Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar
Mesterséges Intelligencia és Rendszertervezés Tanszék

VIMIAC16 2025/26/I.

Regressziós modellek és neurális hálók

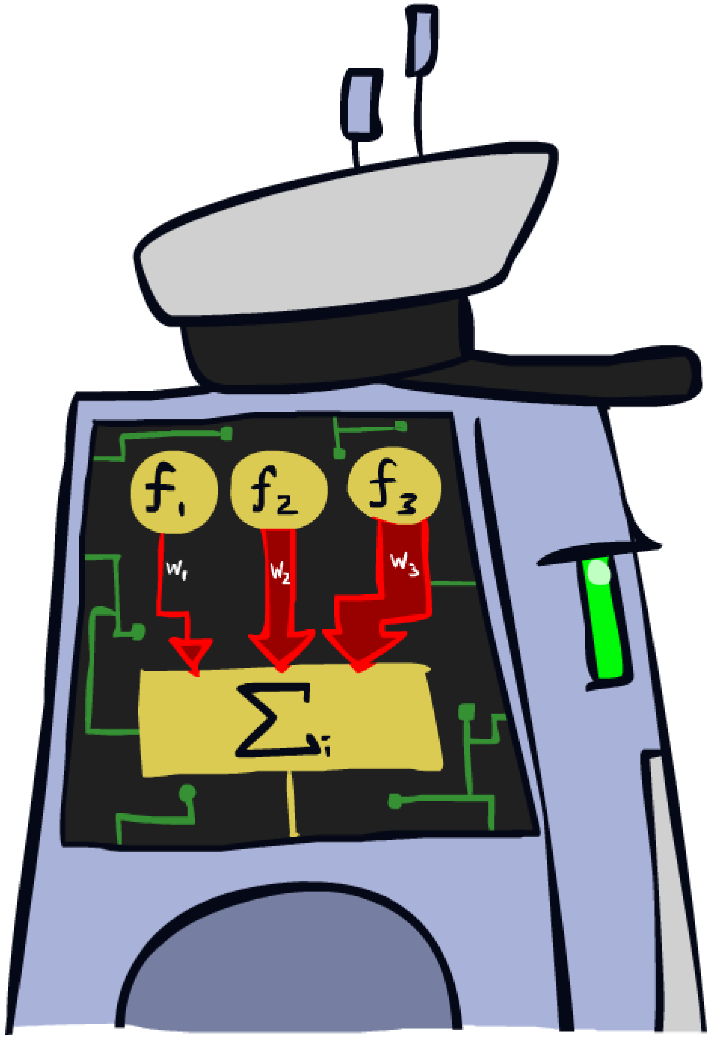
Előadó: Dr. Hullám Gábor

Mesterséges intelligencia előadássorozat

Az előadás diái az AIMA könyvre épülve (http://aima.cs.berkeley.edu) készültek a University of California, Berkeley mesterséges intelligencia kurzusának anyagainak felhasználásával (http://ai.berkeley.edu).

These slides are based on the AIMA book (http://aima.cs.berkeley.edu) and were adapted from the AI course material of University of California, Berkeley (http://ai.berkeley.edu).

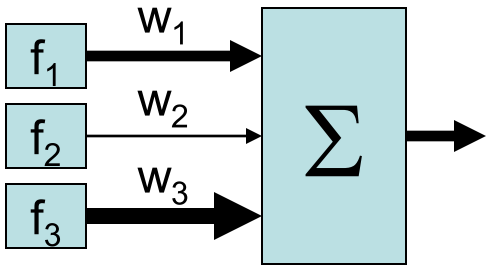
Lineáris modellek



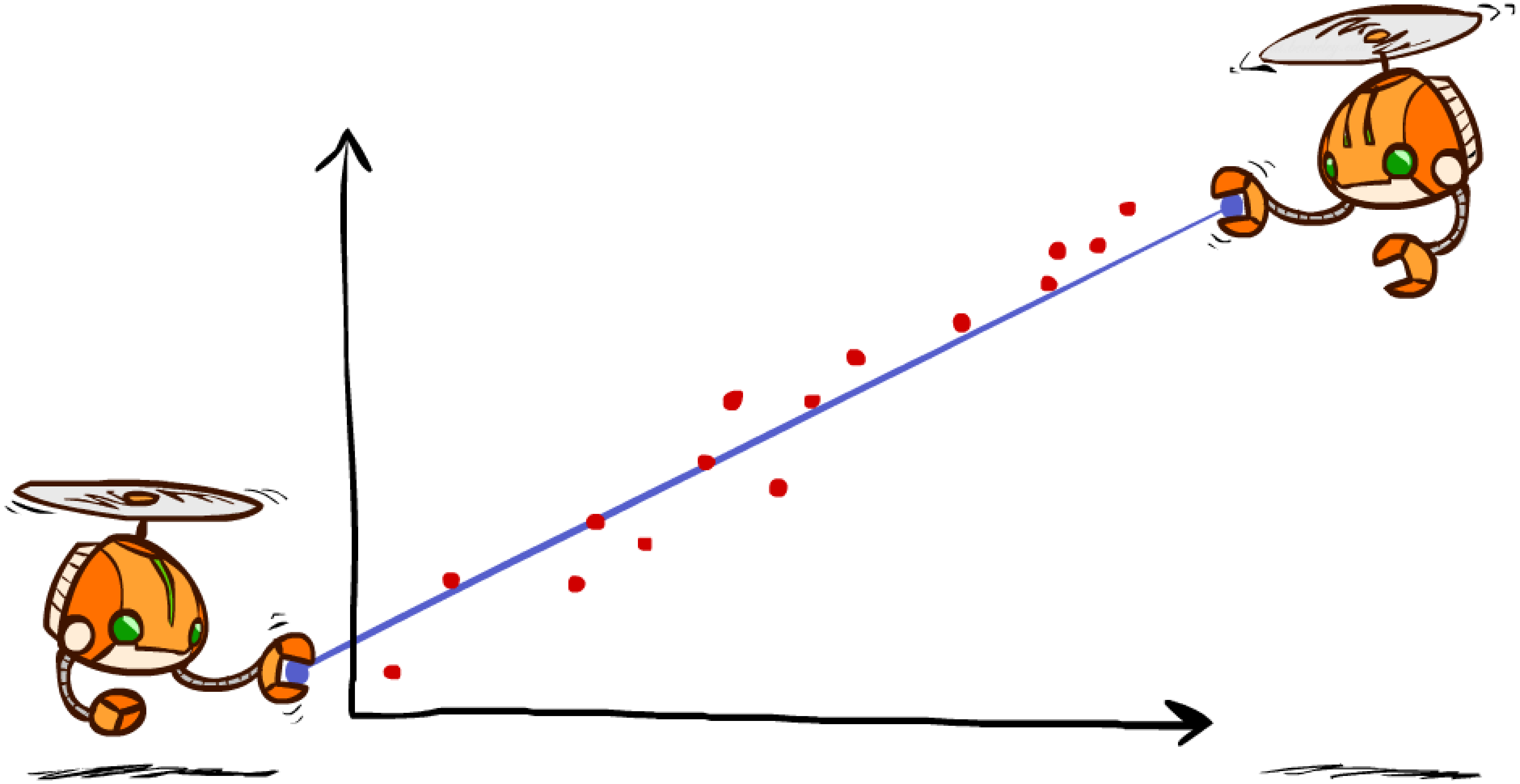
Lineáris modellek

- A bemenetek változó (jellemző, feature) értékek
- Minden változónak van egy súlya
- Az összegzett eredmény (lineáris kombináció) a kimenet

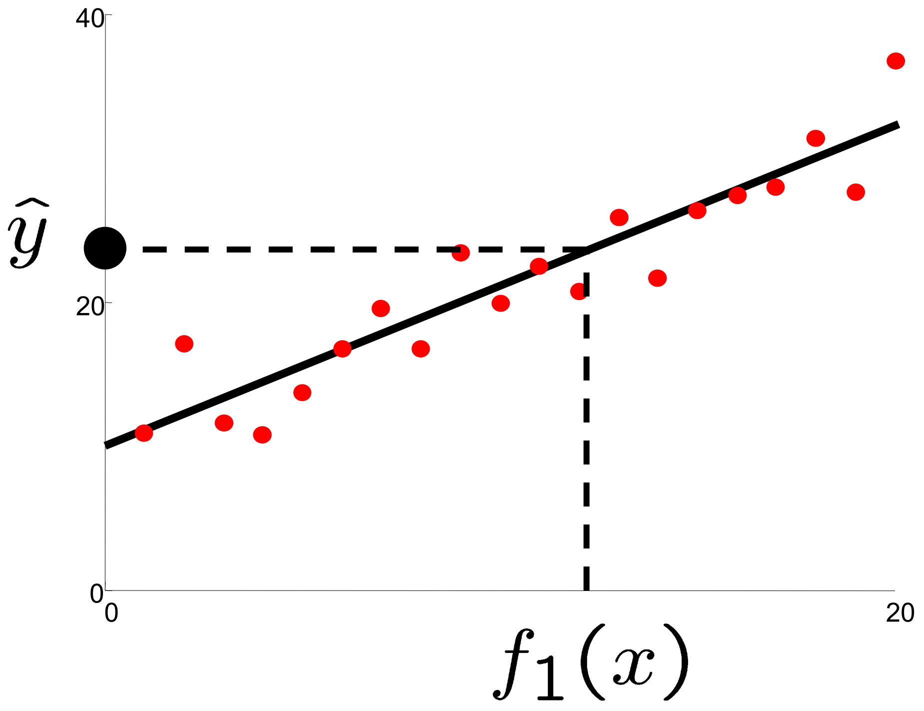
$$y = \sum_i w_i \cdot f_i(x) = w \cdot f(x)$$

