

Umjetne neuronske mreže

Zavod za elektroniku, mikroelektroniku,
računalne i inteligentne sustave

v4.8, 17. svibnja 2012.

© Bojana Dalbelo Bašić

© Marko Čupić

© Jan Šnajder



- Automatiziranu obradu podataka danas uglavnom rade digitalna računala.
- Ipak, još je uvijek daleko više podataka čija obrada nije automatizirana. Te podatke obrađuju ...



... živčani sustavi
živih organizama!



- Razvoj jedne grane računarstva motiviran je razmatranjem prevladavajućeg načina obrade podataka u svijetu u kojem živimo.
- Tražimo drugačiji koncept obrade podataka koji bi bio sličniji funkcioniranju biološkog mozga.
- **A.I.** - sustav koji uspješno oponaša rad mozga bio bi inteligentan.



1. Uvod u neuro-računarstvo

Povijesni pregled • Pravci AI • Definicija i vrste • Primjena • Učenje • Živčani sustav i biološki neuron • Model neurona i vrste • Arhitektura mreža

2. Procesni elementi

Perceptron • Pravilo perceptrona • Primjer • Učenje booleovih funkcija • Linearna odvojivost • XOR problem

3. Backpropagation algoritam

BACKPROPAGATION algoritam i interpretacija skrivenog sloja • Primjeri uporabe neuronskih mreža • Primjer sustava za raspoznavanje

1.

Motivacija razvoja neuro-računarstva

- Istraživanja u neurofiziologiji i kognitivnoj znanosti upućuju: mozak se sastoji od velikog broja neurona koji rade paralelno.
- Najsličniji model: računalni sustav u kojem brojni procesni elementi podatke obrađuju paralelno.
- Paradigma:

umjetna neuronska mreža - UNM
[artificial neural network – ANN]

- Područje koje se bavi tim aspektom obrade:

neuro-računarstvo

- Grane računarstva iz skupine tzv. 'mekog računarstva' (engl. soft computing).

1.

Usporedba ljudskog mozga i računala

- Arhitektura i organizacija mozga gotovo je potpuno različita od arhitekture konvencionalnih računala koja su danas u širokoj uporabi (von Neumannova računala):

atribut	mozak	računalo
gradbeni element	neuron (>100 vrsta)	logička vrata
brzina prijenosa	2 ms ciklus	ns ciklus
broj procesora	oko 10^{11}	$\leq 32^*$
broj veza	$10^3 - 10^4$	$\leq 32^*$
način rada	serijski, paralelno	serijski
signali	analogni	digitalni
informacije	ispravne/neispravne	ispravne

1.

...mozga i super-računala?

- Danas najbrže računalo: Tianhe-1A (11/2010)
 - Intel EM64T Xeon X56xx (Westmere-EP)
2930 MHz (11.72 GFlops)
 - 186 368 jezgri
 - 2 331 000 GFlopsa = $2,6 \cdot 10^{15}$ Flopsa
 - Linux OS
- <http://www.top500.org>
(koristi se LINPACK benchmark test)



1.

Pravci razvoja umjetne inteligencije

- Od prvih dana razvoja umjetne inteligencije (rane '50) postoje **dva pristupa** razvoju inteligentnih sustava:
 - **Simbolički pristup:** znanje iz neke domene nastoji se obuhvatiti skupom atomičkih semantičkih objekata (simbola) i zatim činiti manipulacija tih simbola pomoću algoritamskih pravila
 - **Konektivistički pristup:** temelji se na izgradnji sustava arhitekture slične arhitekturi mozga koji, umjesto da ga se programira, uči samostalno na temelju iskustva
- Različita područja zahtijevaju različite pristupe.
- Simbolički pristup je dobar u mnogim područjima (osobito isplativ postao je razvojem ekspertnih sustava), ali nije ispunio rana ekstravagantna obećanja.
- Neuspjeh leži u pogrešnoj pretpostavci da je svako znanje moguće formalizirati i da je mozak stroj koji podatke obrađuje formalnim pravilima.

1.

Konektivistički pristup

- Mnogi su svakodnevni zadaci previše složeni za simboličko predočavanje, npr. raspoznavanje uzoraka...

- Majku možemo prepoznati u **0.1 s**
- Neuroni u mozgu pale svake **ms**
- U seriji, dakle, pali **samo 100** neurona
- Očigledno **paralelna** obrada!



1.

Umjetne neuronske mreže vs. von Neumann

- Von Neumannova računala odlična su za simbolički pristup jer se problemi rješavaju algoritamski na sekvencijalnom stroju.
- Umjetne neuronske mreže su distribuirani i paralelni sustavi.
- Bitne karakteristične razlike dviju paradigmi/arhitektura:

von Neumann	ANN
Unaprijed detaljno opisujemo algoritam kroz korake	Uči samostalno ili s učiteljem
Samo se precizni podaci adekvatno obrađuju	Podaci mogu biti nejasni (šum) ili neizraziti
Funkcionalnost ovisi o svakom elementu	Obrada i rezultat ne ovisi mnogo o jednom elementu
Eksplicitna veza: sematički objekt - sklopovi računala	Implicitno znanje (teška interpretacija)

1.

