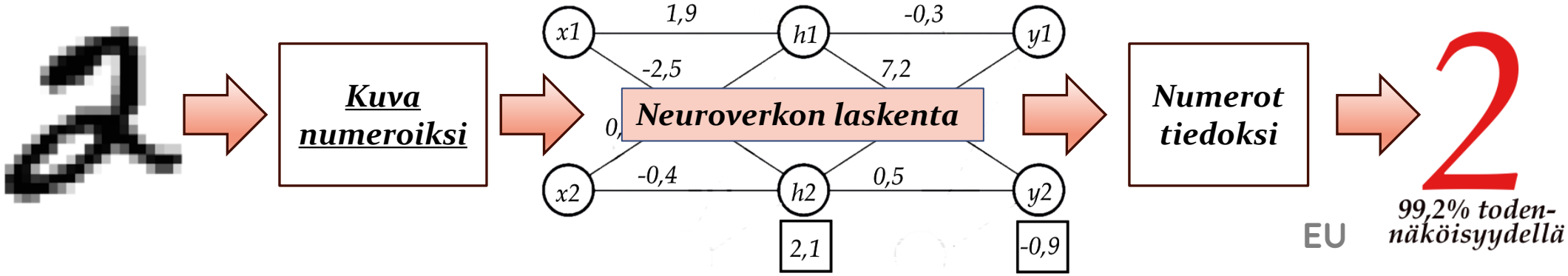
# Kuinka neuroverkon laskenta toimii

# Neuroverkko, tekoälyn sydän ja moottori

- Keinotekoinen neuroverkko on tekoälyn toiminnan perusta, eräänlainen laskentakehikko, joka laskee tekoälylle annetusta lähtötiedosta tekoälyn tuottamat tulostiedot. Neuroverkossa kaikki tieto on numeerista ja kaikki operaatiot matemaattisia operaatioita.
- Neuroverkko suunnitellaan ja koulutetaan aina tiettyä tekoälyn tehtävää varten.
- Kun neuroverkko on valmis suorittamaan sille koulutettua tehtävää, käyttää tekoälysovellus sitä seuraavasti:
  1. Tekoälyn lähtötieto, esimerkiksi teksti, ääni tai kuva muutetaan numeeriseen muotoon ja syötetään sisään neuroverkkoon.
  2. Neuroverkko suorittaa saamallaan numeerisella syötteellä sarjan laskutoimituksia, jotka perustuvat neuroverkon sisältämiin lukuihin eli laskentaparametreihin. Laskennan lopputuloksena syntyy neuroverkon määrittämä tulostieto numeerisessa muodossa.
  3. Lopuksi neuroverkon suorittaman laskennan numeerinen tulos muutetaan selväkieliseen muotoon, esimerkiksi tekstiksi, ääneksi tai kuvaksi. Tämä on tekoälyn vastaus annettuun syötteeseen.

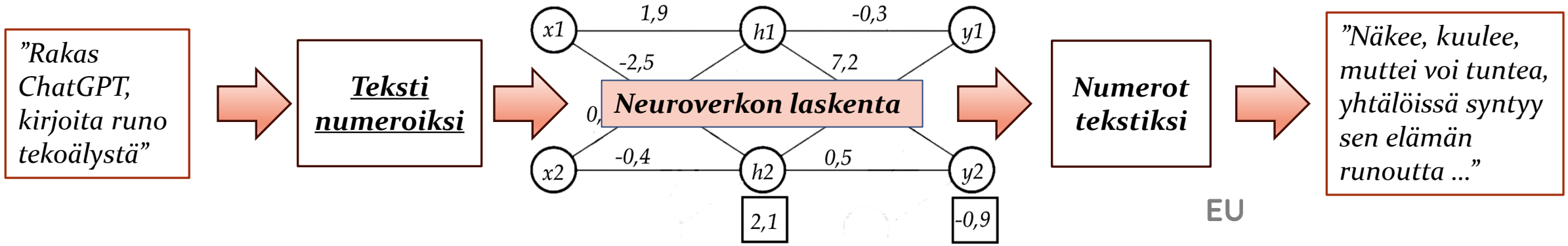


# Esimerkkejä neuroverkon tehtävistä

| Lähtötieto / kehote | Tehtävä |
|---|---|
| Valokuva ihmisestä | Tee arvio henkilön iästä ja sukupuolesta |
| Äänitiedosto linnunlaulusta | Tee arvio linnun lajista |
| Laboratoriotuloksia, röntgenkuva | Tee diagnoosi |
| Kehoteteksti kuvan luontiin | Luo kuva |

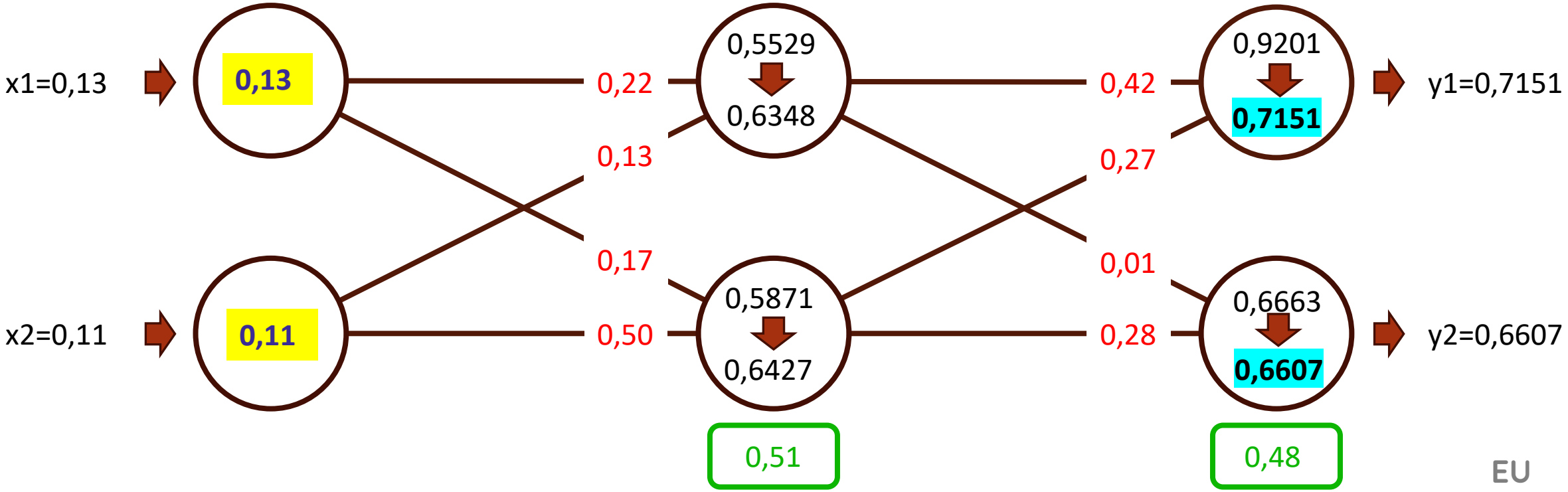
Yllä esitettyjen esimerkkineuroverkkojen tehtävät ovat selkeitä ja helposti ymmärrettäviä. Generatiivisten tekstiä tuottavien tekoälysovellusten tehtävä sen sijaan ymmärretään helposti väärin, sillä niissä neuroverkon tehtävä on laskea todennäköisin jatko annetulle kehotteelle, ei antaa oikeaa vastausta tai järkevää kommenttia syötetekstiin:

| Lähtötieto / kehote | Tehtävä |
|---|---|
| Kehote generatiiviselle tekoälysovellukselle | Luo todennäköinen jatko annetulle kehotteelle |



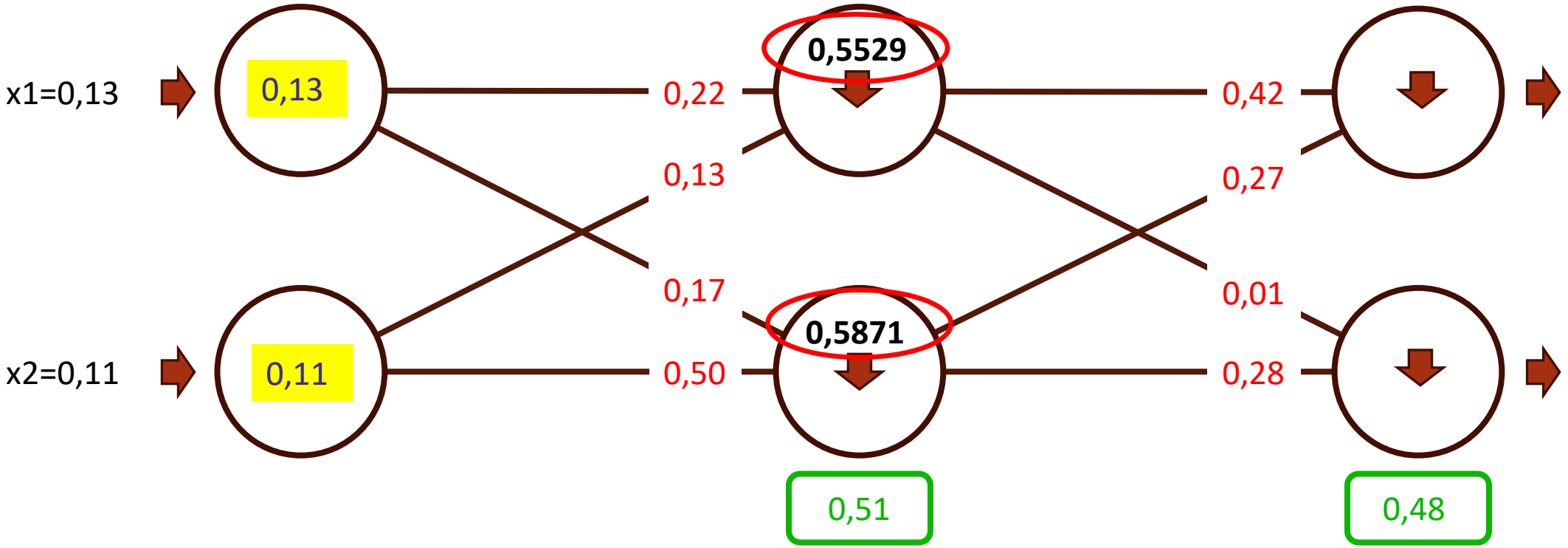
# Neuroverkon laskenta 1/10, neuroverkon rakenne

- Keinotekoinen neuroverkko koostuu **neuroneista (ympyrät)** sekä niitä yhdistävistä **synapseista**.
- Neuroverkon laskenta perustuu verkon **laskentaparametreihin**, joita ovat synapseihin liitetyt **painoarvot** sekä verkon kuhunkin kerrokseen liitetyt **bias- eli poikkeama-arvot**.
- Neuroverkon laskenta etenee vasemmalta **sisään tulevista arvoista** (esimerkissä x1 ja x2) oikealle, kunnes päästään neuroverkon tulosneuroneiden **ulostuloarvoihin** (esimerkissä y1 ja y2).



