

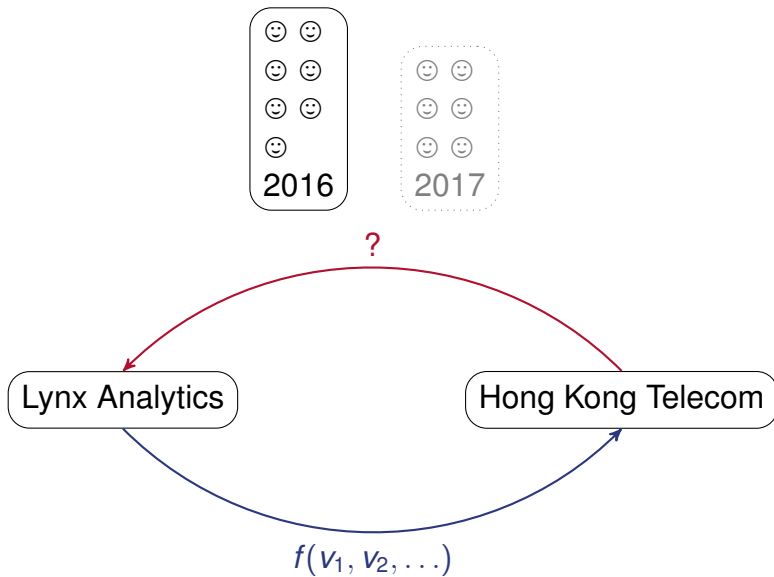
# Egy adatelemző projekt

a kérdésfeltevéstől az automatizálásig

Erben Péter


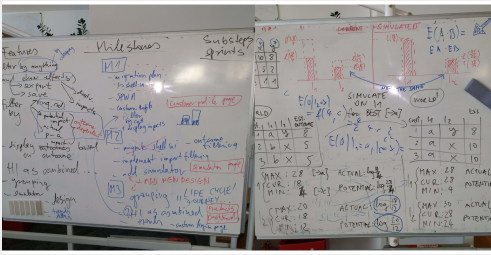
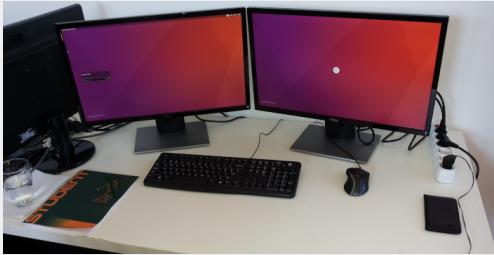
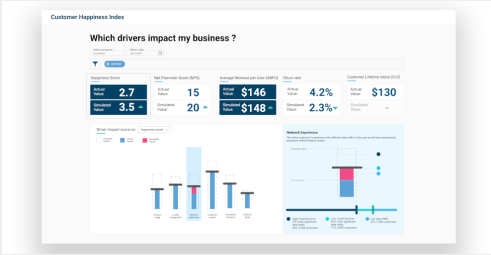
A Gondolkodás Öröme Alapítvány: Beszéljük meg!  
2023. június 26.

