



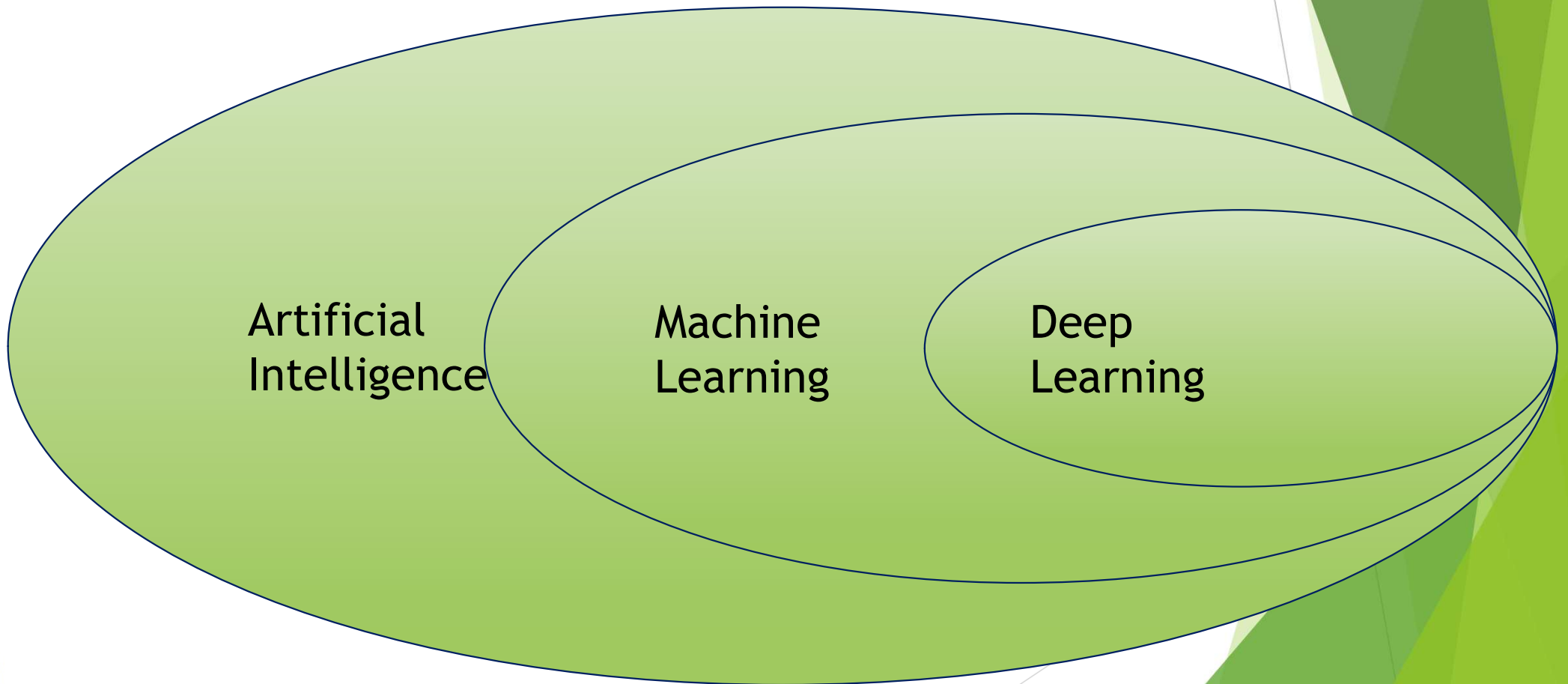
# Mesterséges intelligencia alkalmazások az ELKH Cloudon