Umjetna neuronska mreža - definicija

- U širem smislu: umjetna replika ljudskog mozga kojom se nastoji simulirati postupak učenja i obrade podataka.
- Zapravo dosta klimava analogija.
- **Neuronska mreža** ▶ skup međusobno povezanih jednostavnih procesnih elemenata (jedinica, čvorova) čija se funkcionalnost temelji na biološkom neuronu i koji služe **distribuiranoj paralelnoj** obradi podataka.



1.

Primjena umjetnih neuronskih mreža

- Odlično rješavaju probleme **klasifikacije i predviđanja** – sve probleme kod kojih postoji složena (nelinearna) veza ulaza i izlaza.
- Neke osobitosti:
 - Dobre u procjeni nelinearnosti. Mogu raditi s nejasnim ili manjkavim podacima (*sensor data*). Robusne na pogreške u podacima. Rade s velikim brojem varijabli i parametara. Prilagodljive okolini. Sposobne učiti. ...
- Najčešći zadaci:
 - Raspoznavanje uzoraka. Obrada slike i govora. Problemi optimizacije. Nelinearno upravljanje. Obrada nepreciznih i nepotpunih podataka. Simulacije. Prognoza vremenskih serija. ...

1.

Učenje umjetne neuronske mreže

- Dvije faze rada s ANN:
 - Faza učenja (treniranja) i
 - Faza obrade podataka (iskorištavanja, eksploatacije).
- **Učenje** ▶ iterativan postupak predočavanja ulaznih primjera (uzoraka, iskustva) i eventualno očekivana izlaza pri čemu dolazi do postupnog prilagođavanja težina veza neurona
- Jedno predočavanje svih uzoraka naziva se epohom
- Razlikujemo:
 - Pojedinačno učenje (on-line)
 - ▶ za svaki primjer podešavamo faktore
 - Grupno učenje (batch)
 - ▶ cijela epoha u jednoj iteraciji
- Znanje o izlazu kao funkciji ulaza pohranjeno je implicitno u težinama veza neurona

1.

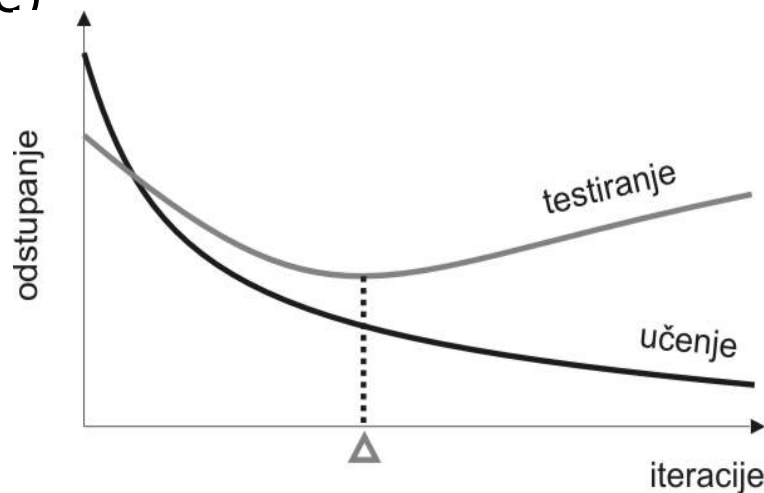
Učenje umjetne neuronske mreže

- Dva načina učenja:
 - Učenje s učiteljem (*supervised learning*) – postoje primjeri oblika (*ulaz, izlaz*)
 - Učenje bez učitelja (*unsupervised learning*) – izlaz je a priori nepoznat

1.

