C. Neironu tīklu uzbūve un darbība

Neironu tīkls ir skaitļošanas sistēma, kas sastāv no liela daudzuma skaitļošanas elementu, sauktu par neironiem, kas noteiktā veidā savienoti viens ar otru, un kopā veic kādu noteiktu uzdevumu. Atšķirībā no tradicionālajām skaitļošanas sistēmām, neironu tīkls ir drīzāk līdzīgs melnajai kastei – katra atsevišķa neirona loma kāda konkrēta uzdevuma risināšanā nav precīzi nosakāma. Šī neironu tīklu īpašība krietni apgrūtina to konstruēšanu, testēšanu un ekspluatēšanu, bet tajā pašā laikā rada papildus intrigu neironu tīklu izzināšanā.

Ir daudz dažādu neironu tīklu modeļu, tomēr noteiktā abstrakcijas līmenī to uzbūve ir līdzīga. Trīs galvenie neironu tīkla konstruktīvie bloki ir:

1. Neirons;
2. Tīkla topoloģija;
3. Apmācības algoritms.

C1. NEIRONS UN TĀ DARBINĀŠANA.

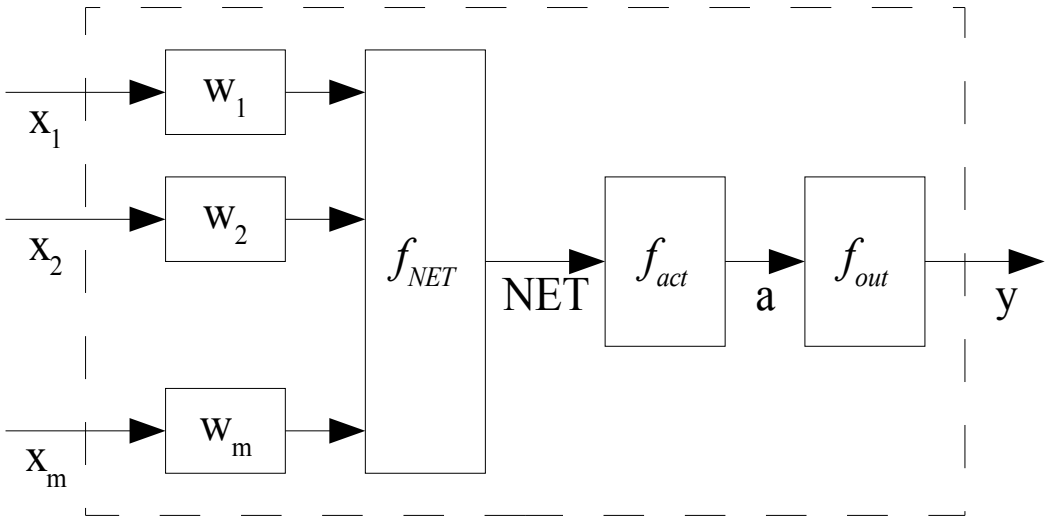
C1.1. VISPĀRĪGS NEIRONA APRAKSTS.

Neirons (*Neuron, Processing Unit*) ir salīdzinoši vienkāršs skaitļošanas elements, kam ir vairākas ieejas (*inputs*) un tikai viena izeja (*output*). Matemātiski neirons atgādina daudzargumentu funkciju.

Neirons ietver sevī šādus konstruktīvos elementus (sk. arī attēlu C1):

1. (Sinaptiskie) svari (*[Synaptic] weights*),
2. Summēšanas (izplatīšanās) funkcija (*Propagation function*),
3. Aktivizācijas funkcija (*Activation function*),
4. Izejas funkcija (*Output function*),
5. Apmācības likums (*Learning rule*).

Šajā nodaļā netiks aprakstīts apmācības likums, kas tiks vispārīgākā veidā darīts nodaļā C3.



Attēls C1. Neirona vienkāršota modelis ar m ieejām.

Neirona darbināšana.

Uz neirona ieejām tiek padotas vērtības. Summēšanas funkcija, izmantojot neirona svarus, visas ieejas sakombinē vienā vērtībā, kas tālā tiek apstrādāta, izmantojot aktivizācijas funkciju un izejas funkciju.

C1.2. NEIRONA KOMPONENTES.

Svari (W*eights*).

Svari ir (skaitliskas) vērtības, kas veido katra neirona un arī visa neironu tīkla galveno atmiņas daļu, kas nodrošina to, ka neironu tīkls funkcionē tā, kā tas funkcionē un kuru **vērtības tiek uzstādītas apmācības procesa ceļā**. Svari kopā ar citām šādām vērtībām dažreiz tiek saukti par neironu tīkla parametriem. Parasti katram neirona ievadam (ienākošajam signālam) atbilst viens svars. Ienākošie signāli (*inputs*), pateicoties atbilstošajām svaru vērtībām, tiek izmainīti un kombinēti, ar ko nodarbojas **summēšanas funkcija**. Aprakstot neironu darbību, svari parasti tiek apzīmēti ar burtu w un indeksu (svara numuru), piemēram, w_i. Ieejas signālus bieži apzīmē ar burtu x un indeksu (vispārīgākā gadījumā tiek izmantots arī burts o), piemēram x_i vai o_i.

Summēšanas funkcija (*Propagation function*).

Summēšanas funkcija, kombinējot ieejas signālus un svarus, izrēķina **vienu** vērtību, kas tālāk piedalās neirona izejas vērtības izskaitļošanā. Summēšanas funkcijas izrēķināto

vērtību parasti apzīmē ar burtu virkni NET, bet pašu summēšanas funkciju kā f_{NET} vai arī vienkārši kā NET.

Vistipiskākā summēšanas funkcija parādīta formulā C1 – ieejas signāli tiek reizināti ar attiecīgajiem svariem un visi šie reizinājumi tiek summēti.

$$NET = \sum_{i=1}^{m} w_i x_i \tag{C1}$$

kur *NET – neirona summēšanas vērtība,*

w_i *– neirona i-tais svars,*

x_i *– neirona i-tā ieeja.*

Tiek izmantotas arī citas līdzīgas summēšanas funkcijas (C1a, C1b, C1c).

$$NET = \prod_{i=1}^{m} w_i x_i \tag{C1a}$$

$$NET = max(w_i x_i); i = 1..m \tag{C1b}$$

$$NET = min(w_i x_i); i = 1..m \tag{C1c}$$

Par summēšanas funkciju dažos modeļos arī Eiklīda attālumu (C1d) vai tā kvadrātu:

$$NET = \|W - X\| = \sqrt{\sum_{i=1}^{m} (w_i - x_i)^2} \tag{C1d}$$

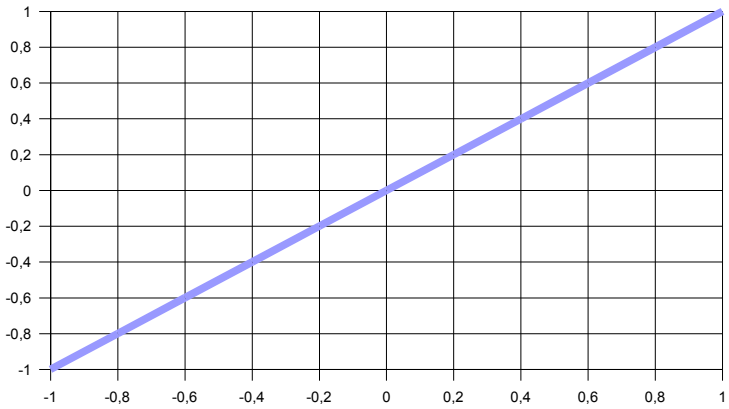
Aktivizācijas funkcija (*Activation function*).

