

Vještačka inteligencija

Predavanje 5: Uvod u neuronske mreže 2

*Učenje nije produkt nastavnčkog predavanja.
Učenje je produkt aktivnosti onih koji uče.
~John Holt*

Odgovorna nastavnica: Vanr. prof. dr Amila Akagić

Univerzitet u Sarajevu



Uvodne informacije

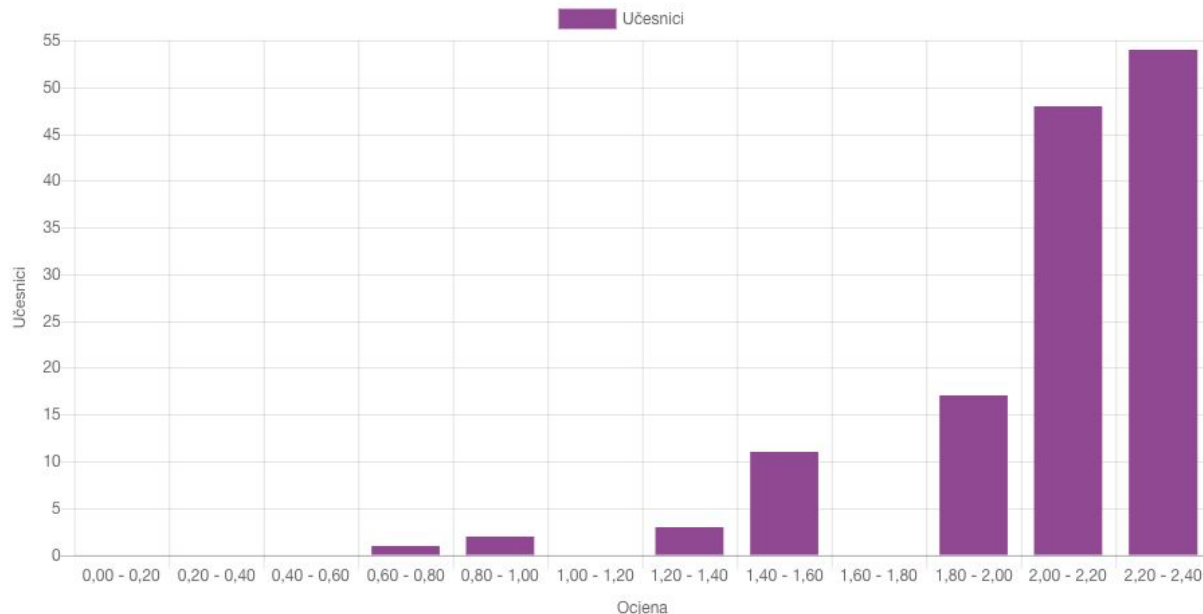
- This work is licensed under a Creative Commons 'Attribution-NonCommercial-ShareAlike 4.0 International' license. EN: <https://creativecommons.org/licenses/by-nc-sa/4.0/>



- Ovaj rad je licenciran pod međunarodnom licencom 'Imenovanje-Nekomercijalno-Dijeli pod istim uvjetima 4.0' od strane Creative Commons. HR: <https://creativecommons.org/licenses/by-nc-sa/4.0/deed.hr>

Najave

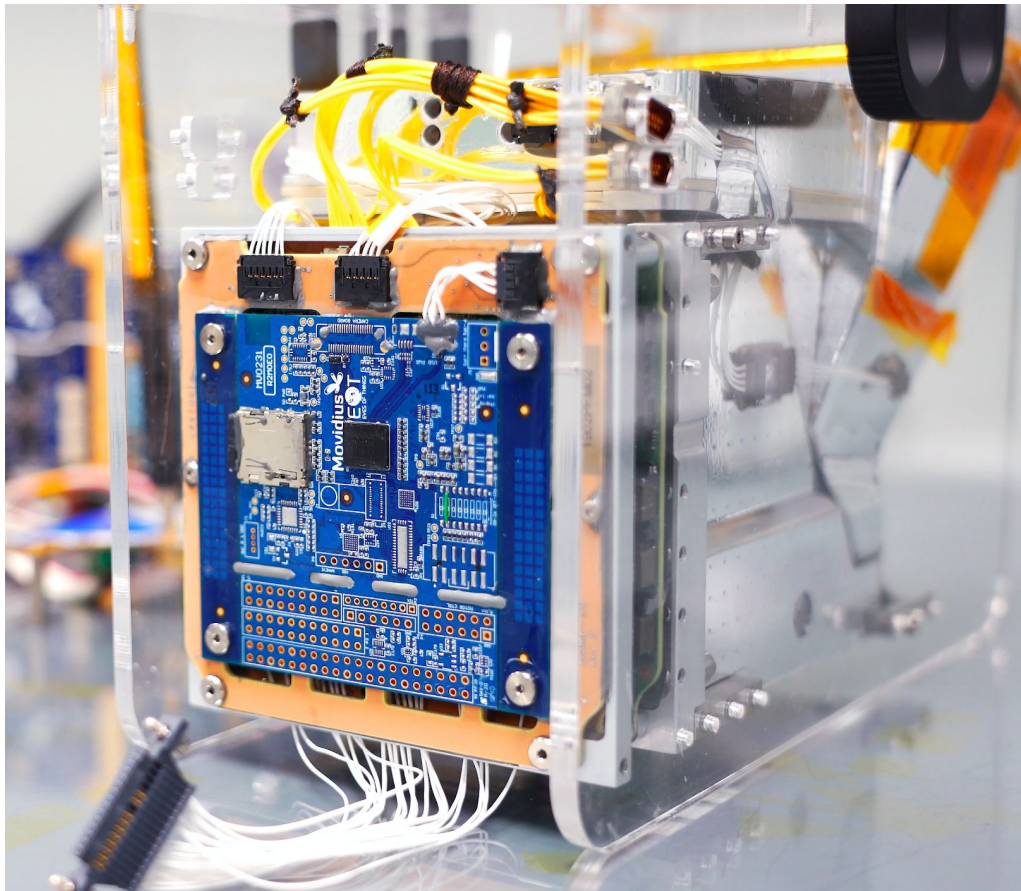
- ❑ Danas u 14:00 je treći kviz u okviru realizacije vježbi.
 - ❑ 8 pitanja, 5 minuta vremena.
 - ❑ Automatsko ocjenjivanje nakon isteka vremena.
- ❑ Sa predavanjima nastavljamo u 14:10+.



Aktuelnosti / Novosti

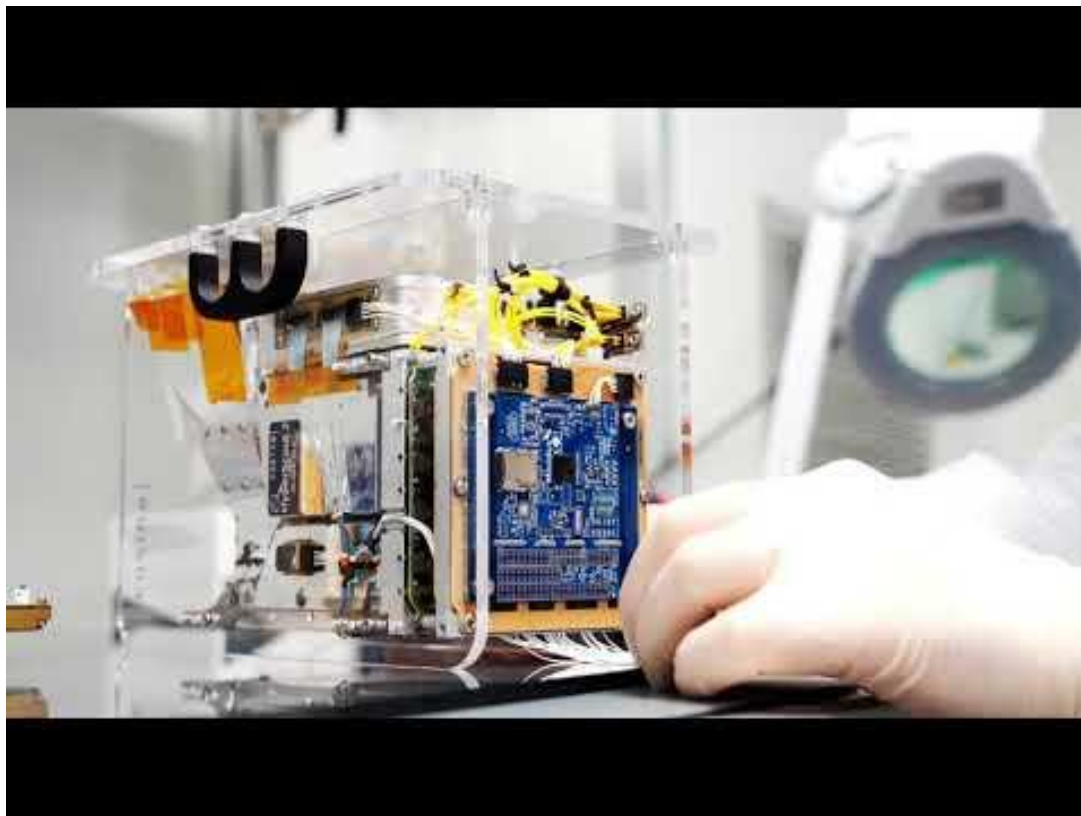
PhiSat-1 contains a new **hyperspectral-thermal camera** and onboard AI processing thanks to an **Intel® Movidius™ Myriad™ 2 Vision Processing Unit (VPU)** — the same chip inside many smart cameras and even a \$99 selfie drone here on Earth.