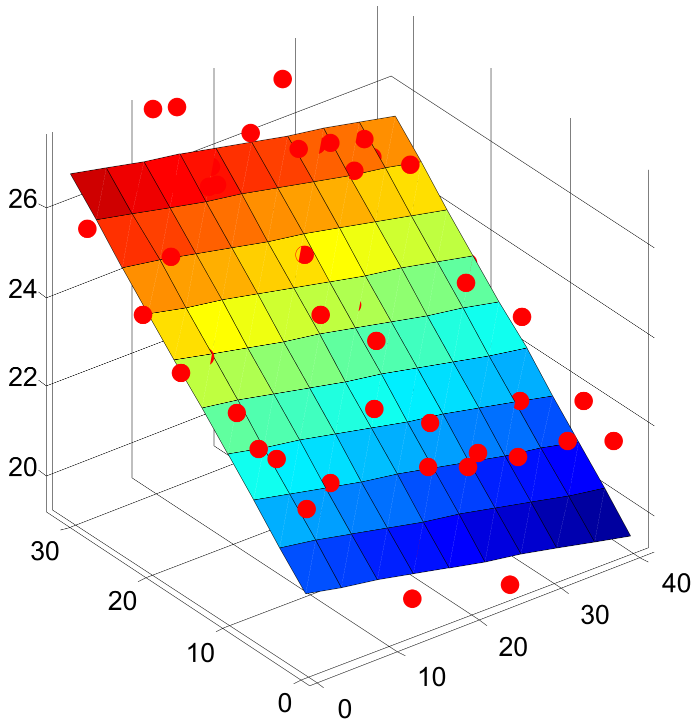


Lineáris regresszió



Lineáris közelítés: regresszió





Predikció:

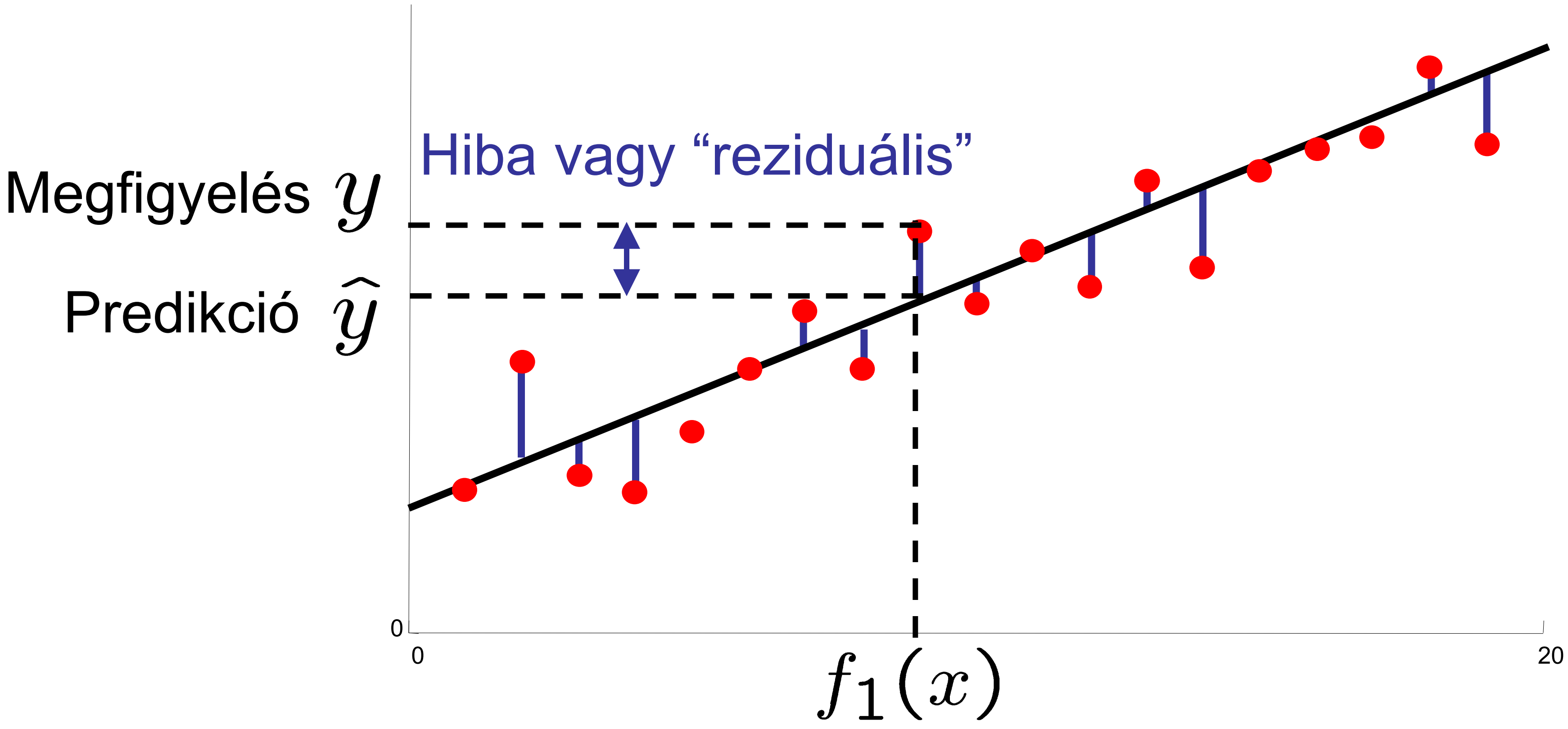
$$\hat{y} = w_0 + w_1 f_1(x)$$

Predikció:

$$\hat{y}_i = w_0 + w_1 f_1(x) + w_2 f_2(x)$$

Optimalizálás: Legkisebb négyzetek (Least Squares)

$$\text{total error} = \sum_i (y_i - \hat{y}_i)^2 = \sum_i \left(y_i - \sum_k w_k f_k(x_i) \right)^2$$



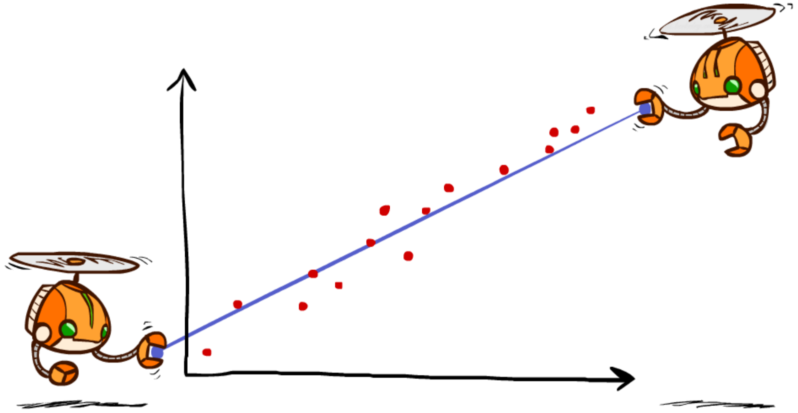
A hiba minimalizálása

Képzeljük el, hogy csak egy x pontunk van, f(x) jellemzőkkel, y célértékkel és w súlyokkal:

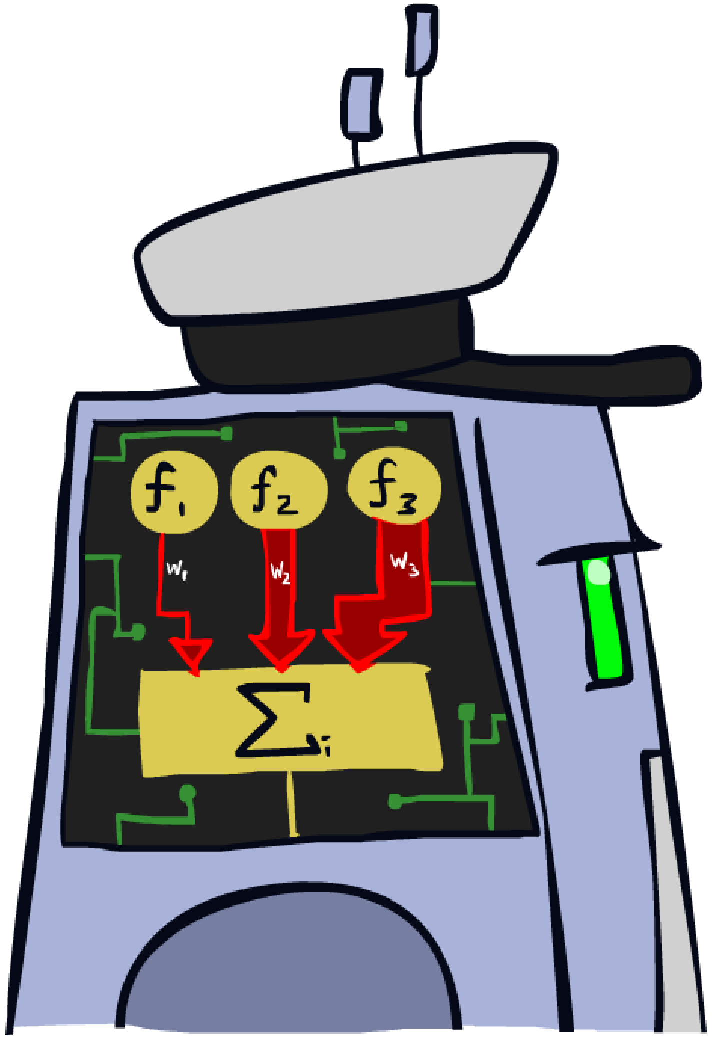
$$\text{error}(w) = \frac{1}{2}\left(y - \sum_k w_k f_k(x)\right)^2$$

$$\frac{\partial\ \text{error}(w)}{\partial w_m} = -\left(y - \sum_k w_k f_k(x)\right) f_m(x)$$

$$w_m \leftarrow w_m + \alpha \left(y - \sum_k w_k f_k(x)\right) f_m(x)$$



Lineáris osztályozók



Jellemző vektorok

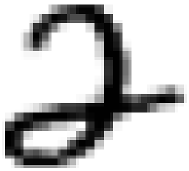
x → $f(x)$ → y

```
Hello,

Do you want free printr
cartriges?  Why pay more
when you can get them
ABSOLUTELY FREE!  Just
```

```
# free       : 2
YOUR_NAME    : 0
MISSPELLED   : 2
FROM_FRIEND  : 0
...
```

