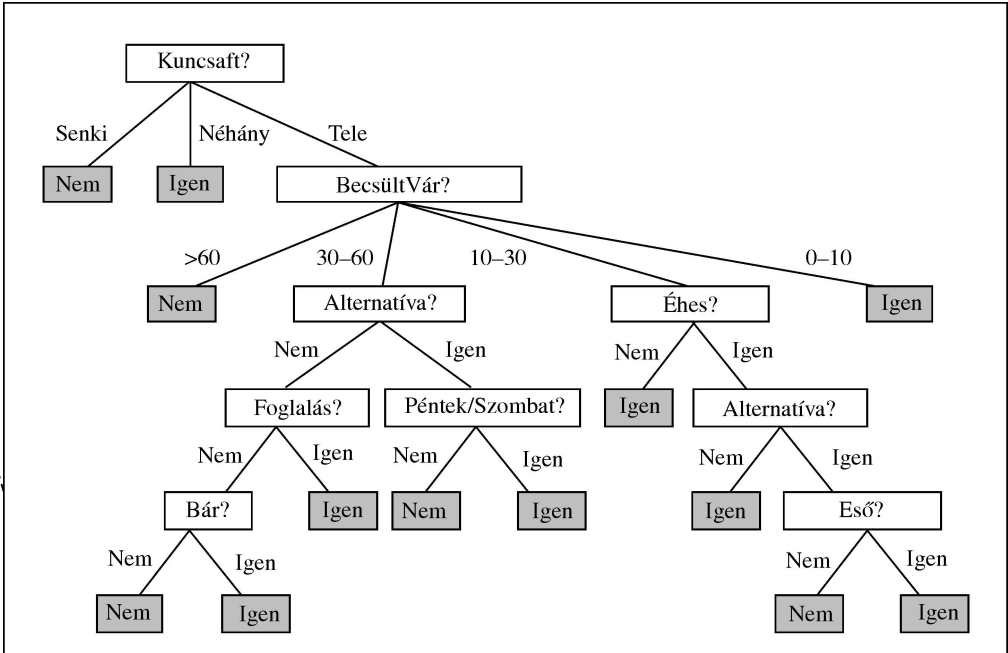
fa egyszerűen memorizálja a példát, nem alakít ki jellegzetes mintákat
a példákból nem általánosít (nem tud ismeretlen példákra extrapolálni)

„igazi” fa - a jellegzetes minták kinyerése, a módszer képes nagyszámú
esetet **tömör** formában leírni, és így az egyelőre ismeretlen példákra is
általánosítani



bemagoló, ügyetlen,
nem általánosító

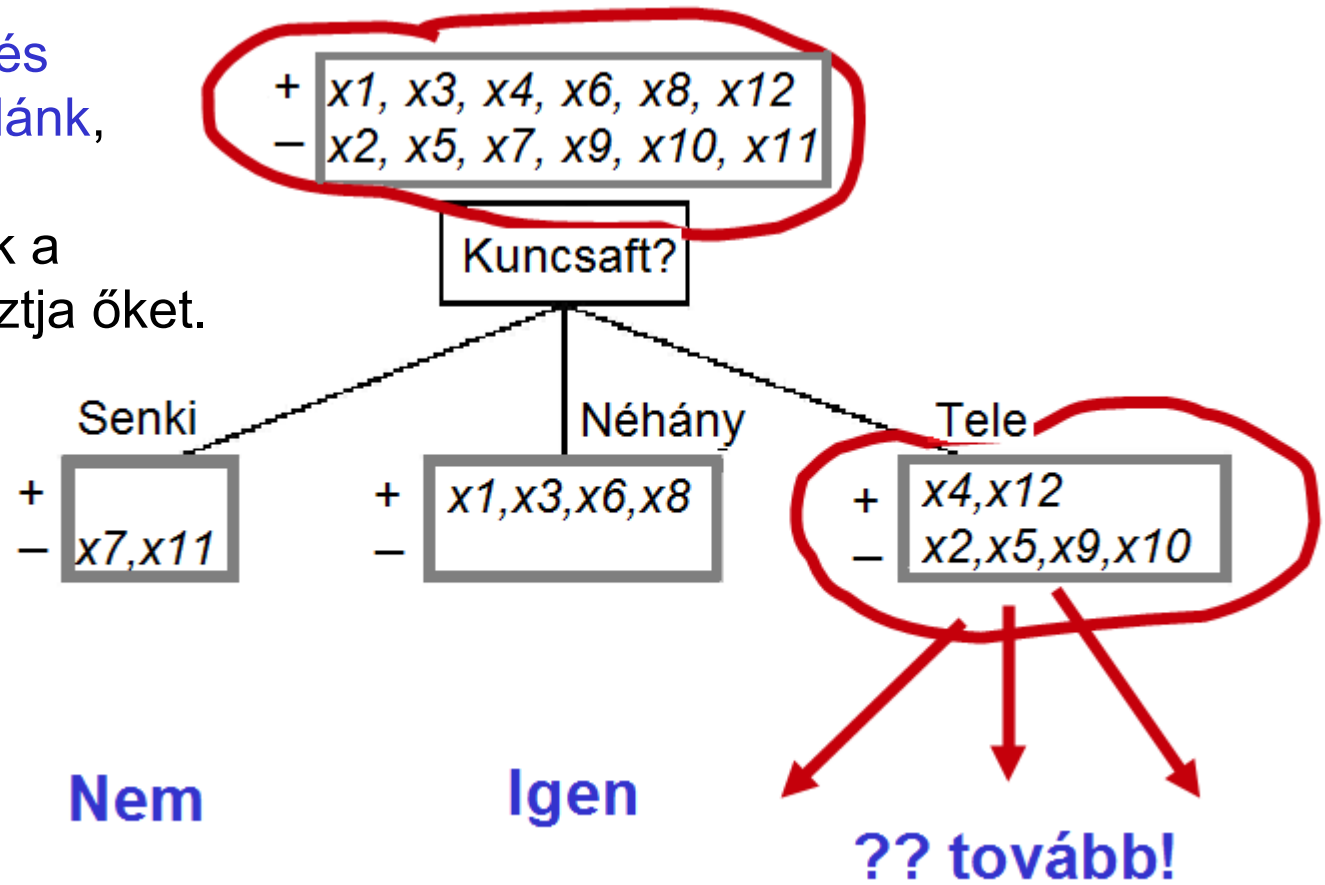
ugyanabból a példahalmazból



ügyes, általánosító

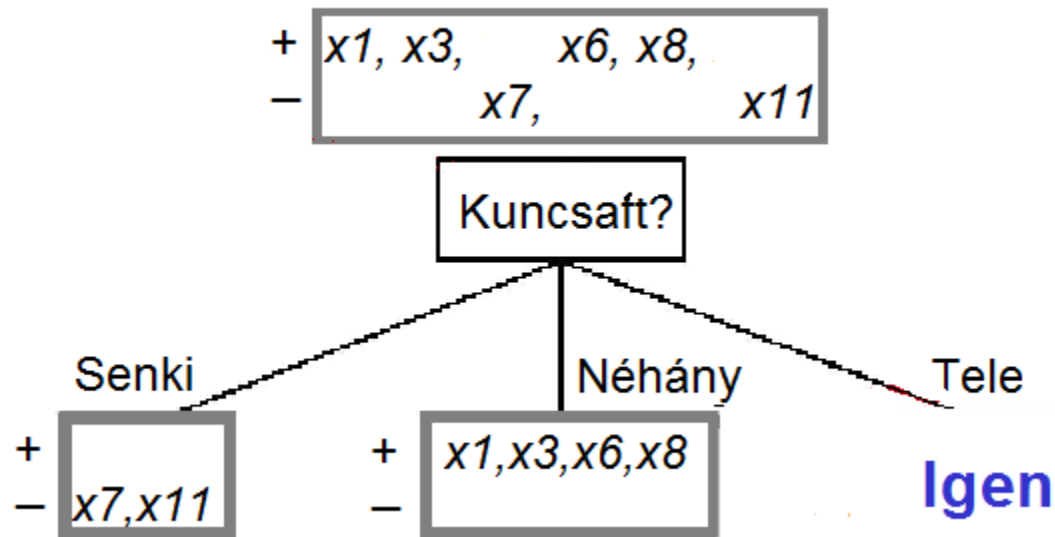
A döntési fa – általános jelenségek

Van **néhány pozitív és néhány negatív példánk**,
válasszuk azt az
attribútumot, amelyik a
legjobban szétválasztja őket.



Ha az **összes** megmaradt eset **pozitív**, (vagy az **összes negatív**),
akkor készen vagyunk: a válasz **Igen** vagy **Nem**.

A döntési fa – általános jelenségek

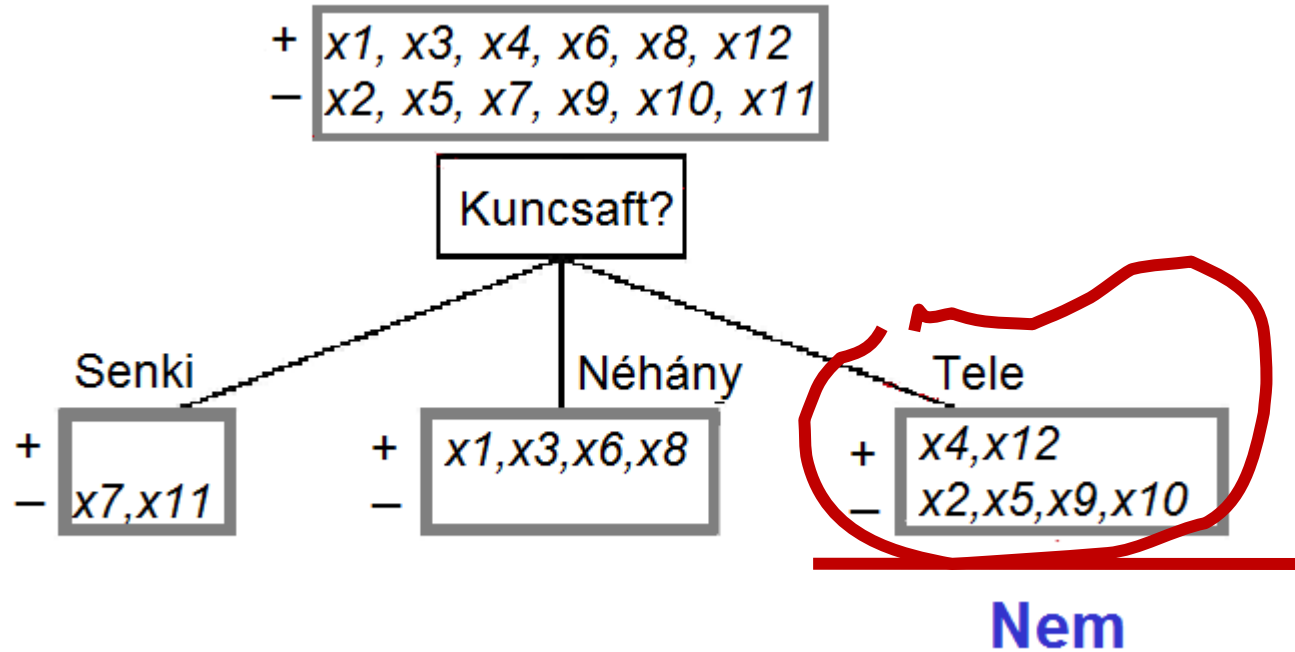


Ha **nem maradt egyetlen példa sem**, ez azt jelenti, hogy ilyen példát nem figyeltünk meg eddig.

Ilyenkor valamilyen alapértéket adunk vissza, amelyet a csomópont szülőjének többségi besorolási válaszából származtatunk.

(**induktív tanulás nem egy deduktív módszer,**
milyen veszély leselkedik itt ránk?)

A döntési fa – általános jelenségek



Nem maradt teszteletlen attribútum, de maradtak pozitív és negatív példák. Baj van.

Ezeknek a példáknak pontosan megegyezik a leírása, de különböző a besorolásuk.

Néhány adat tehát nem korrekt: az ún. **tanulási zaj** torzítja az adatokat. Megoldás? Pl. a többségi szavazás használata.

Döntési fák kialakítása példák alapján

A legkisebb döntési fa megtalálása - általánosságban nem megoldható

Heurisztika: **mohóság** - egy meglehetősen egyszerű (jó) fa is jó lesz!

Az alapötlet: először a „legfontosabb” attribútumot teszteljük.

„**legfontosabb**” = a **legnagyobb eltérést** okozza példák besorolásában

Elvárás: **kisszámú teszt** alapján korrekt besoroláshoz jutunk:

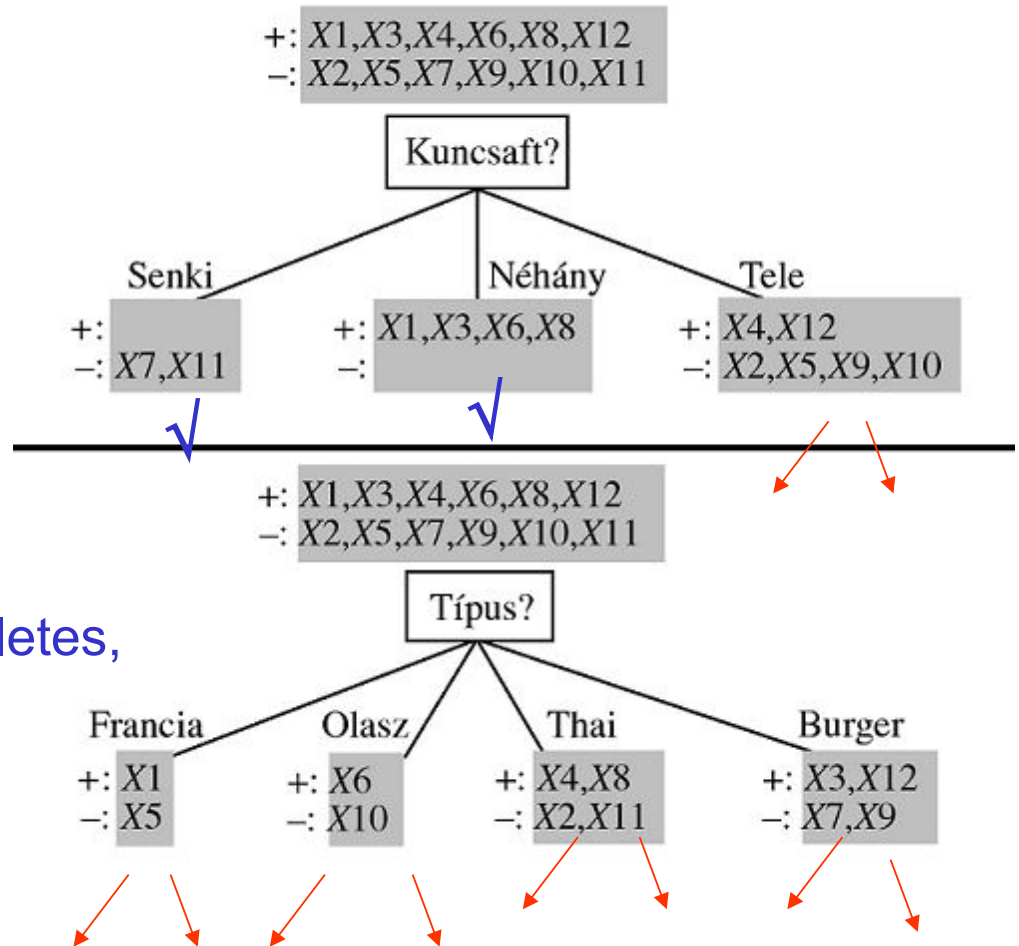
a **bejárési utak rövidek** lesznek, és így az **egész fa kicsi** (tömör) lesz.

Az információelmélet felhasználása

Olyan attribútumot válasszunk, amely a lehető legmesszebb megy a pontos besorolás biztosításában.

A **tökéletes attribútum** a példákat két csoportra bontja,
az egyikbe **csak pozitív**,
a másikba **csak negatív** példák tartoznak.

Ezzel be is lehetne fejezni a fa tanítását



A *Kunceaft* attribútum nem tökéletes, de elég jó.

Egy **nagymértékben haszontalan attribútum**, mint pl. a *Típus* olyan példa halmazokat hoz létre, amelyek nagyjából ugyanolyan arányban tartalmaznak pozitív és negatív példákat, mint az eredeti halmazok.

„Elég jó?" „nagyértékben haszontalan?"