Kertész Gábor  
tudományos munkatárs, SZTAKI-LPDS

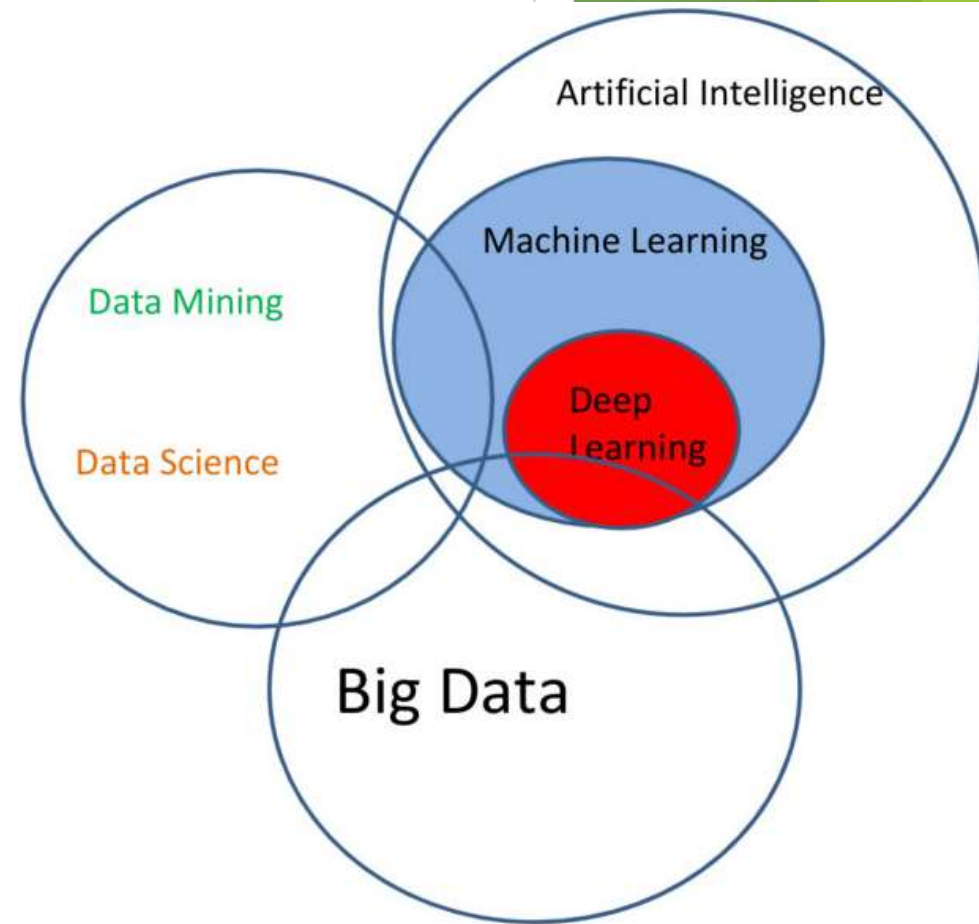
# Terminológia

Mély neurális hálózatokon alapuló gépi tanulási algoritmusok  
A magasabb rétegek egyszerű, a mélyebb rétegek összetett jellemzők detektálására alkalmasak



# Adatok

- ▶ Data Science, Big Data
  - ▶ Nagy mennyiségű adat tárolása, kezelése, feldolgozása
- ▶ Nagy számításigényű feladat
  - ▶ Felhők, szuperszámítógépek szükségesek



# Gépi tanulás



Jellemzően két nagy csoportra szokás bontani:

- ▶ Felügyelt tanulás
- ▶ Felügyelet nélküli tanulás

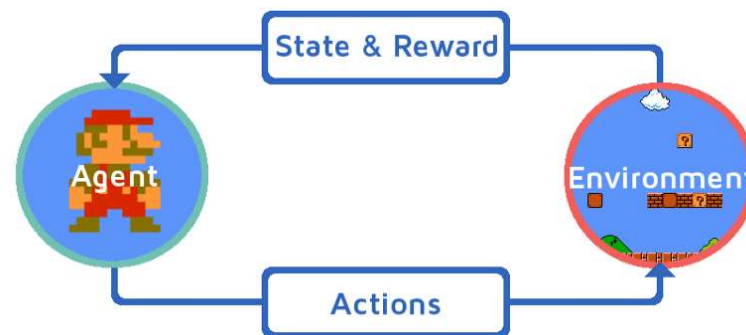
Felügyelt tanulás esetén rendelkezésre állnak olyan tanítóminták, ahol **ismert az elvárt kimenet** is, így tanításkor mérhető a pontosság.

Felügyelet nélküli tanítás esetén **címkezetlen adatok** közötti összefüggések felismerése a cél, amely jellemzően csoportosítást, klaszterezést jelent.