SPAM
or
+

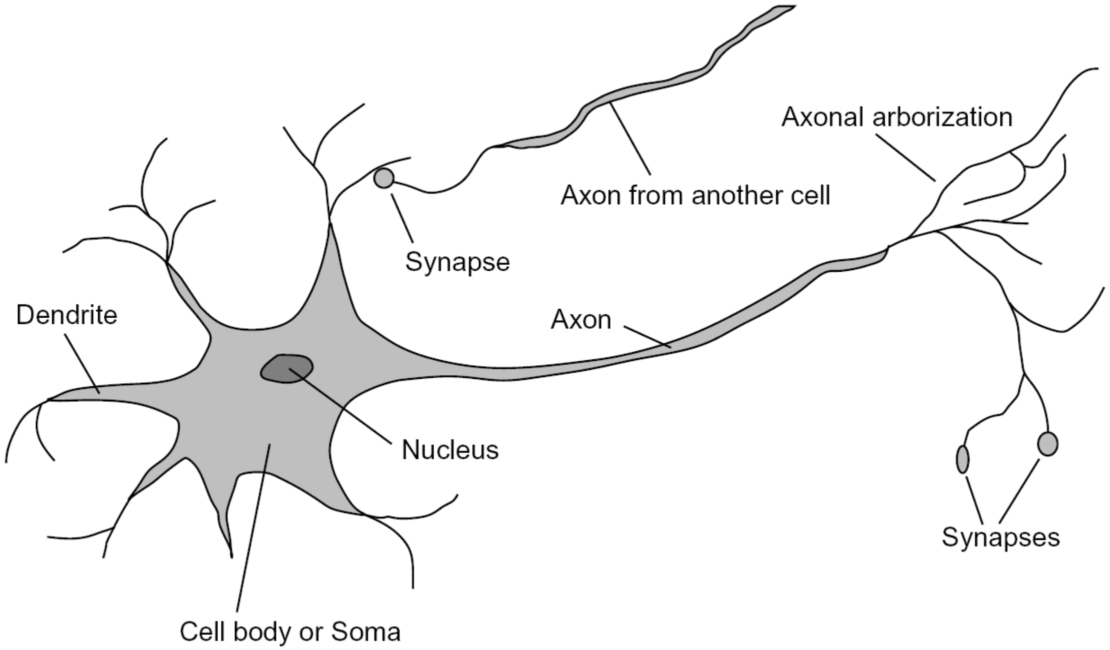


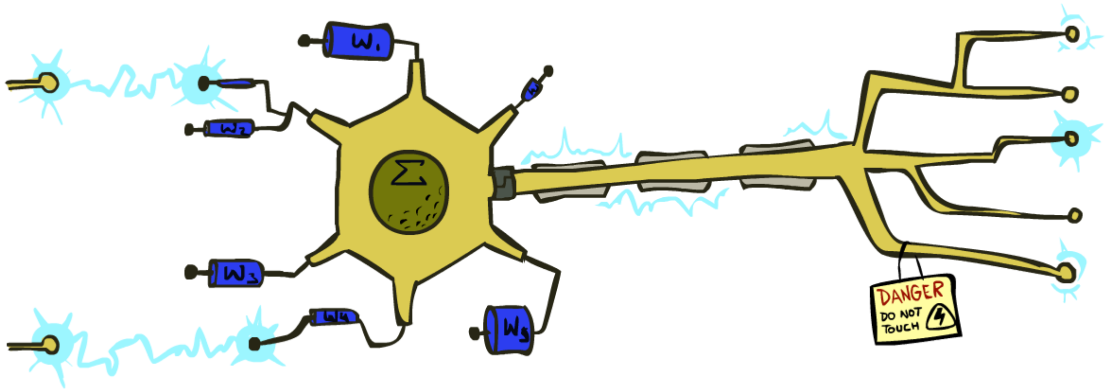
```
PIXEL-7,12   : 1
PIXEL-7,13   : 0
...
NUM_LOOPS    : 1
...
```

“2”

Némi (egyszerűsített) biológia

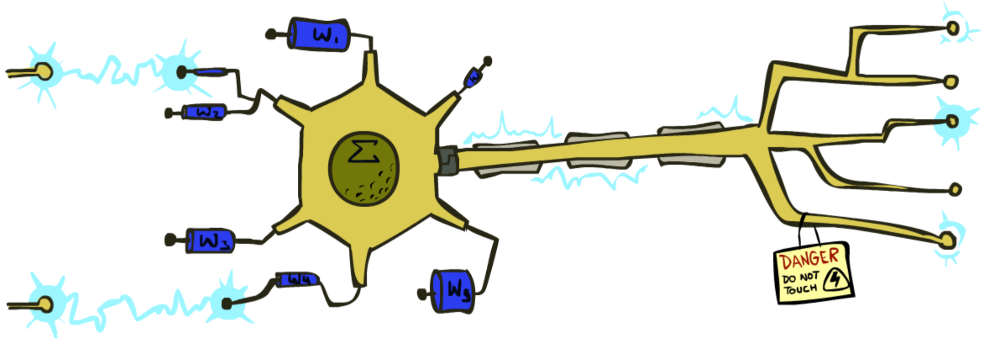
- inspiráció: emberi neuronok





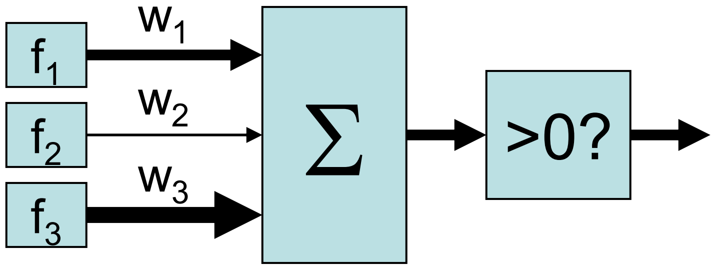
Lineáris osztályozók

- A bemenetek változó (jellemző, feature) értékek
- Minden változónak van egy súlya
- Az összegzett eredmény (lineáris kombináció) az aktiváció



$$\text{activation}_w(x) = \sum_i w_i \cdot f_i(x) = w \cdot f(x)$$

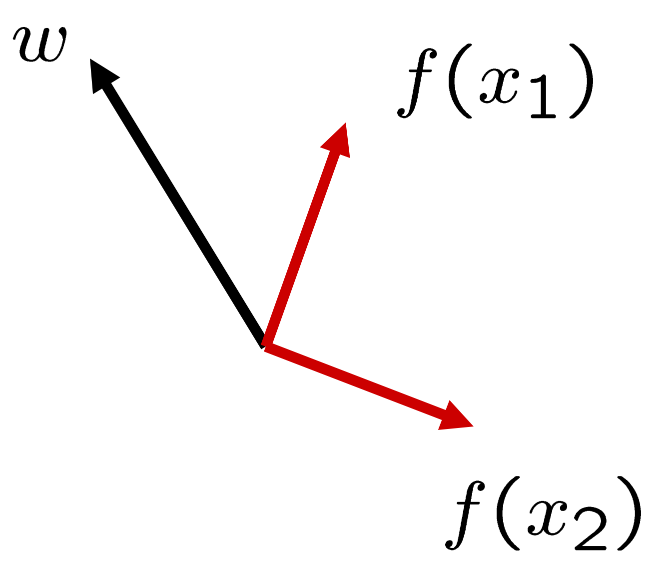
- Kimenet az aktiváció szerint:
  - Pozitív, output: +1
  - Negatív, output: -1



Súlyok

- Bináris eset: hasonlítsuk össze a jellemzőket egy súlyvektorral
- Tanulás: súlyvektor meghatározása példákból

```
# free       : 4
YOUR_NAME    :-1
MISSPELLED   : 1
FROM_FRIEND  :-3
...
```