Aktivizācijas funkcija, izmantojot summēšanas funkcijas vērtību NET, izrēķina izrēķina t.s. aktivitātes stāvokli (*Activation state*). Aktivitātes stāvokli parasti apzīmē ar burtu a, bet pašu aktivitātes funkciju kā f_{act}. Aktivizācijas funkcijai ir ļoti liela ietekme uz neirona skaitļošanas spēju. Aktivizācijas funkcija var būt lineāra, vai nelineāra. Starp nelineārām funkcijām izšķir sigmoidālās un sliekšņveida.

Formulā C2 parādīts lineāras aktivizācijas funkcijas triviālais variants.

$$f_{lin}(NET) = NET \tag{C2}$$

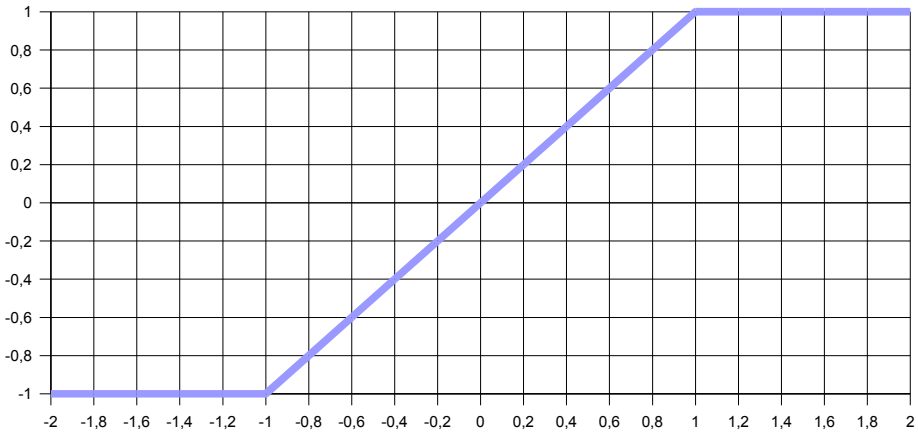
kur *NET – neirona summēšanas funkcija.*



Attēls C1. Lineāra aktivizācijas funkcija.

Formulā C2a parādīts lineāras aktivizācijas funkcijas variants, ierobežojot vērtības intervālā [A..B].

$$f_{lin2}(NET)=\left\{\begin{array}{c} A\,;\,NET<A \\ B\,;\,NET_j>B \\ NET\,;\,NET_j\in[A..B] \end{array}\right\} \qquad \text{(C2a)}$$

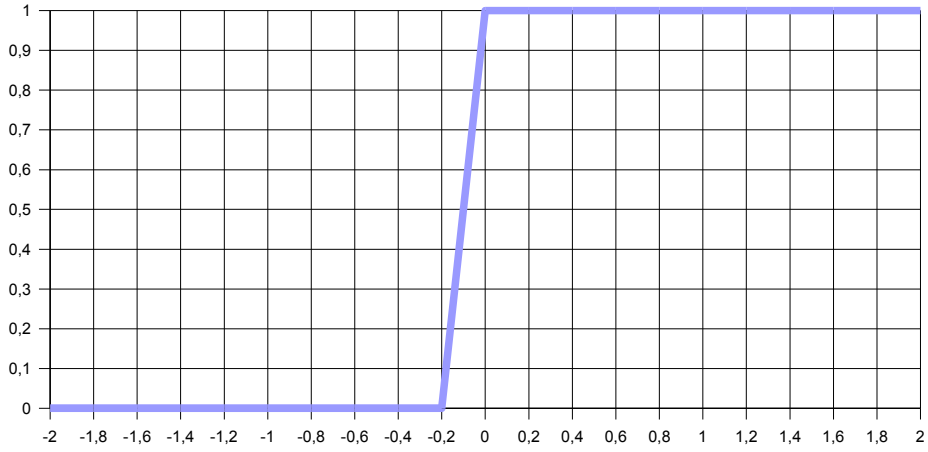


Attēls C2. Lineāra aktivizācijas funkcija ar ierobežojumu intervālā. A=-1, B=1.

Formulā P2d parādīta sliekšņveida aktivizācijas funkcija.

$$f_{\Theta}(NET)=\left\{\begin{array}{c} 0\,;\,NET\leqslant\Theta \\ +1\,;\,NET>\Theta \end{array}\right\} \qquad \text{(C2b)}$$

kur Θ *- sliekšņa vērtība.*

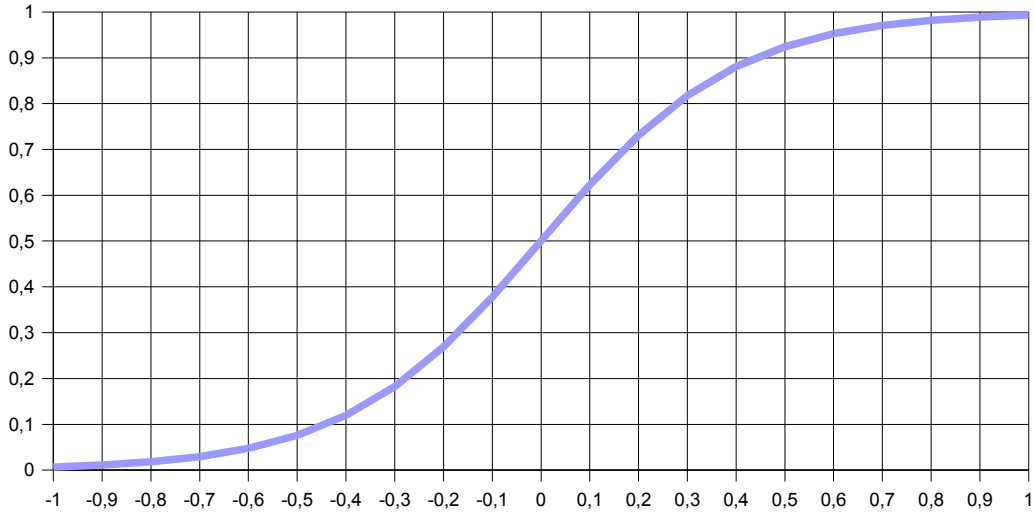


Attēls C3. Sliekšņveida aktivizācijas funkcija. $\Theta = 0$.

Starp sigmoidālajām aktivizācijas funkcijām vispopulārākā ir t.s. loģistiskā aktivizācijas funkcija (C2c), bet tiek lietots arī hiperboliskais tangenss (C2d).