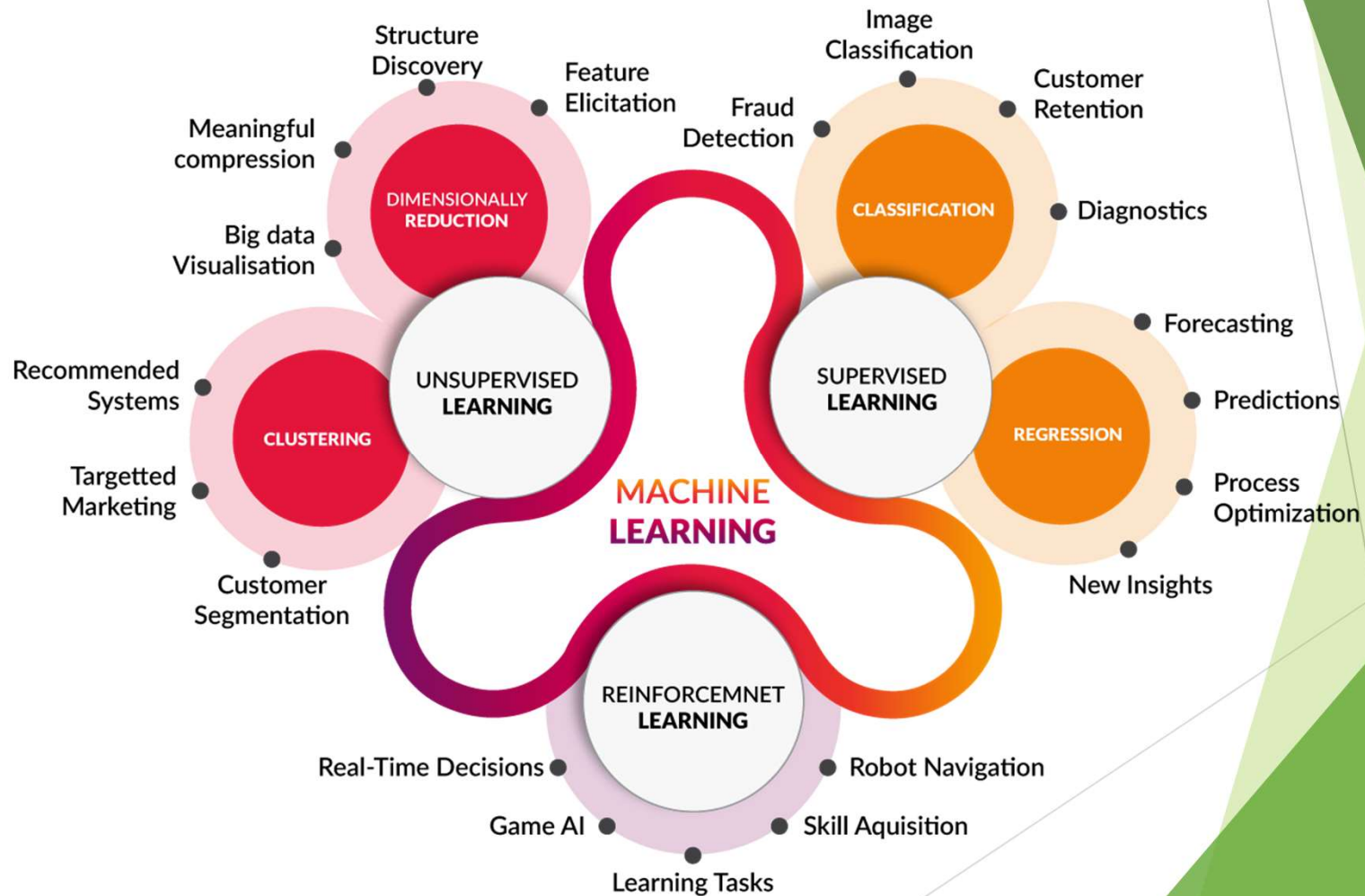
# Megerősítéses tanulás

A gépi tanulás egy különleges kategóriája a megerősítéses tanulás: ebben az esetben a feladat egy ügynök **dinamikus környezetben** való döntéshozatalának optimalizálása a környezettől kapott jutalmak maximalizálására.

Tipikus példák ilyen esetekre a különböző játékokat automatikusan játszó mesterséges ügynökök, vagy épp a Go-világbajnokot legyőző számítógépes program.



# Gépi tanulás



# A mesterséges neuron

A mesterséges neuronok a biológiai neuron által ihletett egyszerű, bemenettel és kimenettel rendelkező feldolgozó egységek

A neuron az egyes bemenetekhez súlyértékeket tárol, amelyekből súlyozott összeget kalkulál:

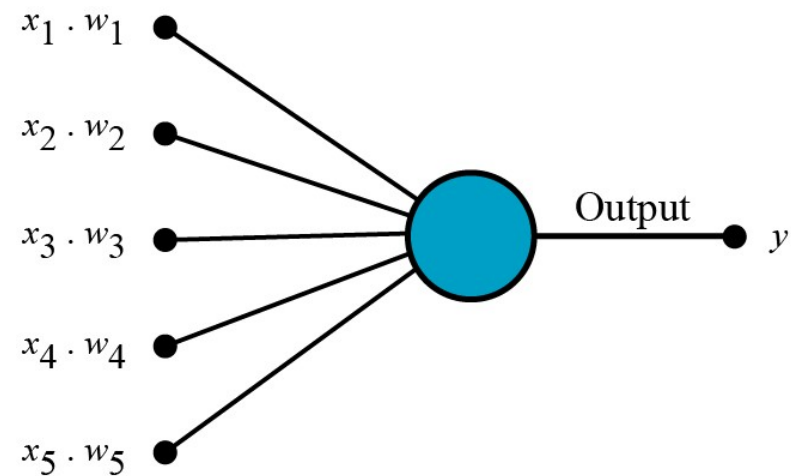
$$y = \sum_{i=1}^N w_i x_i$$

Ahol  $y$  a kimenet,  $x_i$  az  $N$  darab bemeneti paraméter  $i$ -edik eleme,  $w_i$  pedig az ehhez tartozó súlyérték.

# A mesterséges neuron

$$y = \sum_{i=1}^N w_i x_i$$

A neuron tanítása a  $w_i$  paraméterek olyasfajta módosítását jelenti, hogy a kimeneti  $y$  érték hibája minimális legyen.