The first problem the Myriad 2 is helping to solve? How to handle the large amount of data generated by high-fidelity cameras like the one on PhiSat-1. “The capability that sensors have to produce data increases by a factor of 100 every generation, while our capabilities to download data are increasing, but only by a factor of three, four, five per generation,” says Gianluca Furano, Data Systems and Onboard Computing lead at the European Space Agency, which led the collaborative effort behind PhiSat-1.



<https://newsroom.intel.com/news/intel-powers-first-satellite-ai/>

Aktuelnosti / Novosti



<https://newsroom.intel.com/news/intel-powers-first-satellite-ai/>

Motivacija

- ❑ Mozak “računa” na sasvim drugačiji način od konvencionalnih digitalnih računara.
- ❑ Neuroni su pet-šest redova veličine sporiji od digitalne logike (ms i ns).
- ❑ Međutim, ovaj nedostatak u brzini mozak nadoknađuje ogromnim brojem neurona:
 - ❑ Mozak ima oko 100 milijardi neurona i oko 10^{15} konekcija među neuronima.
- ❑ Mozak je izuzetno **energetski efikasan** (10^{16} J po operaciji u sekundi prema 10^{-6} J po operaciji u sekundi) .
- ❑ Mozak je veoma kompleksan, nelinearan, paralelan računar.



Broj zvijezdi
u galaksiji
Andomeda.



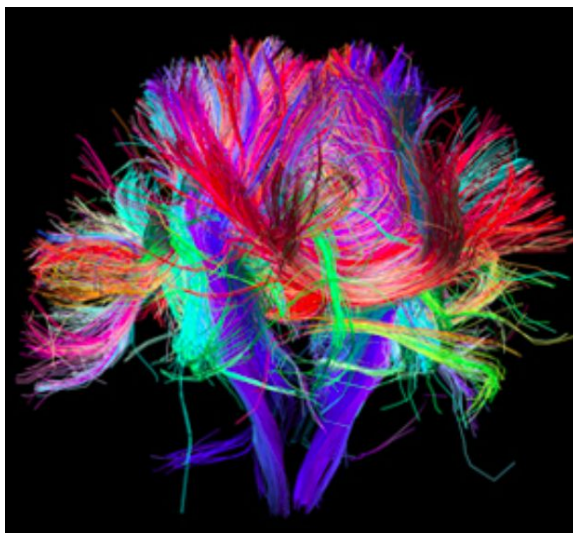
Broj mrava
na Planeti
zemlji.

Motivacija

Za one koje interesuju detalji funkcionisanja ljudskog mozga in-vivo:

Projekt Human Connectome (HCP) je projekat za izgradnju mape kompletnih strukturnih i funkcionalnih neuronskih veza in vivo unutar i između pojedinaca. HCP predstavlja prvi veliki pokušaj prikupljanja i razmjene podataka opsega i detalja dovoljnih za započinjanje procesa rješavanja temeljnih pitanja o ljudskoj vezivnoj anatomiji i razlikama.

<http://www.humanconnectomeproject.org/about/>



Podjela neuronskih mreža

- ❑ Biološke (prirodne) neuronske mreže:
 - ❑ To su biološki organizmi.
 - ❑ Primjeri su mozak ljudi i životinja.
 - ❑ Visoko su složene i ima mogućnost paralelnog izvršavanja.
- ❑ Vještačke neuronske mreže:
 - ❑ Nastale na osnovu motivacije bioloških neuronskih mreža.
 - ❑ Smatra se da su za sada dosta primitivne imitacije bioloških mreža.
 - ❑ Implementacija na digitalnim računarima opće namjene ili pomoću specijaliziranih kola (analognih, digitalnih, hibridnih).
 - ❑ Izučavamo ih na ovom predmetu.

Motivacija: biološki neuron

Biološki	Vještački
Soma	Čvor
Dendridi	Ulazi
Akson	Izlaz
Sinapsa	Koeficijent
Sporo	Brzo
Veliki broj neurona (10^9)	Nekoliko neurona (desetina ili stotine hiljada)

Dendrit

Tijelo ćelije

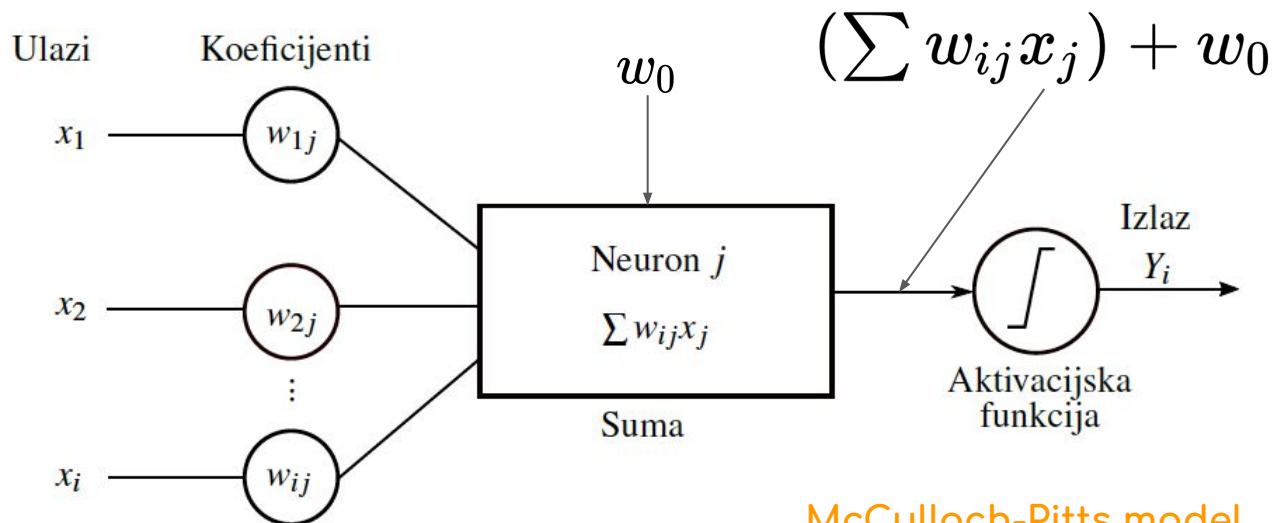
Akson

**Električni
impulsi**

**Presinaptički
terminal**

Prvi model biološkog neurona

- ❑ Threshold Logic Unit ili TLU-perceptron predstavljen je od strane Warren McCulloch i Walter Pitts, 1943 godine.
- ❑ Predstavlja prvi model biološkog neurona.