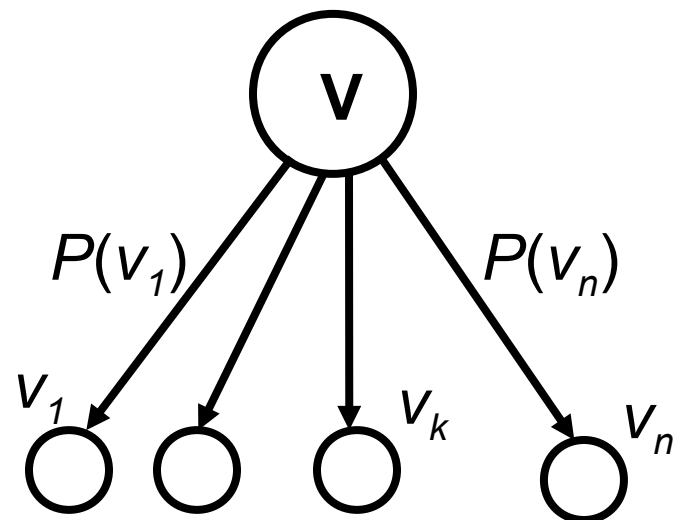
A mérték **maximuma**: a fenti értelemben **tökéletes** attribútumra
minimuma: olyan attribútumra, aminek egyáltalán
nincs értéke számunkra.

Egy megfelelő mérték: **az attribútum által adott információ
várható értéke, információ átlagos tartalma, entrópia, ...**

Ha a lehetséges v_k válaszok valószínűsége $P(v_k)$,
akkor az adott konkrét válasz információ tartalma:

$$I(P(v_1), \dots, P(v_n)) = \sum_{i=1}^n -P(v_i) \log_2 P(v_i)$$

pl. szabályos pénzérme dobás?



A döntési fa információ tartalma = a tanító halmazban található pozitív és negatív példák aránya

Tanító halmaz **p** pozitív és **n** negatív példát tartalmaz:

két válasz: v_1 , v_2 , valószínűség: $P(v_1) = p / (p+n)$, $P(v_2) = n / (p+n)$,

Ekkor a **fa információ tartalmának** becslése:

$$I\left(\frac{p}{p+n}, \frac{n}{p+n}\right) = -\frac{p}{p+n} \log_2 \frac{p}{p+n} - \frac{n}{p+n} \log_2 \frac{n}{p+n}$$

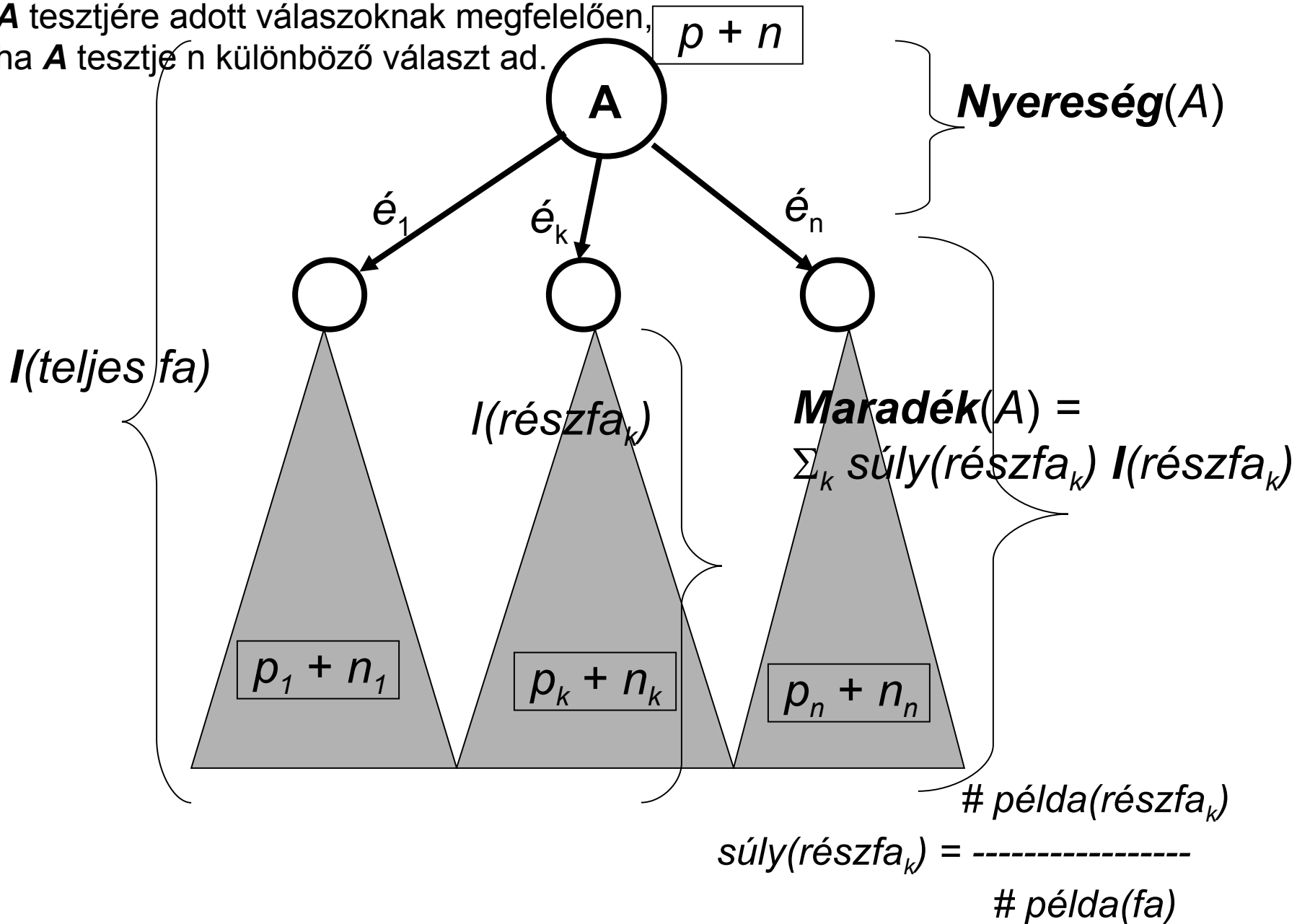
(Az étterem tanító halmaz: $p = n$, **1 bit információt** képvisel)

Hogyan válasszunk attribútumot?

Mennyi információt ad egy attribútum tesztelése?

Mennyi információra van **még** szükségünk az attribútum tesztelése **után**?

Bármelyik **A** attribútum az **E** tanító halmazt $E_1, E_2, \dots, E_n$ részalmazokra bontja az **A** tesztjére adott válaszoknak megfelelően, ha **A** tesztje n különböző választ ad.



Hogyan válasszunk attribútumot?

$$Maradék(A) = \sum_{i=1}^v \frac{p_i + n_i}{p + n} I\left(\frac{p_i}{p_i + n_i}, \frac{n_i}{p_i + n_i}\right)$$

bit információra van szükségünk a példa besorolásához.

Az attribútum tesztjének **információ nyeresége** az **eredeti** információ igény és a **teszt utáni** új információ igény **különbsége**:

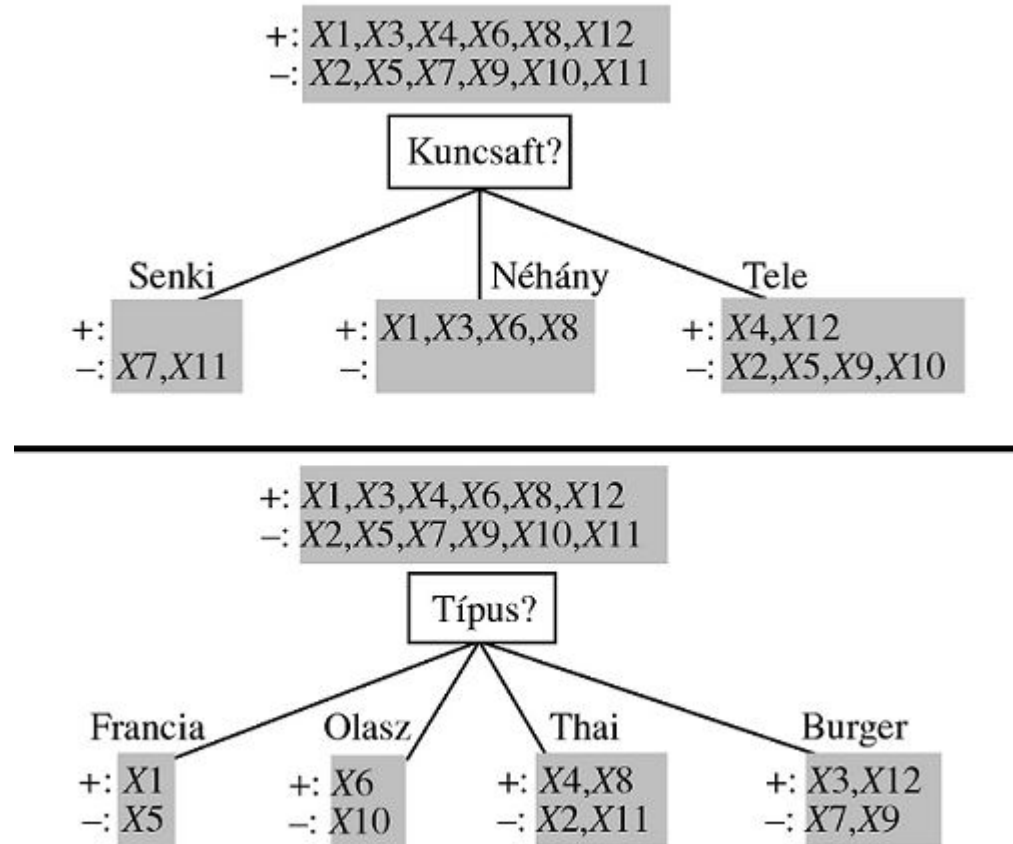
$$Nyeresség(A) = I\left(\frac{p}{p + n}, \frac{n}{p + n}\right) - Maradék(A)$$

Nézzük meg a *Kuncsaft* és a *Típus* attribútumokat

$$I(0,1) = -0 \cdot \ln 0 - 1 \cdot \ln 1 = -0 \cdot \ln 0 = ?$$

$$0 \cdot \ln 0 = \lim_{x \rightarrow 0} \frac{\ln x}{1/x} = \lim_{x \rightarrow 0} \frac{(\ln x)'}{(1/x)'} = \lim_{x \rightarrow 0} \frac{1/x}{-1/x^2} = \lim_{x \rightarrow 0} -x = 0$$

$$I(0,1) = ?$$



$$Nyeresség(Kuncsaft) = 1 - \left[\frac{2}{12} I(0,1) + \frac{4}{12} I(1,0) + \frac{6}{12} I\left(\frac{2}{6}, \frac{4}{6}\right) \right] \approx 0,541 \text{ bit}$$

$$Nyeresség(Típus) = 1 - \left[\frac{2}{12} I\left(\frac{1}{2}, \frac{1}{2}\right) + \frac{2}{12} I\left(\frac{1}{2}, \frac{1}{2}\right) + \frac{4}{12} I\left(\frac{2}{4}, \frac{2}{4}\right) + \frac{4}{12} I\left(\frac{2}{4}, \frac{2}{4}\right) \right] = 0 \text{ bit}$$

Több attribútumra is kellene a számítás, de a Kuncsaft kimagasló

(**Kuncsaft = Tele**) ágon még fennmaradó példák

Példa	Attribútumok									Cél
	<i>Altern</i>	<i>Bár</i>	<i>Pént</i>	<i>Éhes</i>	<i>Ár</i>	<i>Eső</i>	<i>Fogl</i>	<i>Típus</i>	<i>Becs</i>	<i>VárniFog</i>
X_2	<i>Igen</i>	<i>Nem</i>	<i>Nem</i>	<i>Igen</i>	<i>Olcsó</i>	<i>Nem</i>	<i>Nem</i>	<i>Thai</i>	<i>30–60</i>	<i>Nem</i>
X_5	<i>Igen</i>	<i>Nem</i>	<i>Igen</i>	<i>Nem</i>	<i>Drága</i>	<i>Nem</i>	<i>Igen</i>	<i>Francia</i>	<i>>60</i>	<i>Nem</i>
X_9	<i>Nem</i>	<i>Igen</i>	<i>Igen</i>	<i>Nem</i>	<i>Olcsó</i>	<i>Igen</i>	<i>Nem</i>	<i>Burger</i>	<i>>60</i>	<i>Nem</i>
X_{10}	<i>Igen</i>	<i>Igen</i>	<i>Igen</i>	<i>Igen</i>	<i>Drága</i>	<i>Nem</i>	<i>Igen</i>	<i>Olasz</i>	<i>10–30</i>	<i>Nem</i>

X_4	<i>Igen</i>	<i>Nem</i>	<i>Igen</i>	<i>Igen</i>	<i>Olcsó</i>	<i>Nem</i>	<i>Nem</i>	<i>Thai</i>	<i>10–30</i>	<i>Igen</i>
X_{12}	<i>Igen</i>	<i>Igen</i>	<i>Igen</i>	<i>Igen</i>	<i>Olcsó</i>	<i>Nem</i>	<i>Nem</i>	<i>Burger</i>	<i>30–60</i>	<i>Igen</i>

Részfában: $p_1 = 2$, $n_1 = 4$, $p_1/(p_1+n_1) = 1/3$, $n_1/(p_1+n_1) = 2/3$

$$I = I(\text{részfa}) = I(2/6, 4/6) = .9183$$

$$Ny(AI) = I - \left[\underbrace{5/6 I(2/5, 3/5)}_{AI = Igen} + \underbrace{1/6 I(0, 1)}_{AI = Nem} \right]$$

AI = Igen

AI = Nem

$$\begin{aligned} p_2 &= 2, \\ n_2 &= 3 \end{aligned}$$

$$\begin{aligned} p_3 &= 0 \\ n_3 &= 1 \end{aligned}$$



6 példa

$$Ny(AI) = I - \left[5/6 I(2/5, 3/5) + 1/6 I(0, 1) \right]$$

$$= .92 - .81 \approx .11$$

$$Ny(Al) = I - [5/6 I(2/5, 3/5) + 1/6 I(0,1)] = .92 - .81 \approx .11$$

$$Ny(Bár) = I - [1/2 I(1/3, 2/3) + 1/2 I(1/3, 2/3)] = .92 - .92 = 0$$

$$Ny(Péntek) = I - [5/6 I(2/5, 3/5) + 1/6 I(0,1)] = .92 - .81 \approx .11$$

$$Ny(Éhes) = I - [4/6 I(1/2, 1/2) + 2/6 I(0,1)] = .92 - .66 \approx .25$$

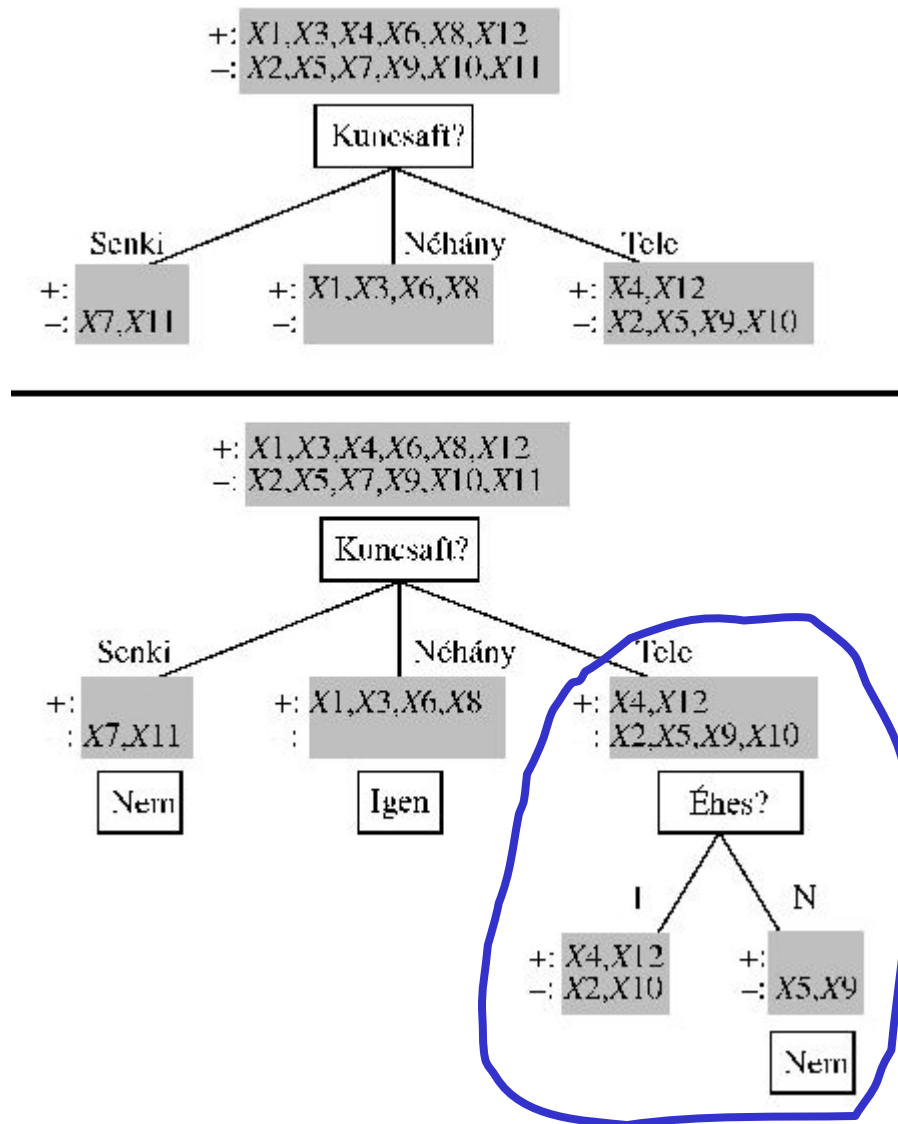
$$Ny(Ár) = I - [4/6 I(1/2, 1/2) + 2/6 I(0,1)] = .92 - .66 \approx .25$$

$$Ny(Eső) = I - [5/6 I(2/5, 3/5) + 1/6 I(0,1)] = .92 - .81 \approx .11$$

$$Ny(Foglalt) = I - [2/6 I(0,1) + 4/6 I(1/2,1/2)] = .92 - .66 \approx .25$$

$$Ny(Típus) = I - [2/6 I(1/2, 1/2) + 1/6 I(0,1) + 2/6 I(1/2, 1/2) + 1/6 I(0,1)] = .92 - .66 \approx .25$$

$$Ny(Becs) = I - [2/6 I(1/2, 1/2) + 2/6 I(0,1) + 2/6 I(1/2, 1/2)] = .92 - .66 \approx .25$$