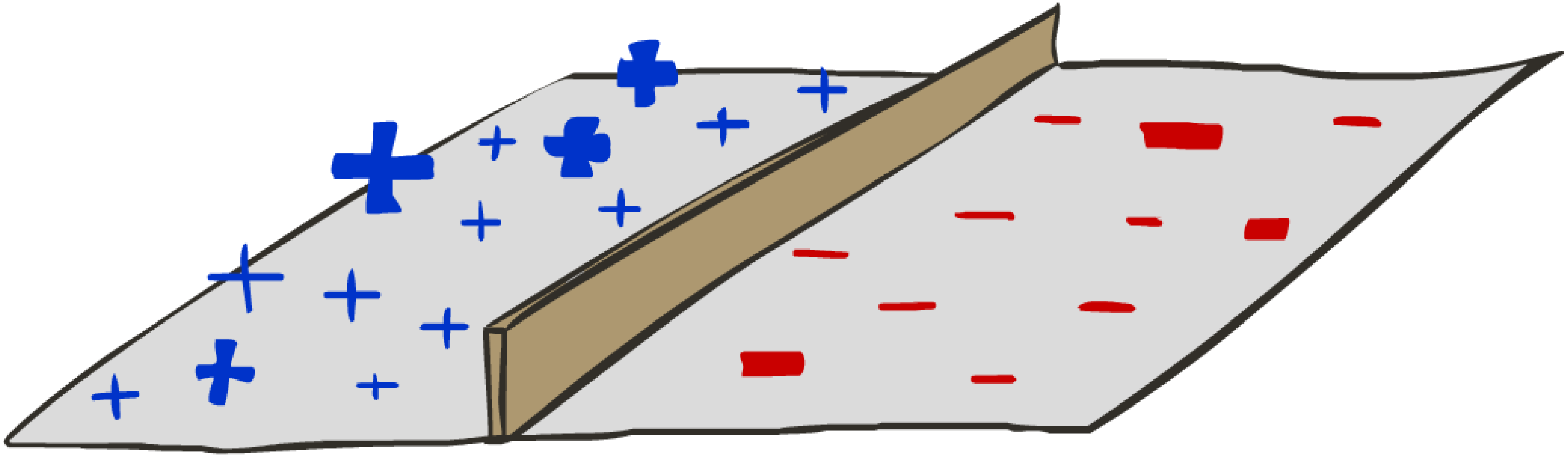

```
# free       : 2
YOUR_NAME    : 0
MISSPELLED   : 2
FROM_FRIEND  : 0
...
```

```
# free       : 0
YOUR_NAME    : 1
MISSPELLED   : 1
FROM_FRIEND  : 1
...
```

$w \cdot f$

Skalár szorzat pozitív, akkor pozitív osztályú besorolást jelent

Döntési szabályok

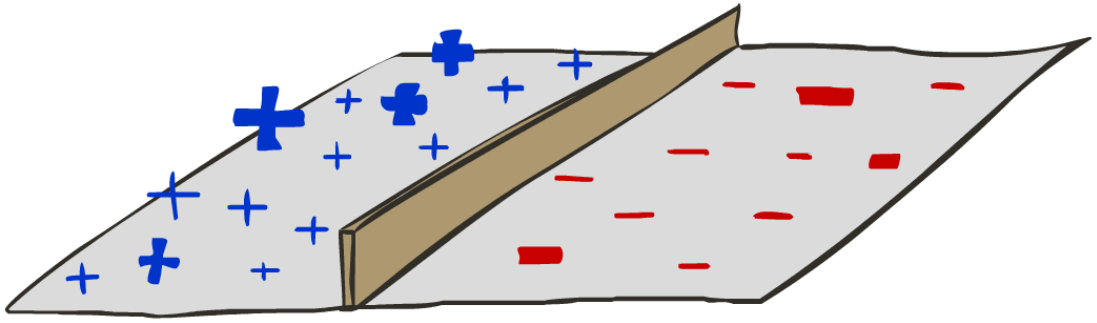


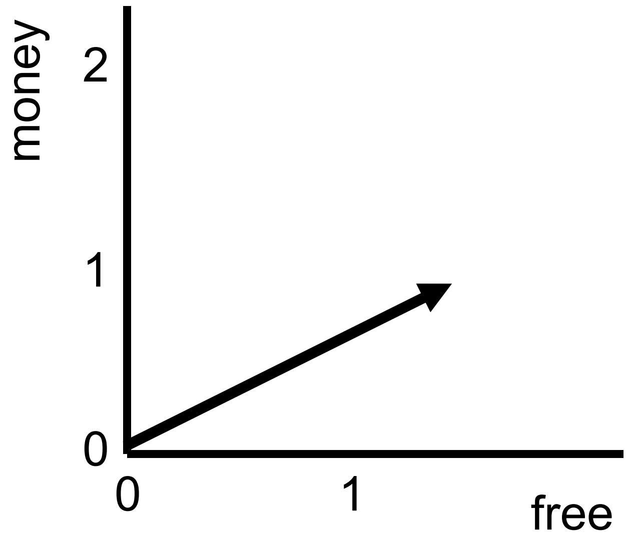
Bináris döntési szabály

- A jellemzővektorok terében
  - A minták a pontok
  - Bármely súlyvektor egy hipersík
  - Az egyik oldal megfelel Y=+1 -nek
  - A másik megfelel Y=-1 -nek

$$w$$

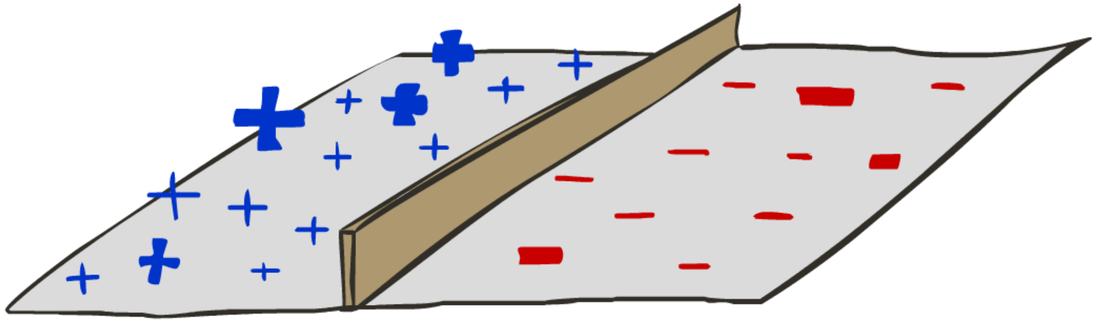
```
BIAS  : -3
free  :  4
money :  2
...
```





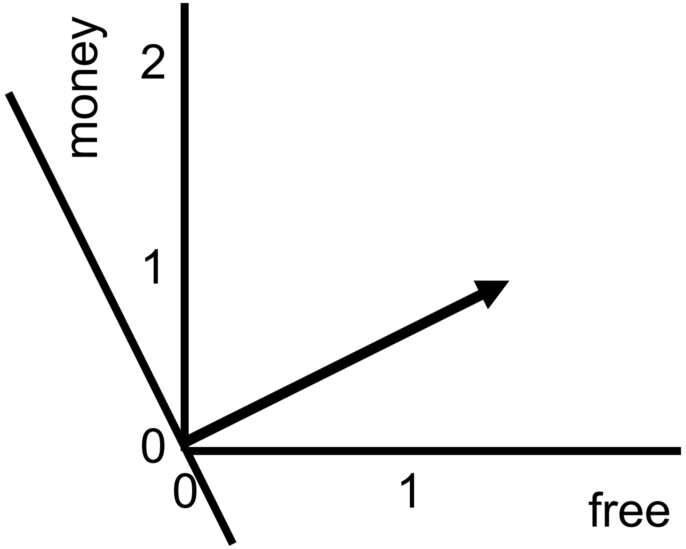
Bináris döntési szabály

- A jellemzővektorok terében
  - A minták a pontok
  - Bármely súlyvektor egy hipersík
  - Az egyik oldal megfelel Y=+1 -nek
  - A másik megfelel Y=-1 -nek



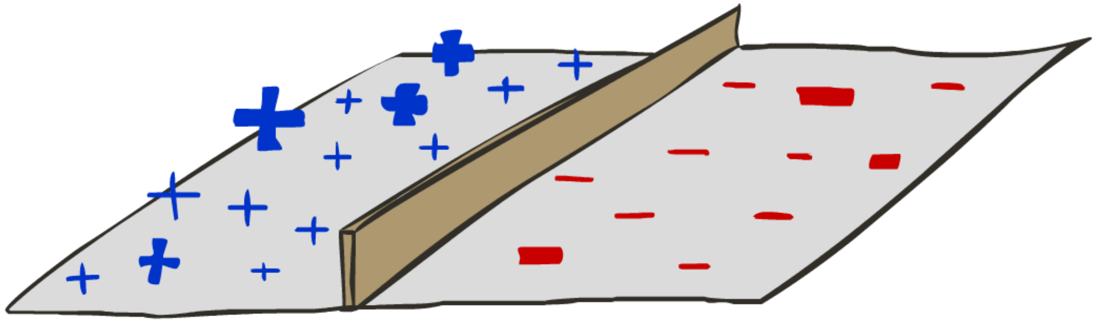
$$w$$

```
BIAS  : -3
free  :  4
money :  2
...
```