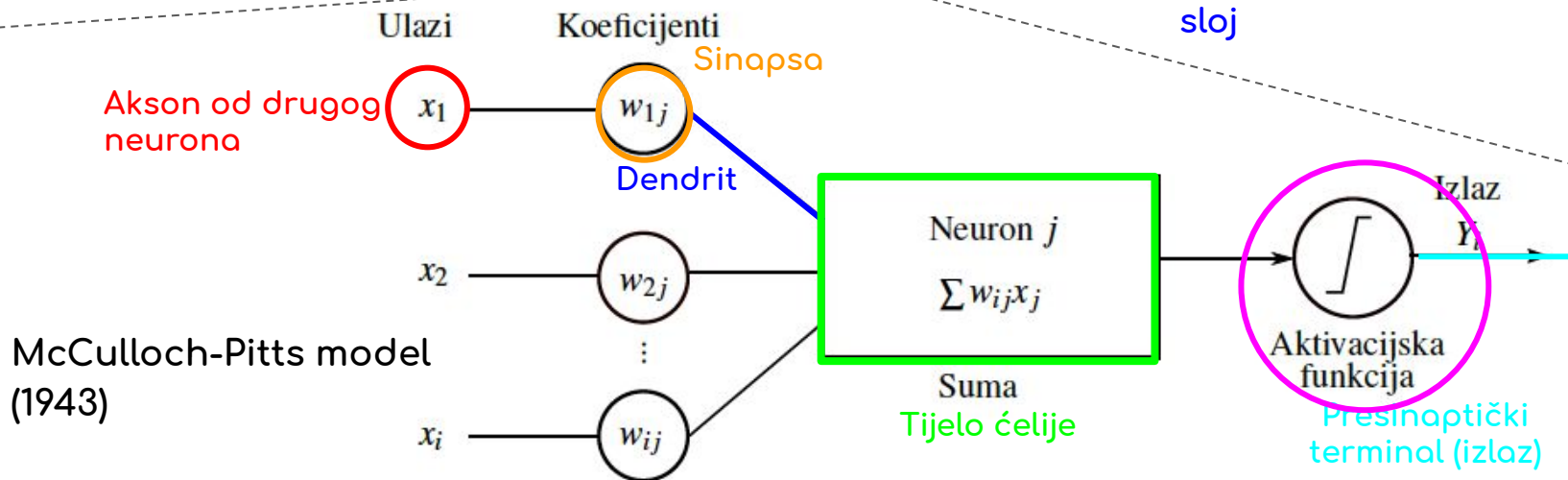
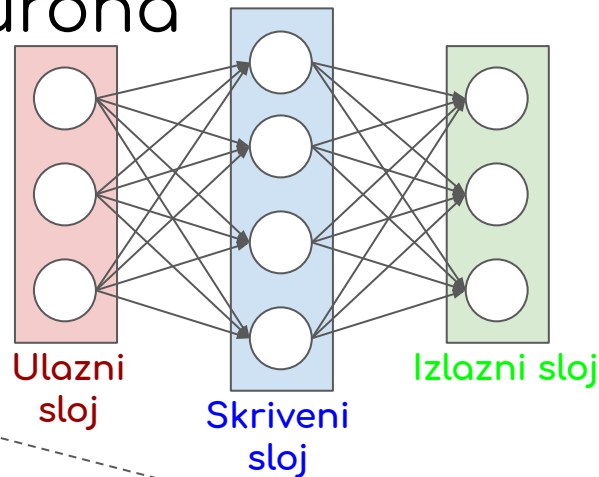
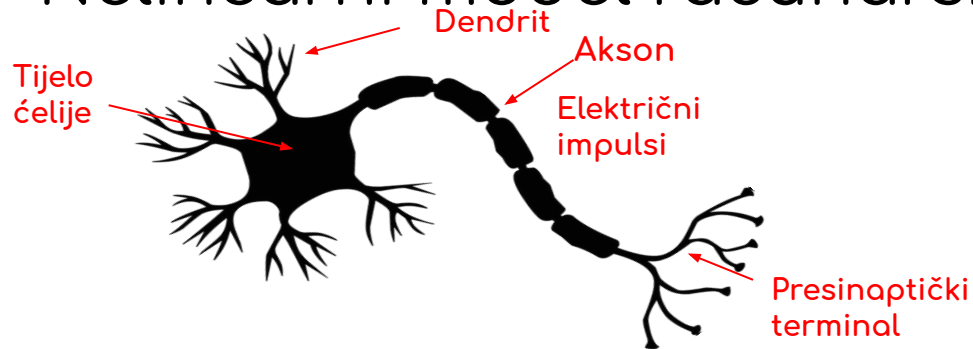
McCulloch-Pitts model

Model će generisati izlaz u slučaju da suma umnožaka ulaza i koeficijenata pređe određeni prag.

Neuronska mreža

- ❑ Neuronska mreža nije ništa drugo do skup jednostavnih komponenti koje nazivamo vještačkim **neuronima**, koji su povezani na neki predefinisani način. Zato ovu strukturu nazivamo **mrežom**.
- ❑ Osobine mreže utvrđene su njenom **topologijom**, kao i **osobinama neurona**.
- ❑ **Računarska neuronauka** (*computational neuroscience*): izučavanje sistema baziranih na osobinama neurona i topologijama neuronskim mreža.
- ❑ **Neke osobine neuronskih mreža:**
 - ❑ Mogućnost distribuiranih proračuna.
 - ❑ Otpornost na šum kod podataka (robustnost).
 - ❑ Sposobnost učenja.
- ❑ Postoje i neki drugi sistemi koji imaju mogućnost učenja iz podataka, poput **Bayesove mreže**, međutim neuronske mreže su najpopularniji i efektniji oblici sistema učenja.

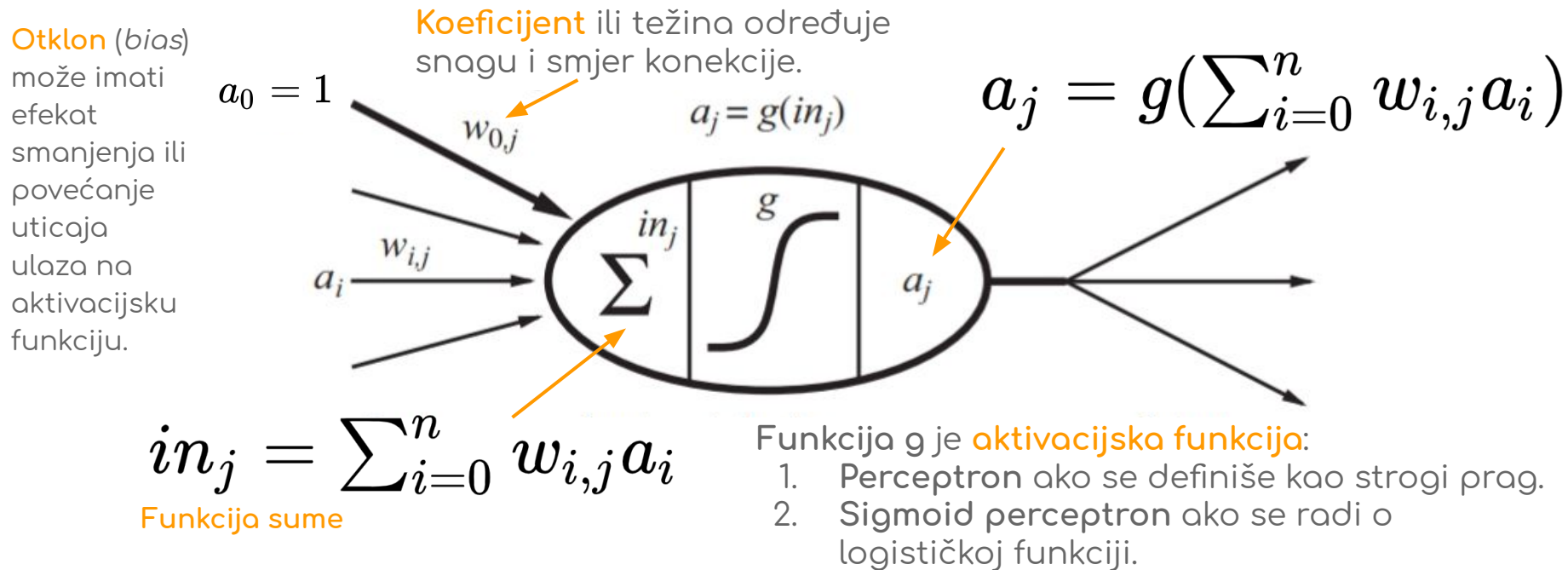
Nelinearni model računarskog neurona



McCulloch-Pitts model
(1943)