(Éhes = Igen) ágon még fennmaradó példák

Példa	Attribútumok								Cél
	<i>Alt</i>	<i>Bár</i>	<i>Pént</i>	<i>Ár</i>	<i>Eső</i>	<i>Fogl</i>	<i>Típus</i>	<i>Becs</i>	<i>VárniFog</i>
X_2	<i>Igen</i>	<i>Nem</i>	<i>Nem</i>	<i>Olcsó</i>	<i>Nem</i>	<i>Nem</i>	<i>Thai</i>	<i>30–60</i>	<i>Nem</i>
X_{10}	<i>Igen</i>	<i>Igen</i>	<i>Igen</i>	<i>Drága</i>	<i>Nem</i>	<i>Igen</i>	<i>Olasz</i>	<i>10–30</i>	<i>Nem</i>

X_4	<i>Igen</i>	<i>Nem</i>	<i>Igen</i>	<i>Olcsó</i>	<i>Nem</i>	<i>Nem</i>	<i>Thai</i>	<i>10–30</i>	<i>Igen</i>
X_{12}	<i>Igen</i>	<i>Igen</i>	<i>Igen</i>	<i>Olcsó</i>	<i>Nem</i>	<i>Nem</i>	<i>Burger</i>	<i>30–60</i>	<i>Igen</i>

Részfában: $p_1 = 2$, $n_1 = 2$, $p_1/(p_1+n_1) = 1/2$, $n_1/(p_1+n_1) = 1/2$

$$Ny(Alt) = I - [1/1 I(1/2, 1/2) + 0] = 1 - 1 = 0$$

$$Ny(Bár) = I - [1/2 I(1/2, 1/2) + 1/2 I(1/2, 1/2)] = 1 - 1 = 0$$

$$Ny(Péntek) = I - [1/4 I(0,1) + 3/4 I(1/3,2/3)] = 1 - .92 \approx .08$$

$$Ny(Ár) = I - [1/4 I(0,1) + 3/4 I(1/3,2/3)] = 1 - .92 \approx .08$$

$$Ny(Eső) = I - [1/1 I(1/2, 1/2) + 0] = 1 - 1 = 0$$

$$Ny(Foglalt) = I - [1/4 I(0,1) + 3/4 I(1/3,2/3)] = 1 - .92 \approx .08$$

$$Ny(Típus) = I - [1/4 I(0,1) + 1/4 I(0,1) + 1/2 I(1/2, 1/2)] = .5$$

$$Ny(Becs) = I - [1/2 I(1/2, 1/2) + 1/2 I(1/2, 1/2)] = 1 - 1 = 0$$

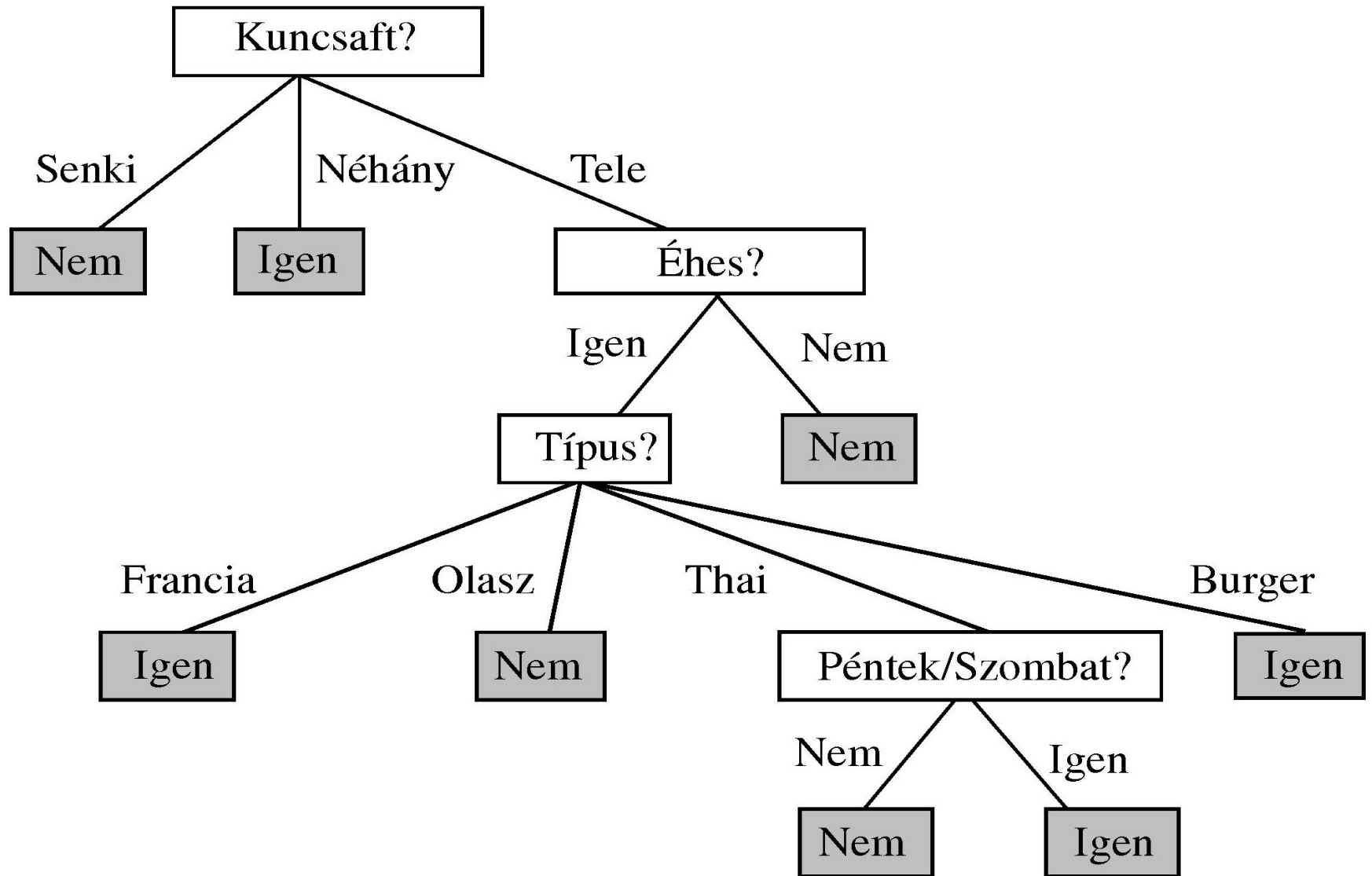
(Típus = Thai) ágon még fennmaradó példák

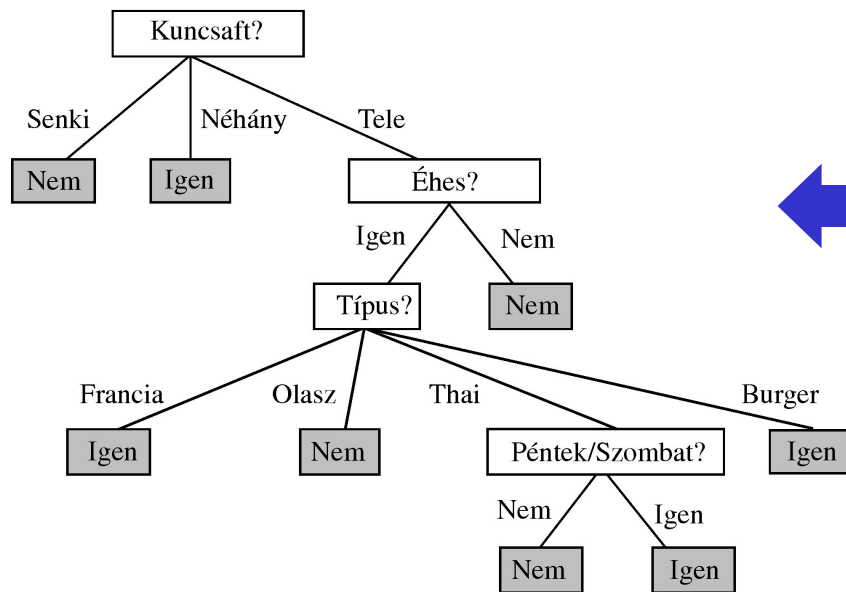
Példa	Attribútumok							Cél
	<i>Alt</i>	<i>Bár</i>	<i>Pént</i>	<i>Ár</i>	<i>Eső</i>	<i>Fogl</i>	<i>Becs</i>	<i>VárniFog</i>
X_2	<i>Igen</i>	<i>Nem</i>	<i>Nem</i>	<i>Olcsó</i>	<i>Nem</i>	<i>Nem</i>	<i>30–60</i>	<i>Nem</i>
X_4	<i>Igen</i>	<i>Nem</i>	<i>Igen</i>	<i>Olcsó</i>	<i>Nem</i>	<i>Nem</i>	<i>10–30</i>	<i>Igen</i>

A többi nem jellemző

Részfában: $p_1 = 1$, $n_1 = 1$, $p_1/(p_1+n_1) = 1/2$, $n_1/(p_1+n_1) = 1/2$

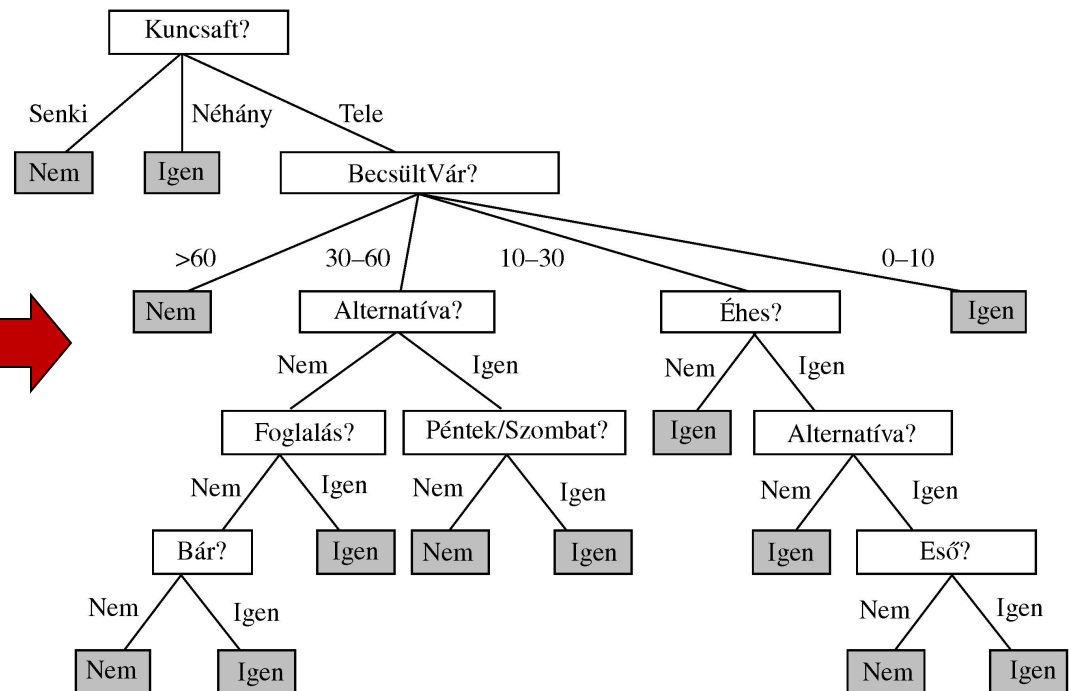
És ez u.u. marad pl. a *Bár = Nem*, vagy *Eső = Nem*, vagy *Ár = Olcsó*, stb. mentén (azaz így tovább nem érdemes építeni)





„Optimális”:
 4 csomópont teszt
 max. 3 mélység
 8 levél csomópont
 = általánosítás, absztrakció

„Bambább”:
 9 csomópont teszt
 max. 4 mélység
 13 levél csomópont
 $\cong$ majdnem a példák
 felsorolása látjuk és
 lehetne ez rosszabb is



Zaj és túlzott illeszkedés

**Rendkívül általános jelenség:
az összes tanulási eljárást sújtja**

Ha alapvető információ hiányzik, a tanuló algoritmus akkor is találhat olyan döntési fát, amely az összes példával konzisztens, bár a példák most nem képviselnek értelmes függvényt.

Az algoritmus felhasználhatja az **irreleváns** attribútumokat is - ha vannak - arra, hogy hamis megkülönböztetéseket tegyen a példák közt.

Ha a hipotézisek halmaza nagy, akkor mindig vigyázni kell, nehogy értelmetlen „szabályosságot” találjunk az adatokban = **túlzott illeszkedés**.

Döntési fa nyesése: ismerjük fel a nem releváns attribútumot

Irreleváns attribútum: példahalmaz kettévágásánál a kapott részhalmazok $\cong$ azonos arányban tartalmazznak pozitív és negatív példát, mint az eredeti halmaz. **Az információ nyereség ilyenkor közel 0.** Az információ nyereség hiánya az irrelevancia jó jelzése.

Milyen nagy (nem 0) legyen az információ nyereség, hogy egy attribútumot mégis felhasználjunk a példahalmaz megosztására?

Statisztikai teszt: nincs jellegzetes minta = ún. **nulla hipotézis**.

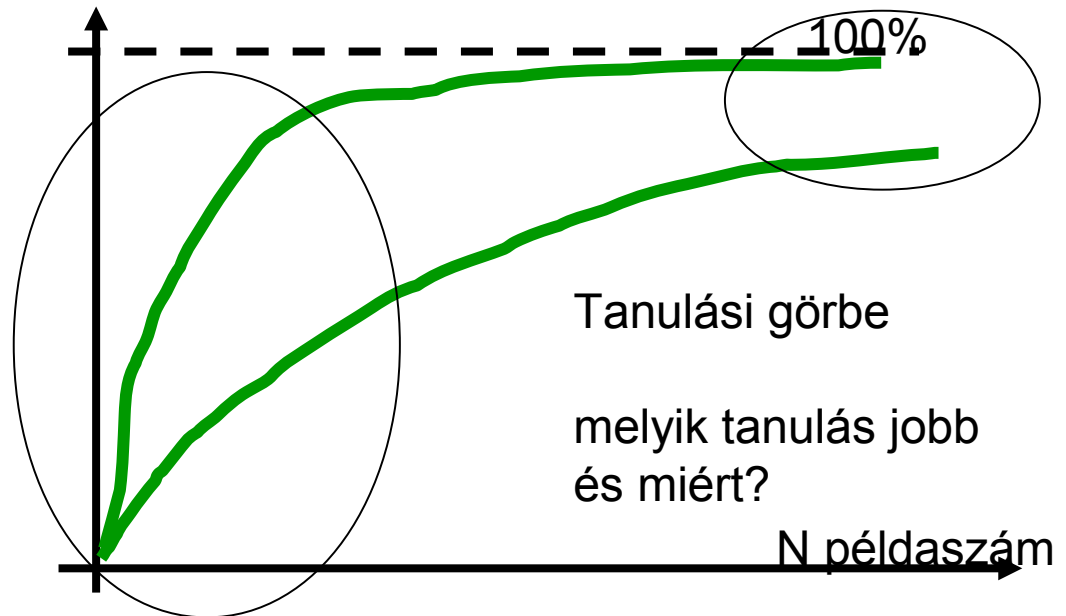
Ha az eltérés nagy (nagyobb, mint 0) $\rightarrow$ nem valószínű, hogy az észlelt minta statisztikai ingadozásból következik $\rightarrow$ lényeges minta van az adatokban.

A tanulási algoritmus teljesítménye

A tanulási algoritmus akkor jó, ha jó hipotéziseket szolgáltat azon esetekre is, amelyeket nem látott előtte.