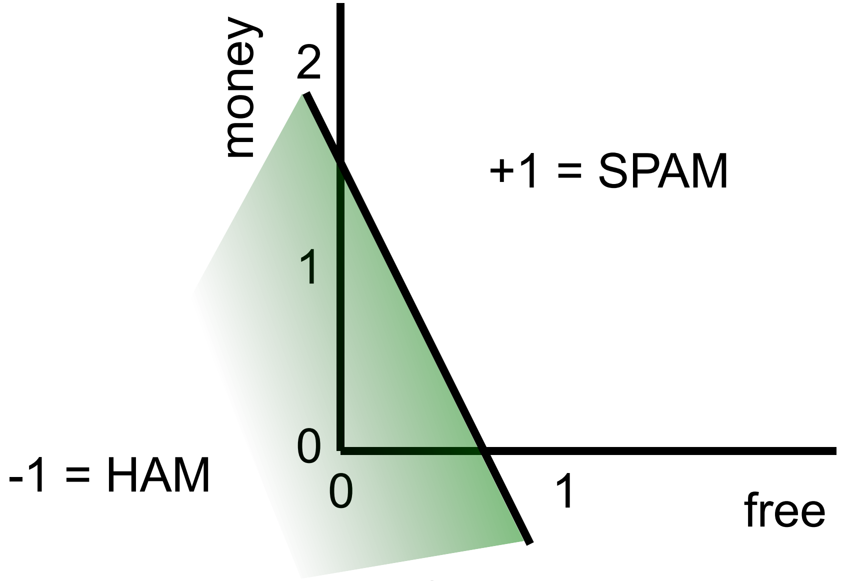
Bináris döntési szabály

- A jellemzővektorok terében
  - A minták a pontok
  - Bármely súlyvektor egy hipersík
  - Az egyik oldal megfelel Y=+1 -nek
  - A másik megfelel Y=-1 -nek



w

```
BIAS  : -3
free  :  4
money :  2
...
```

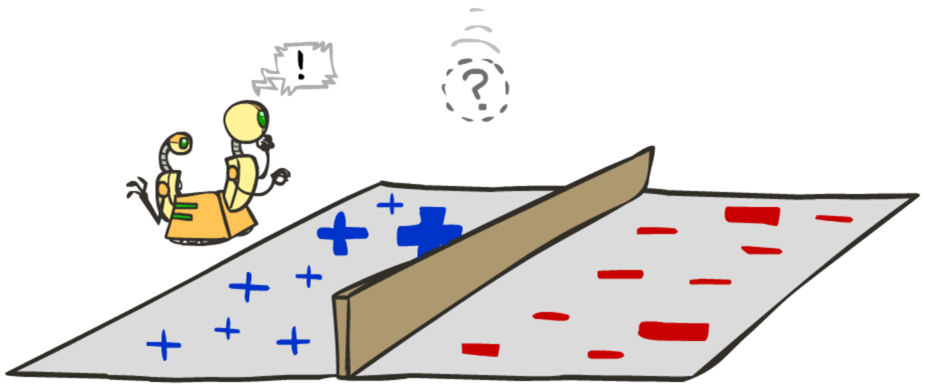


$f \cdot w = 0$

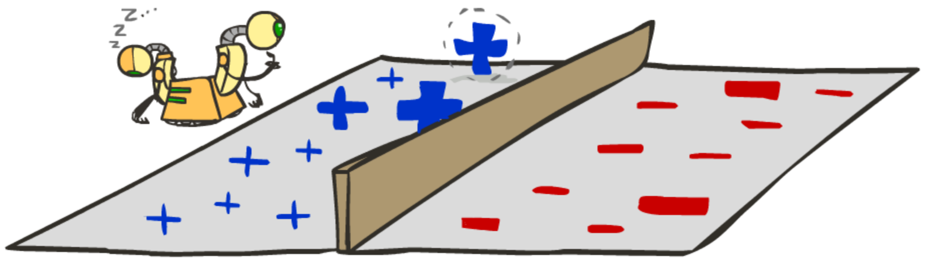
Súly frissítések

Tanulás: Bináris Perceptron

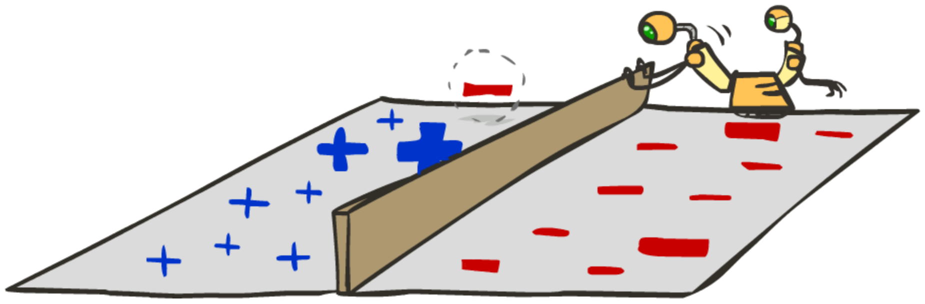
- Kezdetben: Súlyok = 0
- Minden tanítási mintához:
  - Osztályozás az aktuális súlyok alapján



  - Ha helyes (azaz y=y*), nincs változás!



  - Ha rossz: állítsa be a súlyvektort



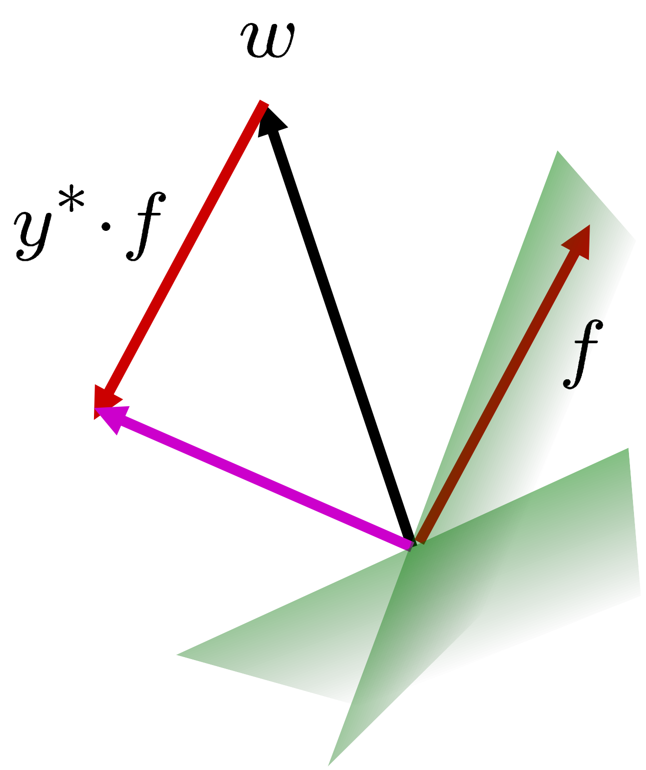
Tanulás: Bináris Perceptron

- Kezdetben: Súlyok = 0
- Minden tanítási mintához:
  - Osztályozás az aktuális súlyok alapján

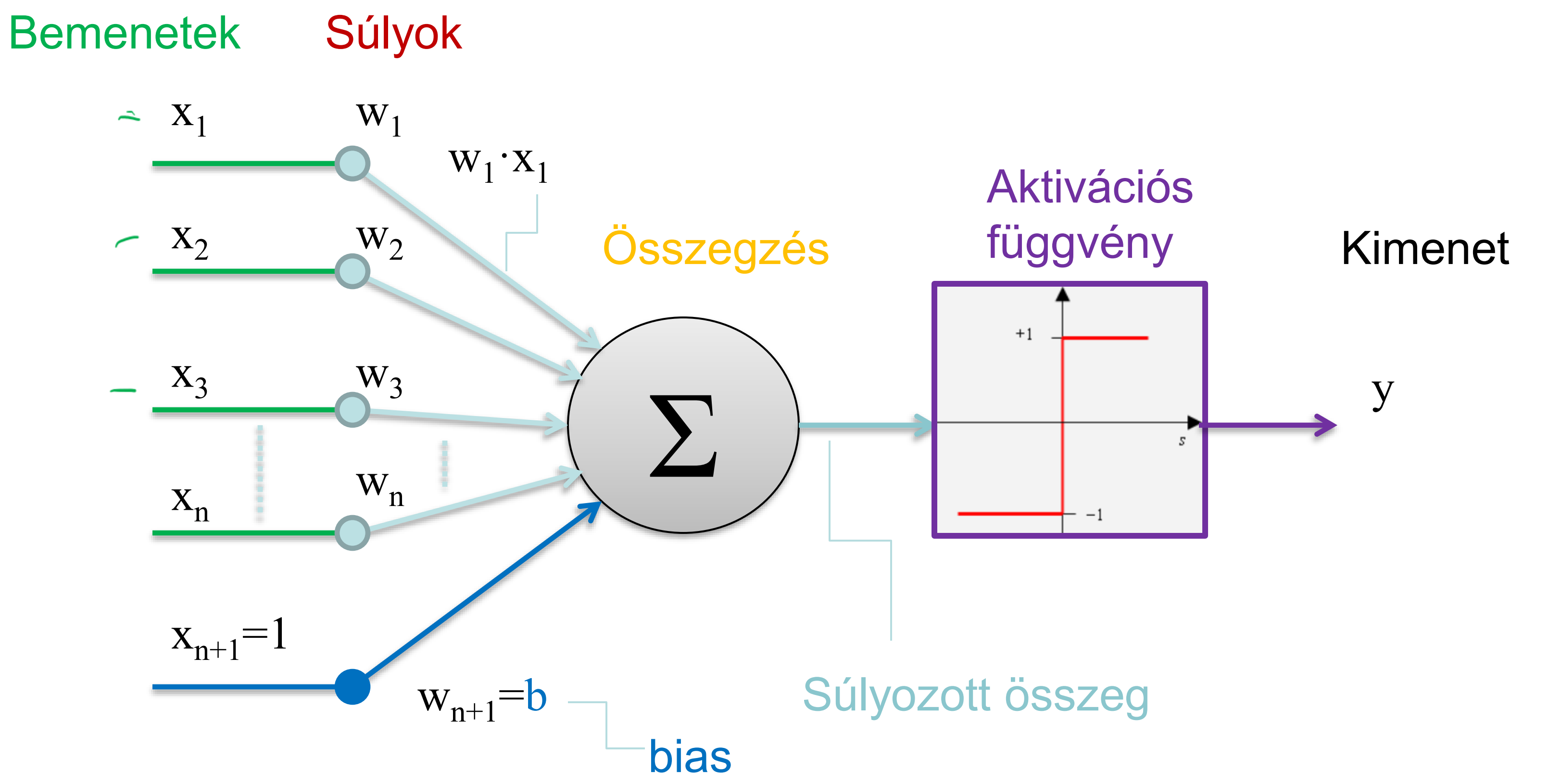
$$y = \begin{cases} +1 & \text{if } \; w \cdot f(x) \geq 0 \\ -1 & \text{if } \; w \cdot f(x) < 0 \end{cases}$$