# A kérdés

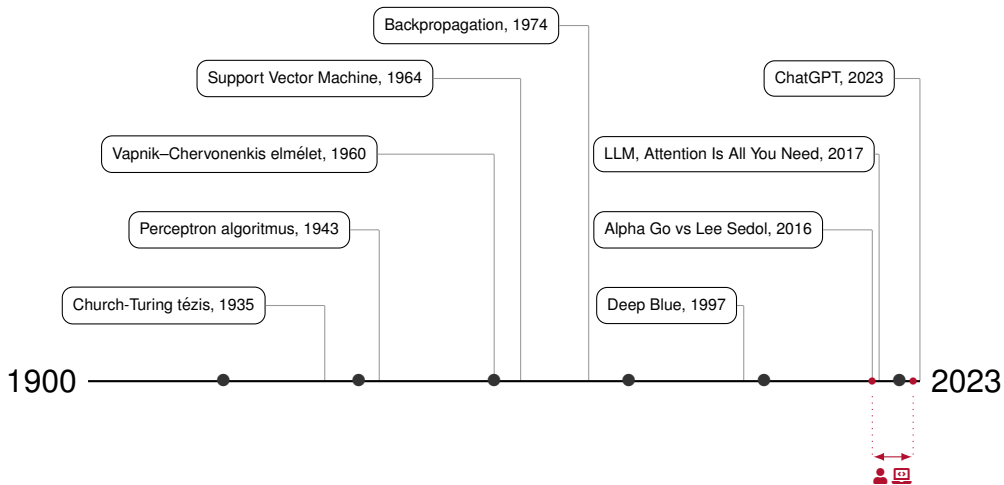


**CASE STUDY:**  
Your next 5G site roll out?  
Let your customers decide:  
How HKT did it.

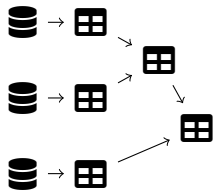
DOWNLOAD NOW >

# AI történet



adat futószalag



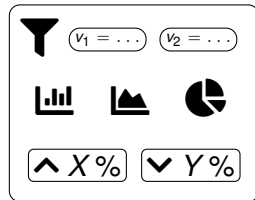
gépi tanulás

$v_1$	$v_2$	...	$t$
.	.	...	.
.	.	...	.
.	.	...	.



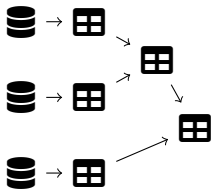
$$f(v_1, v_2, \dots) \approx t$$

a modell használata



⌚ automatizáció, 🛒 terméké alakítás

# Adat futószalag



változók definíciója

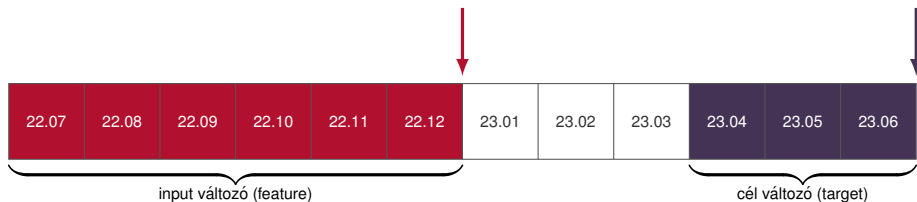
adat transzformációk, adat tisztítás

körmentes irányított gráfok

# Változók definíciója

„Hány ügyfele van a cégnek 2023. június 26-án?”

„Meddig kell visszamenni időben, ha fél évre akarunk előrejelzést adni?”



# Adat transzformációk

Diszkrétizálás, hiányzó adatok

#	életkor		#	életkor
1	24	→	1	'20-30'
2	63		2	'60-70'
3	44		3	'40-50'
4	49		4	'40-50'
5	-		5	'NA'

Optimális vágások?