Učenje umjetne neuronske mreže

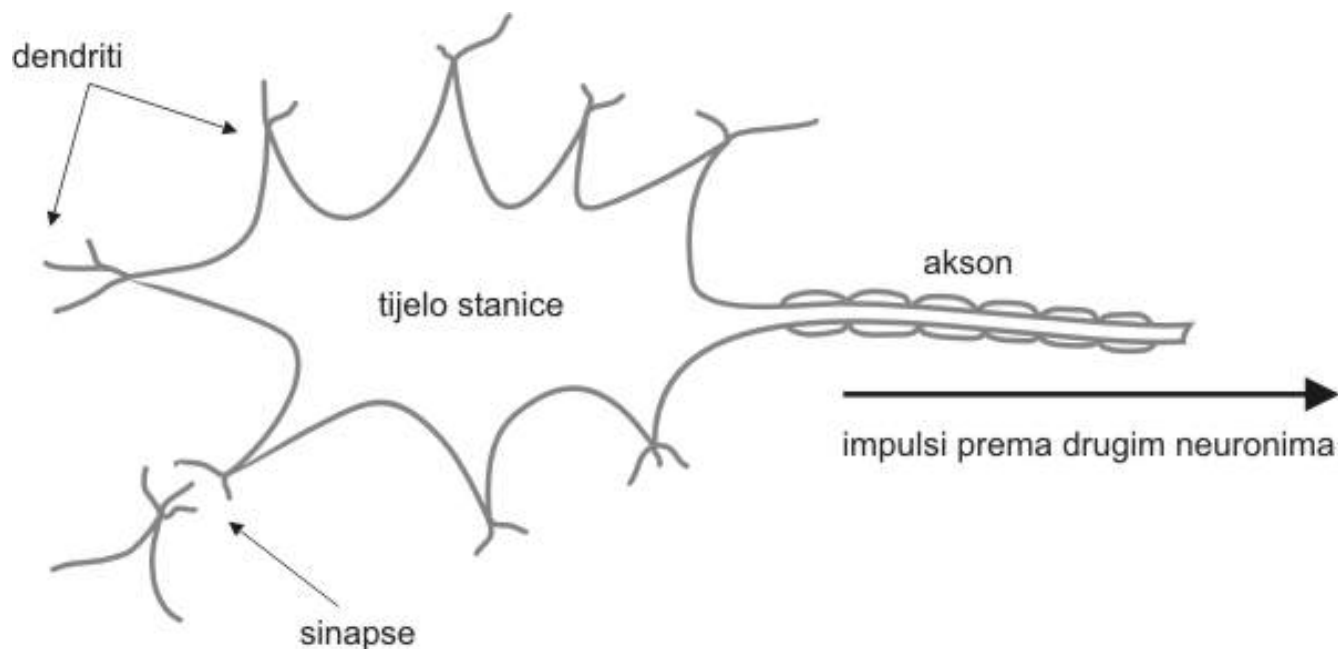
- Skup primjera za učenje često dijelimo na:
 - Skup za učenje – služi za iterativno podešavanje težina
 - Skup za testiranje – provjeravamo rad mreže
 - Skup za provjeru – konačna provjera
- Učenje se provodi dok mreža ne daje odgovarajuću točnost obrade podataka (uvodi se mjera pogreške)
- Pretreniranost ▶
ANN gubi poželjno svojstvo generalizacije i postaje stručnjak za podatke iz skupa za učenje (štreber)



1.

Naš živčani sustav

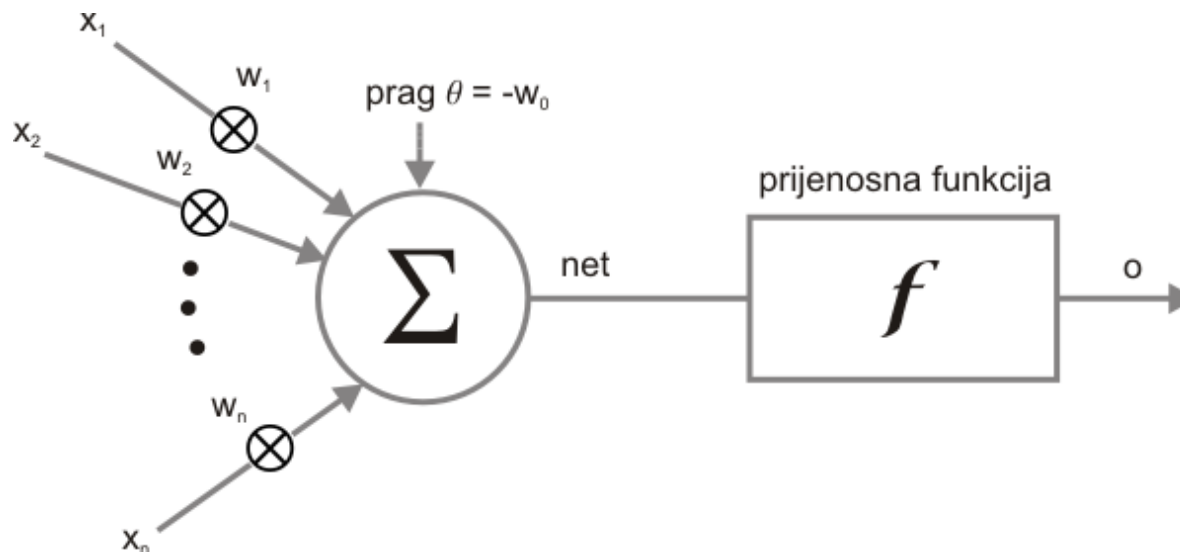
- 10^{11} neurona, 100 različitih vrsta, raspoređeni po definiranom rasporedu, svaki povezan s 10^4 drugih
- Dijelovi: soma, dendriti, akson, završni članci



1.