Model neurona

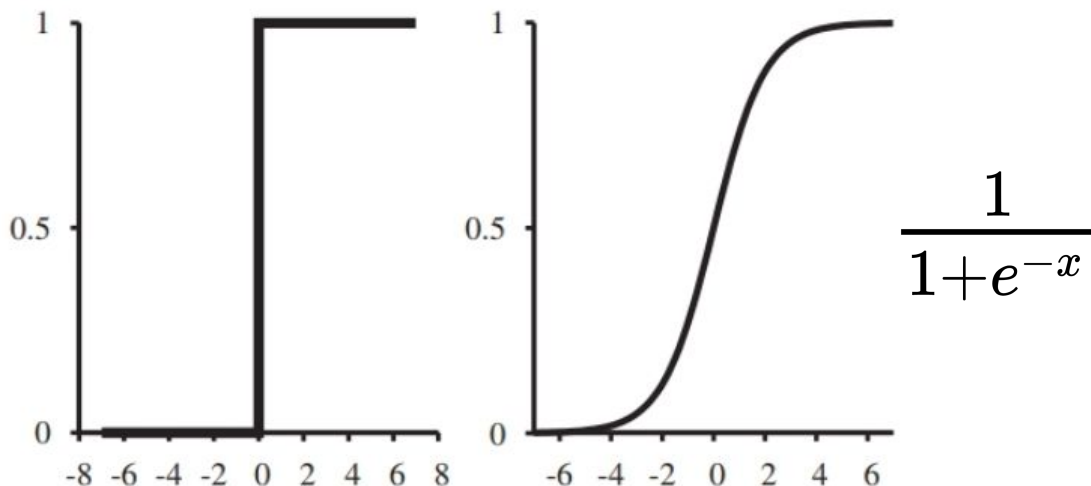
- Neuronske mreže su sačinjene od čvorova (*nodes*) ili jedinica (*units*) povezanih preko grafa sa direktnim linkovima.



Model neurona

Funkcija g je aktivacijska funkcija koja osigurava da mreža može predstaviti nelinearnu funkciju.

1. Perceptron ako se definiše kao strogi prag (0,1). Vrijednost izvoda nije definisana u tački 0.
2. Sigmoid perceptron ako se radi o logističkoj funkciji (sigmoid funkcija).



Strukture neuronskih mreža

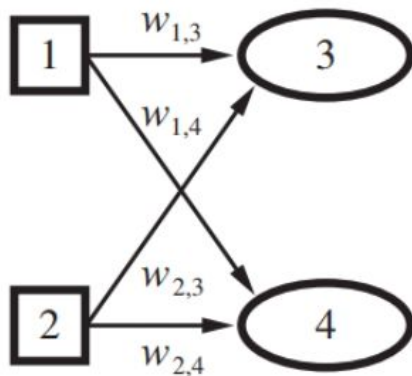
- ❑ Nakon što je utvrđen model neurona, potrebno je definisati topologiju mreže, tj. povezati neurone na odgovarajući način.
- ❑ Postoje dvije osnovne strukture:
 - ❑ **Mreže bez povratnih veza** (*feed-forward network*):
 - ❑ Veze se formiraju samo u jednom smjeru (unaprijed);
 - ❑ Mreža se formira u direktni aciklični graf.
 - ❑ Mreža predstavlja funkciju trenutnih ulaza, tj. ne postoji interno stanje.
 - ❑ **Mreže sa povratnim vezama** (*recurrent network*):
 - ❑ Postoji povratna sprega, tj. izlazi se proslijeđuju na ulaze mreže.
 - ❑ Dinamički sistem koji može biti u stabilnom stanju ili imati određene oscilacije ili biti u stanju haosa.
 - ❑ Nazivaju se još **sistemima sa kratkom memorijom** (*short-term memory*).

Mreže bez povratnih veza

- ❑ Obično se organizuju u nivoima ili slojevima (*layers*), na način da jedan neuron može da prihvati ulaze samo iz prethodnog sloja.
- ❑ Razlikujemo dvije vrste:
 - ❑ Naprijed usmjerena jednoslojna mreža (*single layer feed-forward network*)
 - ❑ Naprijed usmjerena višeslojna mreža (*multiple layer feed-forward network* ili *Multiple Layer Perceptron (MLP)*)

Naprijed usmjerena jednoslojna mreža

- ❑ Naziva se još i **perceptron mreža**: ulazi su direktno povezani na izlaze.
- ❑ Arhitektura se definiše kao broj neurona i način njihovog povezivanja.
- ❑ Koju funkciju možemo naučiti na primjeru ove arhitekture?

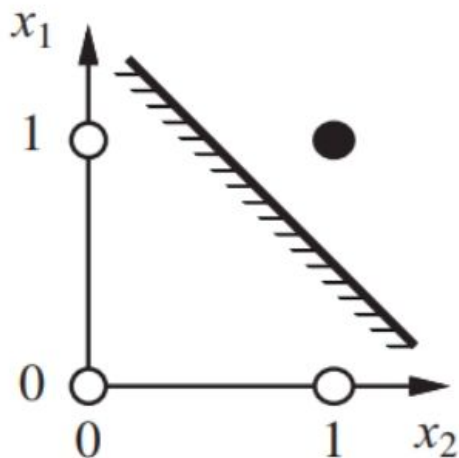


Perceptron mreža sa m izlaza je zapravo m odvojenih mreža zato što jedan koeficijent utiče samo na jedan izlaz.

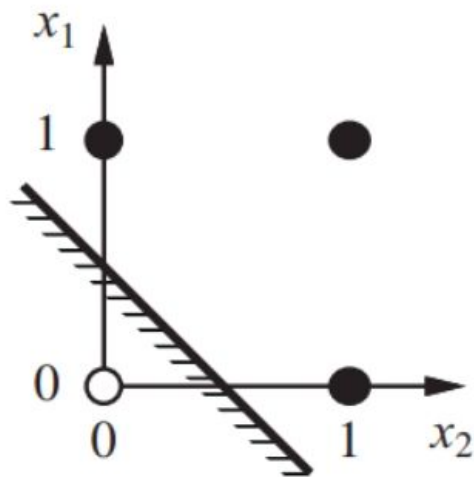
x_1	x_2	y_3 (carry)	y_4 (sum)
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Da li model ove mreže može naučiti dvije funkcije iz primera? Koje su to funkcije?

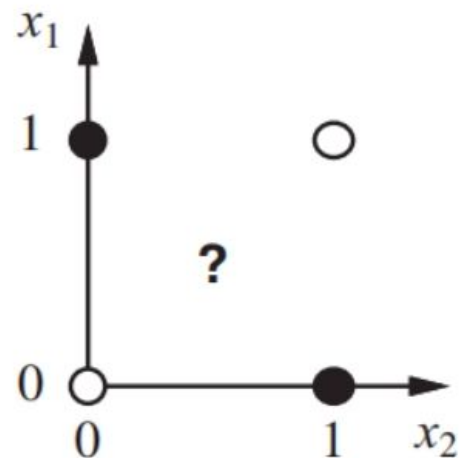
Naprijed usmjerena jednoslojna mreža



x_1 and x_2



x_1 OR x_2



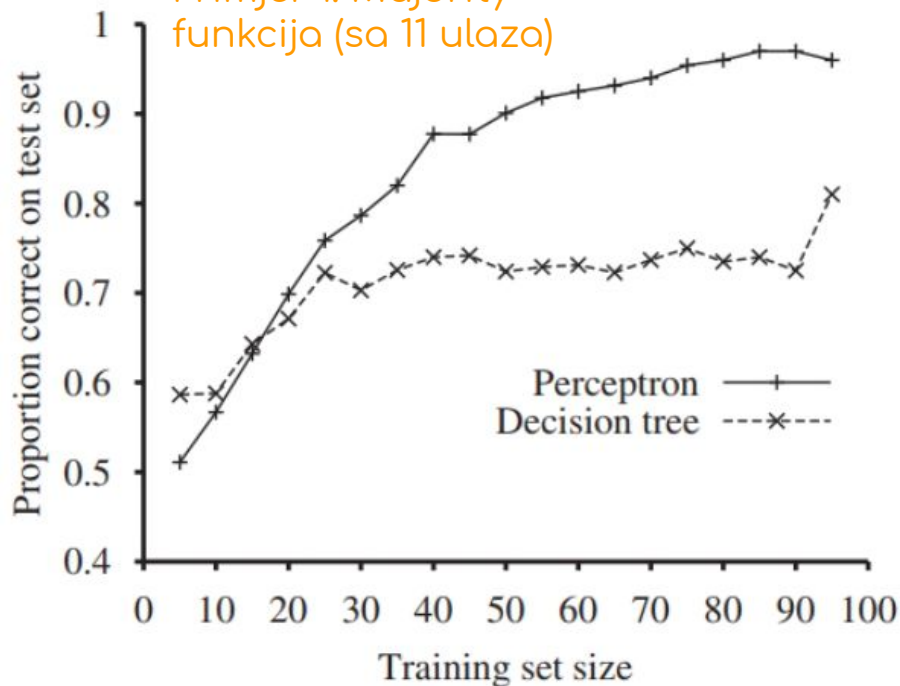
x_1 XOR x_2

x_1	x_2	y_3 (carry)	y_4 (sum)
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Da li su zbog ovoga problema Perceptron mreže beskorisne za rješavanje složenih problema?