  - Ha helyes (azaz y=y*), nincs változás!
  - Ha helytelen: súlyvektor módosítása a jellemzővektor hozzáadásával vagy kivonásával. Kivonás, ha y* = -1.

$$w = w + y^* \cdot f$$



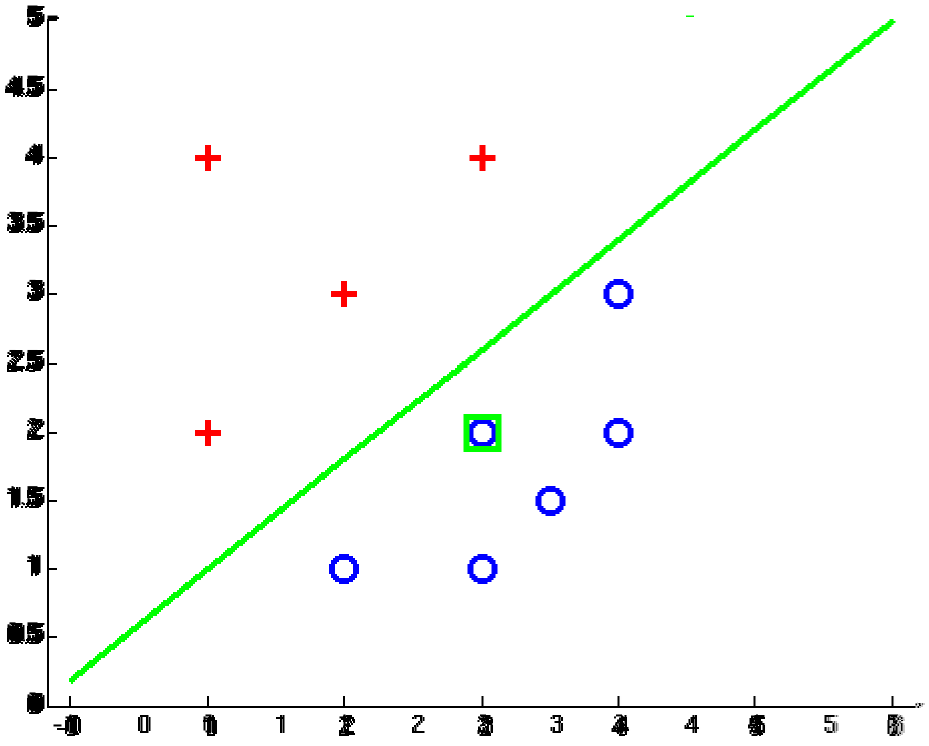
Perceptron



$$y = \mathrm{sgn}\left(\sum_{i=1}^{n+1} w_i\ x_i\right) = \mathrm{sgn}\left(\sum_{i=1}^{n} w_i\ x_i + w_{n+1}\right) = \mathrm{sgn}(\mathbf{x}^T\mathbf{w} + b)$$

Példák: Perceptron

- Elkülöníthető eset



Többosztályos döntési szabály

- Ha több osztályunk van:
  - Minden osztály súlyvektora:

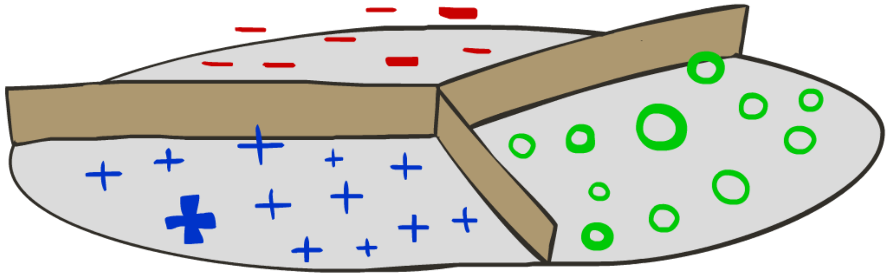
    $$w_y$$

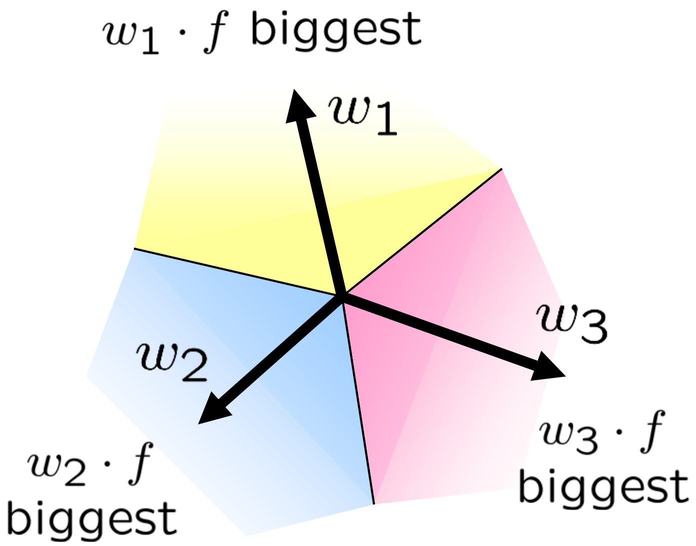
  - Az y osztály (aktivációja):

    $$w_y \cdot f(x)$$

  - Azaz előrejelzés nyer, amelyiknek legmagasabb az aktivációja

    $$y = \arg\max_y \; w_y \cdot f(x)$$



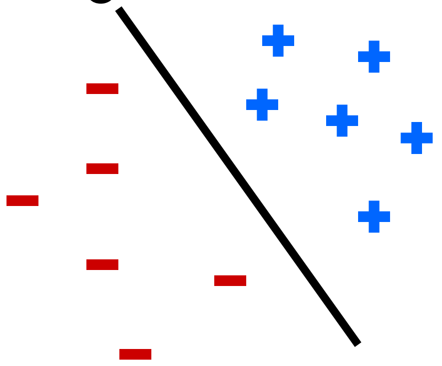


A perceptronok tulajdonságai

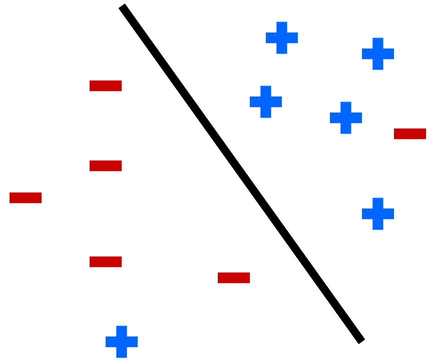
- **Elválaszthatóság:** igaz, ha egyes paraméterek alapján a modell tökéletesen szeparálja a mintákat

- **Konvergencia**: ha a tanító minta elválasztható, a perceptron végül konvergál (bináris esetben)

Elválaszthat
ó

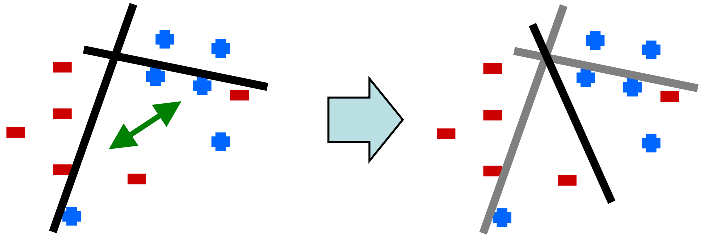


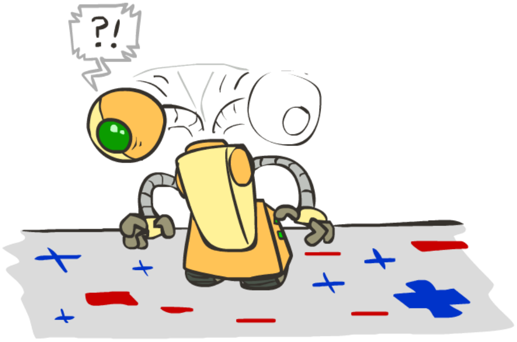
Nem elválasztható



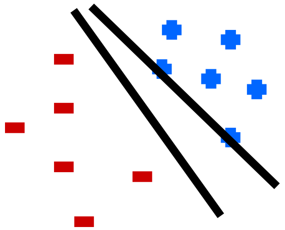
Problémák a Perceptronnal

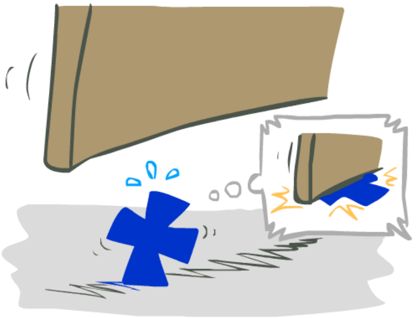
- Zaj: ha az adatok nem különíthetők el, a súlyok „kidőlhetnek”