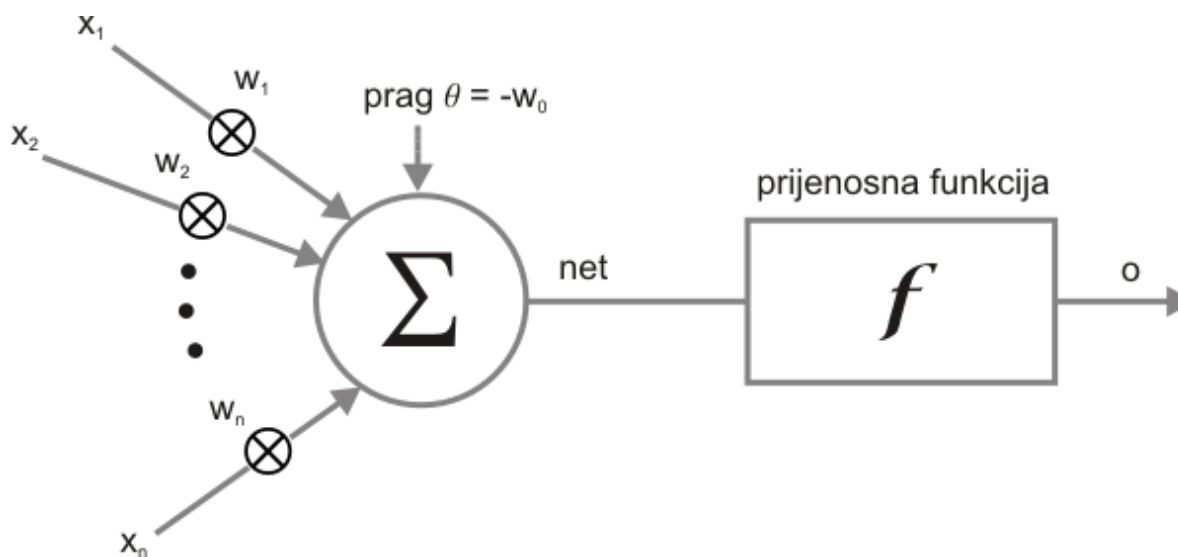
Umjetni neuron

- McCulloch-Pitts model (1943.): *Threshold Logic Unit* (TLU)
- Analogija: signali su numeričke vrijednosti, jakost sinapse opisuje težinski faktor w , tijelo stanice je zbrajalo, akson je prijenosna (aktivacijska) funkcija f



1.

Umjetni neuron



$$net = \omega_1 x_1 + \omega_2 x_2 + \dots + \omega_n x_n - \theta$$

$$net = \omega_0 x_0 + \omega_1 x_1 + \omega_2 x_2 + \dots + \omega_n x_n = \sum_{i=0}^n \omega_i x_i$$

$x_0 = 1, w_0 = -\theta$

$$o = f\left(\sum_{i=0}^n \omega_i x_i\right) = f(net)$$

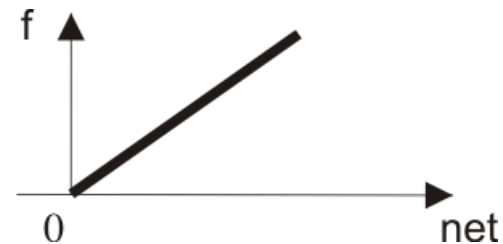
1.

Umjetni neuron

- Različite funkcije dolaze u obzir kao prijenosne funkcije:

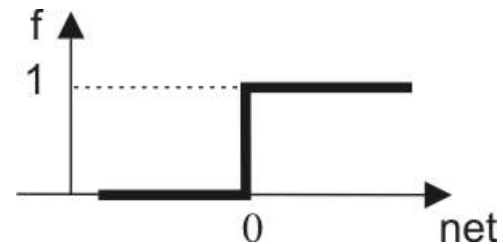
- ADALINE

$$f(\text{net}) = \text{net}$$



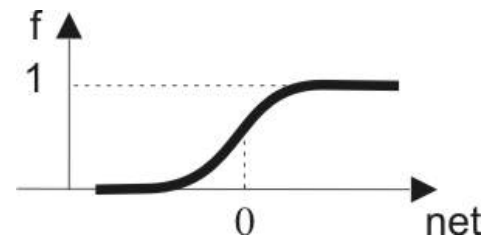
- TLU

$$f(\text{net}) = \begin{cases} 0 & \text{za } \text{net} < 0 \\ 1 & \text{inace} \end{cases}$$



- Sigmoidalna jedinica

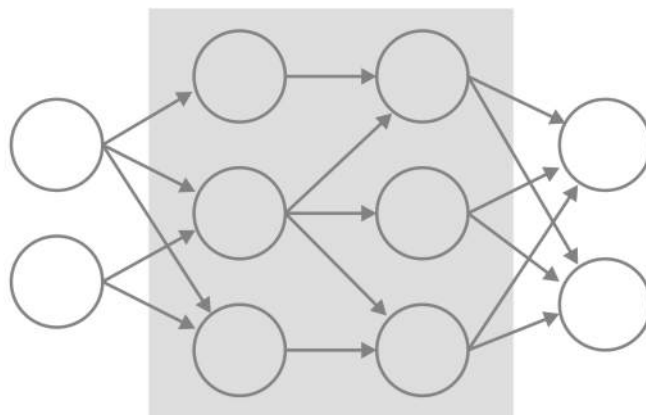
$$f(\text{net}) = \frac{1}{1 + e^{-a \cdot \text{net}}}$$



1.