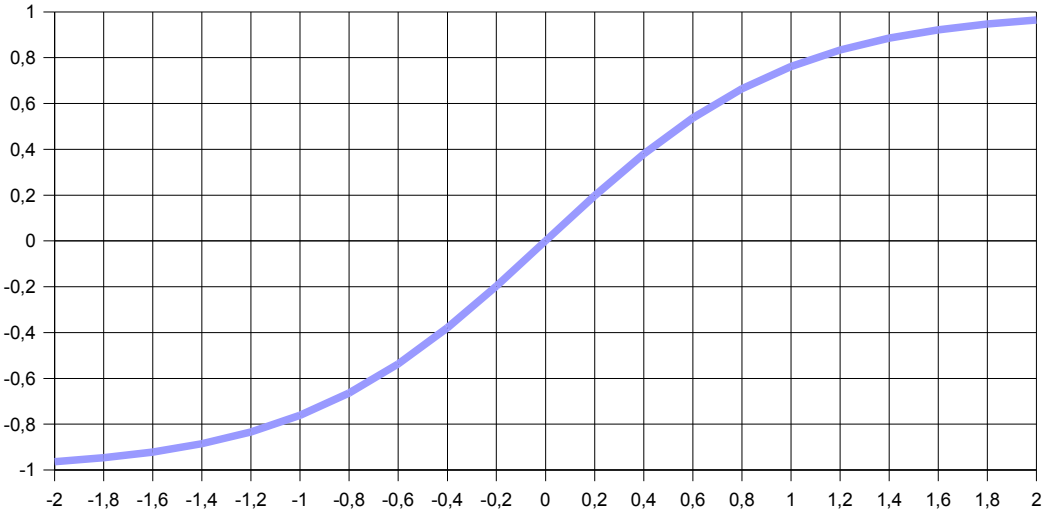
$$f_{\log}(NET) = \frac{1}{1 + e^{-\frac{1}{g} NET}} \qquad \text{(C2c)}$$

kur g *– līknes slīpuma koeficients (gain).*



Attēls C4. Loģistiskā aktivizācijas funkcija. g=0.2.

$$f_{\tanh}(NET) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \qquad \text{(C2d)}$$



Attēls C5. Hiperboliskais tangenss.

Sigmoidālo funkciju ļoti vērtīga īpašība ir tāda, ka to atvasinājums ir izsakāms ar pašas funkcijas vērtību, kas ir ļoti nozīmīgi tāda neironu tīklu modeļa, kā perceptrons apmācībā. Sigmoidālo funkciju atvasinājumi doti formulās C2e un C2f.

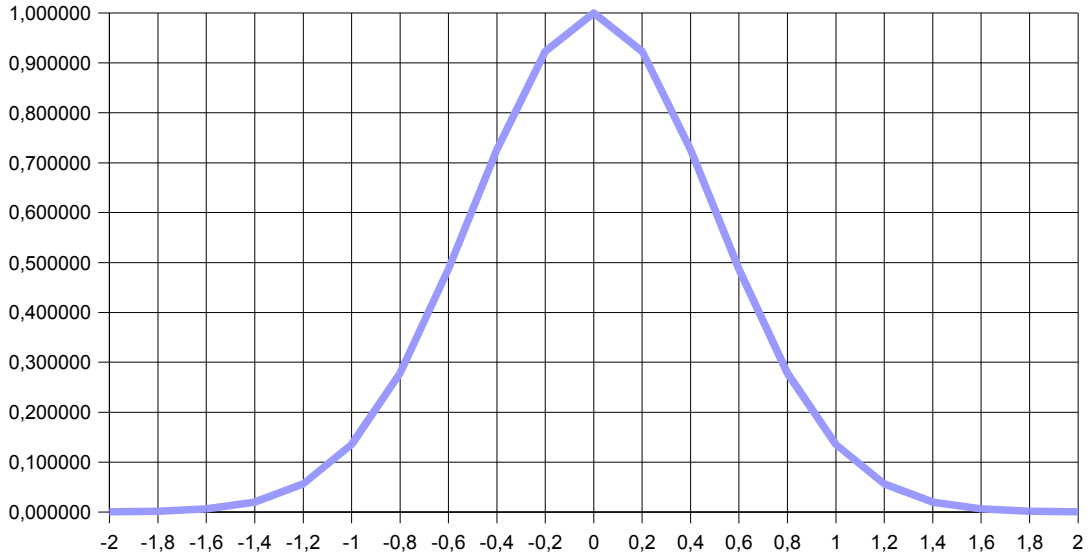
$$f'_{\log} = f_{\log}(1 - f_{\log}) \qquad \text{(C2e)}$$

$$f'_{\tanh} = 1 - f^2_{\tanh} \qquad \text{(C2f)}$$

Starp lietotajām aktivizācijas funkcijām ir arī Gausa funkcija C2g.

$$f_{Gauss} = e^{-\frac{NET^2}{2\sigma^2}} \qquad \text{(C2g)}$$

kur σ *(sigma) – līknes slīpuma (liekuma) koeficients.*



Attēls C6. Gausa funkcija. $\sigma = 0.5$.

Izejas funkcija (*Output function*).

Izejas funkcija, izmantojot aktivitātes stāvokli a, izrēķina neirona izejas vērtību (*output*). Izeju parasti apzīmē ar burtu o vai y, bet pašu izejas funkciju kā f_{out}. Kā redzams, apzīmējums o tiek lietots arī neirona ieejas apzīmēšanā. Tas tāpēc, ka, pastāvot savienojumiem starp dažādiem neironiem, viena neirona izeja var būt cita neirona ieeja.

Lielākajā daļā neironu modeļu izejas funkcija netiek atdalīta no aktivizācijas funkcijas, kas nozīmē, ka aktivizācijas funkcijas f_{act} rezultāts ir nevis tikai aktivitātes stāvoklis, bet arī gala vērtība – neirona izeja.

C1.3. VIENKĀRŠA NEIRONA PIEMĒRS.

Darba uzdevums.

Uzkonstruēt neironu, kas modelē divargumentu loģisko funkciju "UN" un demonstrē atpazīšanu (TRUE apzīmēsim ar 1, bet FALSE ar 0).

Neirona konstruēšana.

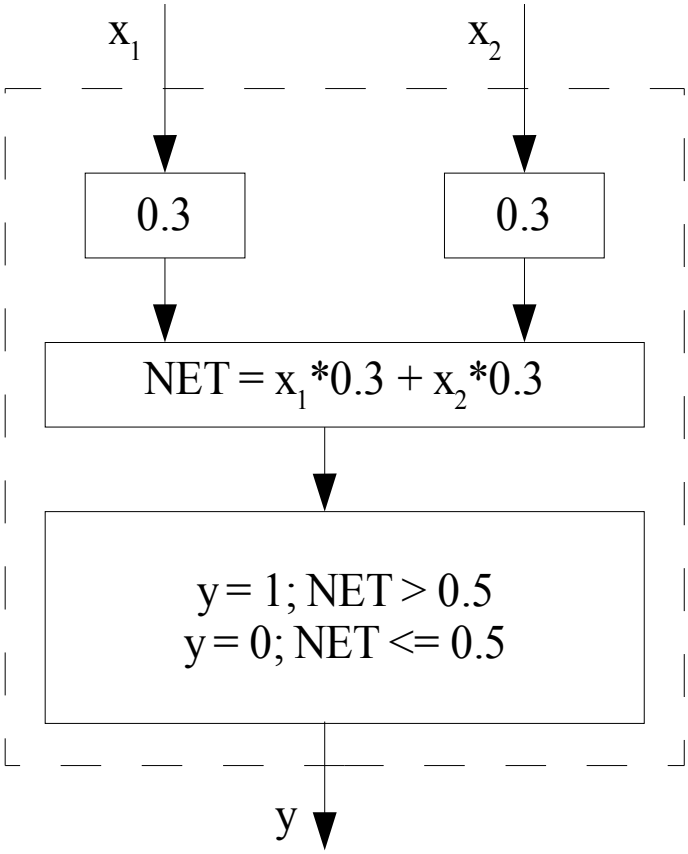
Neironam ir divas ieejas (tātad, arī divi svari w_1 un w_2) un viena izeja.

Par summēšanas funkciju lietosim C1: $NET = x_1w_1+x_2w_2$.

Par aktivizācijas un izejas funkciju lietosim C2b ($\Theta = 0.5$):

$$y = \begin{Bmatrix} 0\,;\, NET \leqslant 0.5 \\ +1\,;\, NET > 0.5 \end{Bmatrix}$$

Aizpildīsim svaru vērtības ar $w_1=w_2=0.3$.



