Time - domain audio features

Amplitude envelope: meghatározza egy framen belül mikor a maximális hang-amplitúdó, milyen erősen szólt a hang framen belül. Egy frame összes sampleje között a legnagyobb amplitúdó érték.

$$AE_t = \max_{k=t\cdot K}^{(t+1)\cdot K-1} s(k)$$

AE_t - a t-edik frame amplitúdó csúcsa

$s(k)$ - a k-adik mintához tartozó amplitúdó

K - frame méret, minták száma

$k = t\cdot K$ - t frame első mintája

$(t+1)\cdot K-1$ - t frame utolsó mintája

max - adott tartományon belül levő minták amplitúdójának maximuma.

- érzékeny a kiugró értékekre: ha egy frameben van egy nagyon erős rövid zaj vagy csúcs, rögtön kiemelődik.
- hangkezdetek felismerése, zenei minták osztályozása: a hangerő gyors változásainak meghatározását
- tipikusan a hangerő időbeli alakjának leírására szolgál, (attack, decay, sustain, release)

Root-mean-square energy: meghatározza RMS energiáját egy adott frame-ben. Adott frame összes mintájának négyzetét összeadjuk, átlagoljuk majd gyököt vonunk.

$$RMS_t = \sqrt{\frac{1}{K} \cdot \sum_{k=t\cdot K}^{(t+1)\cdot K-1} s(k)^2}$$

RMS_t - t-edik frame RMS energiája

K - frame mérete, minták száma

s(k) - k-adik mintához tartozó amplitúdó

- kevésbé érzékeny a kiugró részekre
- a hangerő mutatója, megbízhatóbb, kevésbé zajérzékeny
- hangszegmensek elkülönítése, zenei műfaj osztályozás

Zero crossing rate: meghatározza egy hangjel hányszor metszi a nulla szintet egy adott frame-en belül.

$$ZCR_t = \frac{1}{2} \cdot \sum_{k=t\cdot K}^{(t+1)\cdot K-1} \left| sgn(s(k)) - sgn(s(k+1)) \right|$$

ZCR_t = t edik frame zero crossing értéke

K - frame hossza, minták száma

sgn(s(k)) - s(k) értékének előjele

A képlet minden egymást követő minta előjelét összehasonlítja, ha előjelváltás van, akkor nő a számláló (abszolút értékben). Mindezt összeadja egy adott frame-en belül, majd felezi az értéket. (mert egy előjelváltás két mintát érint)

- ZCR segít megkülönböztetni a zajszerű, ütős hangokat a folytonos, dallamos hangoktól, mert az ütős hangokban nagyon sok nullátmenet van, míg a dallamosakban kevesebb.
- beszédjelekben segít megkülönböztetni a zöngés - zöngétlen tartalmat: zöngétlen hangokban nagyobb a ZCR
  zöngéses hangokban alacsonyabb a ZCR

- durva hangmagasság becslés: ZCR arányos lehet a jel frekvenciájával, egyszerűbb, egyfrekvenciás jeleknél

<u>Fourier transformation</u>

Bármely időfüggvényt frekvencia-összetevőkre bont. Audiojelfeldolgozásban ez lehetővé teszi, hogy:
- azonosítsd egy hang alaphangját és felharmonikusait
- kiszűrd a zavaró zajt
- jellemzőket (pl. spektrumcsúcsokat) adj az ML modellnek

<u>Módszer lényege</u>:
A vizsgált jelet különböző frekvenciájú szinuszokkal hasonlítja össze. Matematikailag ez a belső szorzat (skaláris sz.) - vagyis azt méri, mennyire hasonló a jel az adott frekvenciájú tiszta szinuszhoz

Minden frekvenciához két értéket kapunk:
- amplitúdó (abszolút érték) - milyen erős az adott frekvenciakomponens a jelben
- fázis - azt jelzi, hogy a jel adott frekvenciájú összetevője, mennyivel van eltolva időben a referencia-szinuszhoz képest.