1. A példahalmazt bontsuk szét egy **tanító** és egy **teszt halmazra**.
2. A tanuló algoritmust a tanító halmazra alkalmazva állítsuk elő a **h** hipotézist.
3. Vizsgáljuk meg, hogy **h** a teszt halmaz hány százalékát sorolja be jól.
4. Ismételjük a 2-4 lépéseket különböző tanító halmaz méretekre.

Tanulási görbe: a tanító halmaz méretének függ
(jó, ha a jóslás minősége javul)



Kétértékű függvény = osztályozás = helyes bináris döntés tanulása

paciens **beteg** -
és mi ezt **felismerjük**

kezeljük, helyesen cselekszünk
Igaz Pozitív, True Positive – TP,
 $f(x) = I, h(x) = I$

paciens **egészséges** -
és mi ezt **felismerjük**

nem kezeljük, most is jól cselekszünk
Igaz Negatív, True Negative – TN

paciens **beteg** – és mi
ezt **nem ismerjük fel**

nem kezeljük, nem járunk el jól
Hamis Negatív, False Negative – FN,
 $f(x) = I, h(x) = H$ („elnezett támadás”, 2. tip. hiba)

paciens **egészséges**
- és mi ezt **nem**
ismerjük fel

kezeljük feleslegesen, most sem jól járunk el
Hamis Pozitív, False Positive – FP
 $f(x) = H, h(x) = I$ („hamis riadó”, 1. tip. hiba)

De lehet radarkép, szerver betörés, földcsuszamlás veszélye, ...

ROC: Vevő működési karakterisztika

Receiver Operating Characteristic

true positive rate (TPR)

(hit rate, recall, sensitivity)

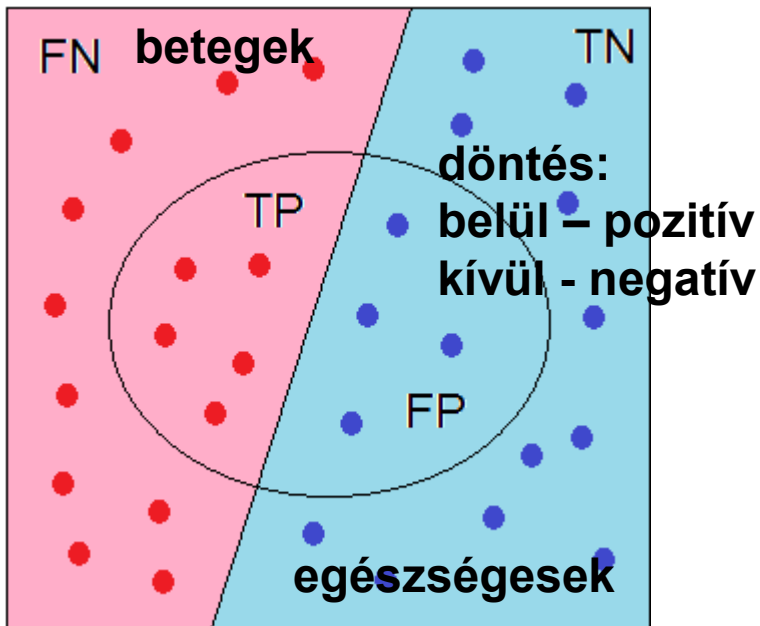
$$\text{TPR} = \text{TP}/P = \text{TP} / (\text{TP} + \text{FN})$$

false positive rate (FPR)

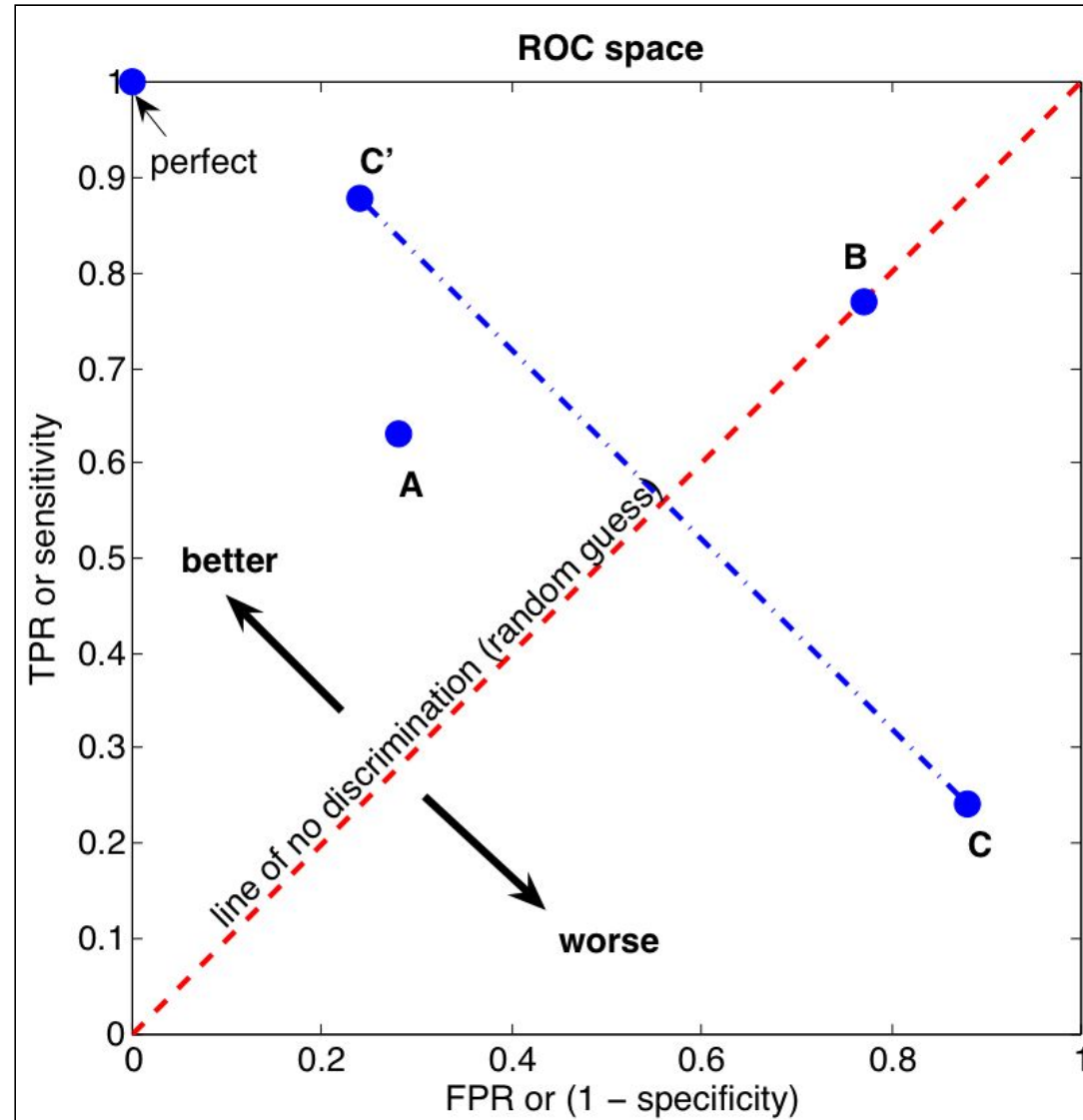
(false alarm rate, fall-out)

$$\text{FPR} = \text{FP}/N = \text{FP} / (\text{FP} + \text{TN})$$

és sok más ...



A.U.C Area Under Curve



Teszt példa

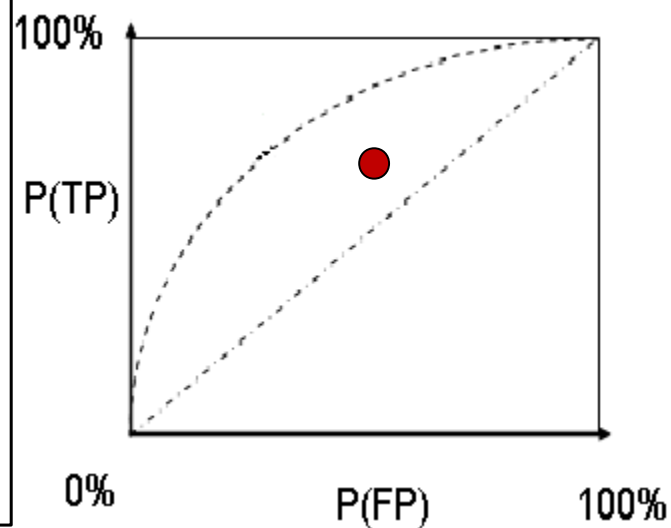
Attribútumok

Cél

	<i>Altern</i>	<i>Bár</i>	<i>Pént</i>	<i>Éhes</i>	<i>Kuncs</i>	<i>Ár</i>	<i>Eső</i>	<i>Fogl</i>	<i>Típus</i>	<i>Becs</i>	<i>VárniFog</i>
X_{13}	<i>Nem</i>	<i>Nem</i>	<i>Nem</i>	<i>Igen</i>	<i>Néhány</i>	<i>Drága</i>	<i>Nem</i>	<i>Igen</i>	<i>Francia</i>	<i>0–10</i>	<i>Igen</i>
X_{14}	<i>Igen</i>	<i>Nem</i>	<i>Igen</i>	<i>Igen</i>	<i>Tele</i>	<i>Olcsó</i>	<i>Nem</i>	<i>Nem</i>	<i>Thai</i>	<i>30–60</i>	<i>Nem</i>
X_{15}	<i>Nem</i>	<i>Igen</i>	<i>Nem</i>	<i>Nem</i>	<i>Tele</i>	<i>Olcsó</i>	<i>Nem</i>	<i>Nem</i>	<i>Burger</i>	<i>0–10</i>	<i>Igen</i>
X_{16}	<i>Igen</i>	<i>Nem</i>	<i>Igen</i>	<i>Igen</i>	<i>Tele</i>	<i>Olcsó</i>	<i>Nem</i>	<i>Igen</i>	<i>Thai</i>	<i>10–30</i>	<i>Igen</i>
X_{17}	<i>Igen</i>	<i>Nem</i>	<i>Igen</i>	<i>Nem</i>	<i>Tele</i>	<i>Drága</i>	<i>Nem</i>	<i>Igen</i>	<i>Olasz</i>	<i>0-10</i>	<i>Igen</i>
X_{18}	<i>Nem</i>	<i>Igen</i>	<i>Nem</i>	<i>Igen</i>	<i>Néhány</i>	<i>Közep</i>	<i>Nem</i>	<i>Igen</i>	<i>Olasz</i>	<i>0–10</i>	<i>Igen</i>
X_{19}	<i>Nem</i>	<i>Igen</i>	<i>Nem</i>	<i>Igen</i>	<i>Senki</i>	<i>Olcsó</i>	<i>Igen</i>	<i>Nem</i>	<i>Burger</i>	<i>0–10</i>	<i>Nem</i>
X_{20}	<i>Igen</i>	<i>Nem</i>	<i>Nem</i>	<i>Igen</i>	<i>Néhány</i>	<i>Közep</i>	<i>Igen</i>	<i>Igen</i>	<i>Thai</i>	<i>0–10</i>	<i>Igen</i>

X_{13}	TP	TP = 4, TN = 1,
X_{14}	FP	FP = 1, FN = 2
X_{15}	FN	TPR = TP / (TP+FN) = 4/6 = .66
X_{16}	TP	FPR = FP / (FP+TN) = 1/2 = .5
X_{17}	FN	ACC = (TP+TN)/(P+N) = 5/8 = .625
X_{18}	TP	...
X_{19}	TN	
X_{20}	TP	

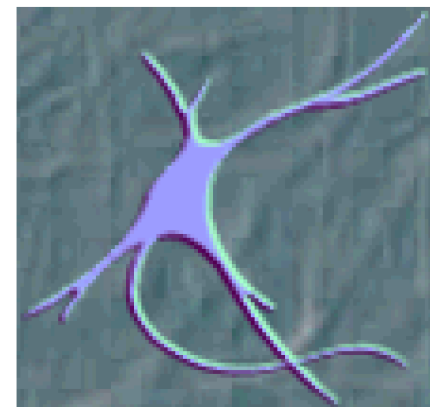
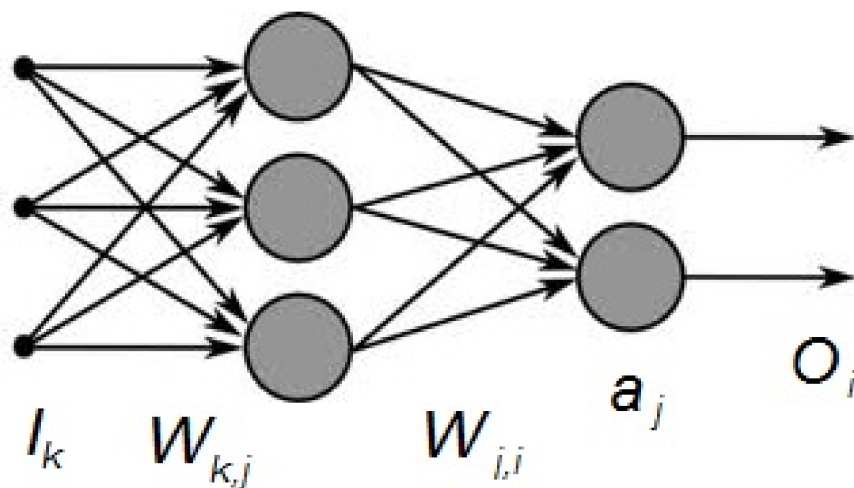
Teszt halmaz



Neurális hálók tanulása

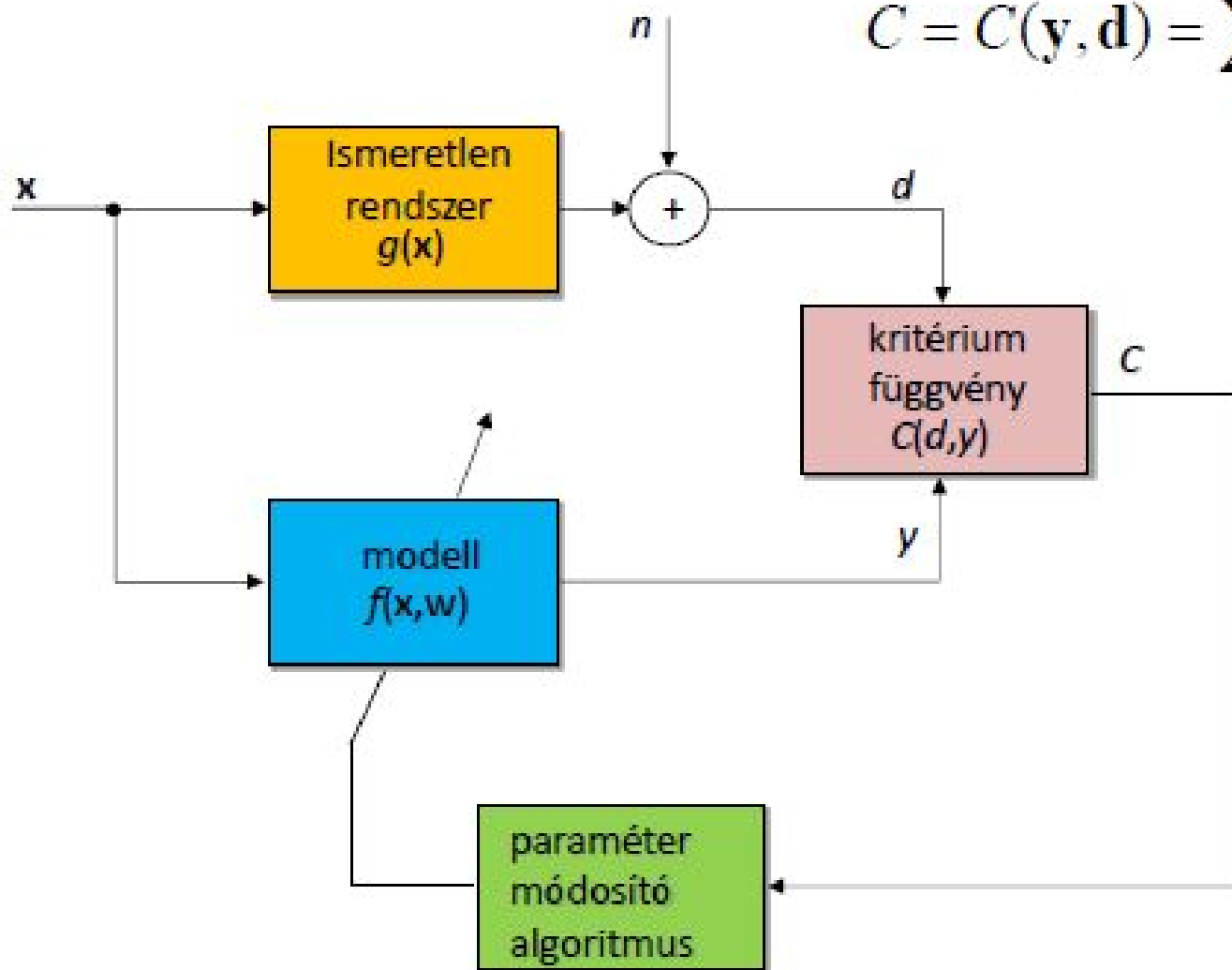
Mesterséges neuronháló

- Az elemi (biológiai) neuron nagyon leegyszerűsített modellje
- Sok elemi neuron megfelelően összekötött rendszere
- Architektúra, szabad paraméterek
- A paraméterek (és az architektúra) módosításával a képességek változnak, javulnak: tanulásra képes rendszer