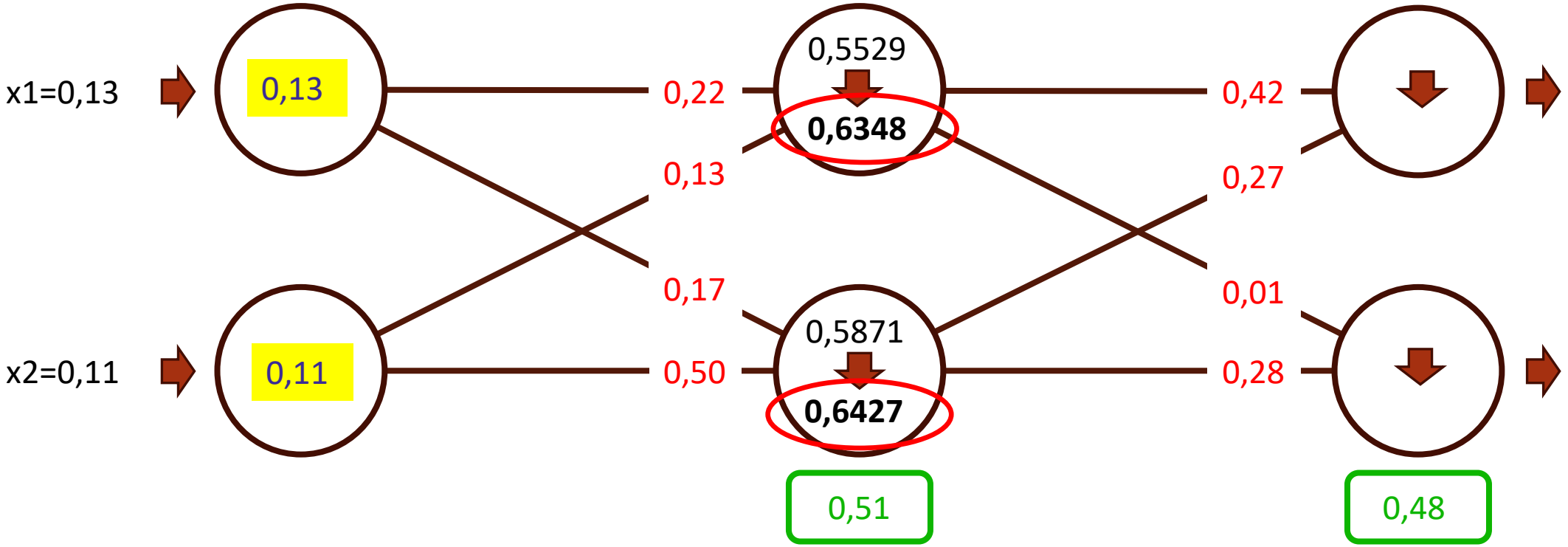
# Neuroverkon laskenta 2/10, neuronin arvon laskenta

- Neuronien arvot saadaan kertomalla edeltävien neuroneiden arvot vastaavien **synapsien painoarvoilla**, laskemalla saadut tulot yhteen ja lisäämällä tuloon neuronin kerrokseen liitetyn **bias- eli vakiotermin arvo**. Esimerkin välineuroneiden arvot saadaan siis seuraavasti:
  - Ylempi neuroni: 0,13*0,22+0,11*0,13+0,51 = **0,5529**
  - Alempi neuroni: 0,13*0,17+0,11*0,50+0,51 = **0,5871**



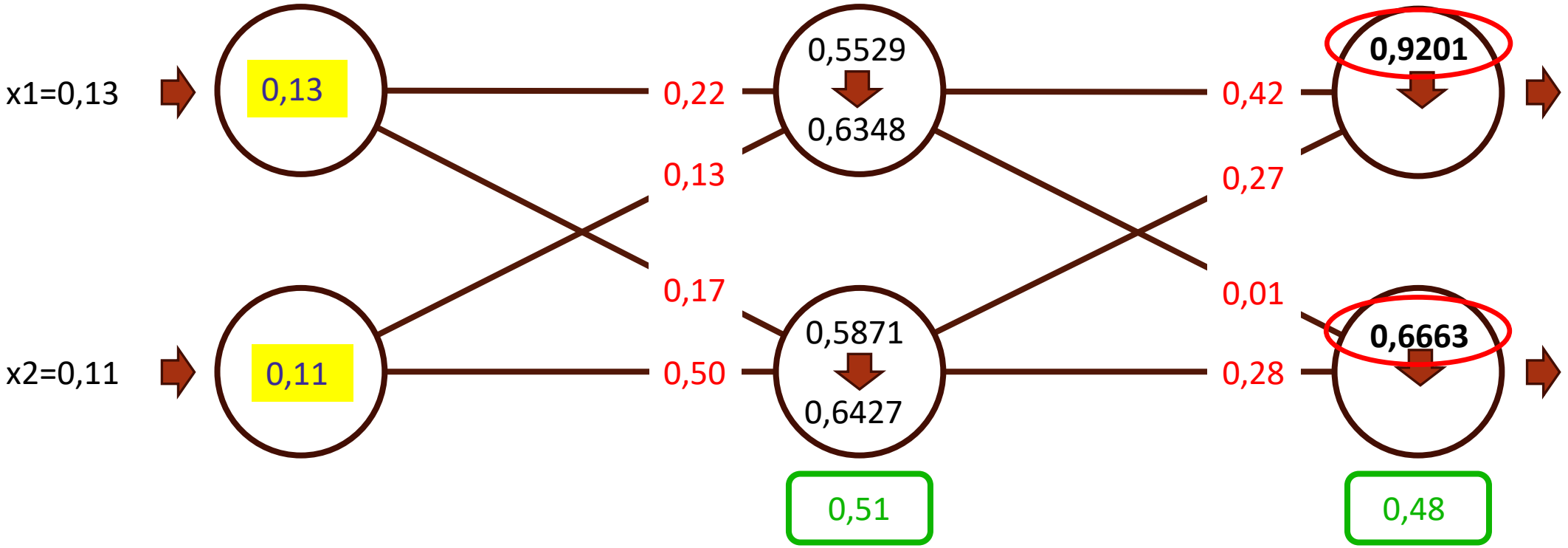
# Neuroverkon laskenta 3/10, aktivaatiofunktio ja neuronin ulostuloarvo

- Ennen kuin neuronin arvo viedään eteenpäin seuraavalle kerrokselle, se sijoitetaan **aktivaatio-funktioksi** kutsuttuun funktioon ja saatu aktivaatiofunktion arvo edustaa neuronin arvoa eteenpäin. Neuroverkoissa käytetään erilaisia aktivaatiofunktioita. Tässä esimerkissä aktivaatio-funktiona käytetään **sigmoid-funktiota** eli **1/(1+e^⁻x)**, joten välineuroneiden ulostuloarvot ovat:
  - Ylempi neuroni: 1/(1+e^⁻ 0,5529) = **0,6348**
  - Alempi neuroni: 1/(1+e^⁻0,5871) = **0,6427**