Attēls C7. Neirons, kas atpazīst loģisko "UN".

Tabula C1. Attēlā C7 aprakstītā neirona pārbaude.

x_1	x_2	NET	y
0	0	0	0
0	1	0.3	0
1	0	0.3	0
1	1	0.6	1

C1.4. VIENKĀRŠA NEIRONA PIEMĒRS AR PAPILDUS SVARU.

Darba uzdevums.

Uzkonstruēt neironu, kas modelē divargumentu loģisko funkciju, kas pretēja "UN" (NOT (A AND B)) un demonstrē atpazīšanu (TRUE apzīmēsim ar 1, bet FALSE ar 0). Izmantojot iepriekšējā piemērā neironu šādu funkciju nevar nomodelēt, jo šī funkcija prasa 1, ja x_1 un $x_2 = 0$, bet ja abas ieejas ir 0, tad arī NET = 0 un y=0, neatkarīgi no svariem. Tāpēc tiek lietots papildus svars b un tiek nedaudz pamainīta summēšanas funkcija.

Neirona konstruēšana.

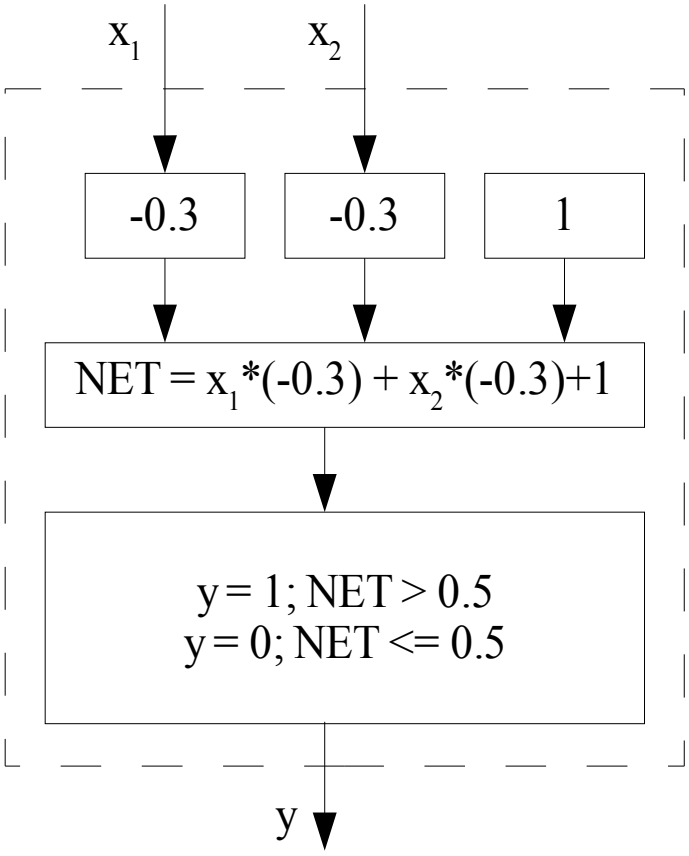
Neironam ir divas ieejas (tātad, arī divi īstie svari w_1 un w_2, kā viens papildus svars b) un viena izeja.

Par summēšanas funkciju lietosim C1: $NET = x_1w_1+x_2w_2+b$.

Par aktivizācijas un izejas funkciju lietosim C2b ($\Theta = 0.5$):

$$y = \begin{cases} 0\,; NET \leqslant 0.5 \\ +1\,; NET > 0.5 \end{cases}$$

Aizpildīsim svaru vērtības ar $w_1=w_2=-0.3$ un b=1.



Attēls C8. Neirons, kas atpazīst NOT (A AND B).

Tabula C2. Attēlā C8 aprakstītā neirona pārbaude.

x_1	x_2	NET	y
0	0	1	1
0	1	0.7	1
1	0	0.7	1
1	1	0.4	0

Varam pārliecināties, ka nav iespējams uzkonstruēt tādu neironu, kas spētu modelēt loģisko funkciju XOR (izslēdzošais VAI). Par to sk. nodaļā C4, kur attēlā C12 parādīts, kā ar 3 neironu palīdzību 2 slāņos var modelēt XOR.

C2. NEIRONU TĪKLA TOPOLOĢIJA.

Neironu tīkla topoloģija jeb arhitektūra ir neironu izvietojums tīklā un to savstarpējās saites.

Kā jau iepriekš tika rakstīts, neironi ir samērā vienkārši skaitļošanas elementi. Neironu tīkla spēku nodrošina tas, ka neironi darbojas kopā, būdami sasaistīti viens ar otru.

Neironu tīklu, līdzīgi kā grafu var definēt neironu un saišu starp neironiem kopumu:

$T=\{N,S\}$; $N=\{n_1, n_2, ..., N_m\}$; $S=\{s_1, s_2, ..., s_p\}$; $S \subset N \times N$

Parasti neironi neironu tīklā tiek grupēti t.s. slāņos (*layers*) un neironu darbināšanu veic pa slāņiem.

Neironu tīkla topoloģiju nosaka:

1. Neironu slāņu izkārtojums vienam aiz otra;
2. Neironu sasaiste starp slāņiem;
3. Neironu sasaiste (sadarbība) slāņa ietvaros.

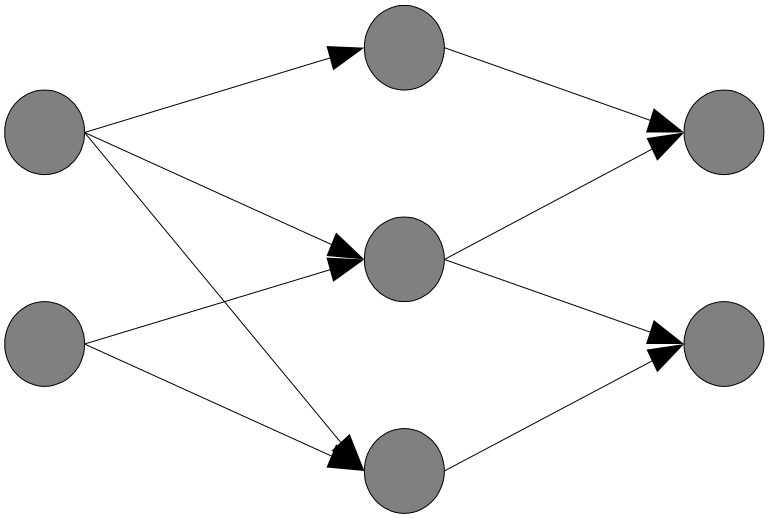
Pēc neironu slāņu izkārtojuma neironu tīkli iedalās divos galvenajos veidos:

1. Vienvirziena tīkli (*Feedforward Networks*);
2. Tīkli ar atgriezeniskajām saitēm (*Feedback Networks*).

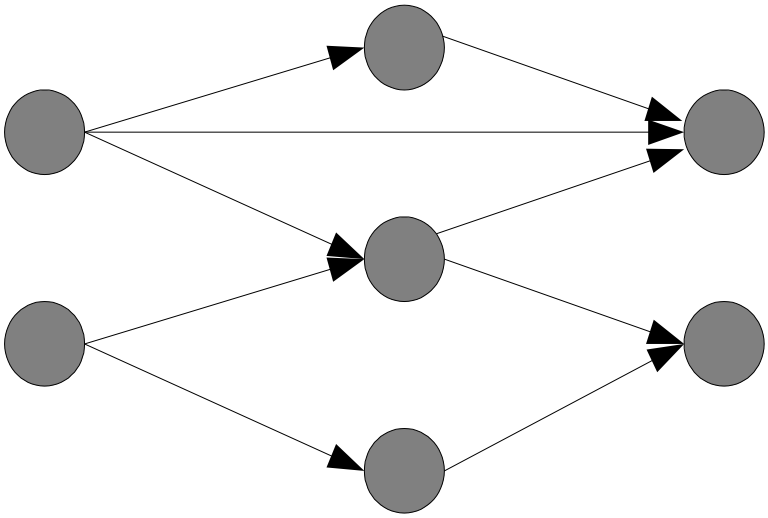
Vienvirziena tīkls.