Ha egy frekvencián nagy az amplitúdó, a jel nagy energiát tartalmaz azon a frekvencián, tehát erősen hasonló az ottani tiszta szinuszjelhez. Frekvenciánként megnyitja szinuszra a jel, ebből megadja az

Szinusz hullám: $y(t) = A \cdot \sin(2\pi f t + \varphi)$

Koszinusz hullám: $y(t) = A \cdot \cos(2\pi f t + \varphi)$

Lépések:

Választasz egy frekvenciát pl: 530 C_5

Optimalizálod a fázist

Kiszámolod az amplitúdót

- Az FT egyszerre elvégzi az összes frekvenciára, a nagyjából a művelet végrehajtását célozza.

$$\varphi_f = \underset{\varphi \in [0,1]}{\operatorname{argmax}} \left(\int s(t) \cdot \sin(2\pi \cdot (ft - \varphi)) \cdot dt \right)$$

$s(t)$ - bemeneti jel argmax - max argumentum, legnagyobb hasonlóság

1. Választ egy frekvenciát amire kíváncsi jelben. 2. Vesz egy referencia sin hullám-ot ezen a frekvencián. 3. Kipróbál minden lehetséges fáziseltolást [0,1) intervallumban. 4. Minden eltolásnál az integrál hasonlóságot számít a jel és a sin között. 5. A végeredmény φ_f fáziseltolás lesz, ahol a hasonlóság a legnagyobb volt.

- A FT pont ezt a feladatot oldja meg elegánsabban, komplex számokkal. Ahelyett, hogy végigpróbálná az összes fázist, a komplex exponenciális használatával egyszerre találja meg.

$$d_f = \max_{\varphi \in [0,1)} \left(\int s(t) \cdot \sin(2\pi \cdot (ft - \varphi)) \cdot dt \right)$$

1. Vesz f frekvenciát. 2. Megkeresi azt a fáziseltolást, amelyre a referencia sin legjobban hasonlít. 3. A legnagyobb hasonlóság lesz az adott f frekvenciájú komponens amplitúdója

- A képlet kiszámítja egy adott f frekvenciájú sin komponens nagyításágát

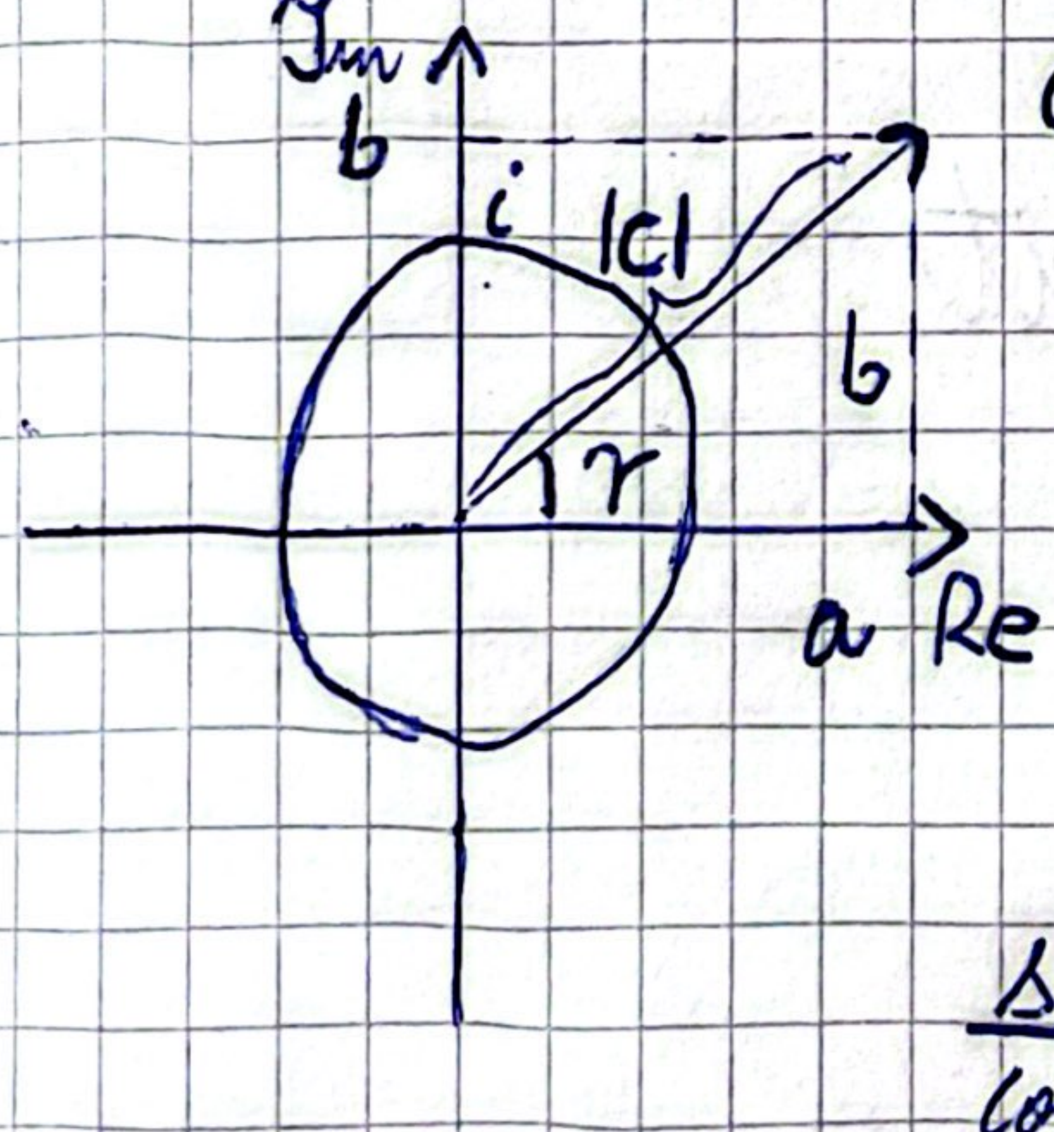
c. Complex Numbers for Audio Signal Processing