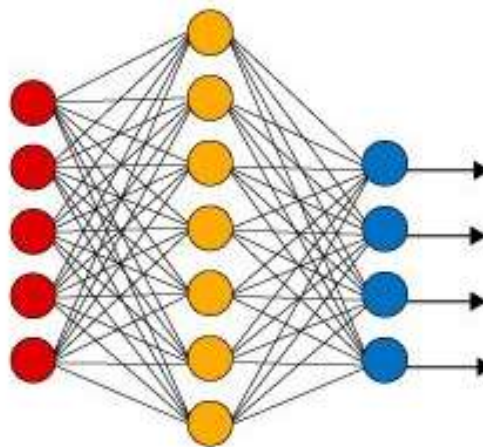




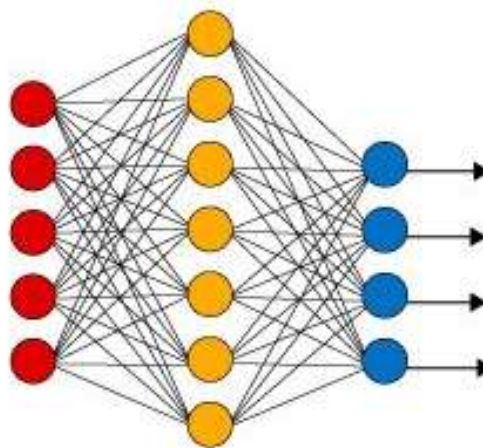
# A neurális hálózatokról dióhéjban

Neurális hálózatoknál tradicionális felépítés a többrétegű előrecsatolt hálózat, ahol több neuron **rétegek**be rendezve kerül összekapcsolásra.

Ezen hálózatok esetén az egyes rétegek neuronjainak kimenetei a következő réteg minden bemenetére lesznek rákötve, kezdve a bemeneti paraméterek számával azonos méretű bemeneti réteggel.



# A neurális hálózatokról dióhéjban



Érdeemes észrevenni a tanítható paraméterek számának alakulását: a sárga színnel ábrázolt úgynevezett rejtett réteg 7 neuronból áll, amelyek mindegyike 5 bemeneti paramétert kap, amely így összesen 35 tanítható paramétert jelent.

A rétegekben egy további, eltolás (bias) nevű, tanítható paramétert is rendelnek az egyes bemenetekhez.

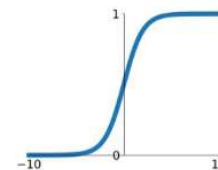
# A neurális hálózatokról dióhéjban

$$A_j = \varphi \left( \sum_{i=1}^N w_i x_i + b \right)$$

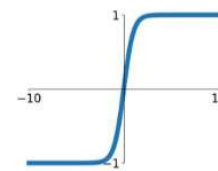
Ahol  $A_j$  jelöli a  $j$ . neuron aktivációját,  $\varphi$  az aktivációs függvényt,  $w_i, x_i$  az  $i$ . súly és bemenet értékeket,  $b$  pedig az eltolást.

- ▶ Bár a paraméterszám növekszik, a **Sigmoid** rétegű hálózatok egyértelműen jól kinyerésében  $\sigma(x) = \frac{1}{1+e^{-x}}$
- ▶ A mélyebb rétegek összetettebb jeleket **tanh**
- ▶ Ha megnöveljük a rétegszámot, a **tanh**( $x$ ) hálózatokról beszélni