Vienvirziena tīklos slāņi ir izkārtoti virknē pēc kārtas, līdz ar to tos var numurēt pēc kārtas. Tīkla darbināšana notiek, sākot ar ieejas slāni (nr. 1) un beidzot ar izejas slāni (nr. m). Saites starp neironiem ir tikai virzienā no slāņa ar mazāku numuru uz slāni ar lielāku numuru.

Attēlā C9a redzams t.s. 1. kārtas vienvirziena tīkls, kurā savienoti tikai neironi starp blakus slāņiem, bet attēlā C9b – 2. kārtas vienvirziena tīkls, kurā iespējami arī citi varianti.



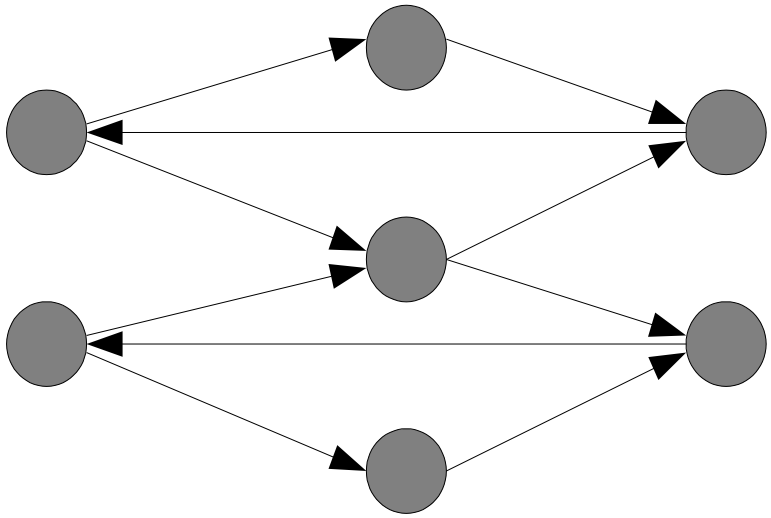
Attēls C9a. Pirmās kārtas vienvirziena neironu tīkls (saites tikai starp blakus slāņiem).



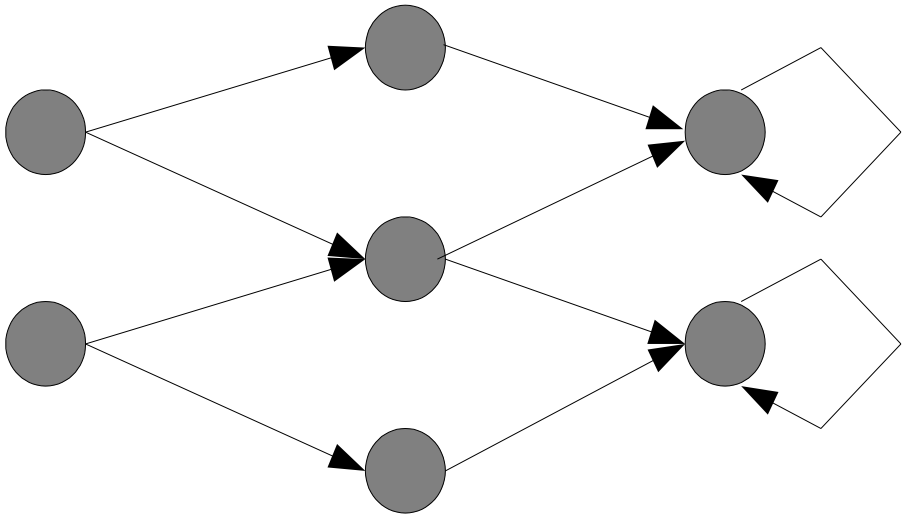
Attēls C9b. Otrās kārtas vienvirziena neironu tīkls (saites ne tikai starp blakus slāņiem).

Tīkli ar atgriezeniskajām saitēm.

Tīkli ar atgriezeniskajām saitēm pieļauj saites no slāņiem ar lielāku numuru uz slāņiem ar mazāku numuru, kā arī slānim pašam uz sevi (attēli C10a, C10b).



Attēls C10a. Tīkls ar atgriezeniskām saitēm.



Attēls C10b. Tīkls ar saitēm uz sevi.

Neironu sasaiste starp slāņiem.

Parasti starp blakus esošajiem slāņiem neironi ir sakārtoti katrs ar katru. Tādējādi neironam parasti ir tik daudz svaru, cik iepriekšējā slānī neironu (katrai neirona ieejai pretī ir svars).

Neironu sadarbība slāņa ietvaros.

Dažos neironu tīklu modeļos (piemēram, Kohonena tīkls) ir nozīme neironu izvietojumam un sadarbībai viena slāņa ietvaros. Var tikt noteikta topoloģija (specifisks neironu izkārtojums) slāņa ietvaros un neironu darbināšanas un apmācības algoritmos var parādīties blakus neironu ietekme.

Ieejas, izejas un slēptie slāņi.

Slāni, kuram tiešā veidā var padot vērtības sauc par ieejas slāni (*Input Layer*), bet slāni, kura izejas ir arī visa neironu tīkla izejas, sauc par izejas slāni (*Output Layers*). Visi pārējie slāņi (tādi, kuriem nav tiešas sasaistes ar apkārtni) tiek saukti par slēptajiem slāņiem (*Hidden Layers*).

Neīsto jeb pseido slāņu izmantošana.

Ja neironi ir savienoti viens ar otru, tad viena neirona izeja var būt cita neirona ieeja. Bet kā ir ar ieejas neironam (tie, kas ir pašā sākumā) – tiem būtu jāievieš speciāls jēdziens "ieeja" – respektīvi, vieta, kur uzstādīt ieejas vērtības. Lai tehniski būtu vieglāk realizēt neironu tīklus, bieži tiek ieviesti **speciāli ieejas slāņi**, kuru neironiem nav neviena svara, bet ir tikai izejas, kuras nepieciešamas tikai neironu tīkla ieejas vērtību uzstādīšanai.

Šādus slāņus varētu nosaukt par neīstajiem jeb pseido slāņiem. Pseido slāņu dēļ bieži vien ir neviennozīmīgi teikt, cik slāņu ir tīklā – vai tas domāts ieskaitot pseido slāni vai nē. Piemēram, ja tīklā ir pseido slānis un vēl divi slāņi – vai uzskatīt, ka tīklā ir 2 vai 3 slāņi. Dažādi autori izvēlas kādu no šiem variantiem, bet noteiktas vienošanās nav, tāpēc vēl viens variants, kā iziet no situācijas, ir nosaukt, cik slēpto slāņu ir tīklā.

Šajā materiālā pseido slānis tiek pieskaitīts kopējā slāņu skaitam.

C3. APMĀCĪBAS ALGORITMS.

Apmācības (*Training*) algoritms ir ļoti svarīga neironu tīkla uzbūves sastāvdaļa, un katram neironu tīklu modelim tas ir atšķirīgs, tomēr var nosaukt vairākas pamatstratēģijas un pamatprincipus, kuras tiešāk vai netiešāk tiek pielietotas vairuma neironu tīklu apmācībā.