- A jelek valódi szerkezetét nem csak amplitudóval, hanem fázissal együtt lehet leírni. Ha csak az amplitudót néznénk, nem tudnánk visszaállítani az eredeti hullámformát, csak erősen megsérült, torz hangot kapnánk.
- A komplex reprezentáció (amplitudó + fázis) teszi lehetővé a hang és más időbeli jelek pontos feldolgozását, szintézisét, visszaállítását.

$$c = a + ib \qquad a, b \in \mathbb{R}$$

a - valós rész
b - képzetes rész

<u>Polár koordinátarendszer</u>



$$c = a + ib$$

$$|c|^2 = a^2 + b^2 \Rightarrow |c| = \sqrt{a^2 + b^2}$$

$$\cos(\gamma) = \frac{a}{|c|} \qquad \sin(\gamma) = \frac{b}{|c|}$$

$$\frac{\sin(\gamma)}{\cos(\gamma)} = \frac{b}{a} = \tan(\gamma) \Rightarrow \gamma = \arctg\left(\frac{b}{a}\right)$$

$$a = |c| \cdot \cos(\gamma) \qquad b = |c| \cdot \sin(\gamma)$$

$$c = a + ib \Rightarrow c = |c| \cdot \underbrace{(\cos(\gamma) + i\sin(\gamma))}_{e^{i\gamma}}$$

$e^{i\gamma}$ - euler formula

Egy időben változó jel felbontása a benne lévő szinuszos frekvenciák összegére.

$$\hat{g}(f) = \int g(t) \cdot e^{-i2\pi f t} \, dt$$

$g(t)$ - időfüggő jel

$\hat{g}(f)$ - frekvenciaspektrum

$e^{-i2\pi f t}$ - komplex hullámot, forgóvektort vagy kört ír le

Mivel $e^{ix} = \cos(x) \cdot i\sin(x)$ a kifejezés egy cos és sin hullám kombinációja. Transzformáció során az eredeti $g(t)$ jelet minden egyes f frekvencián megszorozzuk ezzel a komplex hullámmal. A szorzás eredménye akkor lesz nagy, ha a $g(t)$ jelben erős az adott f frekvenciájú komponens.

$g(t) \rightarrow x(n) \qquad t = nT$

analóg   diszkrét

$$\hat{g}(f) = \int g(t) \cdot e^{-i2\pi ft}\, dt \Longleftrightarrow \hat{x}(f) = \sum_n x(n) \cdot e^{-i2\pi fn}$$

analóg   diszkrét

Discrete Fourier-Transformation: amíg a FT egy teljes, végtelen hosszúságú analóg jelet elemez, a DFT egy véges számú, digitális mintából álló jelsorozatot alakít át frekvencia-komponensekre.

$$\hat{x}\left(\frac{k}{N}\right) = \sum_{n=0}^{N-1} x(n) \cdot e^{-i2\pi fn \frac{k}{N}}$$

A DFT kimenetén kapott frekvencia-komponensek száma (M) megegyezik a bemeneti jelsorozat mintáinak számával. (N)

1. Invertible Transformation: Ha $M = N$ azt jelenti, a transzformáció visszafordítható. Inverz DFT segítségével pontosan vissza lehet állítani az eredeti időtartománybeli jelet frekvenciaspektrumból.