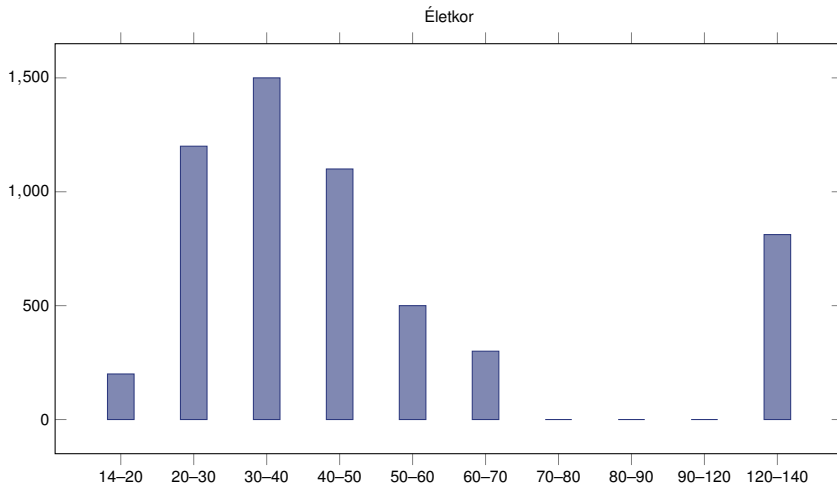
Kategorikusból valós

#	gen		#	3G	4G	5G	NA
1	'4G'	→	1	0	1	0	0
2	'4G'		2	0	1	0	0
3	'5G'		3	0	0	1	0
4	'3G'		4	1	0	0	0
5	'NA'		5	0	0	0	1

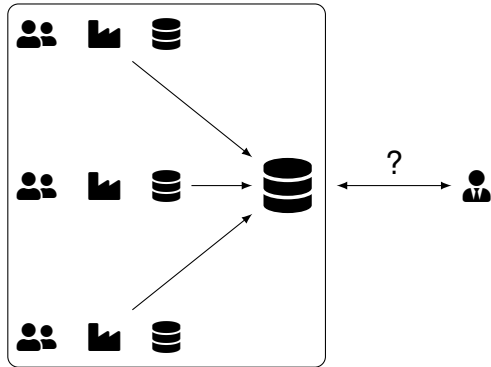
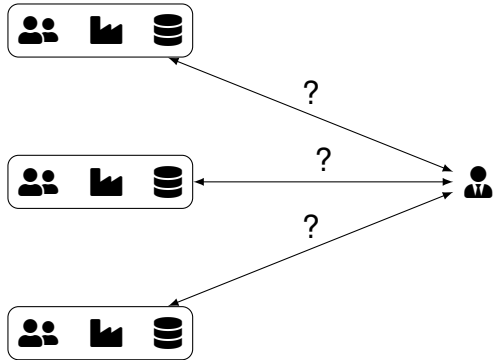
Beágyazás magasabb dimenziós térbe



# Adat tisztítás

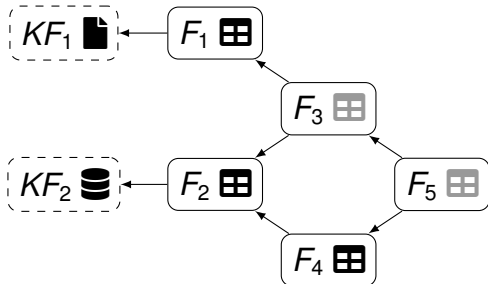


# Adat silók



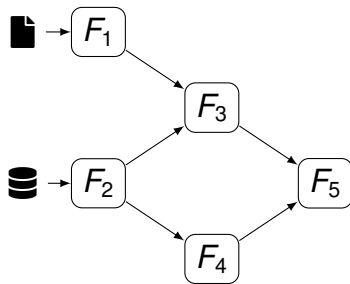
# Írányított körmentes gráfok (DAGs)

## Luigi



A feladat tudja, kitől függ.

## Apache Airflow



Az ütemező tudja, ki következhet.

# Gépi tanulás

$v_1$	$v_2$	...	$t$
.	.	...	.
.	.	...	.
.	.	...	.



$$f(v_1, v_2, \dots) \approx t$$

mesterséges intelligencia

gépi tanulás

(görbeillesztés, statisztikai tanulás)

változók kiválasztása

# Mesterséges intelligencia (Artificial Intelligence)

⚙️ stockfish

## Gépi tanulás (Machine Learning)

### Felügyelt (**supervised**) tanulás

regresszió / klasszifikáció

- ⚙️ lineáris regresszió
- ⚙️ logisztikus regresszió
- ⚙️ support vector machine
- ⚙️ döntési fa
- ⚙️ xgboost

### Felügyelet nélküli (**unsupervised**) tanulás

- ⚙️ principal component analysis
- ⚙️ k-means clustering

### Megerősítéses (**reinforcement**) tanulás

- ⚙️ Q-learning

## Neurális hálók (Deep learning)

⚙️ convolutional neural network

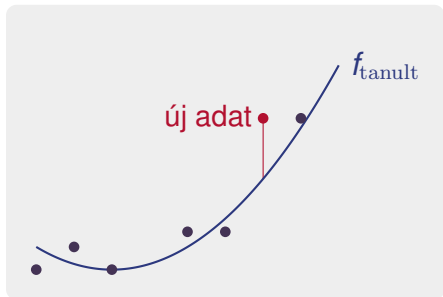
⚙️ autoencoders

⚙️ DeepMind

⚙️ large language models (LLMs)