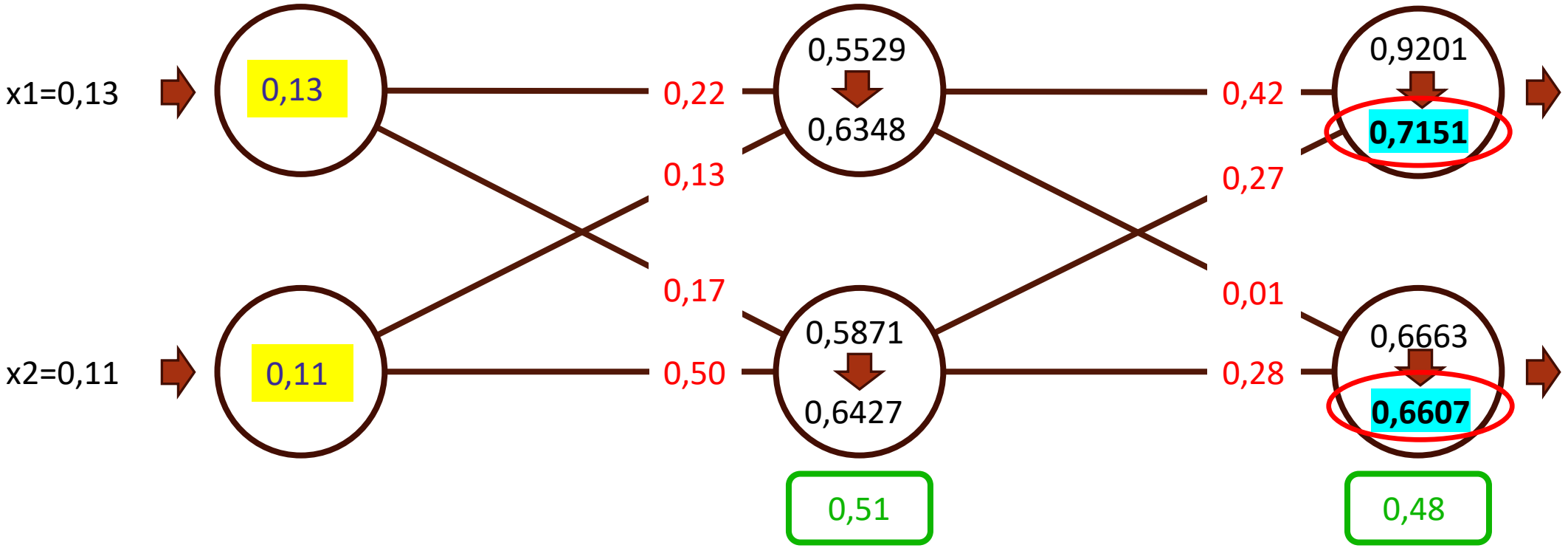
# Neuroverkon laskenta 4/10, laskenta jatkuu samalla tavalla …

- Neuronien ulostuloarvoja käytetään laskettaessa edelleen seuraavan kerroksen neuroneiden sisääntuloarvot:
  - Ylempi neuroni: 0,6348*0,42+0,6427*0,27+0,48 = **0,9201**
  - Alempi neuroni: 0,6348*0,01+0,6427*0,28+0,48 = **0,6663**



# Neuroverkon laskenta 5/10, laskenta jatkuu samalla tavalla …

- … ja neuronien lasketut sisääntuloarvot sijoitetaan taas aktivaatiofunktioon, jolloin saadaan tämän kerroksen neuroneiden ulostuloarvot:
  - Ylempi neuroni: 1/(1+e^⁻ 0,9201) = **0,7151**
  - Alempi neuroni: 1/(1+e^⁻0,6663) = **0,6607**

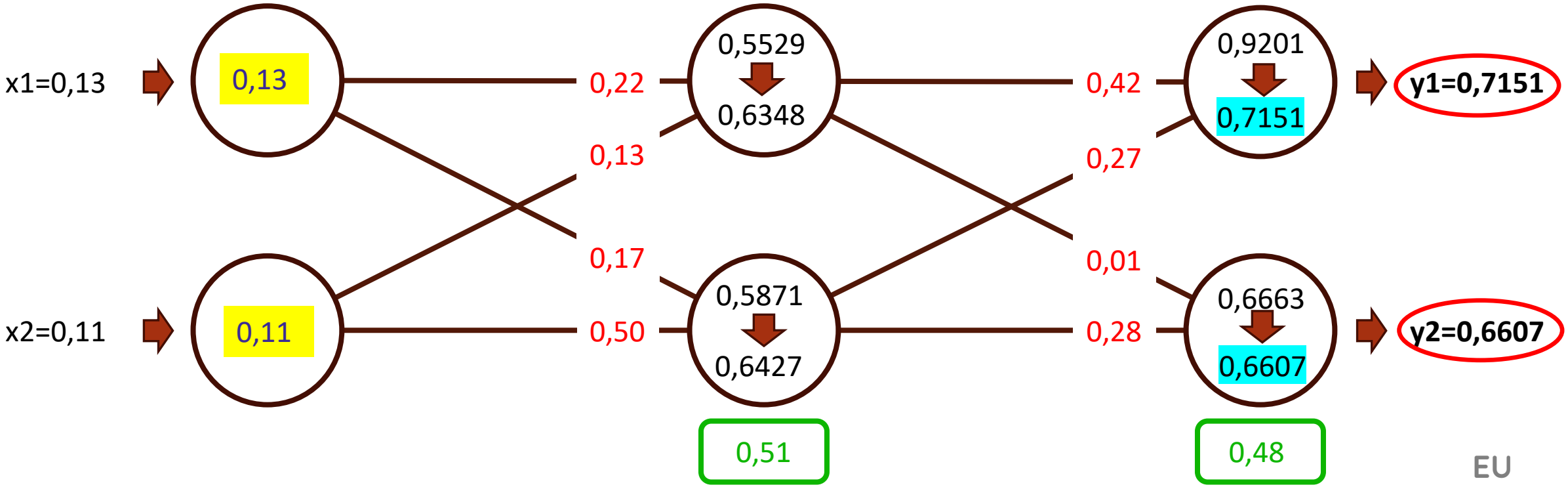


# Neuroverkon laskenta 6/10, laskennan lopputulos

- Laskentaa jatketaan samalla tavalla läpi koko neuroverkon, kunnes saadaan verkon laskemat ulostulosarvot, esimerkissä y1 ja y2.
- Neuroverkko on siis vain sarja laskutoimituksia, jotka voitaisiin esittää myös lausekemuodossa:

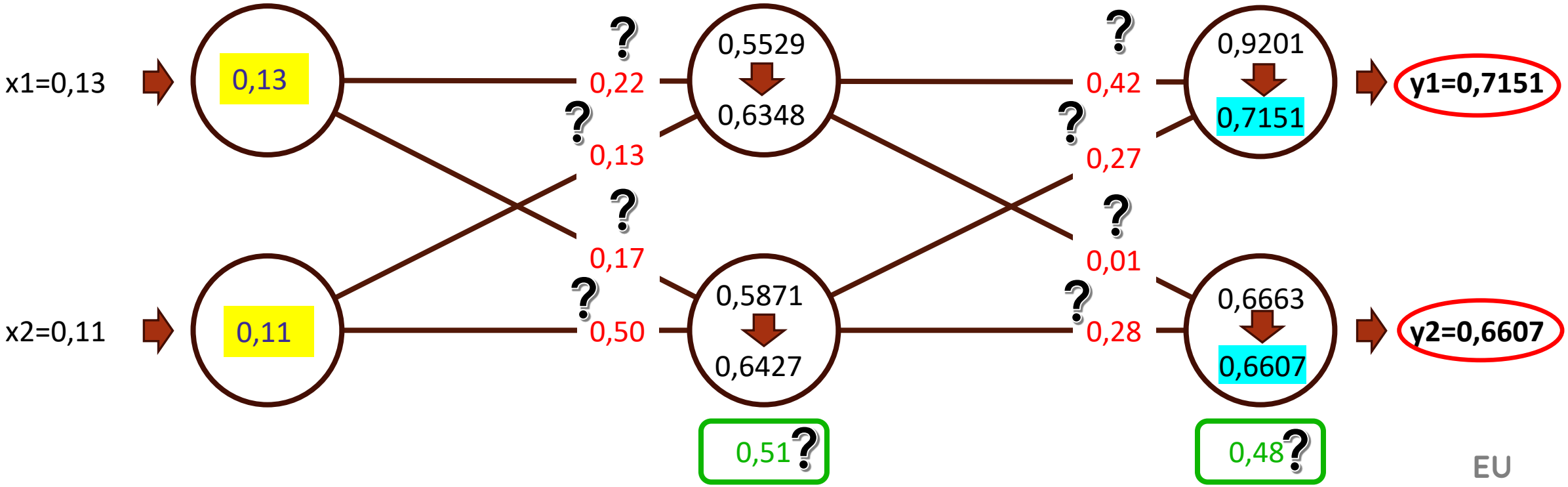
**y1**=1/(1+e^(-1*((1/(1+e^(-1*(**x1***0.22+**x2***0.13+0.51))))*0.42+(1/(1+e^(-1*(**x1***0.17+**x2***0.50+0.51))))*0.27+0.48)))

**y2**=1/(1+e^(-1*((1/(1+e^(-1*(**x1***0.22+**x2***0.13+0.51))))*0.01+(1/(1+e^(-1*(**x1***0.17+**x2***0.50+0.51))))*0.28+0.48)))



# Neuroverkon laskenta 7/10: mistä hyvät arvot laskentaparametreille?

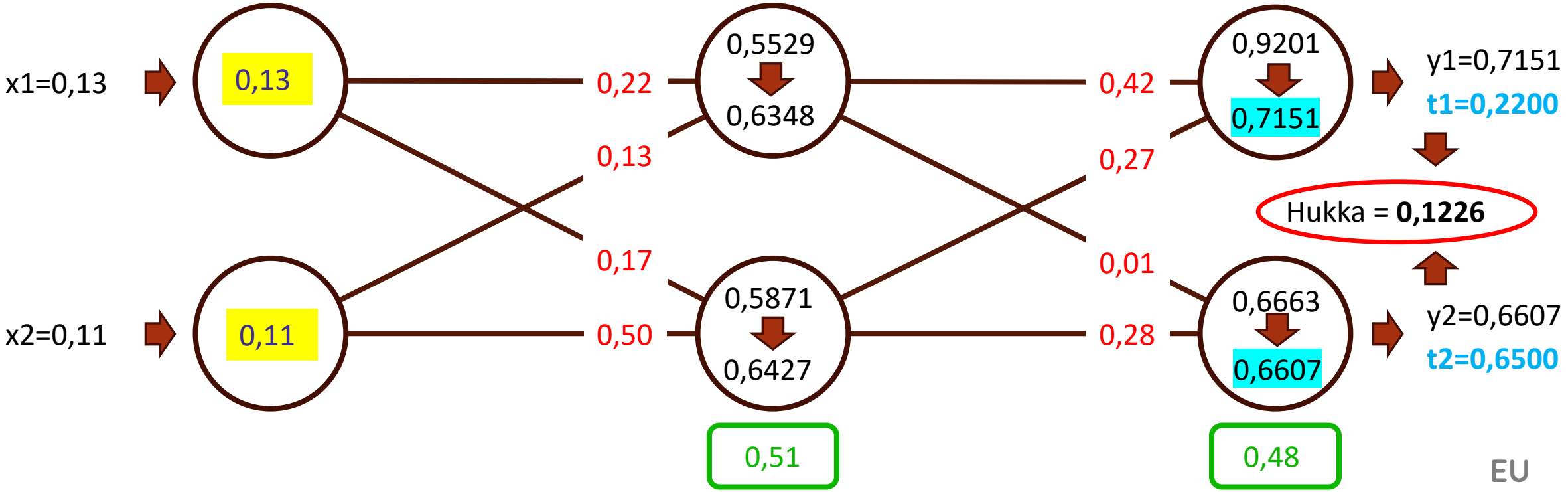
- Kuten huomaamme, määräytyy neuroverkon laskennan lopputulos laskentaparametrien arvoista. **Mistä siis saamme laskentaparametreille arvot, joilla neuroverkko antaa hyviä tuloksia?**
- Ensin pitää määrittää, mikä on hyvä tulos. Tämä taas perustuu **opetusaineistoon**, josta tiedetään sekä lähtötiedot että ennalta tunnetut tavoitetulokset. Hyvä tulos poikkeaa mahdollisimman vähän tunnetuista tavoitetuloksista. Opetusaineisto on siis malli, jonka mukaan neuroverkko rakennetaan.