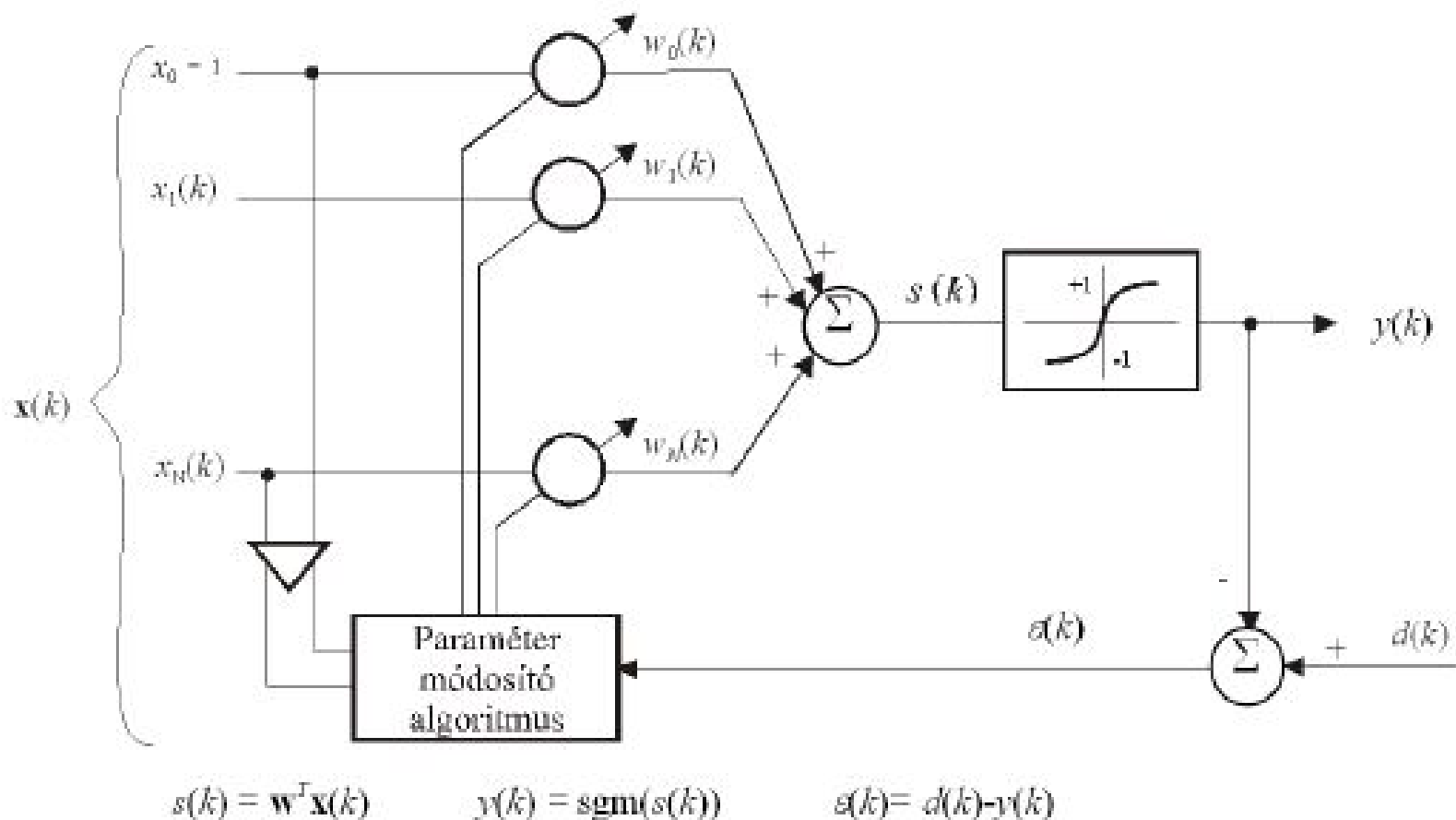
Neurális hálók tanulása (inkr. modell-adaptáció)

$$C = C(\mathbf{y}, \mathbf{d}) = \sum_i (d_i - y_i)^2$$



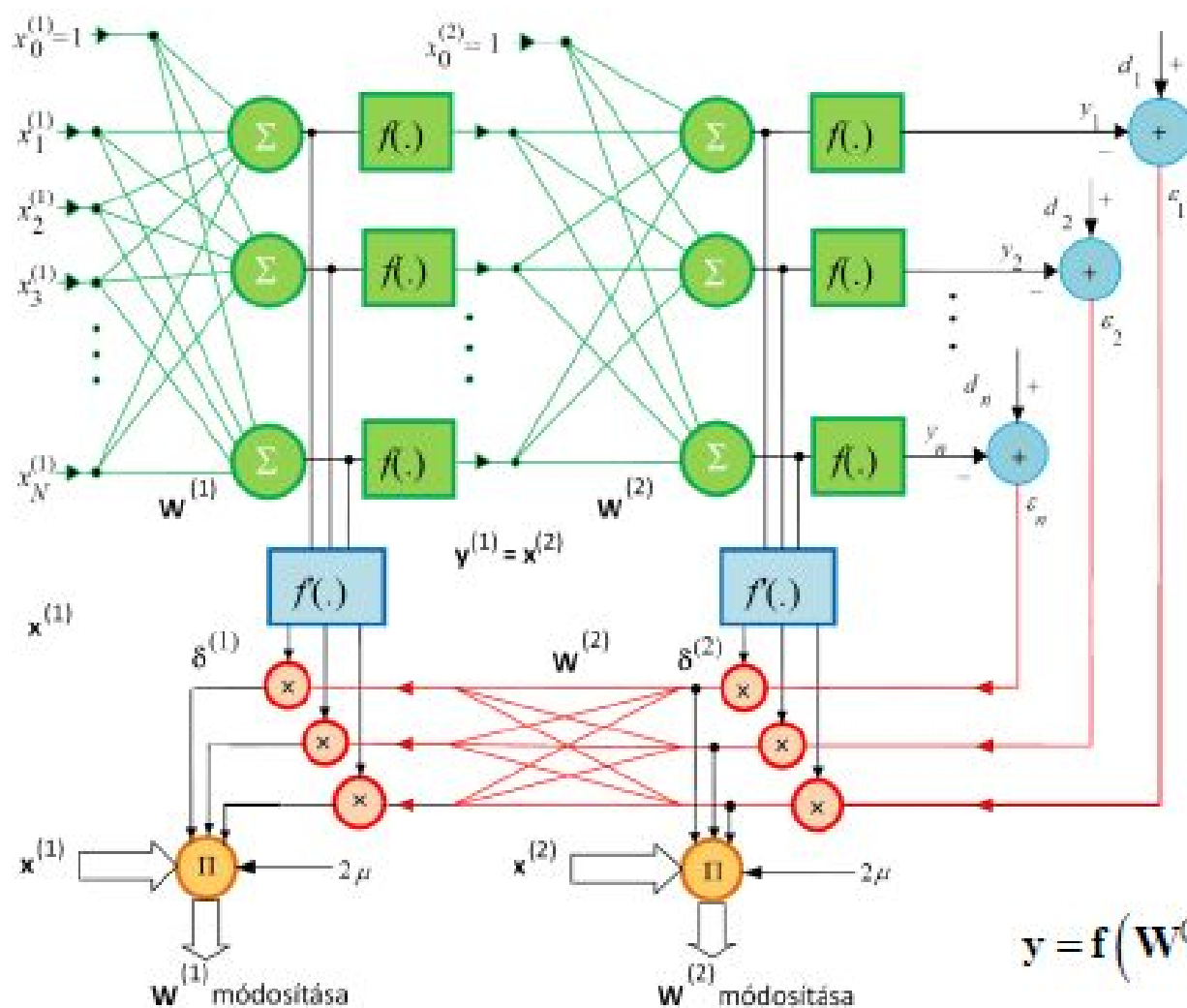
Neurális hálók tanulása

Elemi neuron



Neurális hálók tanulása

Többrétegű perceptron és tanítása



Neurális hálók tanulása

Elemzés

Kifejezőképesség: nem rendelkeznek az általános logikai reprezentációk kifejezőképességével. A többrétegű hálók osztálya egészében, mint osztály az attribútumok bármely függvényének reprezentációjára képes.

Számítási hatékonyság: legrosszabb esetben a szükséges epochok száma a bemenetek számának, n -nek, még exponenciális függvénye is lehet. A hibafelület lokális minimumai problémát jelenthetnek.

Általánosító képesség: jó általánosításra képesek.

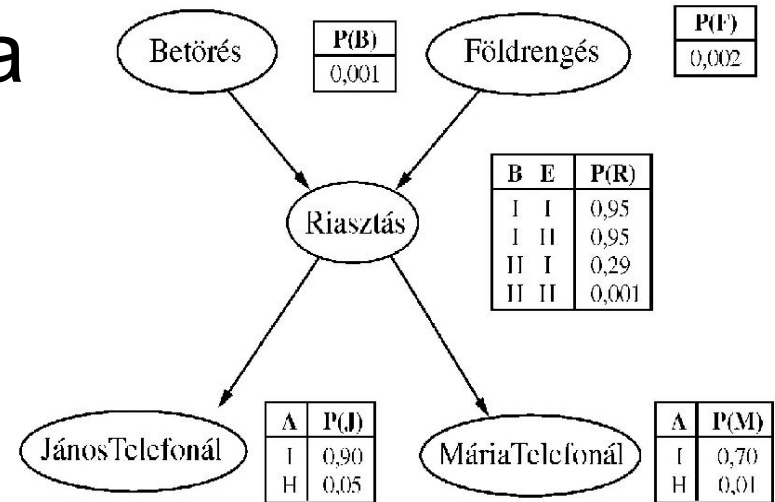
(tanulási) Zajérzékenység: alapvetően nemlineáris regresszió, nagyon jól tolerálják a bemeneti adatok zajosságát.

Átláthatóság: alapvetően fekete doboz.

A priori ismeret: ?

Valószínűségi hálók tanítása

Mit lehet itt tanítani?



Ismert struktúra, teljesen megfigyelhető változók:

tanulandó: a feltételes valószínűségek táblázata, közvetlenül becsülhető a példahalmaz alapján.

Ismeretlen struktúra, teljesen megfigyelhető változók:

tanulandó: háló topológiája

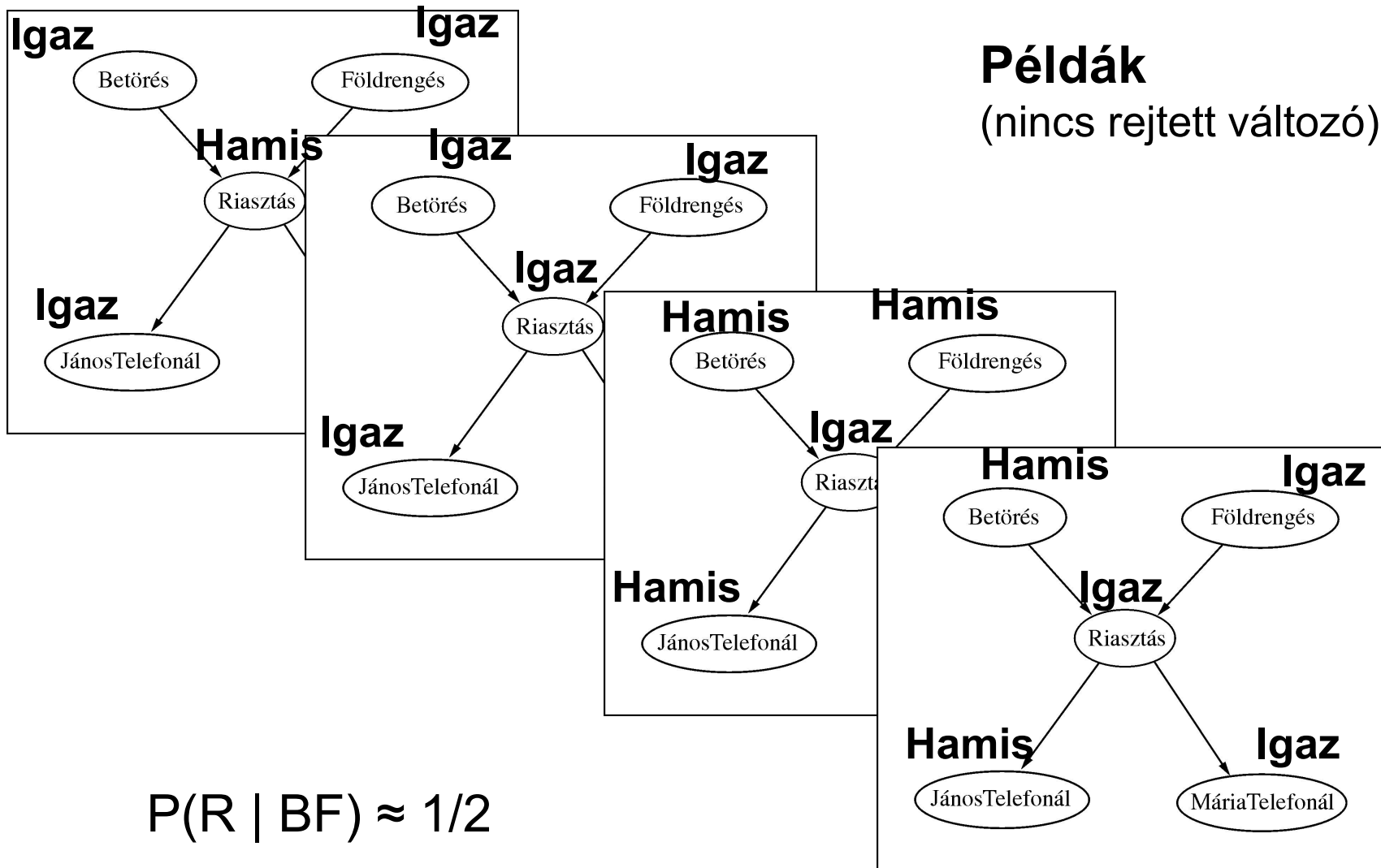
struktúrák terében való keresés, a keresést az egyes struktúráknak az adatok modellezési képessége irányítja.

Ismert struktúra, rejtett változók:

a neurális háló tanulással analóg

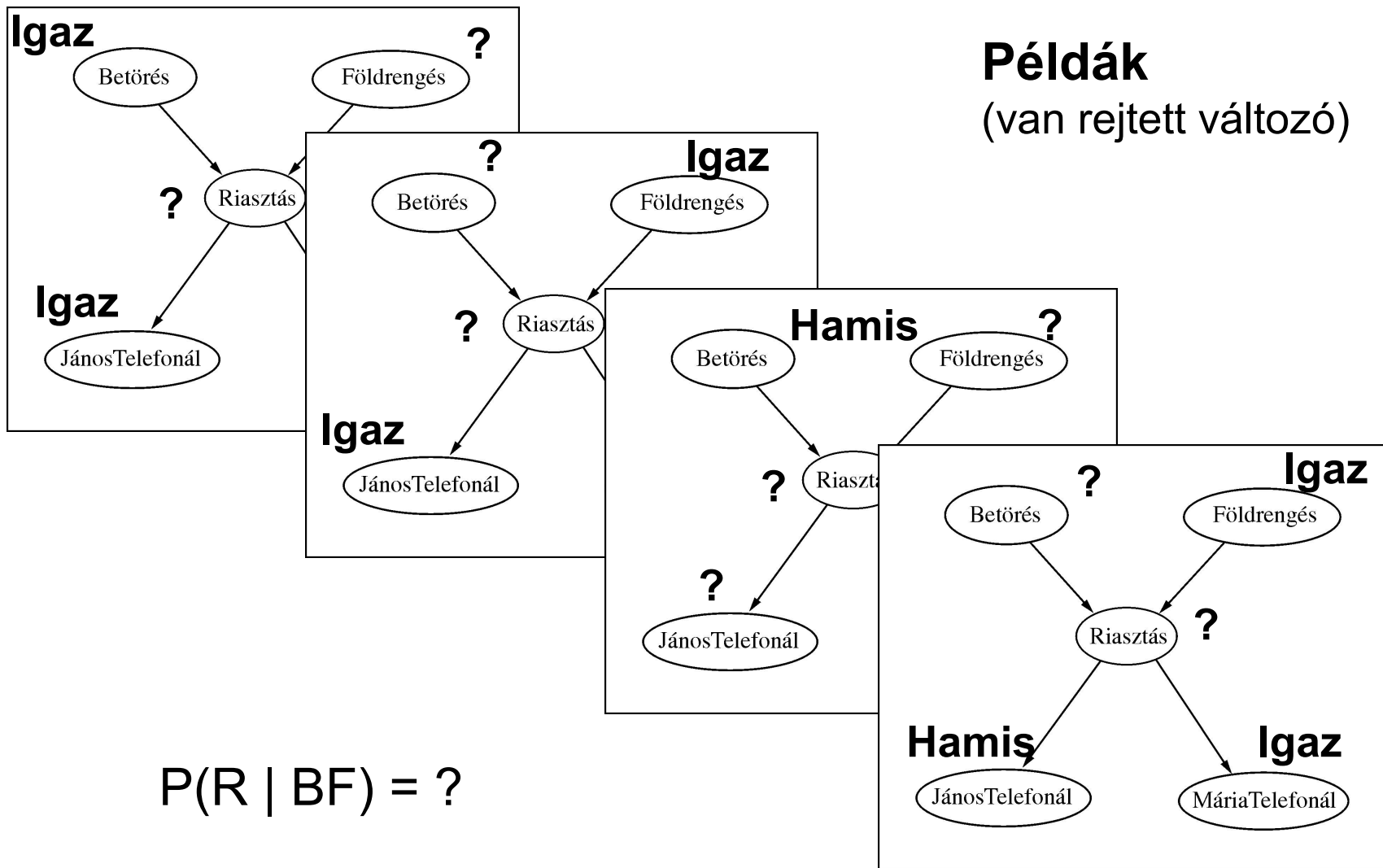
Ismeretlen struktúra, rejtett változók:

jelenleg nem ismert jól működő, általános algoritmus erre az esetre.