C3.1. APMĀCĪBAS MĒRĶI UN UZDEVUMI.

Runājot par neironu tīklu apmācību, būtu jāizšķir divi saistīti, bet atšķirīgu līmeņu jēdzieni – apmācības mērķi un apmācības uzdevumi.

1. Neironu tīkla **apmācības mērķis** – (neironu tīklam) iegūt zināmu spēju risināt noteiktus uzdevumus.
2. Neironu **apmācības** tiešie **uzdevumi** – neironam vei neironu grupai iegūt zināmu spēju, lai neironu tīkls kopumā spētu risināt noteiktus uzdevumus.

Apmācības process ir lokāli globāls, jo mērķi ir globāli (visa neironu tīkla spējas iegūšana), bet līdzekļi to sasniegšanai ir lokāli – atsevišķu neironu vai neironu grupas apmācība.

Nav tiešas saites starp lokālo un globālo apmācības aspektu, jo tāda jau ir paša neironu tīkla būtība, ka:

1. No vienas puses – visi neironi potenciāli piedalās visu problēmu risināšanā;
2. No otras puses – vispārīgā gadījumā principā nav definējama katra atsevišķa neirona loma katras atsevišķās problēmas risināšanā.

Bez tam ne visos neironu tīklu modeļos var viegli novērtēt, vai ir sasniegti apmācības mērķi.

Ja par neironu tīklu ideālu uzskata cilvēka smadzenes, tad var uzskatīt, ka neironu tīklu izpēte vēl ir ļoti zemā līmenī, un to raksturo salīdzinoši zemie apmācības mērķi, kas bieži vien sakrīt ar tiešajiem uzdevumiem. Piemērs. Daudzu neironu tīklu modeļu apmācības mērķis ir noteiktas funkcijas aproksimācija. Lielākajā daļā šo modeļu ir spēkā apgalvojums, ka arī katra atsevišķa neirona tiešais uzdevums ir kādas funkcijas aproksimācija.

Vēl viena lieta, kas raksturo salīdzinoši zemo neironu tīklu attīstības līmeni, ir to pielietojuma pakāpe: neironu tīklu moduļiem lietojumprogrammās parasti ir palīgloma, jo vadību un kontroli veic tradicionāli izveidoti programmas bloki, bet neironu tīkli tiek izmantoti tikai atsevišķu palīgoperāciju izpildei.

Par **neironu tīklu apmācības stratēģijām** varētu nosaukt lokālu mehānismu un principu pielietošanu, lai nodrošinātu (globālos) apmācības mērķus.

C3.2. DIVAS NEIRONU TĪKLU APMĀCĪBAS KATEGORIJAS.

Divas galvenās neironu tīklu apmācības kategorijas ir apmācība "ar skolotāju" (*with a teacher*) un "bez skolotāja" (*without a teacher*).

Neironu tīklu apmācība "ar skolotāju".

Neironu tīkli ar apmācību "ar skolotāju", kuru tipiskākais pārstāvis ir perceptrons, lielāko tiesu darbojas kā funkciju aproksimatori, kas citu funkcijas aproksimācijas metožu vidū izceļas ar to, ka savu funkcionēšanas spēju iegūst apmācības ceļā. Apmācības materiāls sastāv no noteikta skaita "jautājumu" un "pareizo atbilžu". Neironu tīklam tiek uzdots jautājums un tā dotā atbilde tiek salīdzināta ar "pareizo atbildi". Apmācības mērķis ir kļūdas samazināšana. Šo algoritmu priekšrocība ir iespēja precīzi novērtēt tīkla darbības pareizību un precīzi noteikt veicamās darbības pareizības palielināšanā. Pateicoties šādai

noteiktībai, nosauktie algoritmi ir salīdzinoši viegli pielietojami, diezgan efektīvi un tādējādi arī populāri.

Neironu tīklu apmācība "bez skolotāja".

Otrs veids ir apmācība"bez skolotāja". Šādus tīklus bieži sauc par pašorganizējošiem tīkliem. Pašorganizējošo neironu tīklu galvenā iezīme ir tāda, ka tajos neeksistē tāds jēdziens kā "pareizā atbilde", kuram savukārt pie neironu tīkliem ar apmācību "ar skolotāju" ir izšķiroša nozīme. Tas nozīmē, ka nav precīzi definējams kritērijs neironu tīkla funkcionēšanas pareizībai.

Šai iezīmei ir gan pozitīvā, gan negatīvā puse.

1. Pozitīvā puse ir tāda, ka apmācības procesā nav nepieciešami testa piemēri, kuriem būtu dotas arī "pareizās atbildes". Īpaši svarīgi tas ir tādēļ, ka daudzām problēmām nemaz iepriekš nevar noteikt, kādai kurā gadījumā būtu jābūt "pareizajai atbildei".
2. Negatīvā puse ir salīdzinoši sarežģītākais šādu neironu tīklu pielietojums, jo ne vienmēr var saredzēt šādu neironu tīklu pielietojamību konkrētu problēmu risināšanā. Bez tam vispārīgā gadījumā ir daudz sarežģītāk noteikt apmācības procesa uzdevumus. (Tīkliem ar apmācību ar skolotāju tas ir "vienkāršāk" – lai tīkls dotu arvien precīzākas "atbildes" uz uzdotajiem "jautājumiem".)

C3.3. NEIRONU TĪKLU APMĀCĪBAS STRATĒĢIJAS.

Neironu tīkla apmācība ir neironu tīkla parametru (galvenokārt neironu svaru) izmainīšana noteiktā veidā. Tālāk aprakstītās apmācības stratēģijas ir vairāku zināmu apmācības algoritmu vispārinājumi.

Formula C3 parāda viena neirona **svara izmaiņu** apmācības procesa vienā solī. [Haykin, 1999]

$$w_i(t+1) = w_i(t) + \Delta w_i \qquad \text{(C3)}$$

kur $w_i(t+1)$ *– jaunā svara vērtība,*

$w_i(t)$ *– vecā svara vērtība,*

Δw_i *– svara izmaiņa,*

t *– soļa numurs (1..N).*

Ja mums ir zināma "pareizā atbilde", tad svaru vērtības izmaiņa ir **proporcionāla izrēķinātajai kļūdai**, kas vienkāršākajā gadījumā ir starpība, ko veido "pareizā atbilde" un tīkla dotā atbilde (formula C4).

$$\Delta w_i \equiv x_i \, err \qquad \text{(C4)}$$

kur Δw_i *– i-tā svara izmaiņa,*

x *– i-tā ieeja,*

err – neirona izrēķinātā kļūda.

Šo principu sauc arī par Delta likumu (Delta Rule) vai Vidrova-Hofa (Widrow-Hoff) likumu.

Pašorganizējošajiem neironu tīkliem nav tāda jēdziena kā kļūda, tāpēc tiek pielietoti citi apmācības algoritmi.

Viens no vienkāršākajiem apmācības virzieniem ir **Heba apmācība** (*Hebbian learning*). Heba apmācības ideja īsi būtu noformulējama šādi:

Ja divi neironi, kas atrodas vienas sinapses (sinaptiskā svara) abos galos, aktivizējas vienlaicīgi (sinhroni), tad šī sinapse tiek pastiprināta, bet, ja asinhroni, tad pavājināta vai likvidēta.