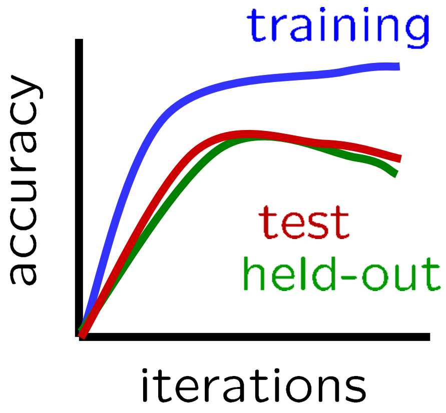


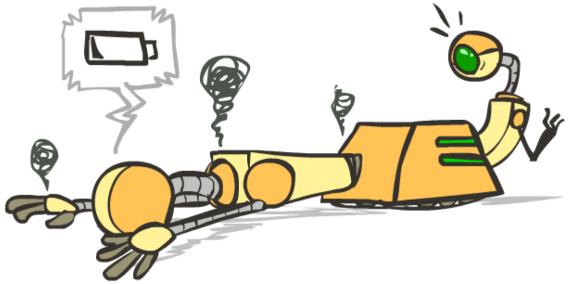
- Középszerű általánosítás: "alig" elválasztó megoldást talál





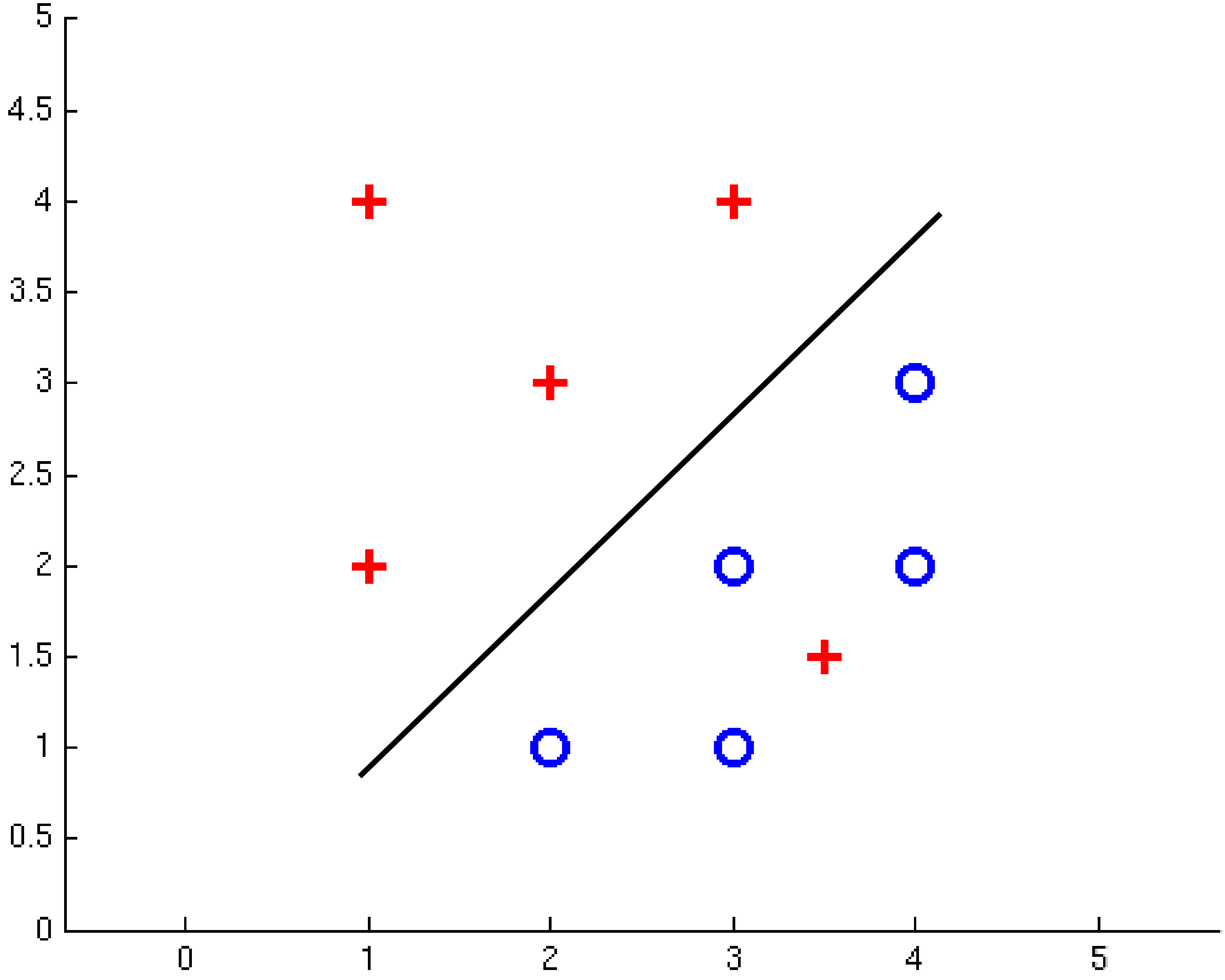
- Túltanulás: a teszt / validációs adathalmazon a modell pontossága általában emelkedik, majd csökken
- A túltanulás egyfajta túlilleszkedés