...

Valószínűségi hálók tanítása



...

Valószínűségi hálók tanítása

Valószínűségi hálók tanítása

Bayes tanulás

Általános predikció - hipotézis konstruálása adatok alapján:

1. az összes hipotézis valószínűségét megbecsüljük az adatokból.
2. a hipotézisek alapján predikciókat készítünk.
3. a predikciókat a hipotézisek a posteriori valószínűségével súlyozzuk.

D adatok, **$H_1, H_2, \dots$** hipotézisek, ismeretlen **X** predikciója

- minden egyes **H_i** az **X** egy teljes eloszlását specifikálja

$$P(X) = \sum_i P(X | H_i) P(H_i)$$

$$P(X | D) = \sum_i P(X | D, H_i) P(H_i | D) = \sum_i P(X | H_i) P(H_i | D)$$

Teljes Bayes tanulás: $P(H_i | D)$ kiszámítása az összes H_i -re.

A legtöbb esetben = kezelhetetlen, de nincs jobb módja az optimalis predikció készítésnek.

A legelterjedtebb közelítés - a **legvalószínűbb hipotézis**: $\max P(H_i | D)$

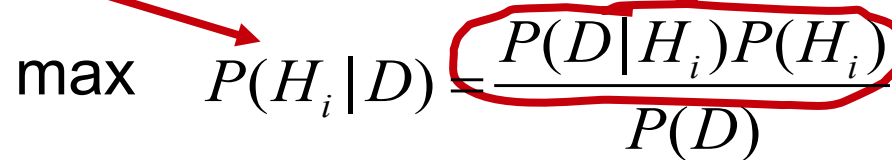
$$P(H_{MAP} | D) \approx 1, \quad P(H_{más} | D) \approx 0$$

Maximum a posteriori (MAP) hipotézis melletti predikció:

$$P(X|D) \approx P(X|H_{MAP})$$

A probléma H_{MAP} megtalálása.

MAP hipotézis:

$$\max P(H_i|D) = \frac{P(D|H_i)P(H_i)}{P(D)}$$


a priori
valószínűségek

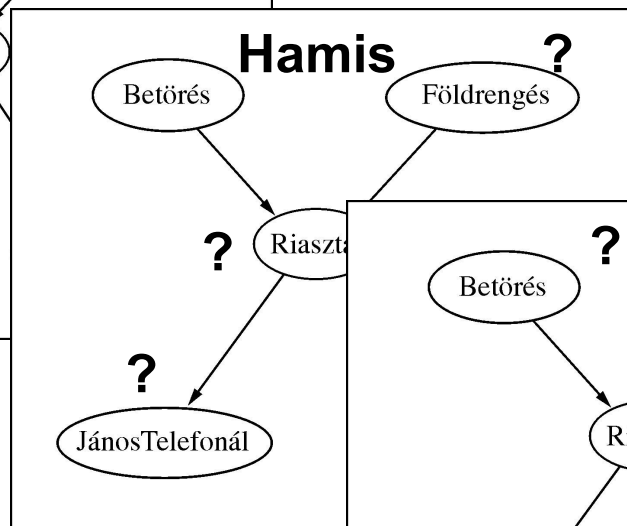
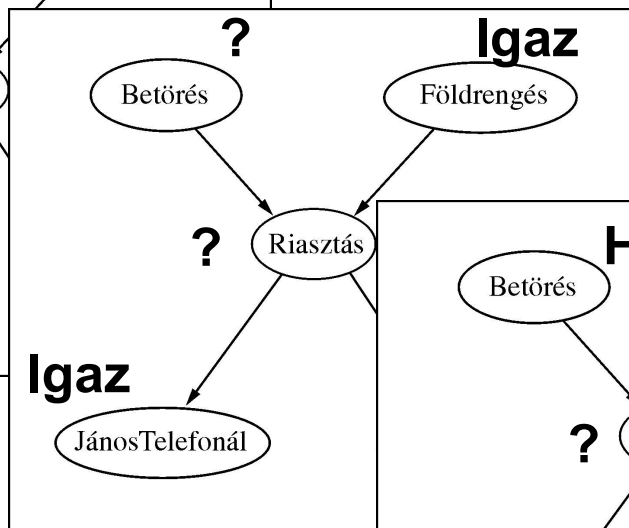
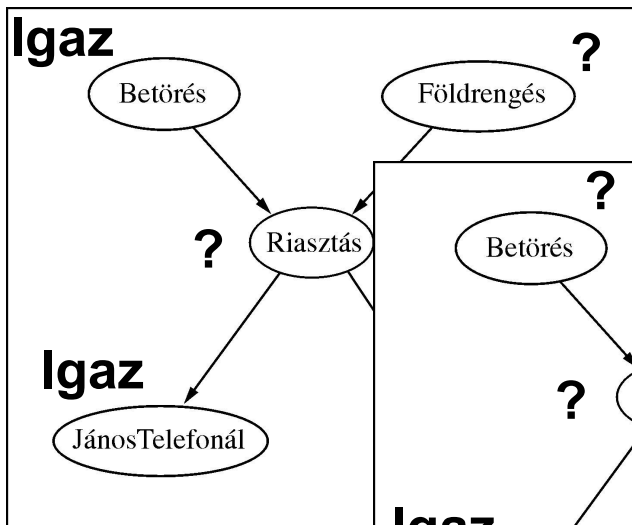
Maximálizálás: probléma az a priori valószínűség

- a priori valószínűségeket rendelünk az egyes hipotézisekhez, összegük 1
- bizonyos esetekben: **egyenletes** a priori eloszlás (tudás hiánya, de néha jobb nem tudni!)

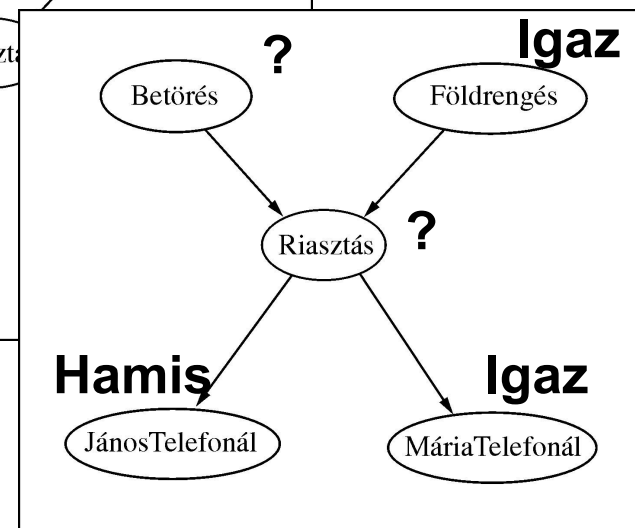
$$\max_i P(H_i|D) = \max_i P(D|H_i)$$

maximum likelihood (ML) hipotézis: H_{ML}

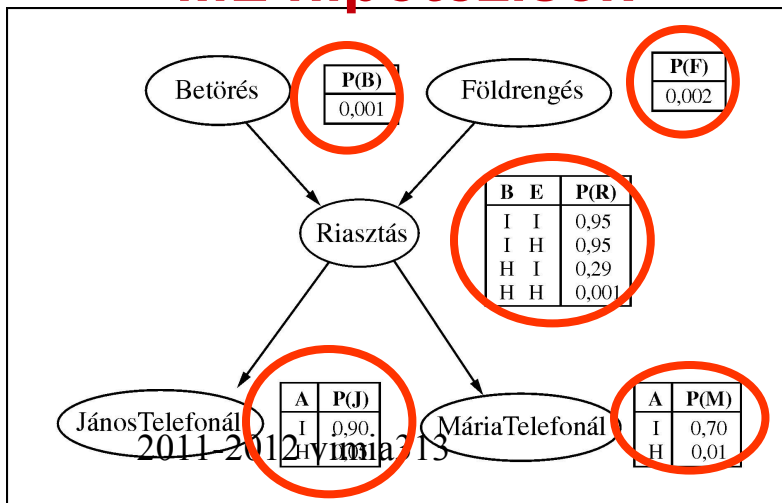
Valószínűségi hálók tanítása



Példák



ML hipotézisek



...

alószínűségi hálók tanítása

Tanulás rögzített struktúrájú hálók esetén

A háló topológiája egyszerűbb feladat. A szakemberek meg tudják mondani, hogy mi minek az oka, de nehéz pontos számokat rendelni a kapcsolatokhoz. Ez különösen is igaz, ha néhány változó a valós esetekben nem figyelhető meg közvetlenül.

A „ismert struktúra, rejtett változók” eset: **FONTOS**

Adaptív valószínűségi háló

H_i hipotézisek = a feltételes valószínűségi táblák (FVT).

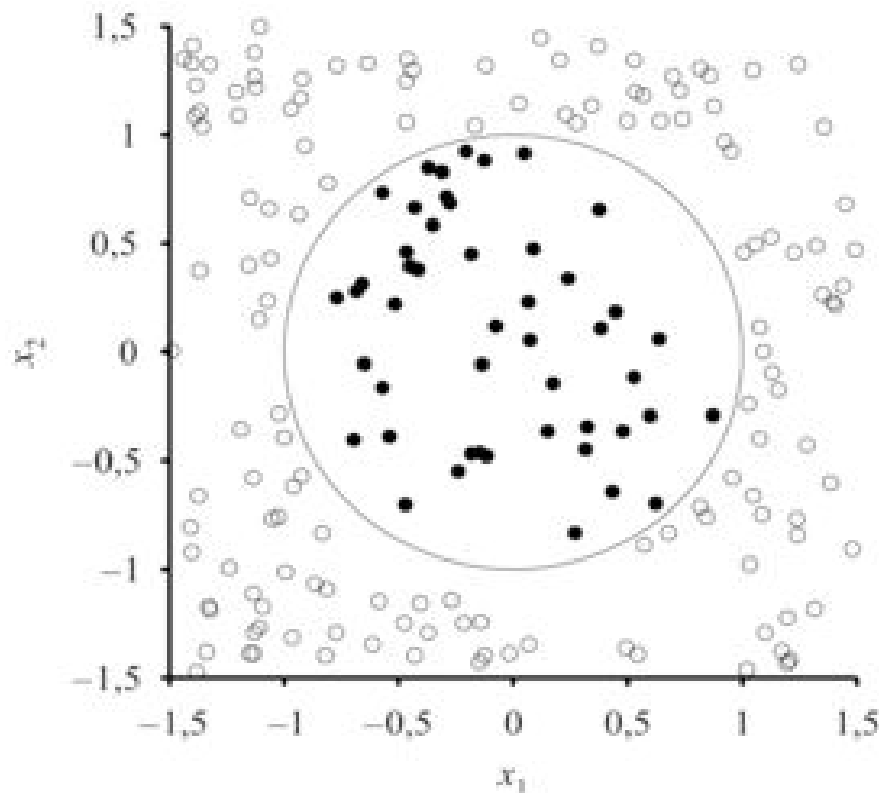
Az összes lehetséges táblázatkitöltés a priori valószínűsége azonos
= **ML** hipotézis.

Az FVT értékeinek egy olyan kombinációja, amely maximálja az adatok valószínűségét, $P(D | H_i)$ –t (analóg az E hibával a neurális hálóban).

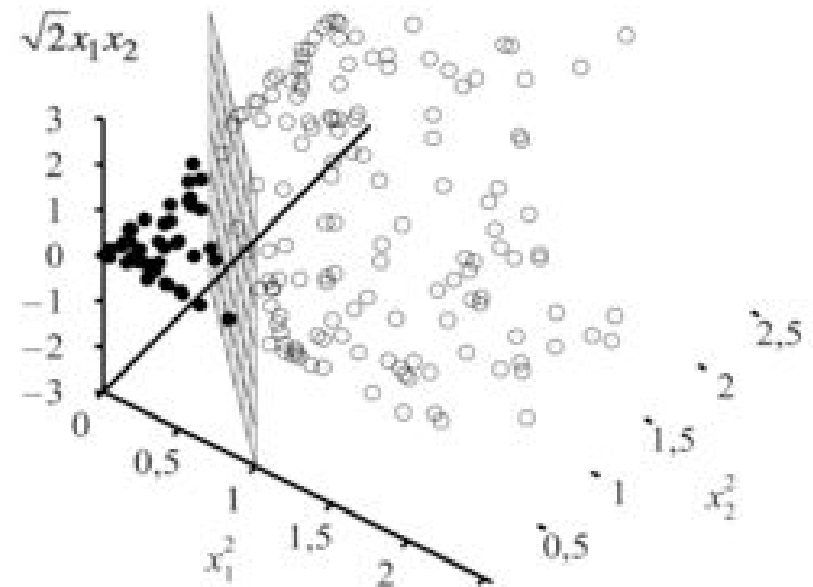
$$W_{i,új} \leftarrow W_{i,régi} - \alpha \frac{\partial E}{\partial W_i}$$
$$W_{i,új} \leftarrow W_{i,régi} - \alpha \frac{\sum_i P(x_i, u_j | D_j)}{W_{i,régi}}$$

Valószínűségi
hálók tanítása

Kernelgépek (SVM, szupport vektor gépek)



(a)



(b)

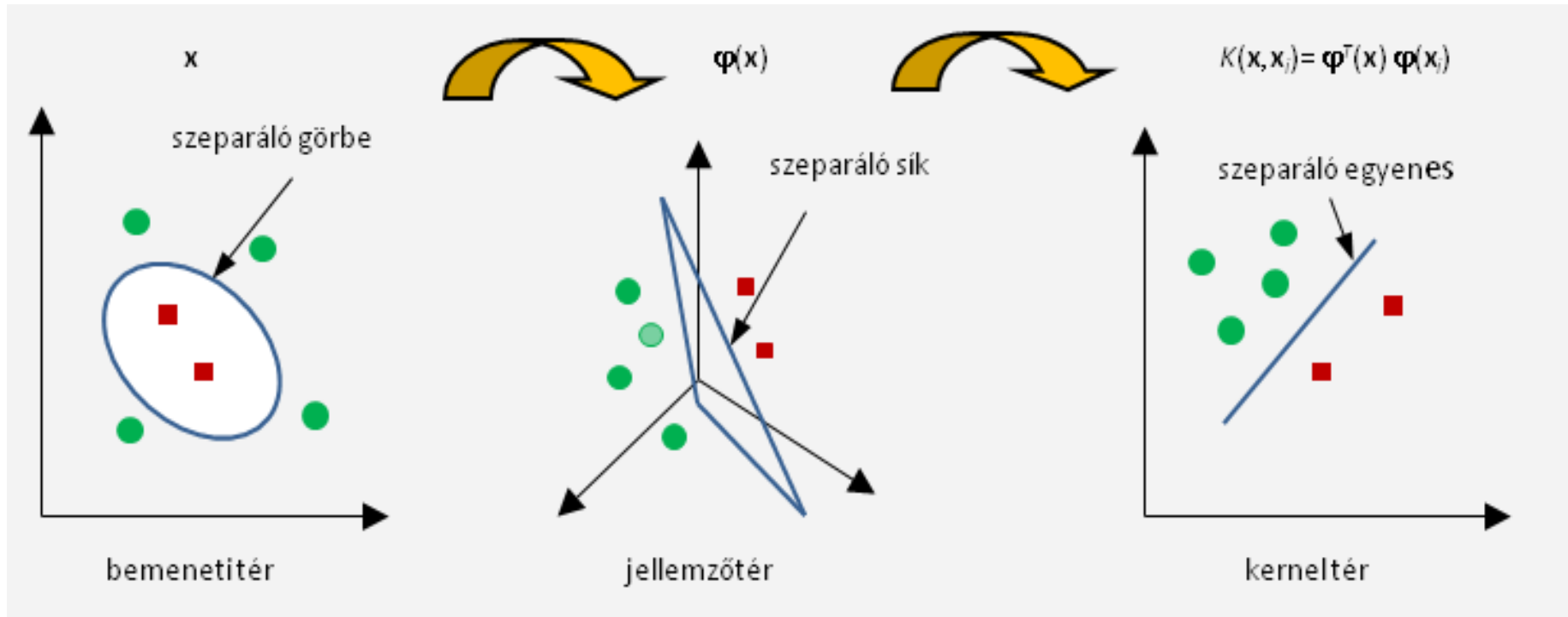
$$x_1^2 + x_2^2 \leq 1$$

Feladattér – nemlineáris osztályozó határfelület

Transzformált (dimenziót növelő) tér – lineáris osztályozó hipersík

$$x_1^2, x_2^2, \sqrt{2}x_1x_2$$

Kernelgépek (SVM, szupport vektor gépek)