Arhitektura mreža

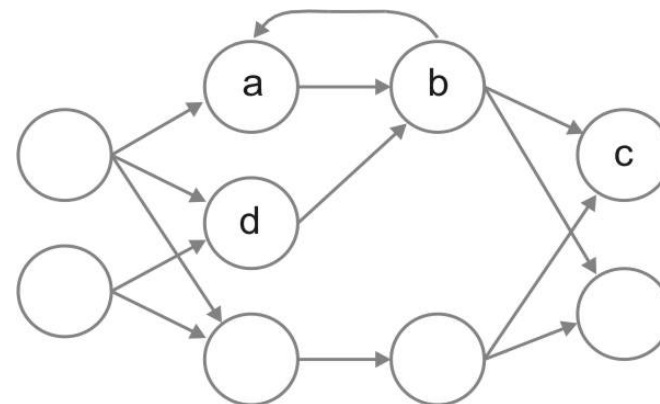
- Moguće arhitekture:



ulazni sloj

skriveni sloj

izlazni sloj



- Aciklička (*feedforward*)
- Druge...

1. Uvod u neuro-računarstvo

Povijesni pregled • Pravci AI • Definicija i vrste • Primjena • Učenje • Živčani sustav i biološki neuron • Model neurona i vrste • Arhitektura mreža

2. Procesni elementi

Perceptron • Pravilo perceptrona • Primjer • Učenje booleovih funkcija • Linearna odvojivost • XOR problem

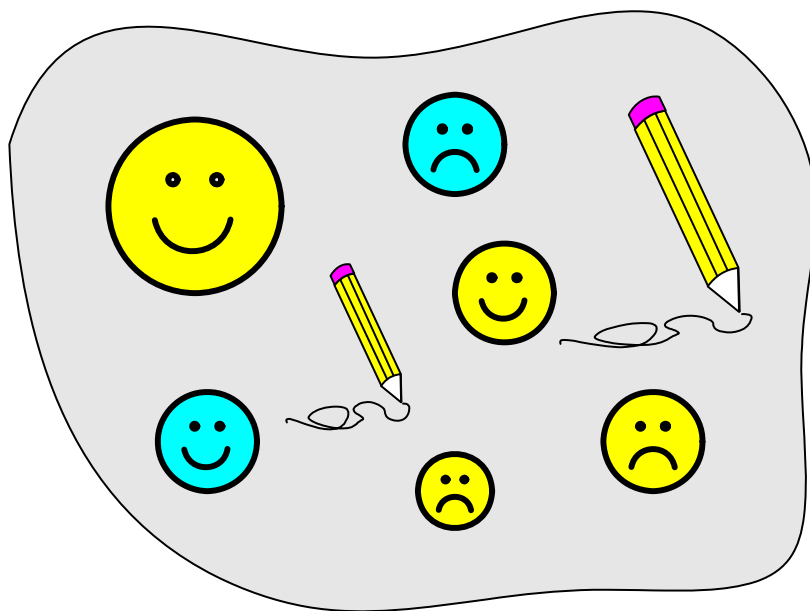
3. Backpropagation algoritam

BACKPROPAGATION algoritam i interpretacija skrivenog sloja • Primjeri uporabe neuronskih mreža • Primjer sustava za raspoznavanje

2.