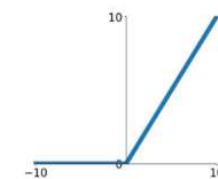
**ReLU**  
 $\max(0, x)$



de a több  
ebb jellemzők

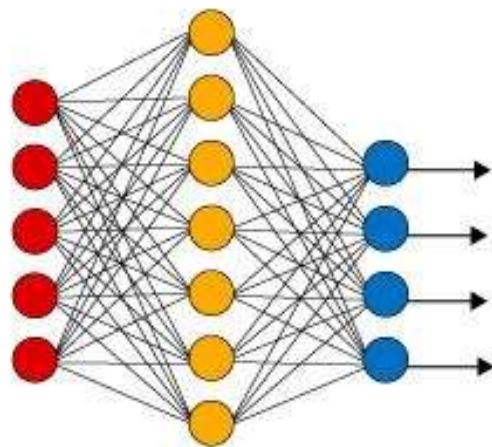


ti  
ális



# Mély Neurális Hálózatok

## Simple Neural Network

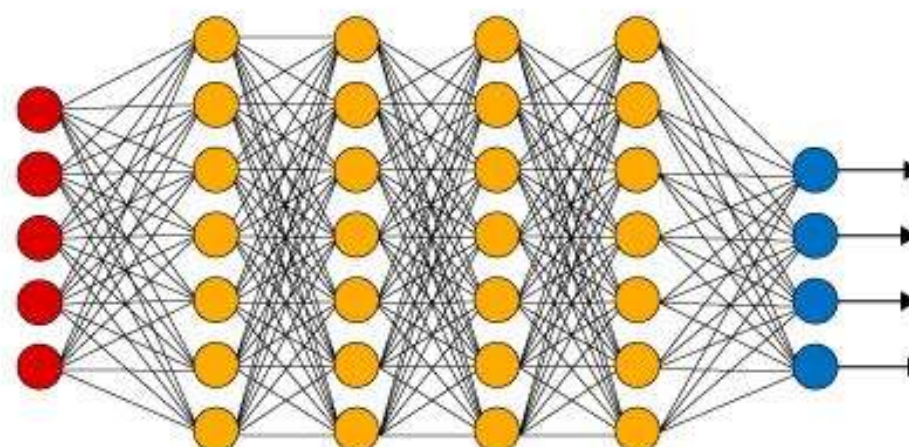


● Input Layer

● Hidden Layer

● Output Layer

## Deep Learning Neural Network



# Mély Neurális Hálózatok

- ▶ A mély hálózat több rétegből áll, a rétegekben nagy számban vannak tanítható paraméterek
- ▶ A tanításhoz nagy számú tanítóminta szükséges (*Big Data*)
  - ▶ Ha nincs elégséges számú tanítóminta, akkor lehet hogy a tanítási adatokra túlságosan illeszkedni fog a modell, és így nem általánosít majd jól (overfitting)
- ▶ A tanítás folyamata így memóriaigényes, és a számítási komplexitás magas
- ▶ **A feladat azonban nagyszerűen párhuzamosítható multiprocesszoros környezetben, kifejezetten GPU gyorsítókkal**



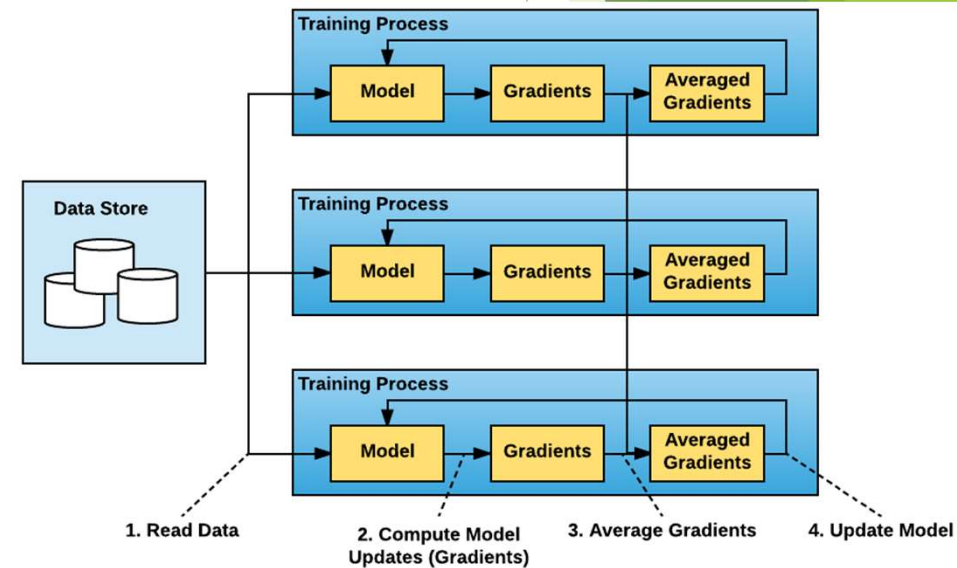
# A mély tanulás számítási komplexitása

- ▶ A mini batcheken alapuló backpropagation algoritmust lehetséges adatrészre bontva implementálni
  - ▶ Erre nagyszerű eszköz a grafikus gyorsító, amely több ezer műveletvégző egységgel rendelkezik
  - ▶ A 2010-es évek elején jelentek meg az első GPGPU-implementációk
- ▶ A hardvergyártók esetenként már dedikáltan multiprocesszoros feldolgozásra terveznek eszközöket
  - ▶ Nem grafikai renderelésre, hanem kutatáshoz készülnek
  - ▶ Ezen költséges eszközök beszerzése helyett felhő alapú, igény szerint történő felhasználásuk a logikus