⚙️ AlphaGo

# Gépi tanulás: görbe illesztés + túltanulás elkerülése



**Hiba:**  $\mathcal{L}$  (loss)

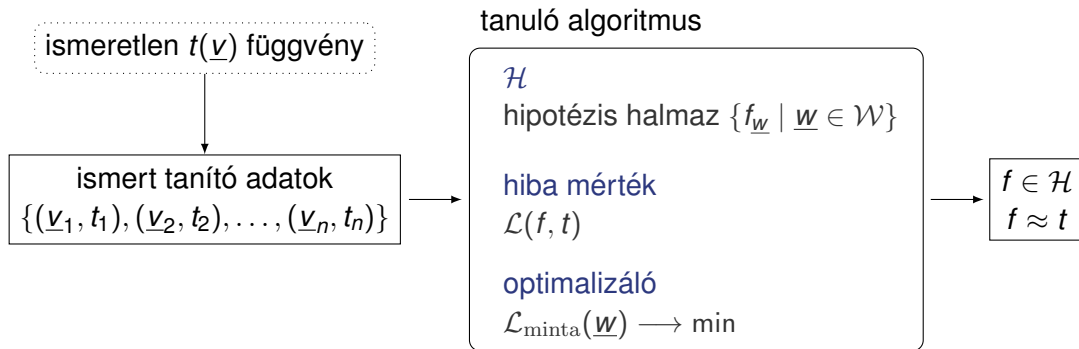
**Algoritmus:**  $\mathcal{L}_{\text{minta}} \rightarrow \min$

**Elmélet:**  $|\mathcal{L}_{\text{minta}} - \mathcal{L}| < \epsilon$ ,  
ha elég sok adaton „tanítottunk”, a  
függvény „bonyolultságához” képest

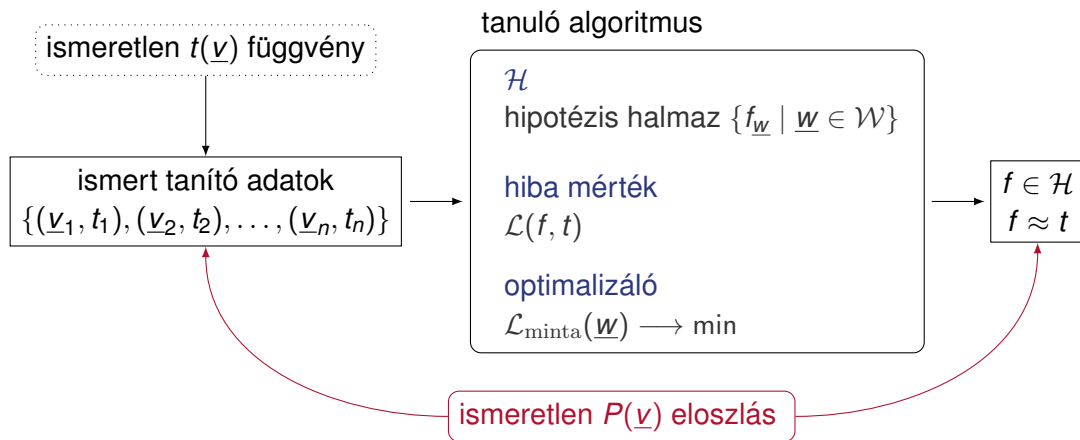
**Túltanulás (overfitting):**

$\mathcal{L}_{\text{minta}} < \epsilon$ , de  $|\mathcal{L}_{\text{minta}} - \mathcal{L}| \gg \delta$

# Felügyelt tanulás

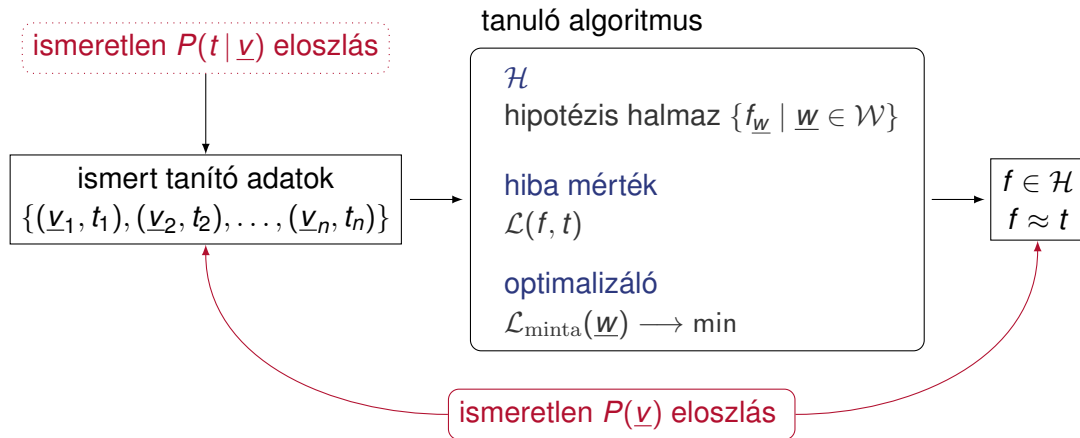


# Felügyelt tanulás





# Felügyelt tanulás



# Áttekintés

Mit tanulunk?

$$P(t \mid \underline{v})$$

Mit feltételezünk?

$$P(\underline{v})$$

Mit remélünk?