Naprijed usmjerena jednoslojna mreža

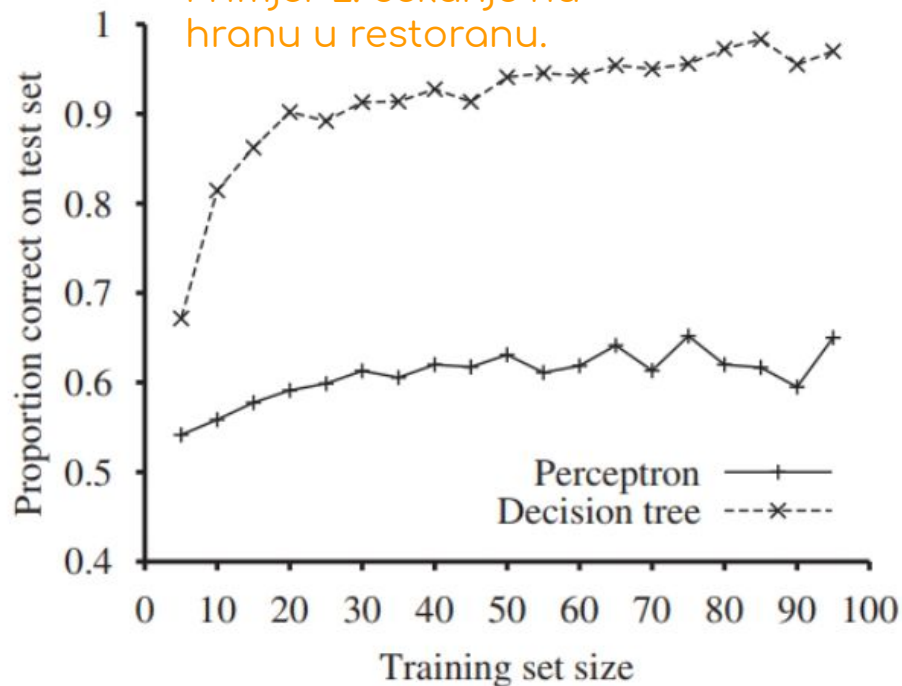
Primjer 1: Majority funkcija (sa 11 ulaza)



(a)

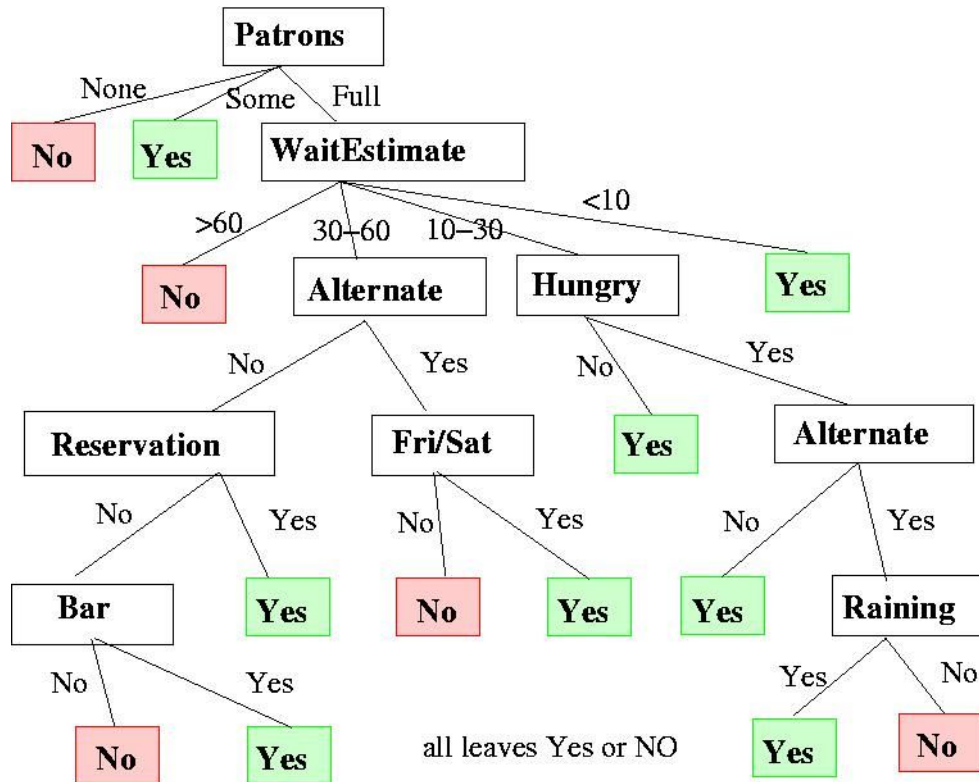
$$\langle p_1, \dots, p_n \rangle = \text{Majority}(p_1, \dots, p_n) = \left\lfloor \frac{1}{2} + \frac{(\sum_{i=1}^n p_i) - 1/2}{n} \right\rfloor.$$

Primjer 2: čekanje na hranu u restoranu.



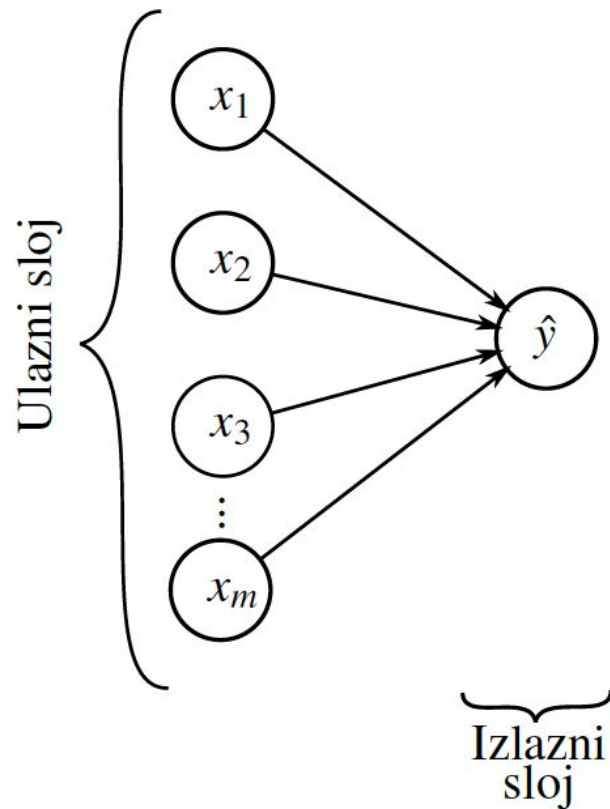
(b)