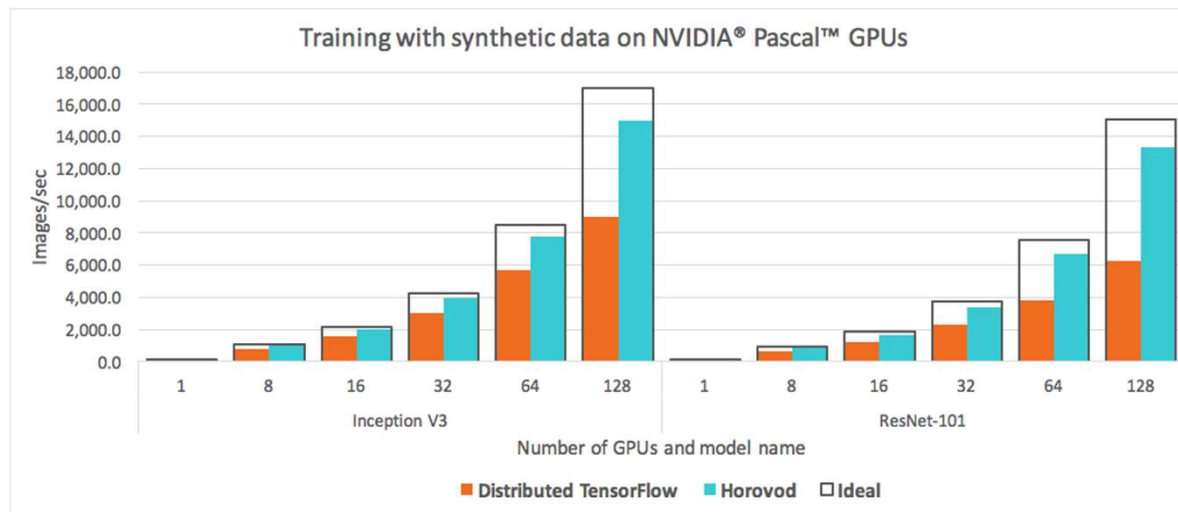
# Elosztott feldolgozás

- ▶ A fizikai korlátok elérésekor érdemes elgondolkodni a horizontális skálázáson, azaz a feladat részfeladatokra bontásán és elosztott feldolgozásán
- ▶ A backpropagation algoritmus esetén sajnos ez nem triviális
- ▶ Az adatdekompozíció alapuló megközelítés népszerű, és könnyen implementálható
  - ▶ A modelltől másolatok készülnek klaszter minden elemére
  - ▶ Minden állomáson lezajlik a tanítás egy-egy kötegni tanítómintán
  - ▶ A kiszámolt gradiensek megosztásra kerülnek egymással, a súlyparaméterek együtt kerülnek módosításra



# Elosztott mély tanulás

- ▶ Az elosztott mély tanulás sarokpontja a kommunikáció
  - ▶ A gradiensek cseréje és a paraméterek módosítása költséges is lehet
  - ▶ Nagyobb, mélyebb modell több paramétert jelent, amely nagyobb átviteli költséget eredményez
- ▶ Léteznek megoldások, amelyek magas hatékonyságot ígérnek a skálázáshoz
- ▶ **Mély tanulási klasztert nem érdemes saját eszközökből kialakítani, az idő jelentős részében kihasználatlan lenne - éppen erre jó a felhő**