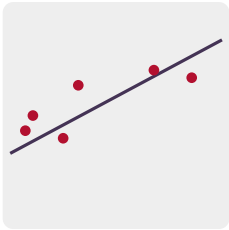
$P(t \mid \underline{v})$  nem változik (gyorsan), akkor sem, ha  $P(\underline{v})$  változik.

Melléktermék

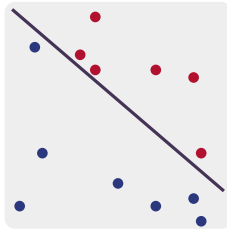
$$P(t, \underline{v}) = P(t \mid \underline{v}) \cdot P(\underline{v})$$

# Példák felügyelt tanulásra

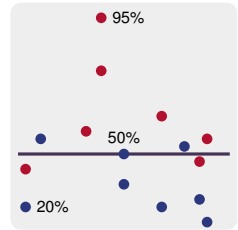
Lineáris regresszió



Perceptron



Logisztikus regresszió



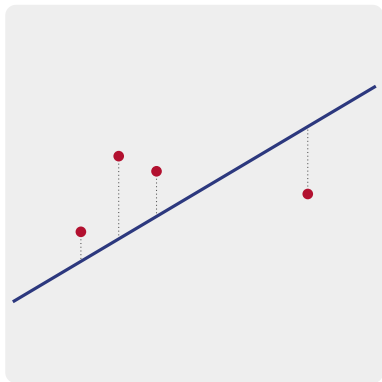
# ⚙️ Lineáris regresszió

$$\mathcal{H}: \{f(x) = mx + b \mid (m, b) \in \mathbb{R}^2\}$$

$$\mathcal{L}(f, t): \frac{1}{n} \sum_{i=1}^n (f(x_i) - t_i)^2$$

optimalizáló:

$\mathcal{L}$  konvex,  $\nabla \mathcal{L}_{m,b} = \underline{0}$





## Lineáris regresszió

$$\mathcal{L} = \frac{1}{n} \sum_{i=1}^n (f(x_i) - y_i)^2 \rightarrow \min$$

$$\begin{aligned}\mathcal{L}(m, b) &= \sum (f(x_i) - y_i)^2 = \sum (mx_i + b - y_i)^2 = \\ &= \sum (m^2 x_i^2 + 2mbx_i + b^2 + y_i^2 - 2y_i mx_i - 2y_i b) = \\ &= m^2 \sum x_i^2 + 2mb \sum x_i + nb^2 + \sum y_i^2 - 2m \sum x_i y_i - 2b \sum y_i\end{aligned}$$

$$\frac{\partial \mathcal{L}}{\partial m} = 2m \sum x_i^2 + 2b \sum x_i - 2 \sum x_i y_i = 0$$

$$\frac{\partial \mathcal{L}}{\partial b} = 2m \sum x_i + 2bn - 2 \sum y_i = 0$$



# Lineáris regresszió

$$m = \frac{n \sum x_i y_i - \sum x_i \cdot \sum y_i}{n \sum x_i^2 - (\sum x_i)^2}$$

$$b = \frac{\sum y_i - m \sum x_i}{n}$$



# ⚙️ Perceptron

$$\mathcal{H}: \{f(x, y) = \text{sgn}(Ax + By + C)\}$$

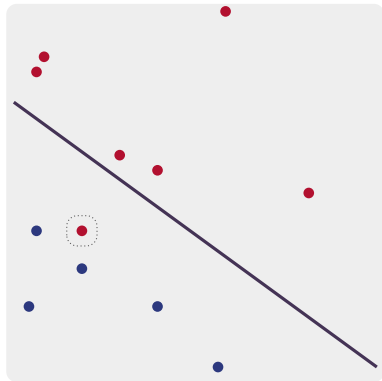
$$(A, B, C) \in \mathbb{R}^3, t_i \in \{+1, -1\}$$

$$\mathcal{L}(f, t): \#\{i \mid t_i \neq f(x_i, y_i)\}$$

optimalizáló:

Ciklus amíg  $\exists i : f(x_i, y_i) \neq t_i$ :

$$\begin{pmatrix} A \\ B \\ C \end{pmatrix} = \begin{pmatrix} A \\ B \\ C \end{pmatrix} + t_i \cdot \begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix}$$





## Perceptron (motiváció)

$\underline{w}(A, B, C)$  az origón átmenő,  $Ax + By + Cz = 0$  egyenletű  $S_B$  sík normálvektora.