Klasifikacija

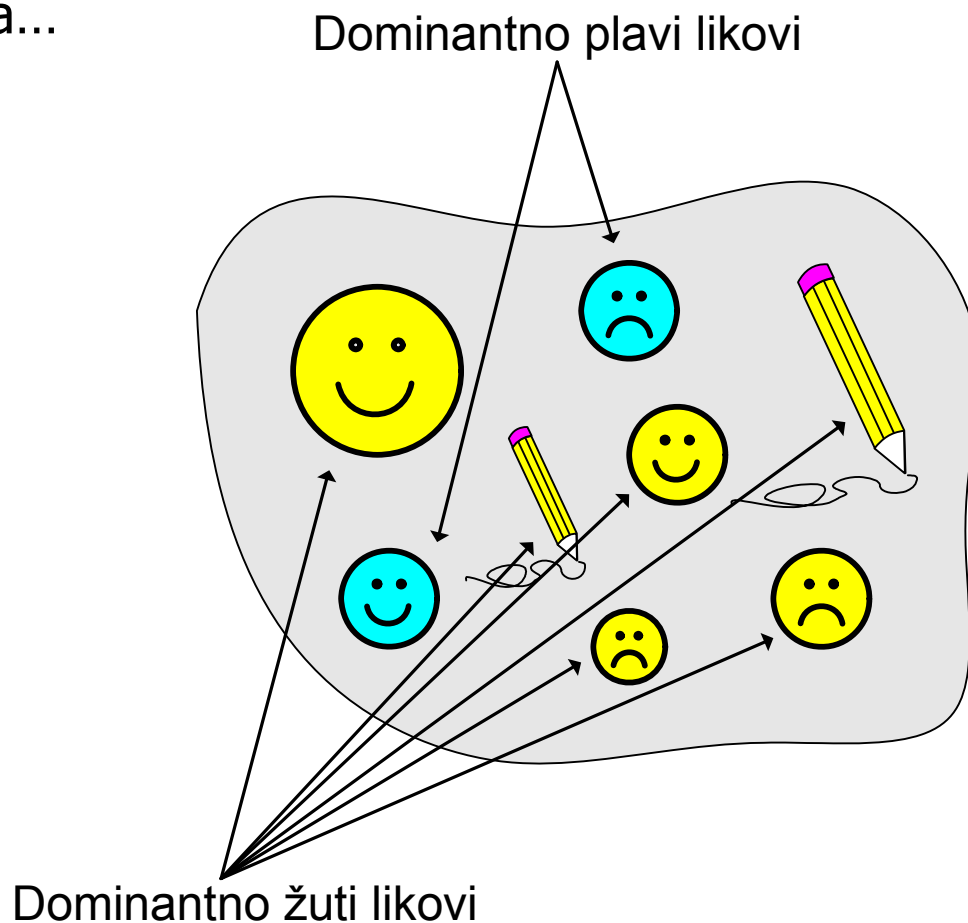
- Čovjek svakodnevno i neprestano obavlja – klasifikaciju!
- Postoji skup objekata/uzoraka koji imaju određen skup značajki.
- Klasifikacija – pridjeljivanje naziva/oznaka svakom uzorku iz prostora uzoraka.



- Uzorci
- Prostor uzoraka

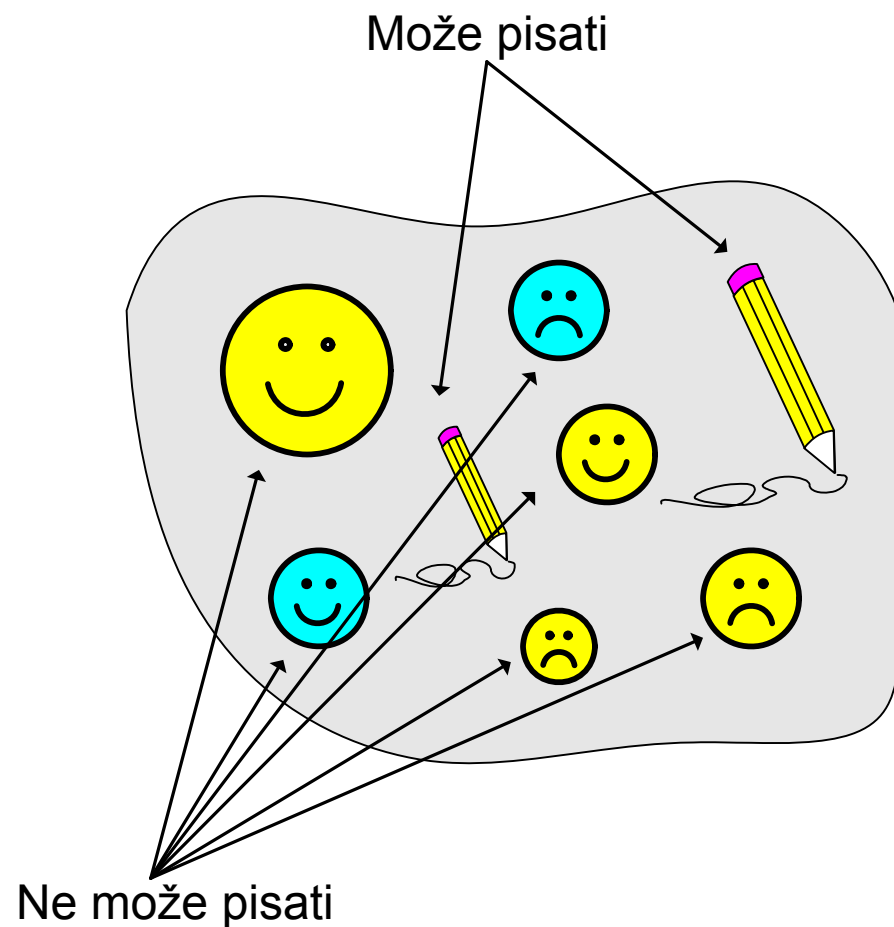
2.

- Jedna moguća klasifikacija...



2.

- Druga moguća klasifikacija...