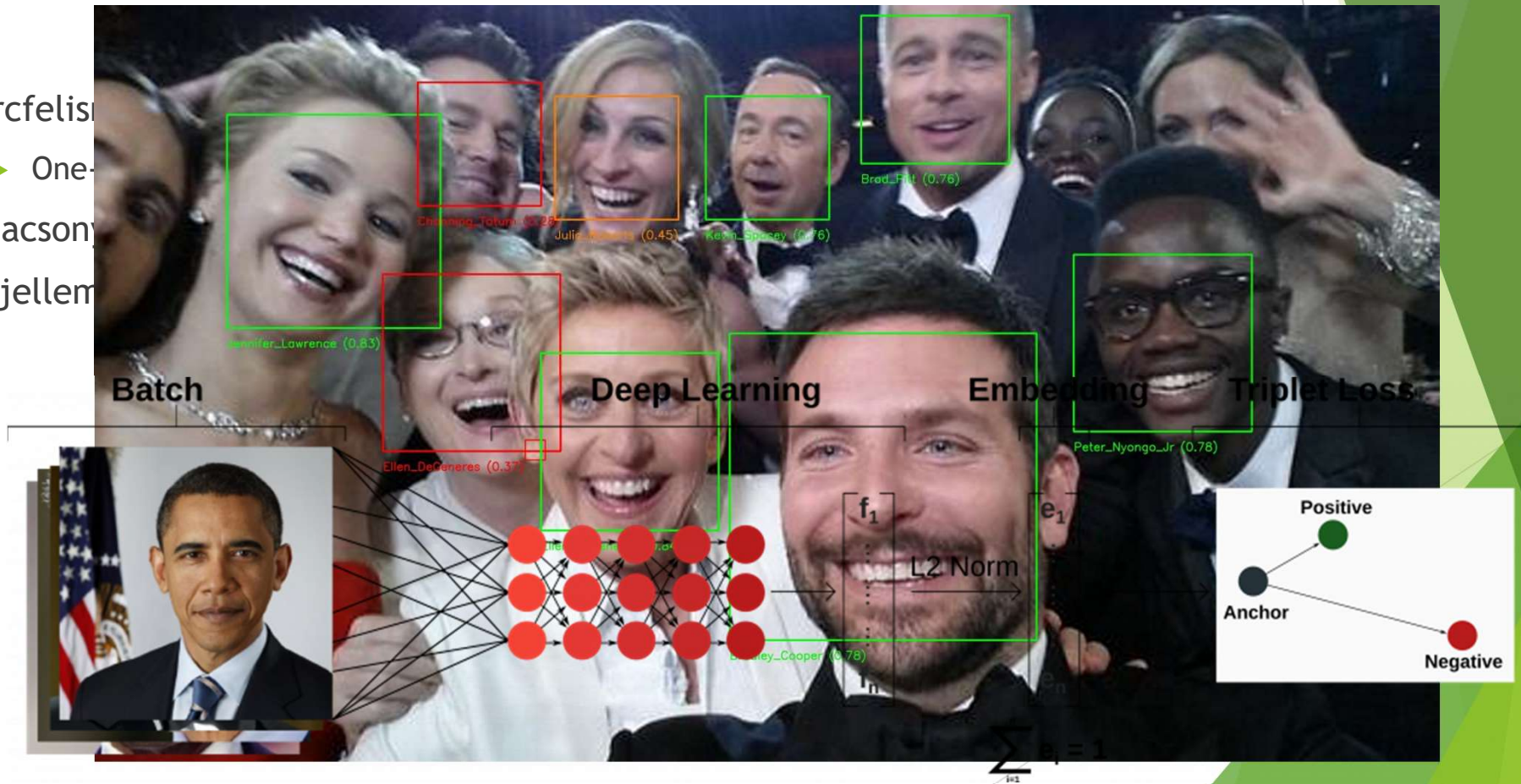
# Alkalmazások

- ▶ Ajánlórendszerek
  - ▶ Anomália detektálás
- ▶ Gépi látás
  - ▶ Képi adatok feldolgozása
  - ▶ Többosztályú klasszifikáció
- ▶ Természetes szöveg feldolgozás
  - ▶ Beszéd felismerés
  - ▶ Szöveg generálás
- ▶ Tartalomgenerálás



# People identification

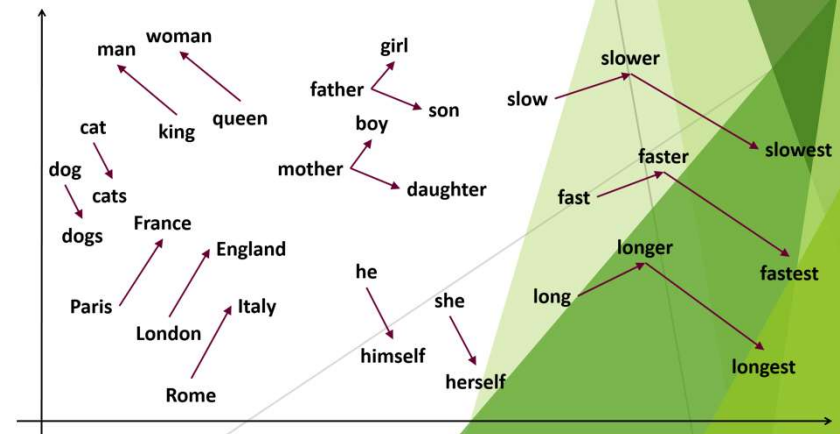
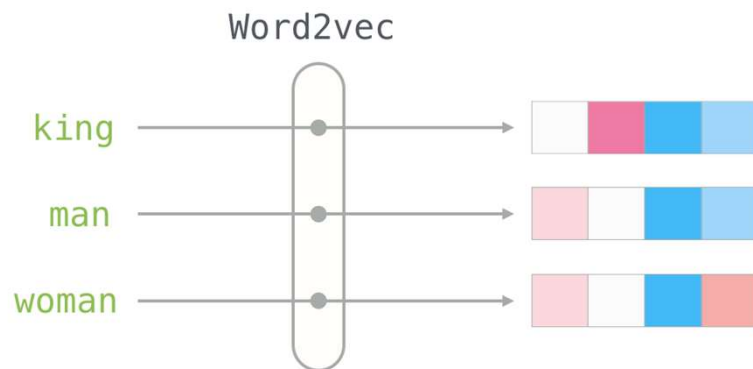
- ▶ Arcfelis
- ▶ One-
- ▶ Alacson
- ▶ A jellem





# Word2Vec, text recognition

- ▶ Hogyan működnek a chatbotok?
- ▶ A neurális hálózatok bemeneteként továbbra is számérték adható meg
- ▶ A szavak vektorokká alakítása az előfordulásuk és környezetük alapján



# OpenAI text generation

- ▶ GPT-3
- ▶ A világhálón elérhető szövegek alapján végzett „önálló” tanulást
- ▶ Az információkat kigyűjtötte és egy összetett állapot térben tárolja

William Shakespeare ¶

Transformer AI poetry: Poetry classics as reimagined and rewritten by an artificial intelligence.

“Sonnet 18”

William Shakespeare

Shall I compare thee to a summer's day?

Thou art more lovely and more temperate:

Rough winds do shake the darling buds of May,

And summer's lease hath all too short a date;

A winter's day, when beams of sun are few,

And skies are grey with clouds intemperate,

Is to thy beauty somehow sordid grown.