Az ismert pontok a  $z = 1$  egyenletű  $S$  síkon helyezkednek el.  $(x_i, y_i, 1)$

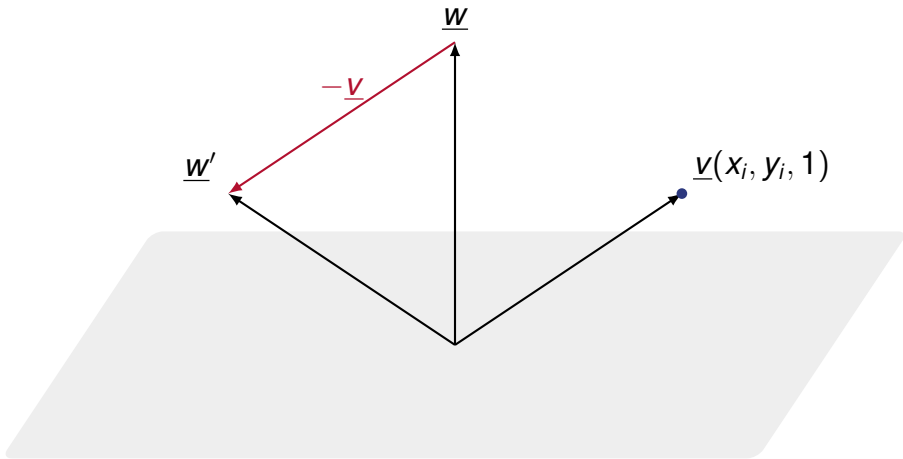
A döntés határvonala  $S_B$  és  $S$  metszészvonala.

Ha a  $\underline{v}(x_i, y_i, 1)$  pontra tévedtünk, akkor  $\underline{w} \cdot \underline{v}$  és  $t_i$  előjele különbözik (tfh.  $t_i = -1$ ).

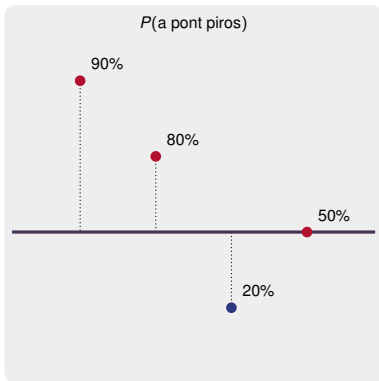
$\underline{w}' = \underline{w} - \underline{v}$  jó irányba forgatta az  $S_B$  síkot  $(x_i, y_i, 1)$  szempontjából.



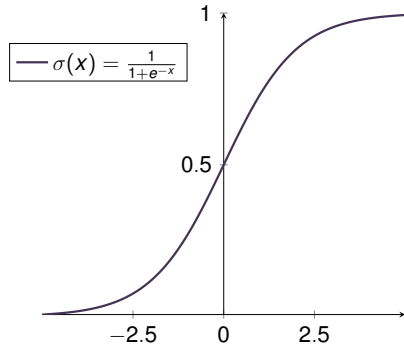
# Perceptron (motiváció)



# Logisztikus regresszió\* (motiváció)



$$u(x, y) = Ax + By + C \in ]-\infty; +\infty[$$



$$f(x, y) = \sigma(u(x, y)) \in ]0; +1[$$

\* Ez egy klasszifikáció. ☺

## Logisztikus regresszió (motiváció)

#	becslés	tény (0 v. 1)
1	$p_1 = f(\underline{v}_1)$	$t_1$
2	$p_2 = f(\underline{v}_2)$	$t_2$
3	$p_3 = f(\underline{v}_3)$	$t_3$
	$\vdots$	

$$P(t_i | f) = \begin{cases} p_i, & \text{ha } t_i = 1 \\ 1 - p_i, & \text{ha } t_i = 0 \end{cases} = \\ = p_i^{t_i} \cdot (1 - p_i)^{(1-t_i)}$$

$$P(\text{minta} | f) = \prod p_i^{t_i} \cdot (1 - p_i)^{(1-t_i)}$$

**Maximum Likelihood Estimate:**

$$P(\text{minta} | f) \rightarrow \max \quad -\ln P(\text{minta} | f) \rightarrow \min$$

$$\mathcal{L}(f, t) = -\ln P(\text{minta} | f) = -\sum (t_i \ln f(x_i, y_i) + (1 - t_i) \ln(1 - f(x_i, y_i)))$$