# Neuroverkon laskenta 8/10, laskennan onnistumisen mittaaminen

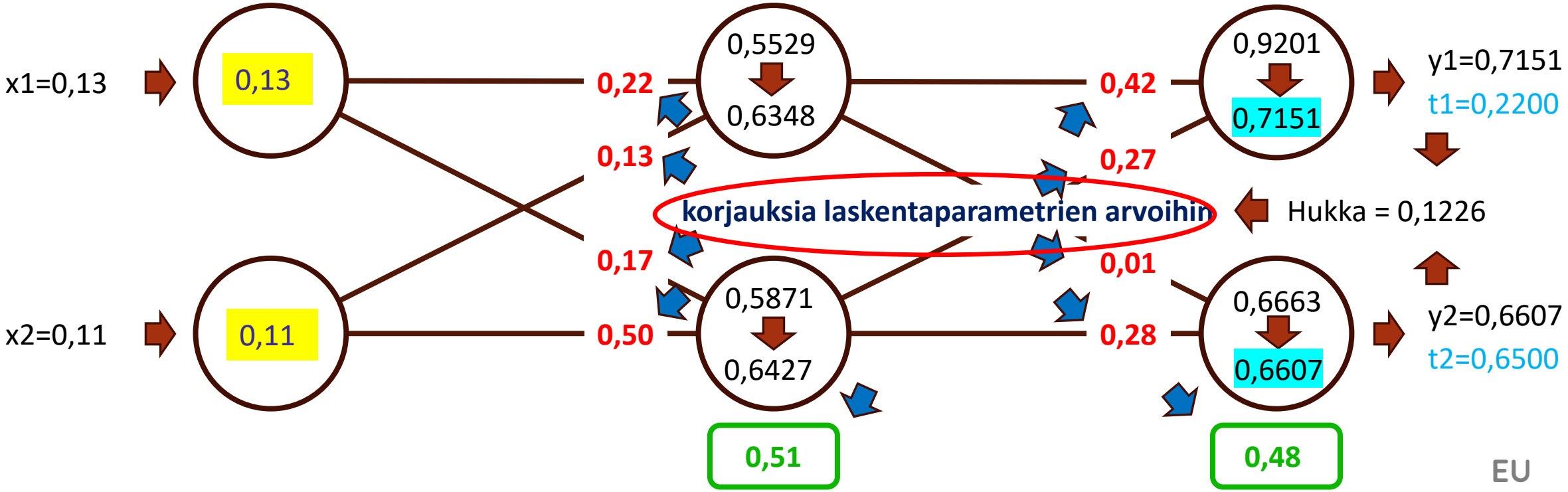
- Kun meillä on opetusaineisto, josta tiedetään sekä lähtötiedot että tavoitetulokset, voimme mitata laskennan onnistumista vertaamalla laskettuja tuloksia tunnettuihin tavoitetuloksiin. Tätä varten määritetään **virheen suuruutta kuvaava ns. hukkafunktio**, jona käytetään usein keskineliövirhettä eli virheiden neliöiden keskiarvoa. Esimerkin lopputuloksista (y1 ja y2) ja **tavoitetuloksista (t1 ja t2)** laskettu keskineliövirheen arvo on:

Hukka eli laskennan virhe = $\frac{1}{2}((y1-t1)^2+(y2-t2)^2) = \frac{1}{2}((0{,}7151-0{,}2200)^2+(0{,}6607-0{,}6500)^2)$ = **0,1226**



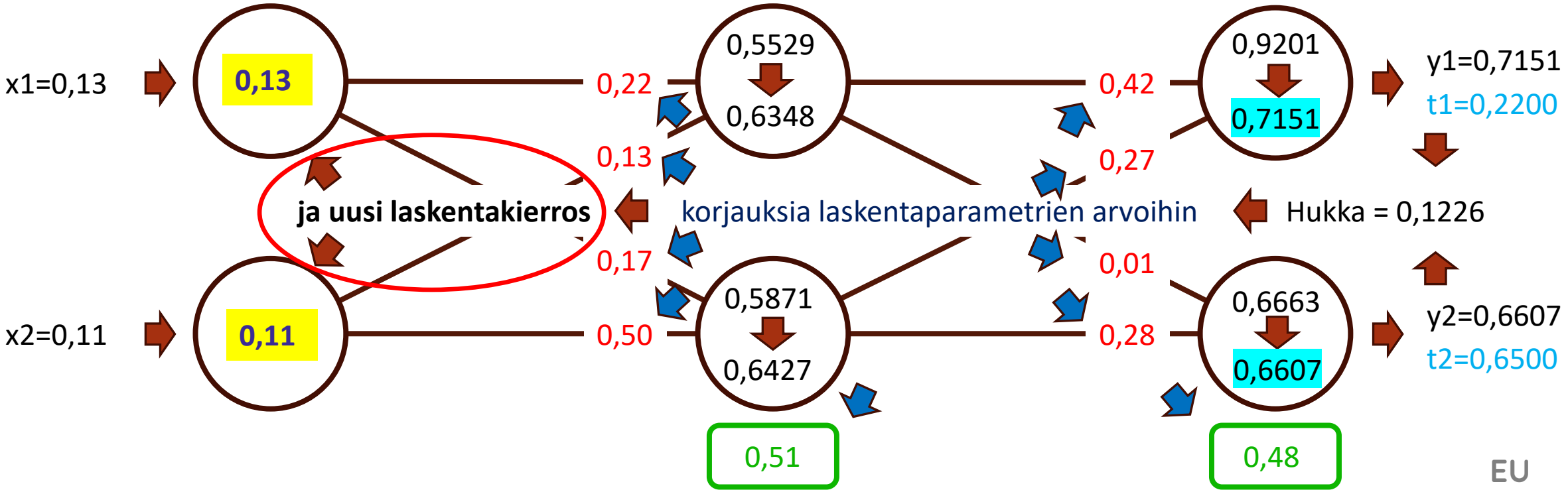
# Neuroverkon laskenta 9/10, verkon koulutus