Primjer 2: Čekanje u restoranu



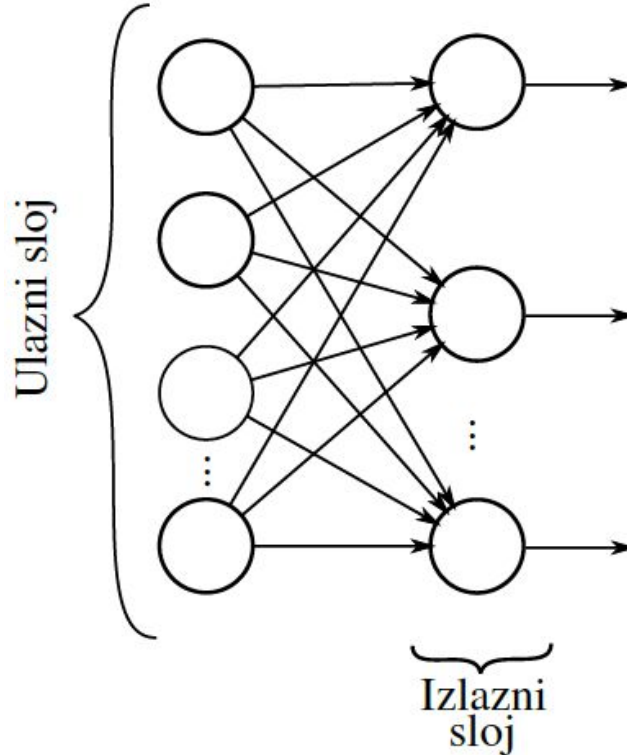
Naprijed usmjerena jednoslojna mreža

- ❑ Jedan ulaz predstavlja jedan atribut.
- ❑ Primjer: potrebno je donijeti odluku da li odobriti ili odbaciti zahtjev za izdavanje pozajmnice.
- ❑ Atributi mogu biti:
 - ❑ dob aplikanta
 - ❑ godišnja plata
 - ❑ bračni status
 - ❑ posjedovanje imovine, itd.
- ❑ Numeričke vrijednosti ovih atributa su ulazi u mrežu.



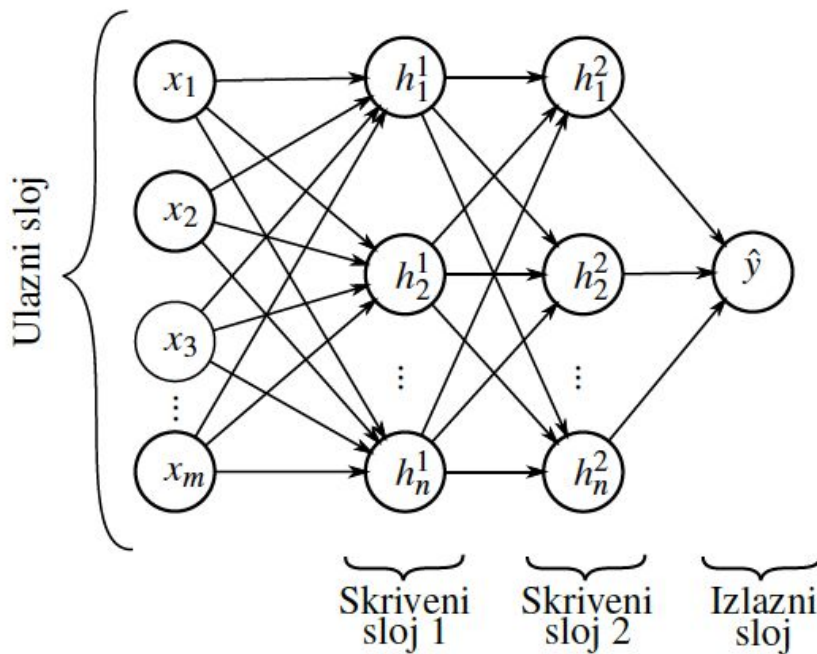
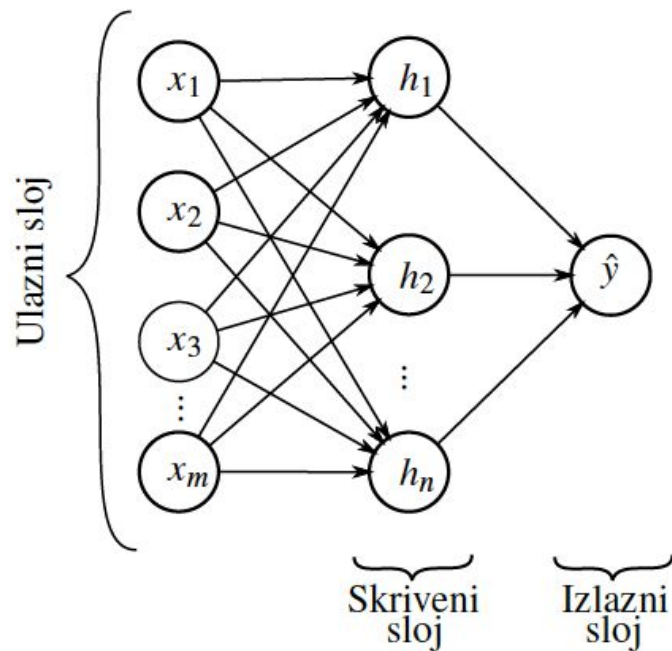
Naprijed usmjerena jednoslojna mreža

- ❑ Važno: izračunavanje se izvodi isključivo u izlaznom čvoru ili čvorovima.



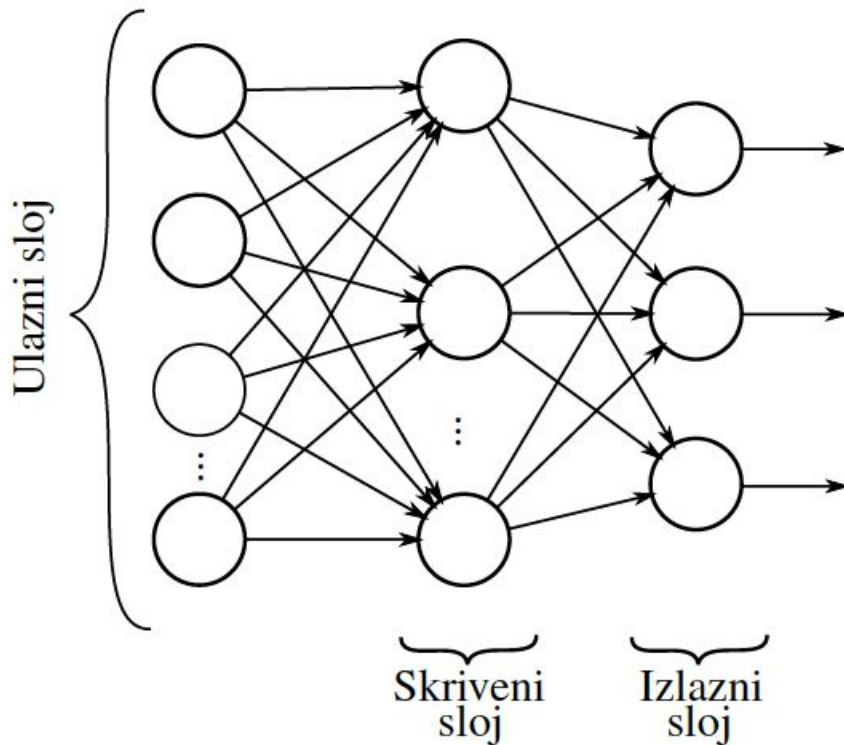
Naprijed usmjerena višeslojna mreža

- ❑ McCulloch i Pitts su 1943 predvidjeli ograničenja njihovom modela, te su teoretisali o ideji povezivanja više neurona u složenije mreže. Međutim, u to vrijeme niko nije znao kako trenirati takve strukture!