# ⚙️ Logisztikus regresszió

$$\mathcal{H}: \left\{ f(x, y) = \frac{1}{1 + e^{-(Ax + By + C)}} \right\}$$

$$(A, B, C) \in \mathbb{R}^3, t_i \in \{1, 0\}$$

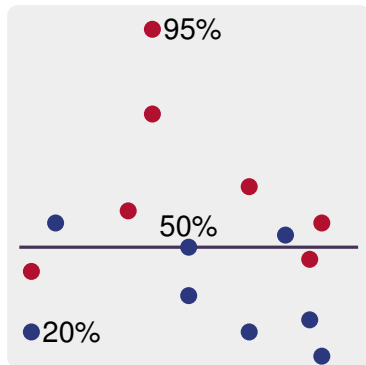
$\mathcal{L}(f, t)$ :

$$- \sum (t_i \ln(f(x_i, y_i)) + (1 - t_i) \ln(1 - f(x_i, y_i)))$$

optimalizáló: gradiens leszállás

Ciklus:

$$\underline{w}' = \underline{w} - \alpha \cdot \nabla \mathcal{L}(\underline{v})$$





## Logisztikus regresszió (hiba gradiens)

$$\begin{aligned}\mathcal{L}(u) &= -t \ln(\sigma(u)) - (1 - t) \ln(1 - \sigma(u)) = \\ &= -t \ln\left(\frac{e^u}{1 + e^u}\right) - (1 - t) \ln\left(\frac{1}{1 + e^u}\right) = \\ &= -tu + \ln(1 + e^u)\end{aligned}$$

$$\frac{\partial \mathcal{L}}{\partial u} = \sigma(u) - t \qquad \nabla \mathcal{L} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \cdot (\sigma(u) - t)$$

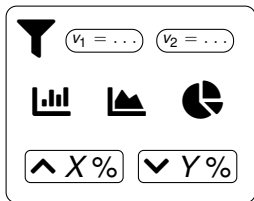
Gradiens leszállás:  $\begin{pmatrix} A' \\ B' \\ C' \end{pmatrix} = \begin{pmatrix} A \\ B \\ C \end{pmatrix} - \alpha \sum \begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix} \cdot (\sigma(u(x_i, y_i)) - t_i)$



# Változók kiválasztása

- ▶ **fontosság** (feature importance)
  - ▶ modeltől független (pl. SHAP)
  - ▶ modell specifikus (pl. döntési fák)
- ▶ **függetlenség**
  - ▶ folytonos (pl. korrelációs együttható)
  - ▶ diszkrét (pl.  $\chi^2$ -teszt)
- ▶ **változtathatóság**
  - ▶ kontextus változók (pl. lakóhely)
  - ▶ szolgáltatás minőségét leíró változók (pl. ügyfélszolgálat válaszigideje)

# A modell alkalmazása



„Mi lenne ha?”

szegmentálás és hasonlóság  
szimuláció

# Mi lenne, ha ... ?

$$f(v_1, v_2, \dots, v_n) \approx t \quad (= \mathbb{E}(t \mid \underline{v}))$$

Ügyfelek:  $\{u_1, u_2, \dots, u_k\}$ , az  $u_j$  ügyfél jellemzői:  $v_{j,1}, v_{j,2}, \dots, v_{j,n}$

Tegyük fel, hogy a  $v_1$  változó lehetséges értékei  $\{e_1, e_2, \dots, e_m\}$

*Ha csak a  $v_1$  változó módosulna*, akkor az  $u_j$  ügyfélre:

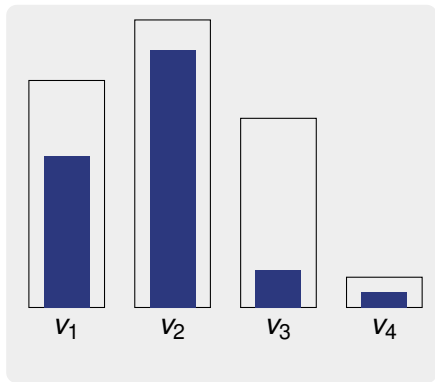
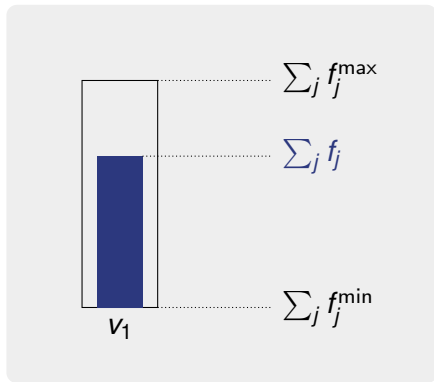
$$f_j^{\min} = \min_{x \in \{e_1, \dots, e_m\}} f(x, v_{j,2}, \dots, v_{j,n}) \leq f_j \leq \max_{x \in \{e_1, \dots, e_m\}} f(x, v_{j,2}, \dots, v_{j,n}) = f_j^{\max}$$

A teljes ügyfél bázisra

$$\sum_j f_j^{\min} \leq \mathbb{E}(\text{kimenet}) = \sum_j \mathbb{E}(\text{kimenet}_j) \leq \sum_j f_j^{\max}$$



Mi lenne, ha ... ?



# Szegmentáció



$v_1 = '40-50' \mid '50-60'$

$v_2 = '4G' \mid '5G'$



$v_1$	$v_2$	$f$
'40-50'	'5G'	5%
'20-30'	'4G'	7%
'50-60'	'5G'	4%
'20-30'	'3G'	9%



$f: 4,5\%$



$5\% \leq f \leq 10\%$

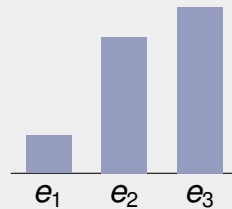
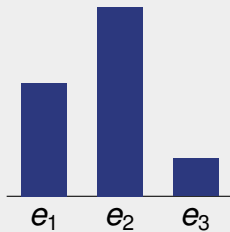
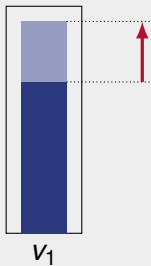


$v_1$	$v_2$	$f$
'40-50'	'5G'	5%
'20-30'	'4G'	7%
'50-60'	'5G'	4%
'20-30'	'3G'	9%

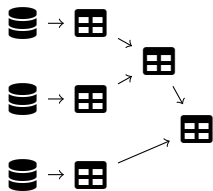


$v_1 = '20-30'$

# Szimuláció



adat futószalag



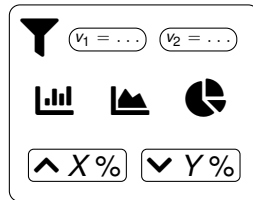
gépi tanulás

$v_1$	$v_2$	...	$t$
.	.	...	.
.	.	...	.
.	.	...	.



$$f(v_1, v_2, \dots) \approx t$$

a modell használata







⌚ automatizáció, 🛒 terméké alakítás



# Adatelemzés és matematika érettségi követelmények

- ▶ **Algebra és aritmetika, kombinatorika:** véges halmazok számossága, átlag
- ▶ **Koordináta geometria:** vektorok, skalárszorzat, egyenes egyenlete, pont és egyenes távolsága
- ▶ **Függvények:** polinomok, exponenciális és logaritmus függvény, derivált és gradiens, szélsőérték
- ▶ **Valószínűségszámítás és statisztika:** feltételes valószínűség, korreláció és függetlenség, várható érték (linearitása)
- ▶ **Gráfelmélet:** körmentes irányított gráfok, topologikus rendezés

## Források

-  David J. C. MacKay (2003) *Information Theory, Inference and Learning Algorithms*, Cambridge University Press.
-  Yaser S. Abu-Mostafa, Malik Magdon-Ismail, Hsuan-Tien Lin (2012) *Learning From Data*, AMLbook.com.
-  Ian Goodfellow, Yoshua Bengio, Aaron Courville (2016) *Deep Learning*, The MIT Press.
-  Avrim Blum, John Hopcroft, Ravi Kannan (2020) *Foundations of Data Science*, Hindustan Book Agency.