Vienkāršākā Heba metodes forma parādīta formulā C5.

$$\Delta w_i \equiv x_i \, y \qquad \text{(C5)}$$

kur Δw_i *– i-tā svara izmaiņa,*

x *– i-tā ieeja,*

y *– neirona izeja.*

Tiešā veidā šāda forma nav izmantojama, jo liek svara vērtībai neierobežoti augt laikā. Tomēr modificēta Heba apmācība tiek lietota vairākos neironu tīklu modeļos. Heba metodes trūkums ir tas, ka grūti intuitīvi saprast šīs metodes efektu (pretstatā, piemēram, metodei ar kļūdu labošanu).

Vēl viens apmācības virziens ir **apmācība ar konkurenci** (*competitive learning*). Šajā metodē neirona svaru apmācību ietekmē arī blakusesošie neironi. Nosacīti var teikt, ka blakusesošos neironus savieno sinapses ar kavējošu iedarbību (*lateral inhibition*). Apmācības ar konkurenci kontekstā parādās jēdziens **neirons – uzvarētājs** (*winning neuron*), kas tiek noteikts pēc neirona aktivitātes pakāpes noteiktā laika momentā.

Neirons – uzvarētājs apmācās ar maksimālo koeficientu, bet blakusesošie neironi – vai nu neapmācās nemaz, vai nu mazāk, un to pakāpi nosaka t.s. **kaimiņa funkcija** (*topological neighbourhood*, formula C6). Koeficients pašam neironam – uzvarētājam parasti ir vienāds ar 1. Apkaimes koeficienta noteikšanai bieži lieto Gausa funkciju.

Apmācība ar konkurenci:

$$\Delta w_i \equiv D(i, j) \tag{C6}$$

kur Δw_i *– i-tā svara izmaiņa,*

j *– neirona uzvarētāja indekss,*

D *– kaimiņa funkcija.*

Viens no intuitīvi visvieglāk saprotamajiem apmācības virzieniem ir **adaptīvais mehānisms**: apmācības gaitā svaru vērtības tiecas ienākošo signālu vērtību virzienā (formula C7).

$$\Delta w_i(t+1) \equiv x_i - w_i(t) \tag{C7}$$

kur $\Delta w_i(t+1)$ *– i-tā svara izmaiņa,*

x_i *– i-tā ieeja,*

$w_i(t)$ *– vecā i-tā svara vērtība.*

C4. LINEĀRAS UN NELINEĀRAS PROBLĒMAS.

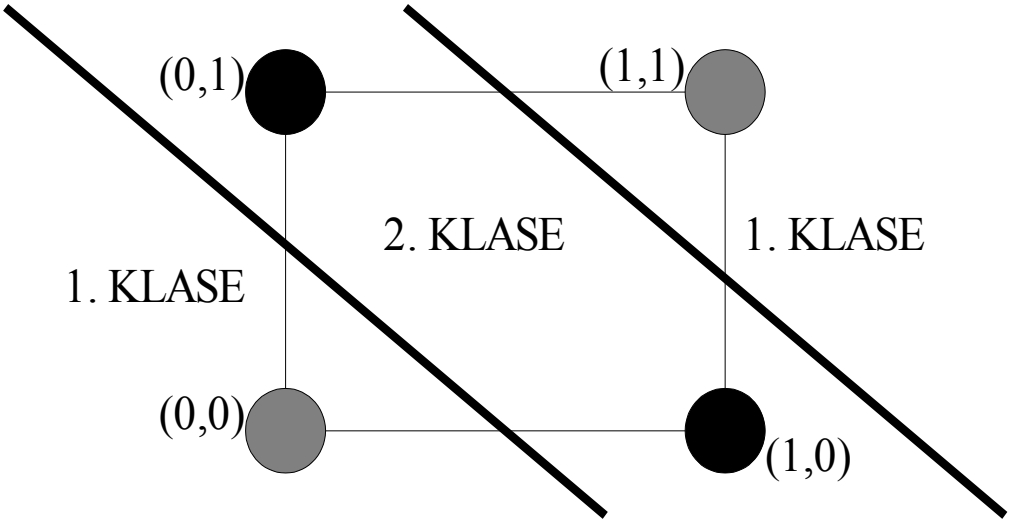
Nodaļās C1.3 un C1.4 aprakstīti vienkārši neironu piemēri. Nav grūti saprast, ka ar šādu neironu palīdzību nevarētu nomodelēt funkciju XOR (izslēdzošais VAI). Tas parāda, ka ir divu veidu funkcijas – tādas, kuras var nomodelēt ar vienu neironu, un tādas, kuras nevar.

Ir divu veidu problēmas – lineāras un nelineāras, un šajā gadījumā mēs esam saskārušies ar vienkāršāko nelineāras problēmas gadījumu funkciju modelēšanā.

Ja mums ir neironu tīkls (vienkāršākajā gadījumā – neirons) ar n ieejām, tad var teikt, ka neironu tīkls modelē n-dimensiju telpu.

Ja klasifikācijas uzdevumā mēs visus paraugus varam sadalīt atbilstošajās klasēs, izmantojot tikai vienu taisni (divdimensiju gadījumā) vai hiperplakni (vairāku dimensiju gadījumā), tad var teikt, ka problēma ir lineāra, citādi tā ir nelineāra. Vienkāršākā nelineārā problēma, kā jau tika minēts, ir XOR problēma – vajadzīgas 2 taisnes, lai punktus sadalītu 2 klasēs (attēls C11).

Vienslāņa perceptrons nespēj risināt nelineārus klasifikācijas uzdevumus. Daudzslāņu perceptrons spēj, bet ar nosacījumu, ka tā neironos tiek izmantotas nelineāras aktivizācijas funkcijas.



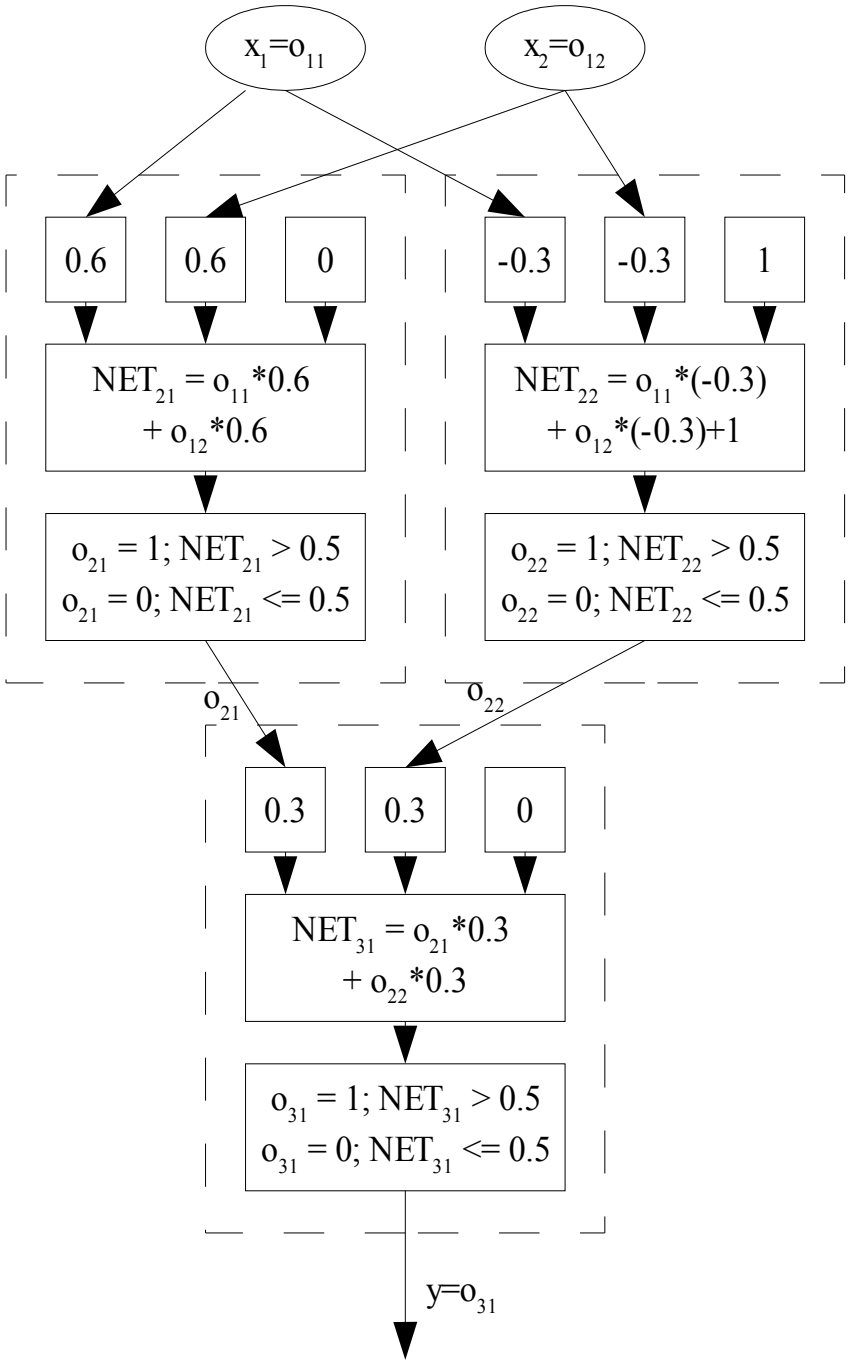
Attēls C11. XOR problēma – lai sadalītu punktus 2 klasēs, nepieciešamas 2 taisnes.