2. Computational efficient: Az $M = N$ korrláthoz a számítási hatékonysághoz, különösen FFT algoritmusnál. Az FFT kihasználja ezt az egyenlőséget, hogy jelentősen csökkentse a szükséges számítási lépések számát.

$k = [0, M-1] = [0, N-1]$

$$F(k) = \frac{k}{NT} = \frac{k \cdot s_r}{N}$$

- Egy adott frekvencia relatív előfordulása vagy intenzitása egy időtartam vagy mintavételezési kontextusban.

Középponti frekvencia:

$$F(k) = \frac{k \cdot Sr}{N} \qquad k = N/2$$

$$\Rightarrow F(N/2) = Sr/2 \quad \Rightarrow \text{Nyquist frekvencia}$$

<u>DFT</u> közvetlen kiszámítása sok műveletet igényel. A műveletszám nagyjából N^2 arányban nő a minták számával.

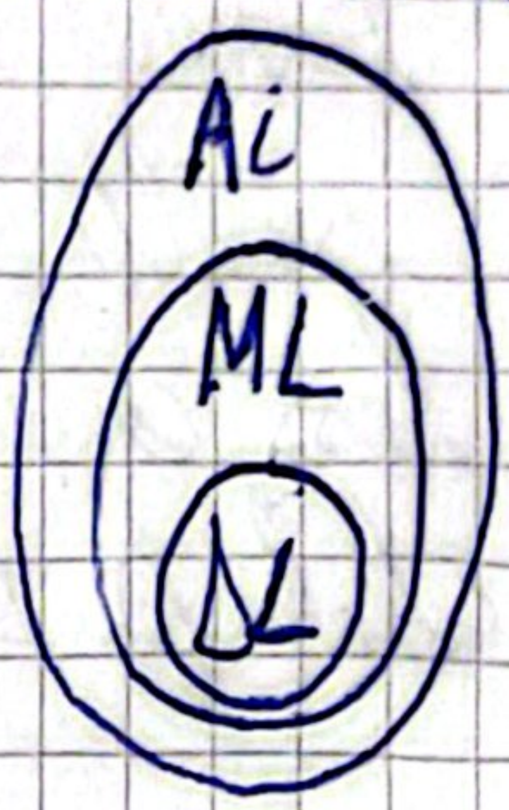
<u>FFT</u> a DFT értékeit sokkal hatékonyabban számítja ki. Műveletszám nagyjából $N \cdot \log_2(N)$ ami jelentős gyorsulás. FFT akkor működik a legjobban ha N kettőnek valamely hatványa. A bemeneti minták számát folyamatosan felezi. Ha N 2-nek hatványa a felbontás végig egyenletes maradhat. Az FFT az „Oszd meg és uralkodj" elven működik, amely az ismétlődéseket és szimmetriákat kihasználva drasztikusan csökkenti a szükséges műveletek számát N^2-ről $N \log_2 N$-re

---

<u>Short-Time Fourier Transformation</u> az eddigiekkel ellentétben az STFT lehetővé teszi a jel frekvenciaösszetevőinek elemzését az idő múlásával. A jelet kis, egymást átfedő szegmensekre (frame) osztja, és minden egyes ablakon külön-külön elvégzi a DFT-t. Az eredmény egy <u>spektrogram</u> amely a frekvenciaváltozásokat mutatja az idő függvényében.

Ai, Machine and Deeplearning

Venn diagram



Ai: Minden olyan ~~kategóriát~~ technológiát, ~~amely~~ és tudományterületet magába foglal, ahol a gépek emberihez hasonló intelligens viselkedést mutatnak, például problémamegoldás, tanulás vagy érzékelés terén

ML: Ai egyik alterülete. A rendszerek nem előre megírt szigorú szabályok alapján működnek, hanem adatokból tanulnak, és azokban mintázatokat fedeznek fel, hogy döntéseket hozzanak.