Kerneltrükk

Optimális szétválasztás

Optimális szeparátor =

(a) max. tartalék

(b) szupport vektorok

$\mathbf{x}_i$ példák, $d_i = \pm 1$ besorolás

$a_i \geq 0$, $\sum_i a_i d_i = 0$

max

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2} \mathbf{w}^T \mathbf{w} - \sum_{i=1}^P \alpha_i [d_i (\mathbf{w}^T \mathbf{x}_i + b)]$$

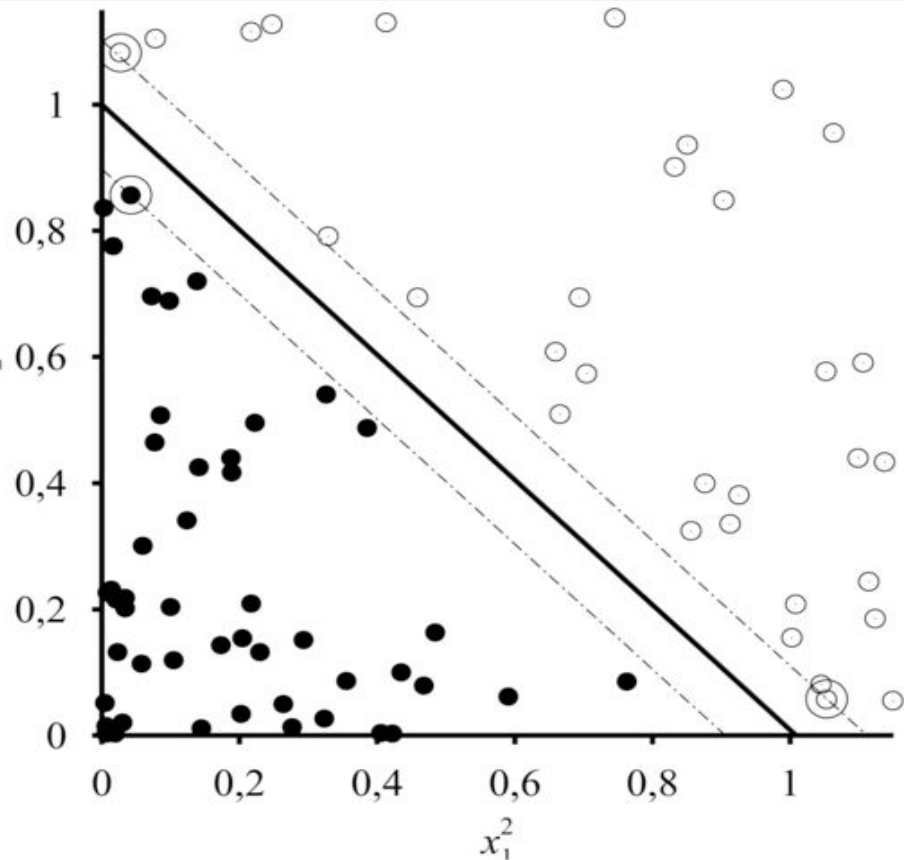
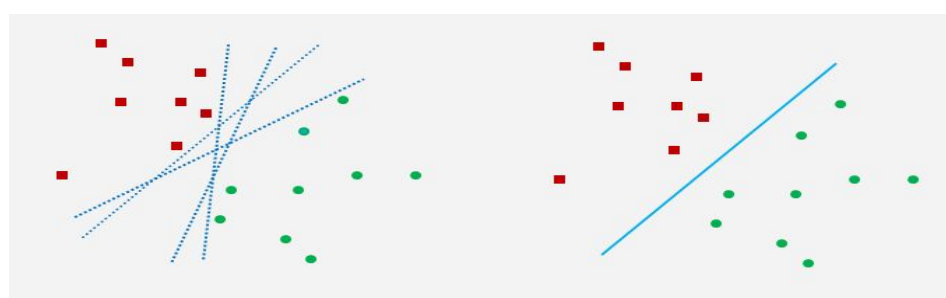
max

$$Q(\alpha) = \sum_{i=1}^P \alpha_i - \frac{1}{2} \sum_{i=1}^P \sum_{j=1}^P \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T \mathbf{x}_j$$

Optimális szeparátor

$$y(\mathbf{x}) = \text{sign} \left[\sum_{i=1}^P \alpha_i^* d_i \mathbf{x}_i^T \mathbf{x} + b^* \right]$$

$$y(\mathbf{x}) = \text{sign} \left[\sum_{i=1}^P \alpha_i^* d_i K(\mathbf{x}_i, \mathbf{x}) + b^* \right]$$



- optim. $\neq 0$ a_i – szupport vektorok
- csak skalárszorzat ($\mathbf{x}_i \mathbf{x}_j$)
- tulajdonságtérben: $F(\mathbf{x}_i) F(\mathbf{x}_j) = K(\mathbf{x}_i \mathbf{x}_j)$

Szekvenciális döntési probléma

Markov döntési folyamat

Kezdőállapot: S_0
Állapotátmenet-modell: $T(s, a, s')$
Jutalomfüggvény: $R(s)$, v. $R(s, a, s')$

Optimális eljárasmód = optimális mozgás, döntés cselekvésszekvencia megválasztására, amíg nincs a probléma vége.

$$\begin{aligned}\pi(s) &\leftarrow U(s), & U(s) &= R(s) + \gamma \max_a \sum_{s'} T(s, a, s') U(s') \\ \pi(s) &\leftarrow U_i(s) & U_{i+1}(s) &\leftarrow R(s) + \gamma \max_a \sum_{s'} T(s, a, s') U_i(s')\end{aligned}$$

Megerősítő tanulás

Kezdőállapot: ismert
Állapotátmenet-modell: **nem ismert, legfeljebb a tapasztalat**
Jutalomfüggvény: **nem ismert, legfeljebb ahogy kapunk**

Optimális eljárasmód = **megtanulni, mindennek ellenére?!**

Megerősítéssel tanulás

Pl. sakkjáték: tanító nélkül is van visszacsatolás a játék végén:
nyert vagy **vesztett** = +/- **jutalom**, azaz **megerősítés**.

... a játék végén = a cselekvés szekvencia végén

Megerősítés: **milyen gyakran? milyen erős?**

A feladat:

(ritka) jutalmakból megtanulni egy sikeres ágens-függvényt
(optimális eljárás mód: melyik állapot, melyik cselekvés hasznos).

Nehéz: az információhiány miatt az ágens sohasem tudja, hogy **mik a jó lépések**, azt sem, **melyik jutalom melyik cselekvésből ered**.

Ágens **tudása**:

induláskor **tudja már** a környezetet és cselekvéseinek hatását,
vagy pedig még ezt is **meg kell tanulnia**.

Megerősítés:

csak a **végállapotban**, vagy **bármelyikben**.

Ágens:

passzív tanuló: figyeli a világ alakulását és tanul.

aktív tanuló: a megtanult információ birtokában cselekednie is kell.
felfedező ...

Ágens kialakítása: megerősítés $\rightarrow$ hasznosság leképezés

U(i) hasznosság függvény tanulása, ennek alapján a cselekvések eldöntése, hogy az elérhető hasznosság várható értéke max legyen (környezet/ágens modellje **szükséges**)

Q(a, i) cselekvés érték függvény (állapot-cselekvés párok) tanulása, valamilyen várható hasznot tulajdonítva egy adott helyzetben egy adott cselekvésnek (környezet/ágens modell **nem szükséges**, közben tanult)
– **Q tanulás**

Fontos egyszerűsítés: a sorozat hasznossága

= a sorozat állapotaihoz rendelt hasznosságok összege.

= a hasznosság **additív**

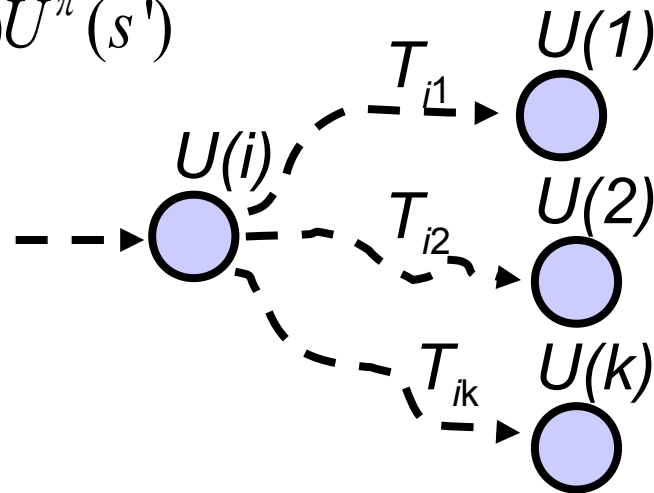
Az állapot **hátralevő-jutalma (reward-to-go)**:

azon jutalmak összege, amelyet akkor kapunk, ha az adott állapotból valamelyik végállapotig eljutunk.

Egy állapot várható hasznossága = a hátralevő-jutalom várható értéke

$$U^\pi(s) = E[\sum_t \gamma^t R(s_t) | \pi, s_0 = s]$$

$$U^\pi(s) = R(s) + \gamma \sum_{s'} T(s, \pi(s), s') U^\pi(s')$$



Adaptív Dinamikus Programozás

Az állapotátmeneti valószínűségek T_{ij} a megfigyelt gyakoriságokkal becsülhetők.

Amint az ágens megfigyelte az összes állapothoz tartozó jutalom értéket, a hasznosság-értékek a következő **egyenletrendszer megoldásával** kaphatók meg:

$$U^\pi(s) = R(s) + \gamma \sum_{s'} T(s, \pi(s), s') U^\pi(s')$$

megfigyelt $\nearrow$ $\nwarrow$ megtanult

$$\begin{pmatrix} U(1) \\ U(2) \\ U(3) \\ \dots \\ U(N) \end{pmatrix} = \begin{pmatrix} R(1) \\ R(2) \\ R(3) \\ \dots \\ R(N) \end{pmatrix} + \gamma \begin{pmatrix} T_{11} & T_{12} & T_{13} & \dots & T_{1N} \\ T_{21} & T_{22} & T_{23} & \dots & T_{2N} \\ T_{31} & T_{32} & T_{33} & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ T_{N1} & T_{N2} & \dots & \dots & T_{NN} \end{pmatrix} \begin{pmatrix} U(1) \\ U(2) \\ U(3) \\ \dots \\ U(N) \end{pmatrix}$$

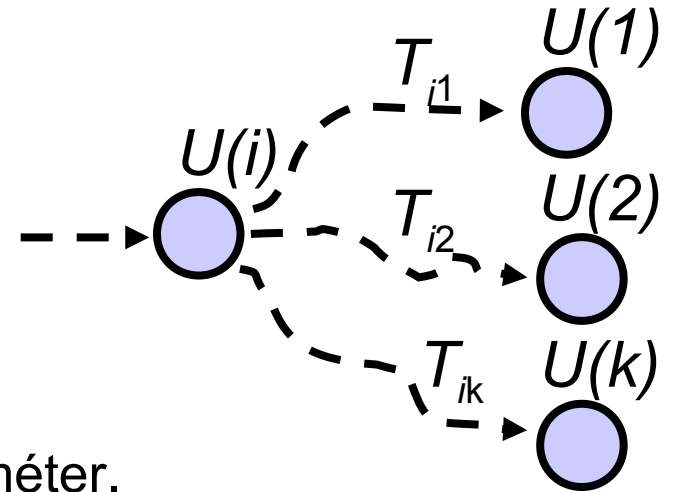
$$\mathbf{U} = \mathbf{R} + \gamma \mathbf{TU} \quad \mathbf{U} = (\mathbf{I} - \gamma \mathbf{T})^{-1} \mathbf{R}$$

és ha nem megy?

Az időbeli különbség (IK) tanulás (TD – Time Difference)

Alapötlet: használjuk fel a megfigyelt állapotátmeneteket a megfigyelt állapotok olyan frissítésére, hogy megfeleljenek a korlátozó egyenleteknek.

$$U(i) \leftarrow U(i) + \alpha (R(i) + U(j) - U(i))$$



α a **bátorsági faktor**, tanulási tényező paraméter.

Az egymást követő állapotok hasznosság-különbsége **időbeli különbség (IK) (temporal difference, TD)**.

$$\begin{aligned} U(i) &\leftarrow (1 - \alpha)U(i) + \alpha (R(i) + U(j)) \\ &= U(i) + \alpha (R(i) + U(j) - U(i)) \end{aligned}$$

Az időbeli különbség (IK) tanulás (TD – Time Difference)

1. korrekt hasznosság-értékek esetén **lokálisan fennálló feltételek rendszere:**

$$U^{\pi}(s) = R(s) + \gamma \sum_{s'} T(s, \pi(s), s') U^{\pi}(s')$$

2. **módosító egyenlet**, amely a becsléseinket ezen „**egyensúlyi**” **egyenlet irányába** módosítja:

$$U(i) \leftarrow U(i) + \alpha (R(i) + \gamma U(j) - U(i))$$

3. Csak a **(A)DP** módszer használta fel a **modell teljes** ismeretét. Az **IK** az állapotok közt fennálló kapcsolatokra vonatkozó információt használja, de csak azt, amely az **aktuális tanulási sorozatból származik**.

Az **IK** változatlanul **működik előzetesen ismeretlen környezet esetén is**.

Ismeretlen környezetben végzett aktív tanulás

Döntés: melyik cselekvés? a cselekvésnek mik a kimenetelei?

hogyan hatnak az elért jutalomra?

A környezeti modell: a többi állapotba való **átmenet valószínűsége egy adott cselekvés esetén.**

$$U(s) = R(s) + \gamma \max_a \sum_{s'} T(s, a, s') U(s')$$

Az állapottér felderítése - milyen cselekvést válasszunk? **Nehéz probléma**

Helyes-e azt a cselekvést választani, amelynek a jelenlegi hasznosság-becslés alapján legnagyobb a várható hasznossága?

Ez figyelmen kívül hagyja a **cselekvésnek a tanulásra gyakorolt hatását.**

Döntésnek kétféle hatása van:

1. Jutalmat eredményez a **jelenlegi** szekvenciában.
2. Befolyásolja az észleléseket, és ezáltal az ágens **tanulási képességét**
– így jutalmat eredményezhet a **jövőbeni** szekvenciákban.

Az állapottér felderítése

Kompromisszum: a **jelenlegi jutalom**, amit a pillanatnyi hasznosság-becslés tükröz, és a **hosszú távú előnyök** közt.

Két megközelítés a cselekvés kiválasztásában:

„**Hóbortos, Felfedező**„: véletlen módon cselekszik, annak reményében, hogy végül is felfedezi az egész környezetet

„**Mohó**„: a jelenlegi becslésre alapozva maximalizálja a hasznot.

Hóbortos: képes jó hasznosság-becsléseket megtanulni az összes állapotról.
Sohasem sikerül fejlődnie az optimális jutalom elérésében.

Mohó: gyakran talál egy jó utat. Utána ragaszkodik hozzá, és soha nem tanulja meg a többi állapot hasznosságát.

Ágens addig legyen hóbortos, amíg kevés fogalma van a környezetről, és legyen mohó, amikor a valósághoz közeli modellel rendelkezik.

Létezik-e optimális felfedezési stratégia?

Ágens súlyt kell adjon azoknak a cselekvéseknek, amelyeket még nem nagyon gyakran próbált, a kis hasznosságúnak gondolt cselekvéseket elkerülje.

felfedezési függvény

$$f(u,n) = \begin{cases} R^+ & \text{ha } n < N_e \\ u & \text{különben} \end{cases}$$

mohóság $\leftrightarrow$ kíváncsiság

$f(u,n)$ u -ban monoton növekvő,
 n -ben monoton csökkenő

$$U^+(s) \leftarrow R(s) + \gamma \max_a f(\sum_{s'} T(s, a, s') U^+(s'), N(a, s))$$

$U^+(i)$: az i állapothoz rendelt hasznosság optimista becslése

$N(a,i)$: az i állapotban hányszor próbálkoztunk az a cselekvéssel

R^+ a tetszőleges állapotban elérhető legnagyobb jutalom optimista becslése

Az a cselekvés, amely felderítetlen területek *felé* vezet, nagyobb súlyt kap.