- Svaki uzorak posjeduje određene **značajke** (boja, oblik, dimenzije, ...).
- *Klasifikaciju* činimo na temelju nekih od *značajki*.
- Uzoraka može biti beskonačno – ne možemo unaprijed generirati sve uzorke i pripadne klasifikacije → čovjek ipak dobro generalizira...
- Želimo postupak kojim bismo na temelju malog broja poznatih uzoraka mogli “naučiti” pravilno klasificirati neviđene primjere.

- Primjer koncepta "Dobra tajnica" – svaki objekt ima dvije značajke:
 - Značajka "Komunikativnost"
 - Značajka "Snalažljivost"
- Za potrebe računala značajke treba kvantizirati – računalo radi s brojevima.
- Ocjenjujemo značajke na skali od 1 do 5.
 - 1 je najlošija vrijednost.
 - 5 je najbolja vrijednost.
- Oznake "Dobra tajnica" i ne-"Dobra tajnica" kodiramo:
 - "Dobra tajnica" → 1
 - Ne-"Dobra tajnica" → -1

- Primjer "Dobra tajnica"

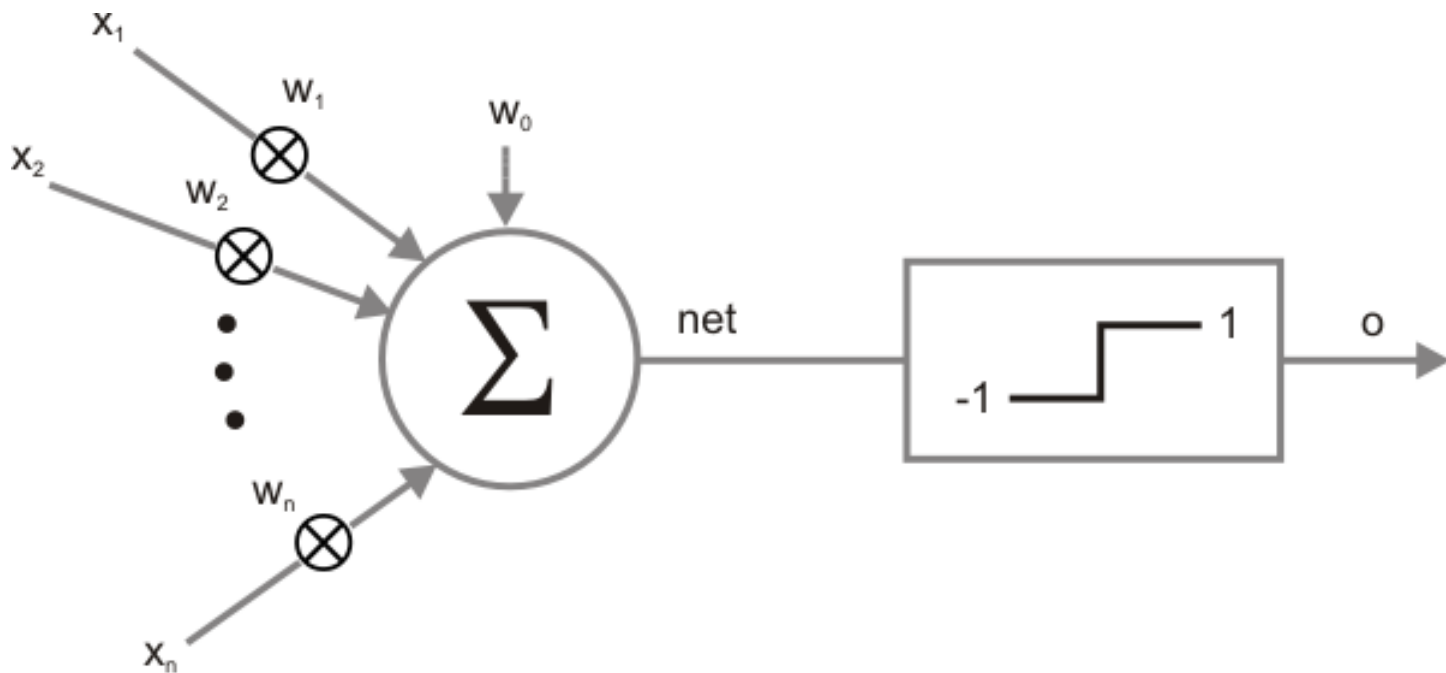
Komunikativnost	Snalažljivost	Oznaka	Kodirano
2	5	"Dobra tajnica"	1
5	2	"Dobra tajnica"	1
1	5	ne-"Dobra tajnica"	-1
5	1	ne-"Dobra tajnica"	-1

- Kada su *ulazi* i *izlazi* brojevi, za klasifikaciju možemo koristiti TLU perceptron.


2.

TLU perceptron

- 1943. McCulloch i Pitts – TLU Perceptron.



2.