DL: az ML specializáltabb ága. Komplex többrétegű neurális hálózatokat használ a tanuláshoz, különösen hatékonyan kezelve nagy mennyiségű adatokat

ML paradigmák

Supervised learning: olyan adatokon tanítják ahol minden bemenethez meg van adva a helyes kimenet (címke).
Pl: kézírás / számjegy felismerés - CNN

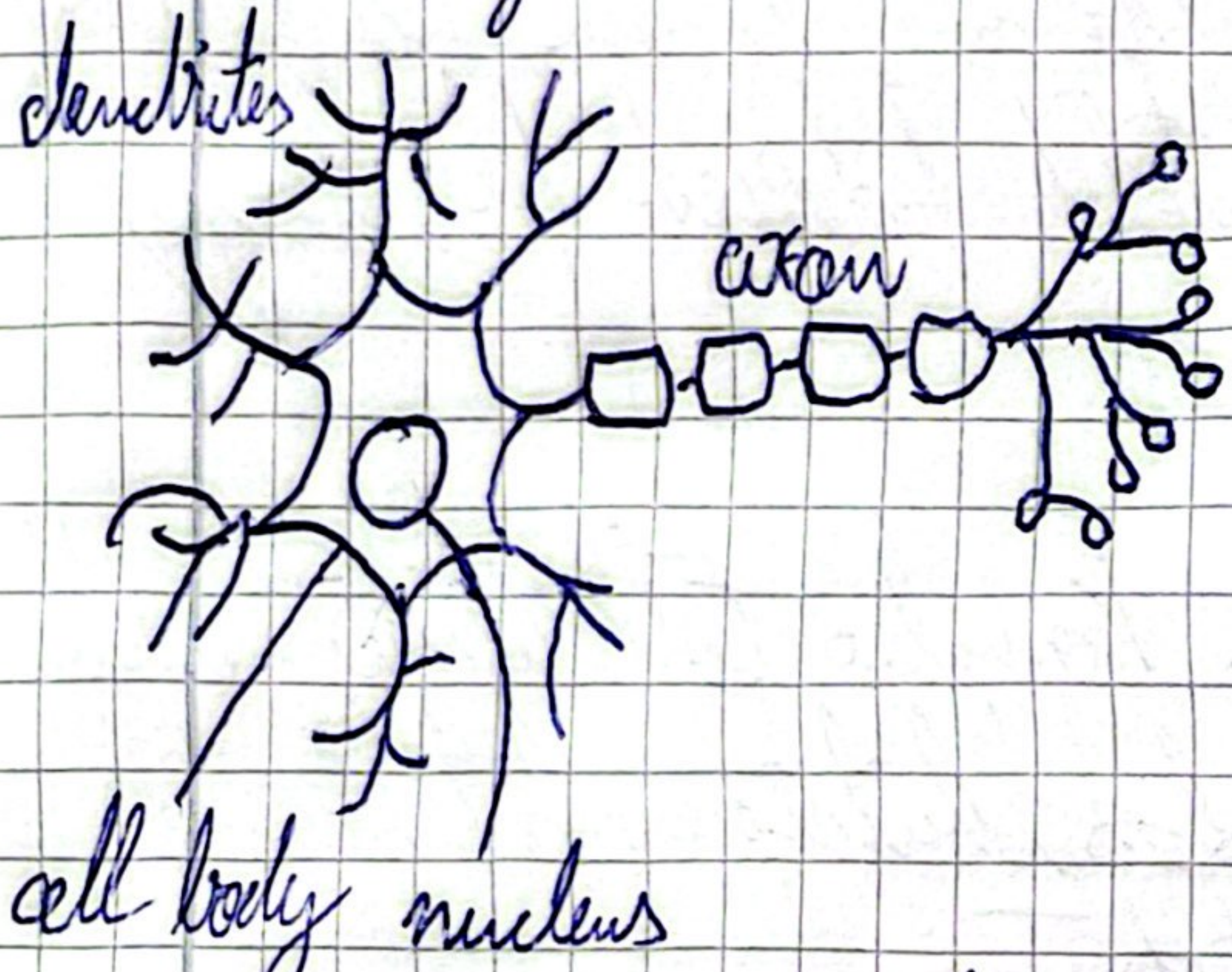
Unsupervised learning: segítség nélkül magától fedezi fel a rejtett összefüggéseket az adatokban. Nincsenek előre megadott helyes válaszok, Pl: Zeneszám csoportosítás - K means clustering