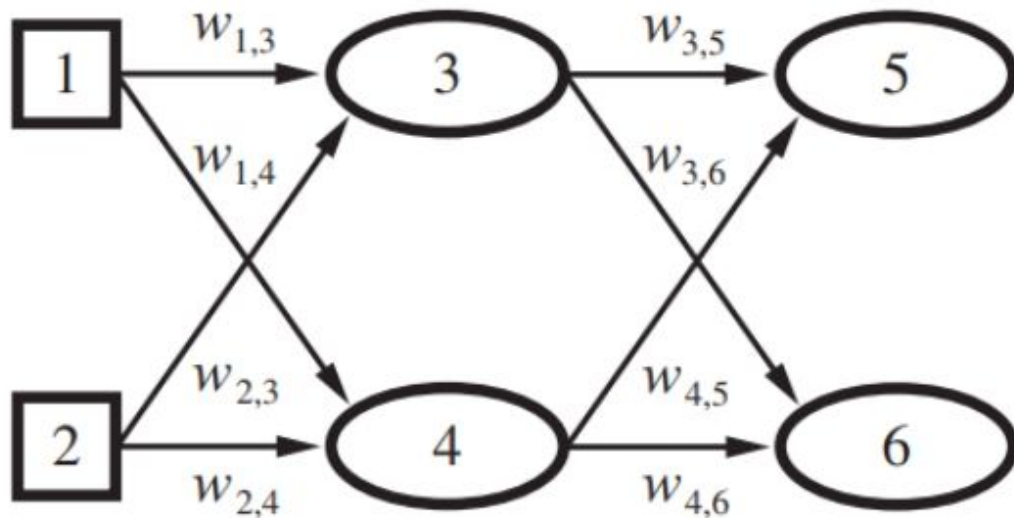


Naprijed usmjerena višeslojna mreža

- ❑ Kada su neuroni u jednom sloju povezani sa svakim neuronom u sljedećem susjednom sloju, onda za takvu mrežu kažemo da je **potpuno povezanom mrežom (fully connected network)**.
- ❑ U suprotnom, mrežu nazivamo **djelimično povezanom mrežom**.
- ❖ Ako se pretpostavi da postoji 10 ulaznih atributa (singala), 6 skrivenih neurona u skrivenom sloju i 3 izlazna neurona, onda se takva mreža naziva 10-6-3 mreža.
- ❖ Općenito za n ulaznih neurona, h_1 neurona u prvom i h_2 u drugom sloju, nazivamo $n-h_1-h_2$ mrežom.



Naprijed usmjerena višeslojna mreža

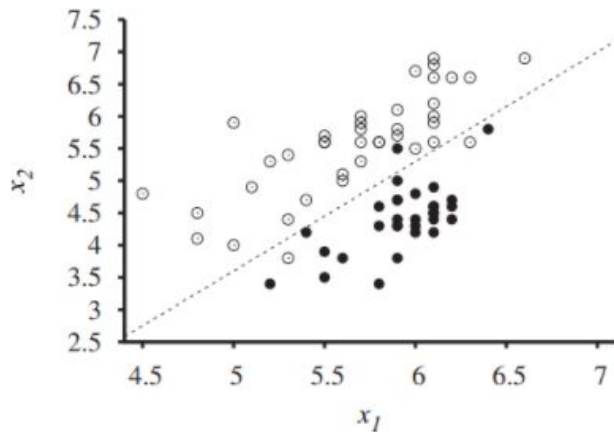


$$\begin{aligned} a_5 &= g(w_{0,5} + w_{3,5}a_3 + w_{4,5}a_4) \\ a_3 &= g(w_{0,3} + w_{1,3}a_1 + w_{2,3}a_2) \\ a_4 &= g(w_{0,4} + w_{1,4}a_1 + w_{2,4}a_2) \end{aligned}$$

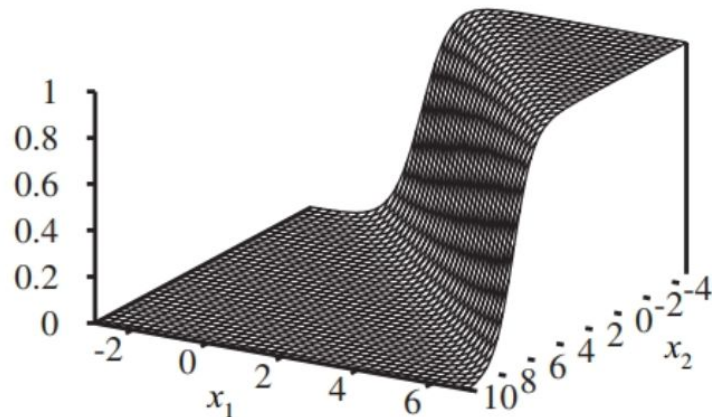
x_1 x_2

Za treniranje modela može se koristiti neka metoda bazirana na gradijentu koja će minimizirati funkciju gubitka (vrijednost se računa kao razlika između predviđenje i poznate izlazne vrijednosti).

Primjer: jednoslojna mreža



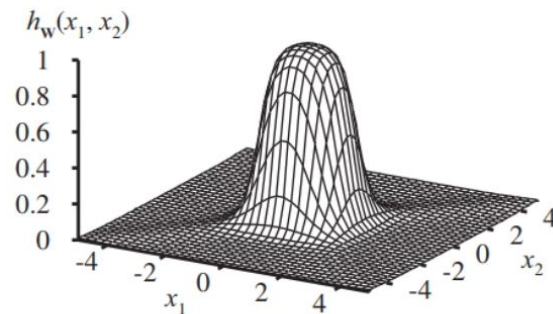
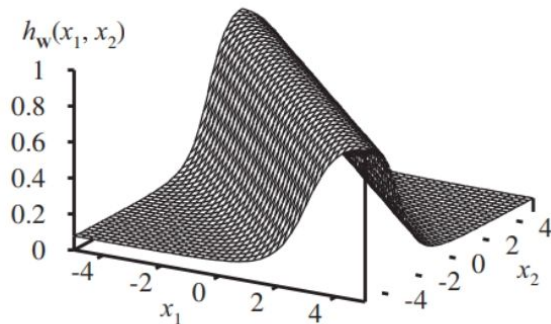
Funkcija praga



- ❑ Podaci su predstavljeni preko dvije promjenjive: x_1 i x_2 , koji predstavljaju veličinu tijela i površina talasa za seizmičke signale.
- ❑ Postoje dvije klase:
 - ❑ Zemljotresi (kružići)
 - ❑ Nuklearne reakcije (crne tačke)
- ❑ Zadatak klasifikatora je da napravi ispravnu klasifikaciju za nove tačke.
- ❑ Potrebno je pronaći granicu (*decision boundary*) između dva skupa.
- ❑ Može biti linija ili površina (ako ima više dimenzija).

Primjer: višeslojna mreža

- ❑ Kombinovanje dvije funkcije praga koje su okrenute nasuprot možemo dobiti funkciju koju još nazivamo **greben** (*ridge*).
- ❑ Povezivanje više grebena (npr. 4) dobit će se **ispupčenje**.
- ❑ Zatim je moguće generisati nekoliko ispupčenja različitih veličina i kombinovati ih po potrebi.
- ❑ Sa jednim skrivenim slojem sa puno neurona moguće je predstaviti bilo koju kontinualnu funkciju, dok je sa više skrivenih slojeva moguće prezentirati i prekide u funkciji.



Funkcija gubitka

- ❑ Funkcija gubitka definiše se kao iznos gubitka korisnosti u odnosu na predviđenu vrijednost funkcije $h(x) = \hat{y}$, kada je tačan odgovor $f(x) = y$.

$$L(x, y, \hat{y})$$

$$L(y, \hat{y})$$

$$L(y, y) = 0$$

$$h(x) = 137.036$$

$$y(x) = 137.035999$$

Kako ocijeniti korisnost ove predikcije?

- ❑ Apsolutna vrijednost gubitka (L1):
- ❑ Kvadratna vrijednost greške (L2):

$$L_1(y, \hat{y}) = |y - \hat{y}|$$

$$L_2(y, \hat{y}) = (y - \hat{y})^2$$

Generalizirani gubitak

- Možemo definisati generalizirani gubitak kada su poznate svi ulazno/izlazni primjeri za neki problem. Neka su ti primjeri označeni sa \mathcal{E} .
- Generalizirani gubitak se definiše na sljedeći način:

$$GenLoss_L(h) = \sum_{(x,y) \in \mathcal{E}} L(y, h(x))P(x, y)$$

- Najbolja hipoteza se zatim računa kao minimalna očekivana vrijednost generaliziranog gubitka:

$$h^* = \operatorname{argmin}_{h \in H} GenLoss_L(h)$$

- Kako nije poznata vjerovatnoća distribucije $P(x,y)$, moguće je definisati samo empirijski gubitak na skupu primjera koji označavamo sa E :

$$EmpLoss_{L,E}(h) = \frac{1}{N} \sum_{(x,y) \in E} L(y, h(x))$$

Empirijski gubitak

- ❑ Kako nije poznata vjerovatnoća distribucije $P(x,y)$, moguće je definisati samo empirijski gubitak na skupu primjera koji označavamo sa E :