Spēja risināt nelineāras problēmas uzskatāma par ļoti būtisku neironu tīklu īpašību nelineāru problēmu salīdzinoši lielā skaita dēļ. Ilustrācijai nelineāru problēmu skaits starp Būla funkcijām (tabula C3).

Tabula C3. Lineāri atdalāmu n-dimensiju Būla funkciju skaits.

n	Skaits kopā	Lineāri atdalāmo funkciju skaits
1	4	4
2	16	14
3	256	104
4	65536	1772
5	$4.3*10^{9}$	94572
6	$1.8*10^{19}$	5028134

Attēlā C12 parādīts neironu tīkls no 3 neironiem, kas atpazīst funkciju XOR. Neironu konstrukcija tāda pati kā nodaļā C1.4.



Attēls C12. Neironu tīkls, kas atpazīst funkciju XOR. A XOR B modelēts kā (A OR B) AND NOT (A AND B)

Tabula C4. Attēlā C12 aprakstītā neironu tīkla pārbaude.

$x_1=o_{11}$	$x_2=o_{12}$	NET_{21}	o_{21}	NET_{22}	o_{22}	NET_{31}	$y=o_{31}$
0	0	0	0	1	1	0.44	0
0	1	0.6	1	0.7	1	0.74	1
1	0	0.6	1	0.7	1	0.74	1
1	1	1.2	1	0.4	0	0.44	0

C5. VIENOŠANĀS PAR APZĪMĒJUMIEM.

Aprakstot neironu tīklu darbību tiek lietoti dažādi apzīmējumi. Runājot par neironu tīkla komponentēm, mainīgie ar lielo reģistru parasti apzīmē vērtību vektoru (masīvu), piemēram, W – visa slāņa svaru kopums, bet mainīgie ar mazo reģistru – atsevišķas vērtības, piemēram, w_{ji} – j-tā neirona i-tais svars. Izņēmums ir vienīgi NET, kas tradicionāli rakstās ar lielajiem burtiem.

Nodaļā C1.2 aprakstīti dažādu neirona komponenšu apzīmējumi. Šie apzīmējumi parādīti fiksētam neironam, piemēram, w_i – dotā neirona i-tais svars, NET – dotā neirona summēšanas funkcija.

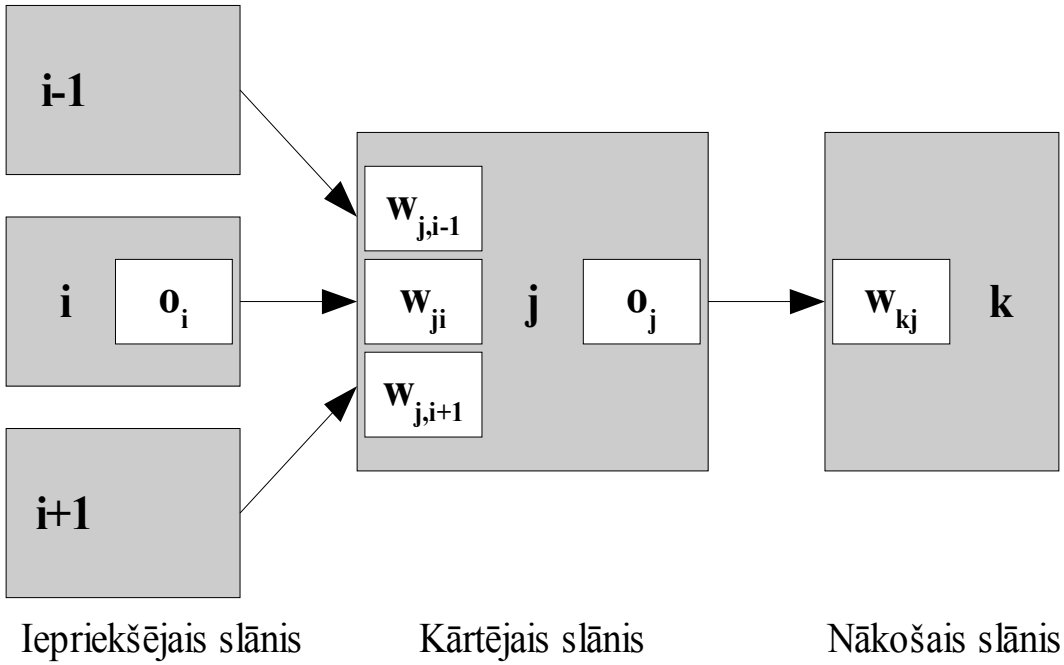
Ja apraksts veidots neironu slāņa kontekstā, nāk klāt papildus indekss – neirona numura apzīmēšanai, piemēram – o_j – j-tā neirona izeja, w_{ji} – j-tā neirona i-tais svars.

Dažkārt apraksts veidots plašākā kontekstā nekā tikai vienam slānim – bez kārtējā slāņa tiek aprakstīts iepriekšējais, un dažreiz pat nākošais slānis (attēls C13).

Klasiskajos neironu tīklu modeļos blakus slāņu neironi ir savienoti katrs ar katru, tādēļ neironā ir tik daudz svaru, cik iepriekšējā slānī neironu.

Kārtējā slāņa kārtējā neirona numurs tiek apzīmēts ar j, iepriekšējā slāņa saistīta neirona numurs ar i, bet nākošā slāņa saistītā neirona numurs ar k (ievērojot šo burtu alfabētisko secību), turklāt pirmais indekss pie svara norāda neirona numuru, bet otrais – svara numuru neironā.

Ļoti līdzīgi apzīmējumi tiek lietoti arī citos avotos, tomēr katrā avotā parasti ir nelielas izklāsta nianses, tāpēc ir svarīgi precizēt izmantoto apzīmējumu lietojumu.



Attēls C13. Neironu un neironu komponenšu numerācija, aprakstot neironu tīkla darbības vairākiem slāņiem.