Reinforcement learning: próbálkozások és hibák útján, jutalmak és büntetések rendszerén tanul. Pl: robotok betanítása - Q-learning

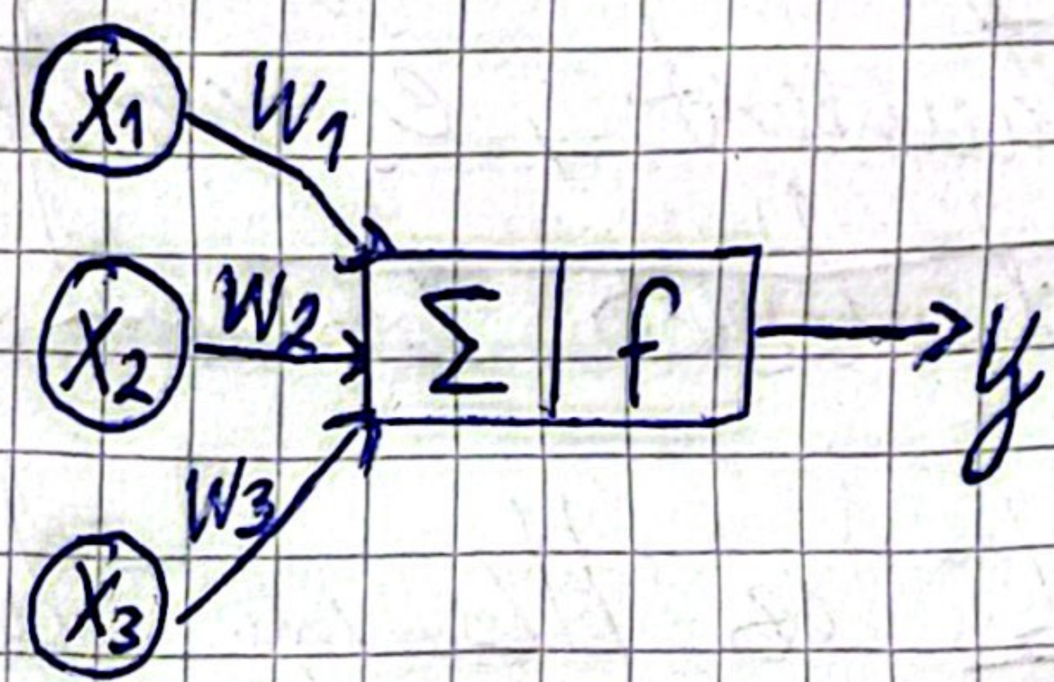
Traditional ML vs DL

feature engineering vs end-to-end
small dataset vs large dataset
less computation intensive vs very resource intensive
ideal for simple problems vs ideal for complex problem

Biological neuron



Artificial neuron



$$h = \sum_i x_i w_i = x_1 w_1 + x_2 w_2 + x_3 w_3$$

$$y = f(h) = f(x_1 w_1 + x_2 w_2 + x_3 w_3)$$

x - bemenet ; w - súly ; h - súlyozott összeg
f - aktiváció ; y - kimenet

Az aktivációs dönti el, hogy a neuron milyen erősségű jelet adjon tovább a következő rétegnek.

1. Nemlinearitás bevezetése: komplexebb problémák megoldása
2. Kimenet normalizálás: neuron kimenete jól definiált standard állapotban.

Vector: matematikai-fizikai fogalom amely olyan mennyiséget ír le, amelynek van:

1. nagysága
2. iránya
3. irányítása

$$a = [a_1 \ a_2 \ a_3] = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

Scalar operations: a vektor és egy szám művelete

Vektor operations: két azonos dimenziójú vektor művelete

Dot product vs **Matrix multiplication**

- két azonos méretű vektor
- elementwise szorzás, majd a szorzatok összegzése
- skalár az eredmény

- két ~~különböző~~ egyenlő méretű mátrix (sum, sub)
- sort oszloppal
- mátrix az eredmény
- A oszlop szám = B sorszám (Mátrixszor.)

~~Mátrix~~ **Matrices**: számokból álló téglalap-alakú táblázat

$$A_{ij} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix}$$

Row vector = (1, n) matrix
Column vector = (n, 1) mátrix