$$EmpLoss_{L,E}(h) = \frac{1}{N} \sum_{(x,y) \in E} L(y, h(x))$$

- ❑ Najbolja estimirana (procjenjena) hipoteza se zatim računa kao minimalna očekivana vrijednost empirijskog gubitka:

$$\hat{h}^* = \operatorname{argmin}_{h \in H} EmpLoss_{L,E}(h)$$

Empirijski gubitak

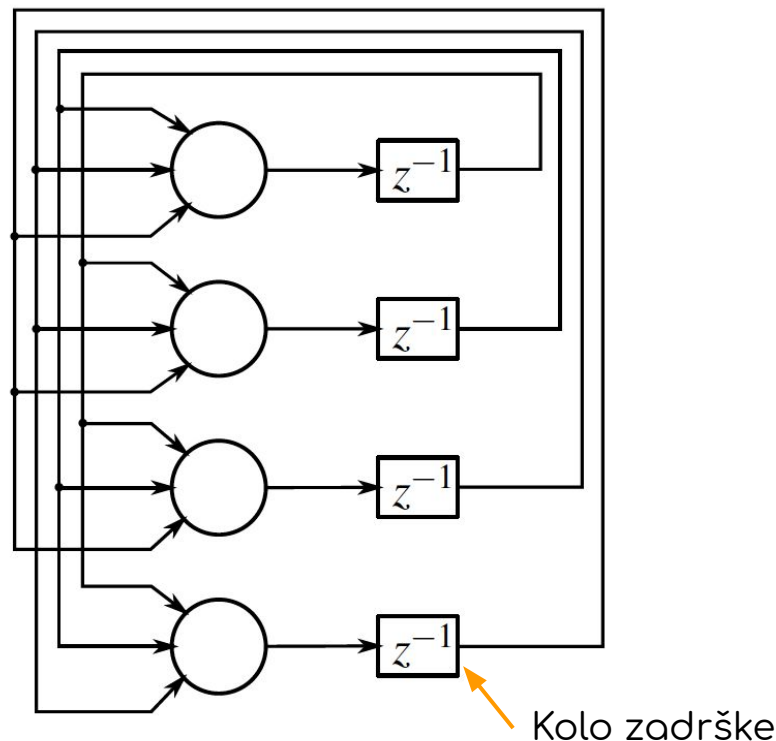
- ❑ Postoje četiri razloga zašto će se pronađena funkcija \hat{h}^* razlikovati od stvarne funkcije f :
 - ❑ Funkcija je neostvariva: moguće da se ne nalazi u prostoru hipoteza. Ili može da postoji, međutim postoje druge funkcije koje su bolje (efikasnije).
 - ❑ Funkcija varira: algoritam učenja može vratiti drugu funkciju za drugi skup primjera, čak iako su ti primjeri nastali sa istom funkcijom f .
 - ❑ Funkcija f je nedeterministička funkcija ili je pod uticajem šuma: vrijednost funkcije može biti drugačije za isti ulaz x . Opće je poznato da šum ne možemo predvidjeti.
 - ❑ Računska kompleksnost (složenost): ako je prostor hipoteza složen, onda pretraživanje tog prostora može biti nerješiv problem. U ovom slučaju moguće je pretraživati samo jedan dio prostora (lokalno pretraživanje)

Učenje i podataka

- ❑ Učenje iz podataka se grubo može podijeliti na dva pristupa:
 - ❑ Učenje iz malog skupa podataka: nekoliko desetina, do maksimalno nekoliko hiljada podataka
 - ❑ Učenje iz velikog skupa podataka: milioni podataka

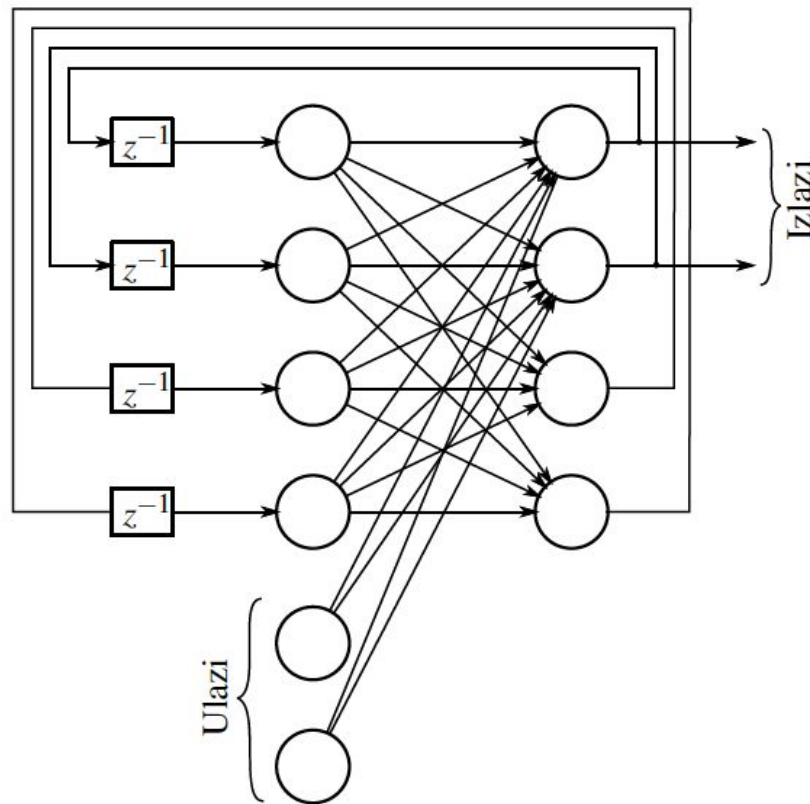
Mreže sa povratnim vezama (rekurentne mreže)

- ❑ Razlikuju se od mreža bez povratne veze u tome što imaju barem jednu povratnu vezu (spregu).
- ❑ Najjednostavniji model prikazan je na slici.
- ❑ U ovom primjeru nema skrivenog sloja, dok se izlazi dovede na ulaze ostalih ulaznih čvorova.



Mreže sa povratnim vezama (rekurentne mreže)

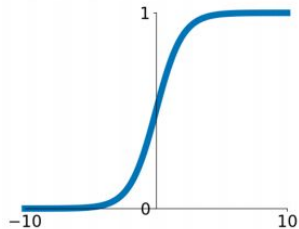
- ❑ Postojanje povratne sprege u neuronskim mrežama ima značajan uticaj na proces učenja, kao i na performanse mreže.
- ❑ Korištenje povratne sprege uslovljeno je korištenjem kola zadržke, koje se označava sa z^{-1} .
- ❑ Kolo zadržke uvodi jedan trenutak kašnjenja za određeni signala.
- ❑ Može uzrokovati nelinearno dinamičko ponašanje mreže.



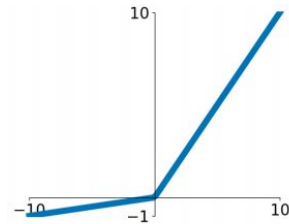
Aktivacijske funkcije: koju funkciju izabrati?

Sigmoid

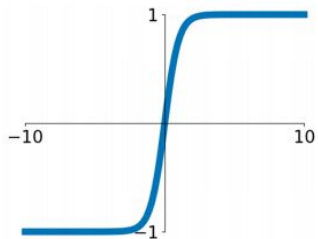
$$\sigma(x) = \frac{1}{1+e^{-x}}$$



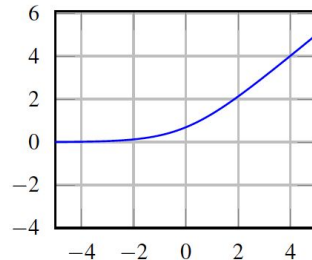
Leaky ReLU
 $\max(0.01x, x)$



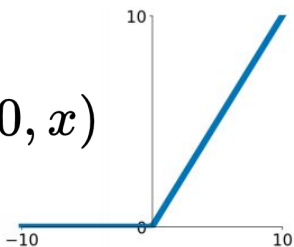
$$\tanh(x) = \frac{e^{2x}-1}{e^{2x}+1}$$



Softplus
 $\log(1 + e^x)$



$$\text{ReLU}(x) = \max(0, x)$$



ELU (*Exponential Linear Unit*)

$$f(x) = \begin{cases} x, & x \geq 0 \\ \alpha(e^x - 1), & x \leq 0 \end{cases}$$