- Kun pystymme mittaamaan laskennan onnistumista, voimme **kouluttaa neuroverkon tuottamaan mahdollisimman hyviä ja osuvia tuloksia.**
- Koulutus aloitetaan tutkimalla, mihin suuntaan neuroverkon laskentaparametreja tulisi muuttaa, jotta hukkafunktion arvo eli verkon laskennan virhe pienenisi ja muuttamalla arvoja pieni askel tuohon suuntaan.



# Neuroverkon laskenta 10/10, verkon koulutus

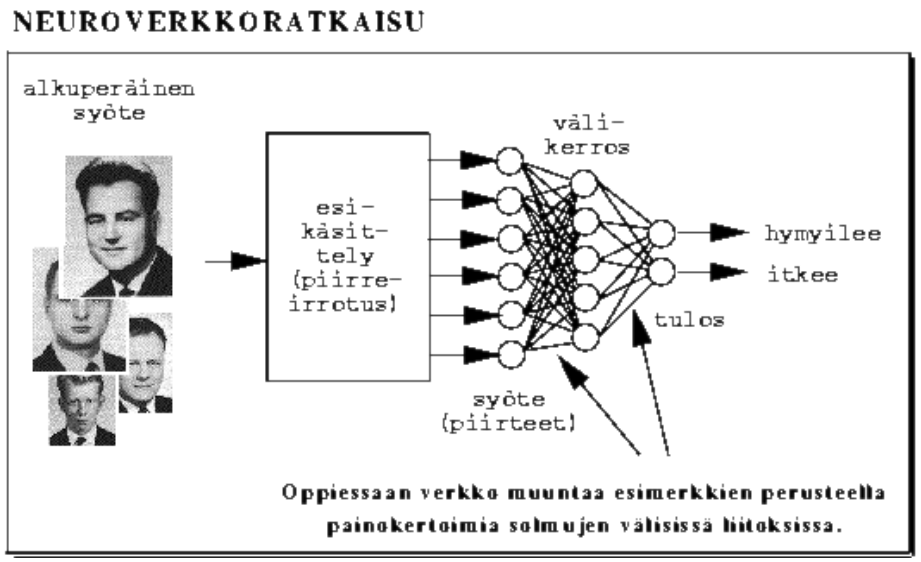
- Kun laskentaparametrien arvoja on muutettu pieni askel oikeaan suuntaan, **lasketaan hukkafunktion arvo uudelleen koko opetusaineistolle** tai sen satunnaisesti valitulle osajoukolle.
- Laskentaa toistetaan pienin muutosaskelin, kunnes hukkafunktion arvo on saatu riittävän pieneksi.
- **Lopputuloksena verkko tuottaa hyviä tuloksia myös täysin uudesta ja tuntemattomasta aineistosta, joka käyttäytyy kuten opetusaineisto.**



# Neuroverkon toiminnan vaiheet

- **Neuroverkon suunnittelu**:
  - Verkon tehtävän määrittely.
  - Verkon rakenne, käytetyt laskentakaavat ym.
  - Lähtö- ja tulostietojen valinta.
  - Määritys, kuinka verkon lähtötiedot muutetaan numeerisiksi ja numeeriset tulostiedot selväkielisiksi.
  - Opetus- ja testiaineiston valinta ja valmistelu.
- **Neuroverkon laskentaparametrien lähtöarvojen asetus:** Lähtöarvoiksi annetaan tyypillisesti täysin satunnaiset arvot, koska parempaa perustetta lähtöarvoille ei ole.
- **Koulutus:** Neuroverkon läpi viedään suuri määrä opetusaineistoa, josta tunnetaan sekä lähtötiedot että tavoitetulokset. Neuroverkko koulutetaan päivittämällä verkon laskentaparametreja matemaattisesti niin, että se tuottaa opetusaineistosta tuloksia, jotka poikkeavat mahdollisimman vähän tavoitelluista tuloksista.
- **Testaus:** Neuroverkkoa testataan aineistolla, josta tunnetaan sekä lähtötiedot että tavoitetulokset, mutta joka ei ollut mukana opetusaineistossa.
- **Käyttö:** Lopputuloksena neuroverkko osaa tuottaa hyviä tuloksia myös uudesta ja tuntemattomasta aineistosta edellyttäen, että se käyttäytyy kuten opetusaineisto.