Her lovely-lovely eye is quite shut up,

Like windows pale before a sunny shower.

The earth that's cold and moist, like to my mistress,

When once that she unbrac'd, her lovely jewel,

Grows as both ugly and uncourtly.

Poor soul, the center of my motley host,

See, now they request thy sovereign presence;

The din of merry hearts hath brought me thus

To greet thee, thou unsoil'd and saint-like beauty.

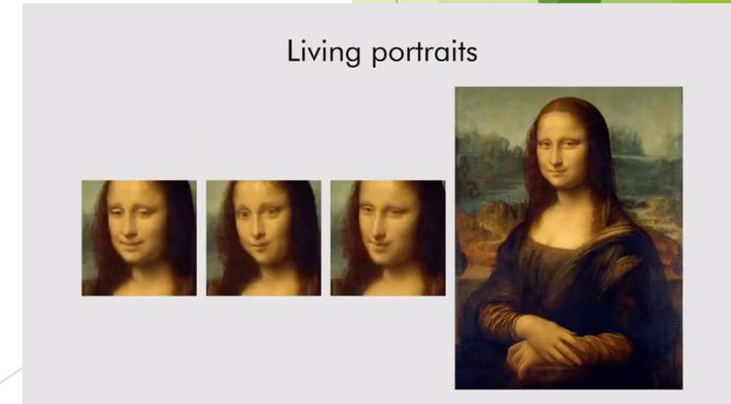
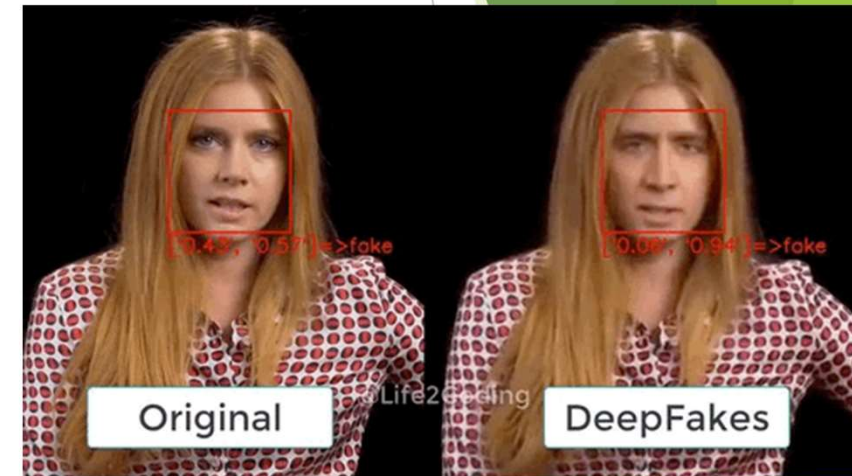
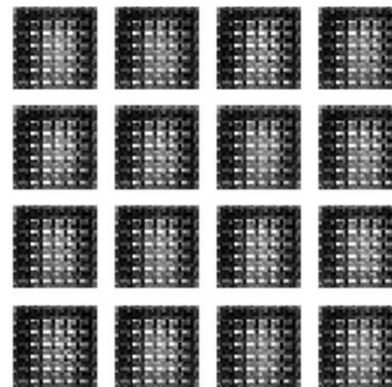
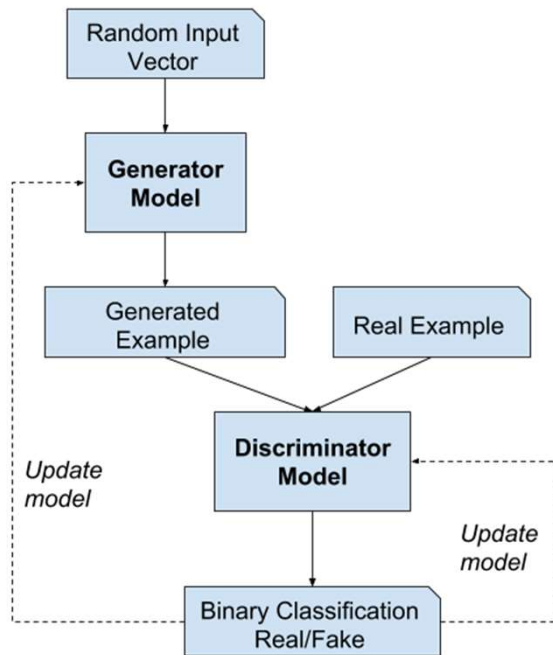


ELKH Cloud

Brown, Tom B., et al. "Language models are few-shot learners." *arXiv preprint arXiv:2005.14165* (2020).

# Generative Adversarial Networks

- ▶ Ian Goodfellow, Yoshua Bengio et al
- ▶ Generator vs Discriminator



„Any sufficiently advanced technology is indistinguishable from magic.”

Arthur C. Clarke, 1973

**Köszönöm a figyelmet!**