ϵ -mohóság: az ágens ϵ valószínűséggel véletlen cselekvést választ, ill. $1-\epsilon$ valószínűséggel mohó

Boltzmann-felfedezési modell

egy a cselekvés megválasztásának valószínűsége

egy s állapotban:

$$P(a, s) = \frac{e^{\text{hasznosság}(a, s)/T}}{\sum_{a'} e^{\text{hasznosság}(a', s)/T}}$$

a T „hőmérséklet” a két véglet között szabályoz.

Ha $T \rightarrow \infty$, akkor a választás tisztán (egyenletesen) véletlen,

ha $T \rightarrow 0$, akkor a választás mohó.

A cselekvés-érték függvény tanulása

A cselekvés-érték függvény egy adott állapotban választott adott cselekvéshez egy várható hasznosságot rendel: **Q-érték**

A Q-értékek fontossága:

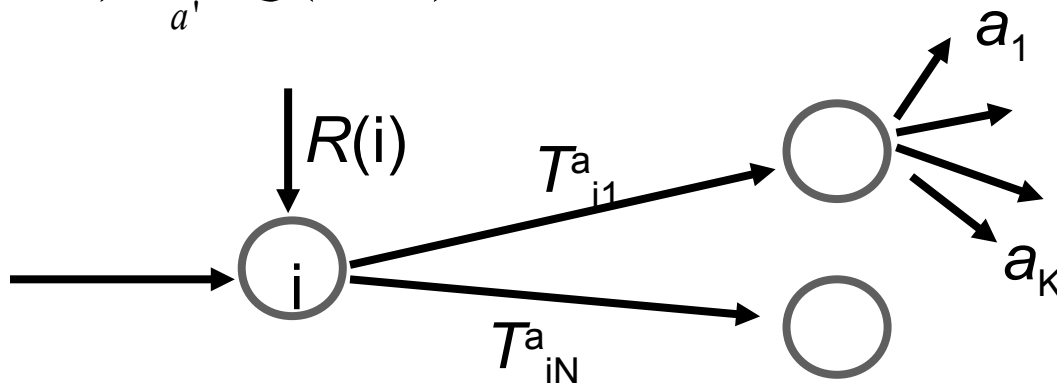
a feltétel-cselekvés szabályokhoz hasonlóan lehetővé teszik a döntést **modell használata** nélkül,

ellentétben a feltétel-cselekvés szabályokkal, **közvetlenül** a jutalom visszacsatolásával **tanulhatók**.

$$U(i) = \max_a Q(a, i)$$

Mint a hasznosság-értékekénél, felírhatunk egy kényszer egyenletet, amely **egyensúlyi állapotban**, amikor a **Q-értékek korrektek**, fenn kell álljon:

$$Q(a, s) = R(s) + \gamma \sum_{s'} T(s, a, s') \max_{a'} Q(a', s')$$



A cselekvés-érték függvény tanulása

Ezt az egyenletet közvetlenül felhasználhatjuk egy olyan iterációs folyamat módosítási egyenleteként, amely egy adott modell esetén a pontos Q-értékek számítását végzi.

Az **időbeli különbség** viszont nem igényli a modell ismeretét. Az **IK** módszer **Q-tanulásának** módosítási egyenlete:

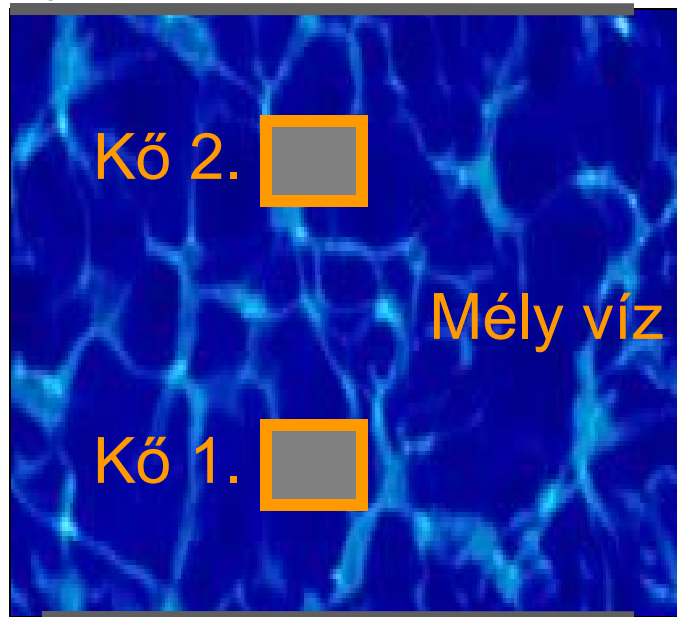
$$Q(a, i) \leftarrow Q(a, i) + \alpha (R(i) + \gamma \max_{a'} Q(a', j) - Q(a, i))$$

SARSA Q-tanulás (State-Action-Reward-State-Action)

$$Q(a, i) \leftarrow Q(a, i) + \alpha (R(i) + \gamma Q(a', j) - Q(a, i))$$

(a' megválasztása pl. Boltzmann felfedezési modellből)

Folyópart B



Folyópart A

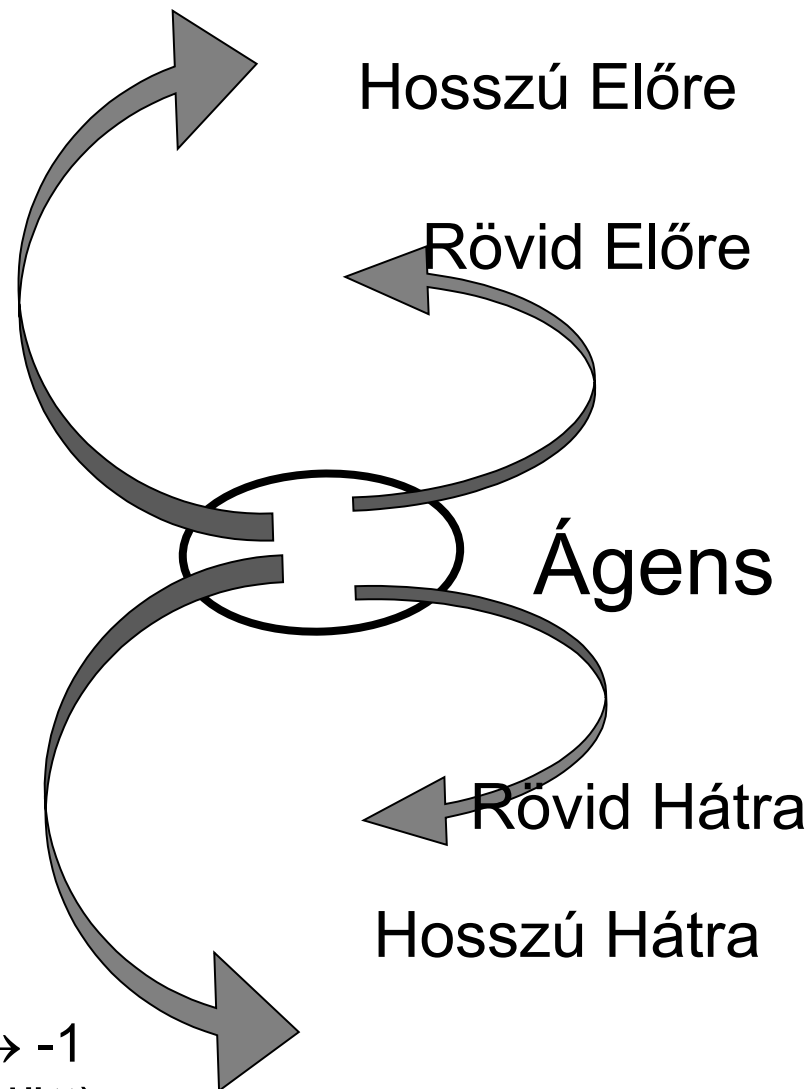
Tanuló szekvencia:

RE → HE → RE → +1 (száraz lábbal át)

RE → HE → HH → HE → HH → RH → HE → -1

RE → HE → RH → -1 (megfürdött)

.....



Part B

Kő 2.



$$Q(a,i) \leftarrow Q(a,i) + \alpha(R(i) + \max_{a'} Q(a',j) - Q(a,i))$$

Kő 1.



Part A

Cselekvés

HH

RH

RE

HE

Állapot

Part A
Kő 1.
Kő 2.

0

0

-0.035

-0.877

-0.520

-0.550

-0.835

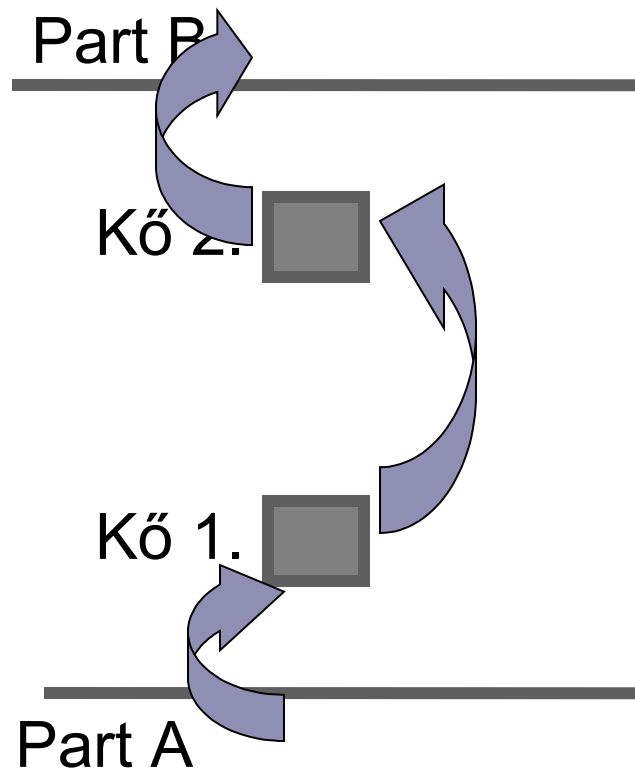
0.555

-0.204

-0.897

1.180

1.158



	HH	RH	RE	HE
Part A	0	0	-0.035	-0.877
Kõ 1.	-0.520	-0.550	-0.835	0.555
Kõ 2.	-0.204	-0.897	1.180	1.158

A megerősítéses tanulás általánosító képessége

Eddig: ágens által tanult hasznosság: **táblázatos** formában reprezentált
= **explicit reprezentáció**

Kis állapotterek: elfogadható, de a konvergencia idő és (ADP esetén) az iterációnkénti idő gyorsan nő a tér méretével.

Gondosan vezérelt közelítő ADP: 10000, vagy ennél is több.

De a való világhoz közelebb álló környezetek szóba se jöhetnek.

(a sakk stb. - állapottere 10^{50} - 10^{120} nagyságrendű)

(az összes állapotot látni, hogy megtanuljunk játszani?!)

Egyetlen lehetőség: a függvény **implicit reprezentációja**:

Pl. egy játékban a táblaállás tulajdonságainak valamilyen halmaza $f_1, \dots, f_n$.

Az állás becsült hasznosság-függvénye:

$$\hat{U}_\theta(s) = \theta_1 f_1(s) + \theta_2 f_2(s) + \dots + \theta_n f_n(s)$$

C. Shannon, Programming a Computer for Playing Chess, Philosophical Magazine, 7th series, 41, no. 314 (March 1950): 256-75

A. L. Samuel, Some Studies in Machine Learning Using the Game of Checkers.

II-Recent Progress, IBM. J. 3, 211-229 (1959), kb. 20 tulajdonság

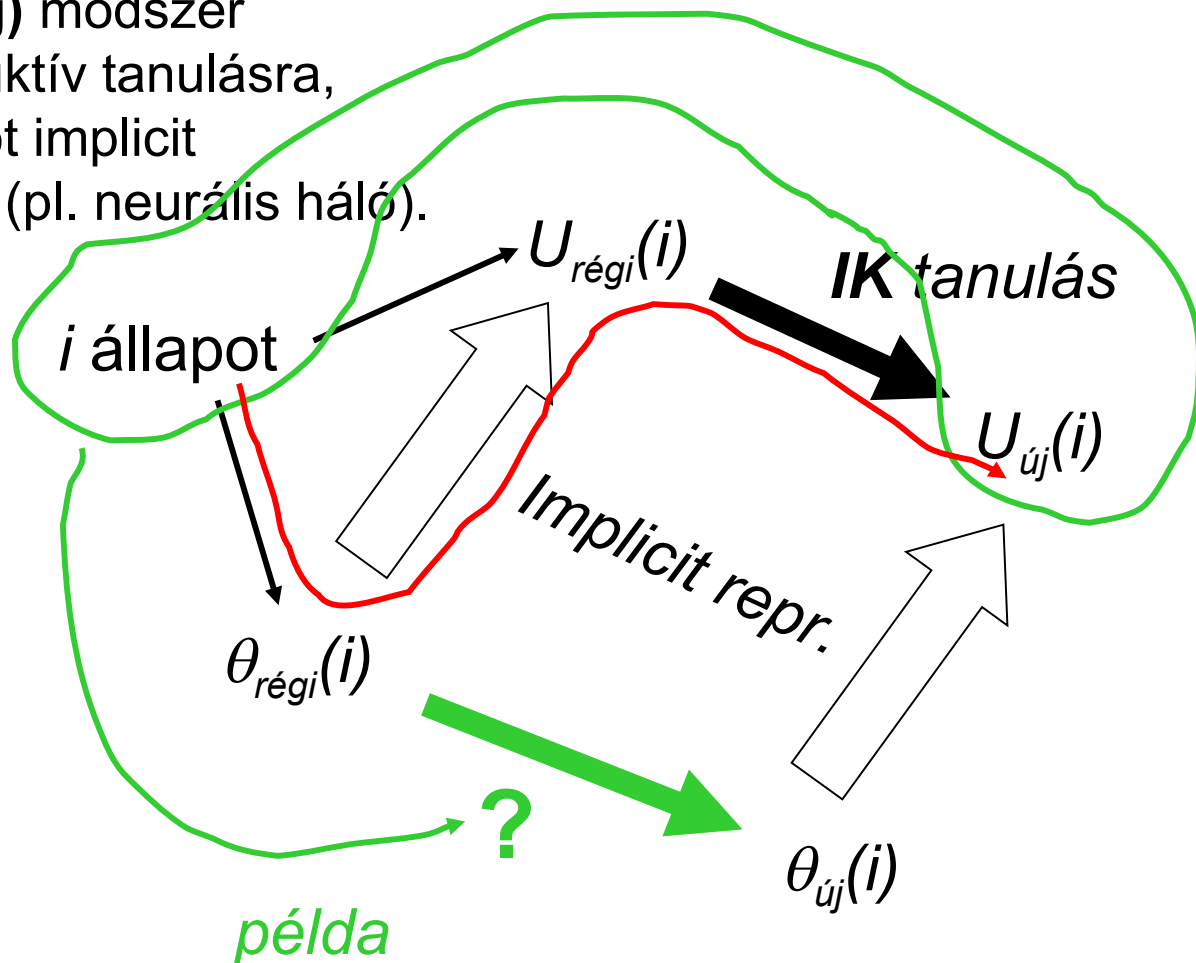
Deep Blue (Feng-Hsiung Hsu) kb. 8000 tulajdonság, spec. sakk-csip

A hasznosság-függvény n értékkel jellemezhető, pl. 10^{120} érték helyett. Egy átlagos sakk kiértékelő függvény kb. 10 súllyal épül fel, tehát *hatalmas* tömörítést értünk el.

Az implicit reprezentáció által elért tömörítés teszi lehetővé, hogy a tanuló ágens általánosítani tudjon a már látott állapotokról az eddigiekben nem látottakra.

Az implicit reprezentáció legfontosabb aspektusa nem az, hogy kevesebb helyet foglal, hanem az, hogy lehetővé teszi a **bemeneti állapotok induktív általánosítását**. Az olyan módszerekről, amelyek ilyen reprezentációt tanulnak, azt mondjuk, hogy **bemeneti általánosítást** végeznek.

Az **IK (időbeli különbség)** módszer szintén alkalmazható induktív tanulásra, ha az U , vagy Q táblázatot implicit reprezentációra cseréltük (pl. neurális háló).



$$\theta_i \leftarrow \theta_i + \alpha \left[R(s) + \gamma \hat{U}_\theta(s') - \hat{U}_\theta(s) \right] \frac{\partial \hat{U}_\theta(s)}{\partial \theta_i}$$

$$\theta_i \leftarrow \theta_i + \alpha \left[R(s) + \gamma \max_{a'} \hat{Q}_\theta(a', s') - \hat{Q}_\theta(a, s) \right] \frac{\partial \hat{Q}_\theta(a, s)}{\partial \theta_i}$$