*Tämän neuroverkkokuvan laati professori Timo Honkela vuonna 1996, ks. http://users.ics.aalto.fi/tho/stes/step96/honkela2/*



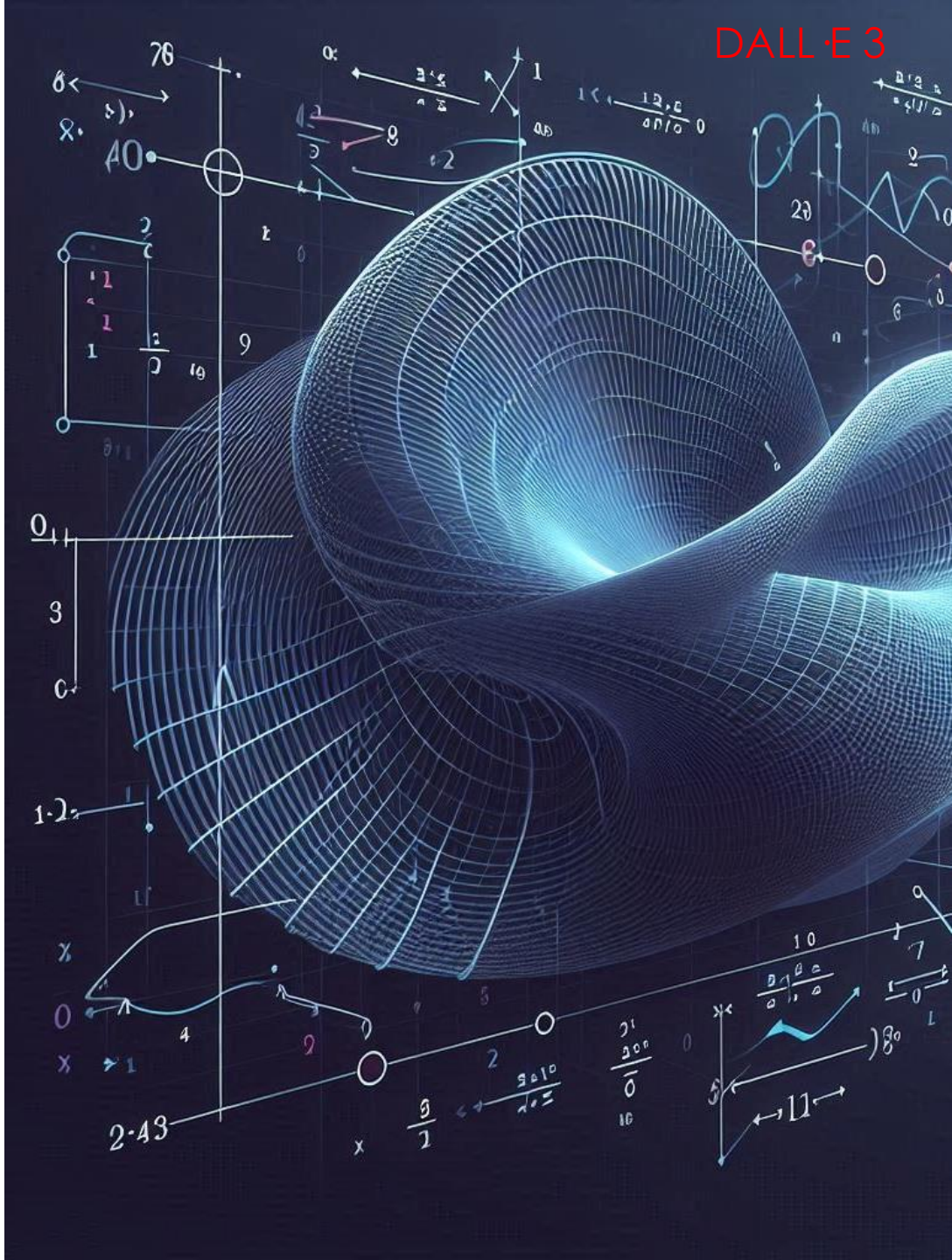

*Neuroverkkojen idea ei siis ole uusi, vaan nykyisen kaltaista neuroverkkotekniikkaa kehitettiin jo 1980-luvulla. Uutta on vain tietokoneiden tehokkuuden kasvu ja tietokoneilla luettavissa olevan opetusaineiston määrän valtava kasvu – vasta nämä mahdollistivat neuroverkkojen tehokkaan hyödyntämisen.*

# Esitelmän esimerkkineuroverkko vs todelliset neuroverkot

Käytimme esimerkissämme käytännön syistä pientä neuroverkkoa, jossa on vain kuusi neuronia ja kymmenen laskentaparametria. Todelliset neuroverkot ovat tyypillisesti huomattavasti suurempia ja voivat poiketa monilla muillakin tavoilla esimerkkinä käytetystä neuroverkosta. Esimerkkineuroverkon avulla oppimamme asiat pätevät kuitenkin myös yleisesti verkon koosta ja monista muista ominaisuuksista riippumatta.

Alla eräitä mahdollisia eroja:

- Esimerkkiverkko on pieni, vain kymmenen laskenta-parametrin verkko, mutta oikeat neuroverkot voivat olla jopa biljoonan parametrin kokoisia.
- Neuroverkoissa voidaan käyttää erilaisia aktivaatio-funktioita, hukkafunktioita ja muita laskentaa ohjaavia parametreja.
- Esimerkkiverkko on täysin kytketty, mutta niin ei ole kaikissa verkoissa.
- Esimerkkiverkko on vain eteenpäin kytketty, mutta verkko voi olla myös osin takaisinpäin kytketty.



# Neuroverkon tavat oppia

- Edellä kuvattiin opetusaineistoon ja sen tunnettuihin tavoitetuloksiin perustuva **ohjattu oppiminen (supervised learning)**. Opetusaineisto voi olla mikä tahansa aineisto, josta tunnetaan lähtötiedot ja tulostiedot. Esimerkiksi
  - joukko laboratoriotuloksia ja niiden perusteella tehtyjä varmistettuja diagnooseja.
  - joukko kuvia ja niihin liitettyjä selityksiä kuvan sisällöstä.
- Oppiminen voi jatkua myös neuroverkon käytönaikaisen palautteen perusteella (**vahvistettu oppiminen, reinforcement learning**). Palaute voi olla mitä tahansa, mikä kertoo verkon antaman tuloksen laadusta tai osuvuudesta. Esimerkiksi
  - ihmisen antama palaute siitä, kuinka hyvin tekoälyn luoma kuva vastaa tiettyjä ennalta annettuja kriteerejä.
  - tieto robotin törmäämisestä esteeseen.
- Oppiminen voi tapahtua myös siten, algoritmi etsii itsenäisesti opetusaineistosta säännönmukaisuuksia ja luokittelee sitä (**ohjaamaton oppiminen, unsupervised learning**).