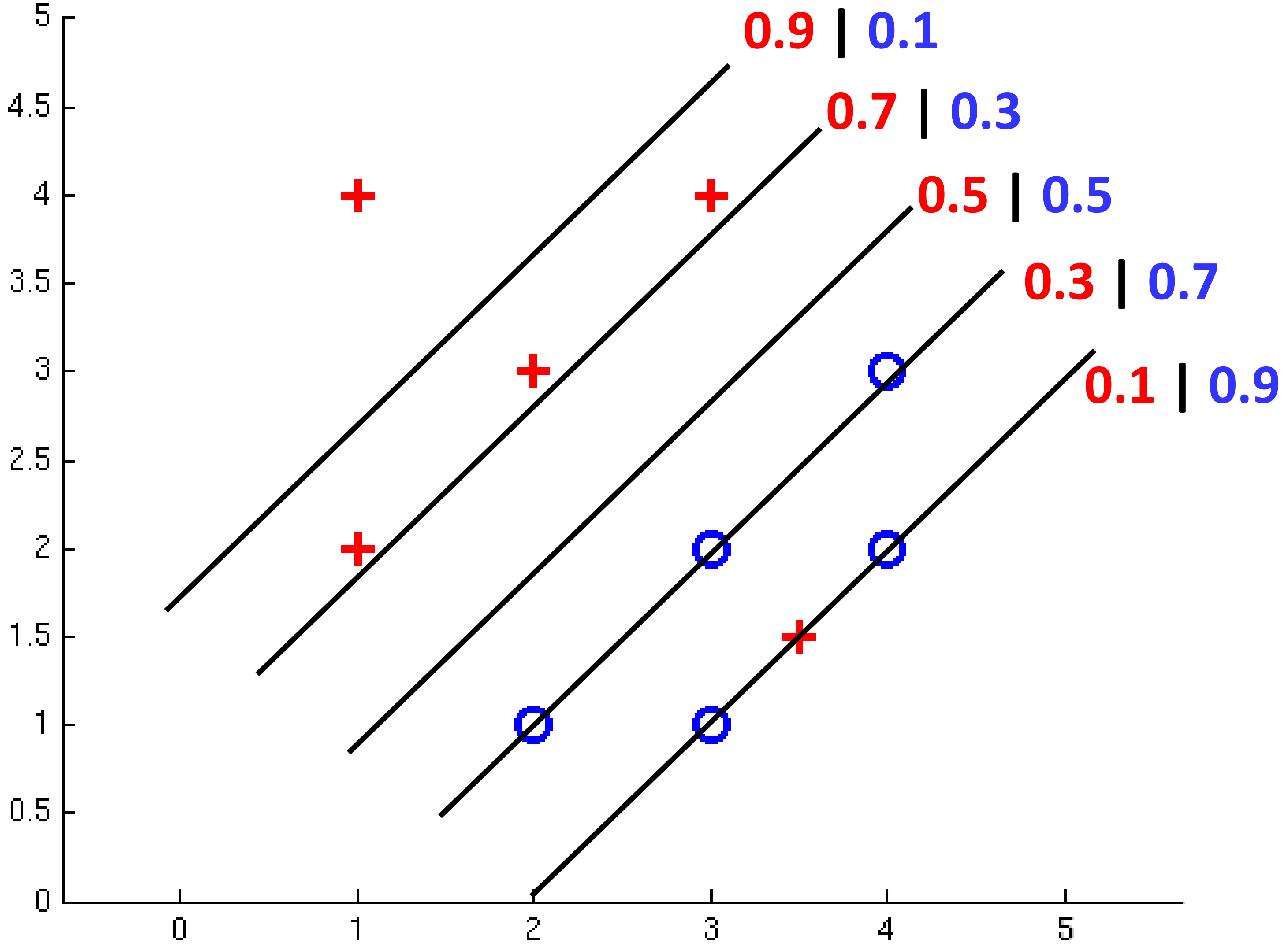


Nem elválasztható eset: determinisztikus döntés

Még a legjobb lineáris határ is elkövet legalább egy hibát



Nem szétválasztható eset: valószínűségi döntés

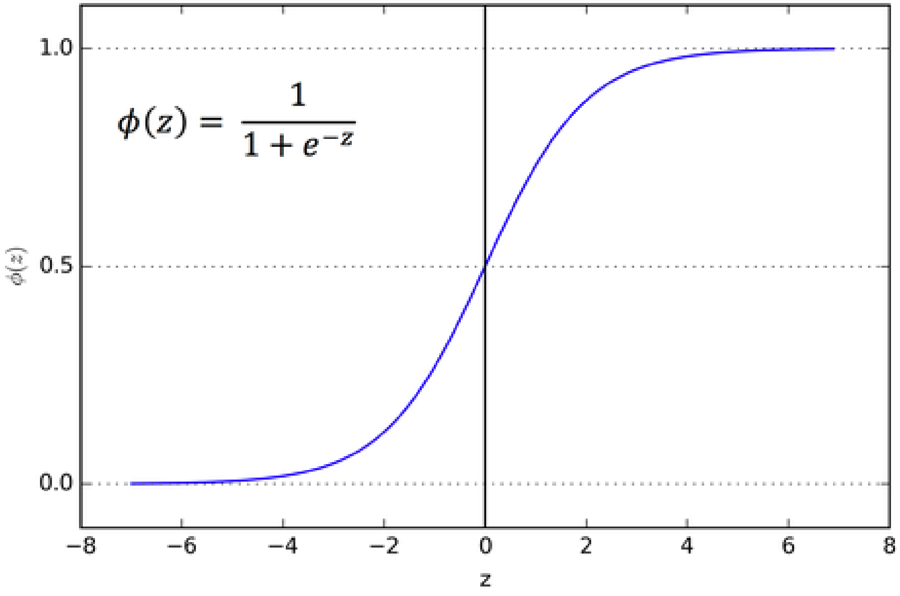


Hogyan lehet valószínűségi döntéseket hozni?

- Aktiváció: $z = w \cdot f(x)$
- Ha $z = w \cdot f(x)$ Pozitív → a valószínűség 1-re közelítsen
- Ha $z = w \cdot f(x)$ Negatív → a valószínűség 0-ra közelítsen

- Szigmoid függvény

$$\phi(z) = \frac{1}{1 + e^{-z}}$$



Legjobb w?

- Maximum likelihood becslés:

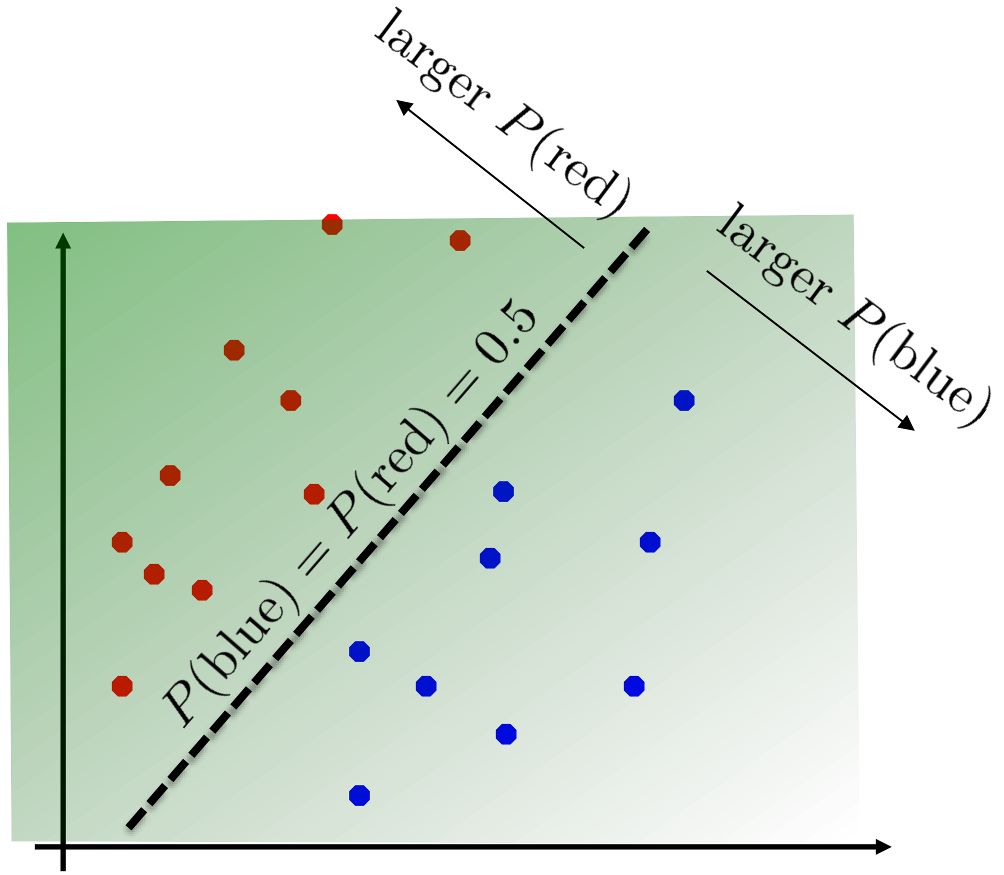
$$\max_{w} \quad ll(w) = \max_{w} \quad \sum_{i} \log P(y^{(i)}|x^{(i)}; w)$$

$$P(y^{(i)} = +1|x^{(i)}; w) = \frac{1}{1 + e^{-w \cdot f(x^{(i)})}}$$

$$P(y^{(i)} = -1|x^{(i)}; w) = 1 - \frac{1}{1 + e^{-w \cdot f(x^{(i)})}}$$

= Logisztikus regresszió

Valószínűségi perceptron



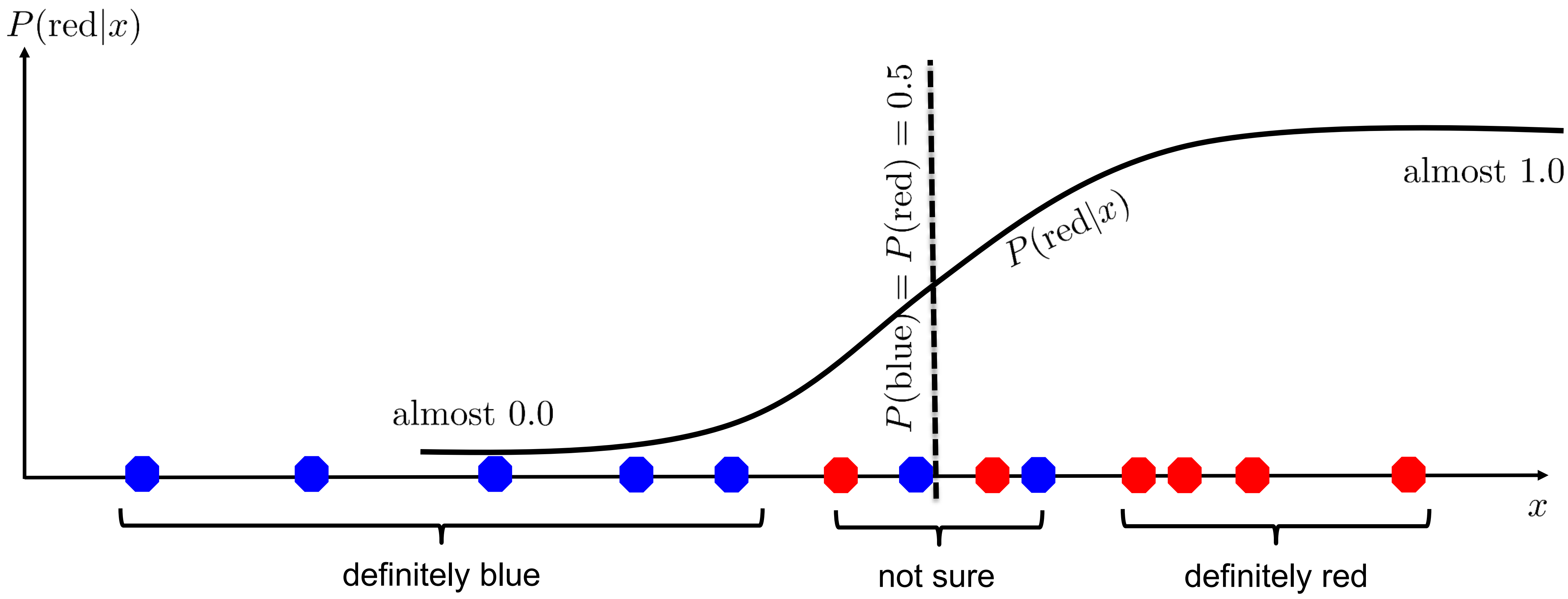
As $w_y \cdot x$ gets bigger, $P(y|x)$ gets bigger

1D Példa

$$P(\text{red}|x) = \frac{e^{w_{\text{red}} \cdot x}}{e^{w_{\text{red}} \cdot x} + e^{w_{\text{blue}} \cdot x}}$$

probability increases exponentially as we move away from boundary

normalizer



Soft Max

$$P(\text{red}|x) = \frac{e^{w_{\text{red}} \cdot x}}{e^{w_{\text{red}} \cdot x} + e^{w_{\text{blue}} \cdot x}}$$