- 1943. McCulloch i Pitts – TLU Perceptron.
- Dvije definicije prijenosne funkcije:
 - $\text{Step}(x) = 1, x \geq 0$
 - 
 - Step(x) = 0, $x < 0$
 - Step(x) = -1, $x < 0$
- Oba oblika koriste se podjednako.
- U primjerima ćemo koristiti TLU:-1,1.

2.

TLU perceptron

- Izlaz računamo u općem slučaju prema formuli:

$$o = \text{Step}(\text{net}) = \text{Step}(\vec{w}^T \cdot \vec{x}) = \text{Step} \left(\begin{bmatrix} w_n & \dots & w_0 \end{bmatrix} \cdot \begin{bmatrix} x_n \\ \vdots \\ x_1 \\ 1 \end{bmatrix} \right)$$
$$= \text{Step}(w_n \cdot x_n + \dots + w_1 \cdot x_1 + w_0)$$

- EksPLICITNO se uvodi: $x_0 = 1$

- Kada dolazi do promjene klasifikacije?

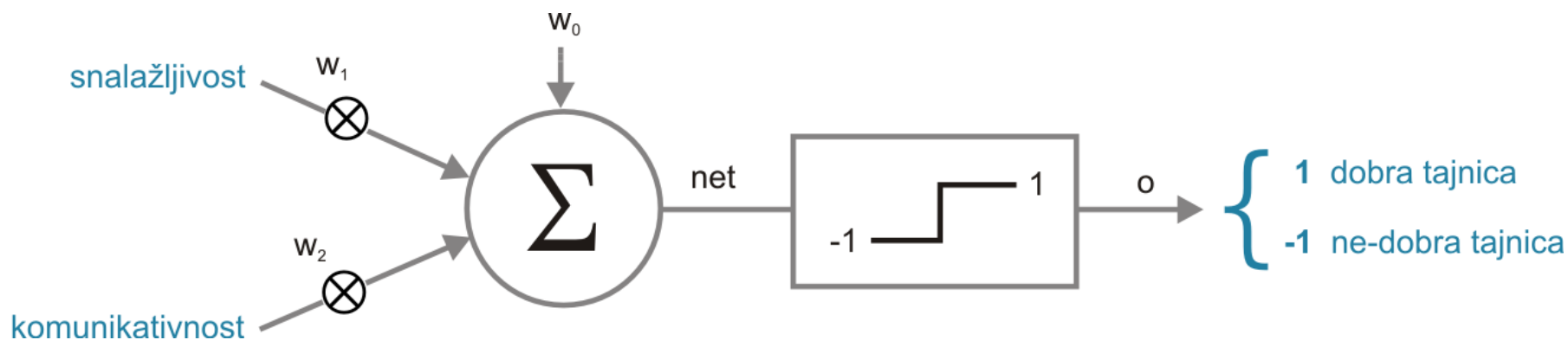
$$net = 0 \Rightarrow w_n \cdot x_n + \dots + w_1 \cdot x_1 + w_0 = 0$$

- Točke koje zadovoljavaju ovu jednadžbu čine granicu! (*decizijska funkcija*)
- Kada imamo dvije značajke granica je pravac.
- Više značajki rezultira hiperravninom.
- Bitno je uočiti – granica je **linearna**.

2.

TLU perceptron

- U našem primjeru perceptron ima dva ulaza:



- Snalažljivost $\equiv x_1$
- Komunikativnost $\equiv x_2$

- Pretpostavimo: $[w_2 \quad w_1 \quad w_0] = [1 \quad 1.3 \quad -5.85]$

2.

TLU perceptron

- U našem primjeru izlaz računamo prema formuli:

$$o = \text{Step} \left(\begin{bmatrix} w_2 & w_1 & w_0 \end{bmatrix} \cdot \begin{bmatrix} x_2 \\ x_1 \\ 1 \end{bmatrix} \right) = \text{Step}(w_2 \cdot x_2 + w_1 \cdot x_1 + w_0)$$

Težinski faktori			Komun.	Snal.	Suma	Izlaz	Točan	Ispravno
w_2	w_1	w_0	x_2	x_1	net	$o=S(\text{net})$	t	
1	1.3	-5.85	2	5	2,65	1	1	DA
1	1.3	-5.85	5	2	1,75	1	1	DA
1	1.3	-5.85	1	5	1,65	1	-1	NE
1	1.3	-5.85	5	1	0,45	1	-1	NE

- 1949. Hebb – “*učiti znači mijenjati jakosti veza*”!
- Potrebno je mijenjati težinske faktore.
- 1958. Rosenblatt: spoj Hebbove ideje i McCulloch-Pitts modela
- **Pravilo Perceptrona**
 - Ukoliko se uzorak klasificira **ispravno** ▶ **ne radi korekciju**
 - Ukoliko se uzorak klasificira **neispravno** ▶ **primjeni korekciju**
 - Ciklički uzimaj sve uzorke redom, a postupak zaustavi kada su svi uzorci za redom klasificirani ispravno
- Korekcija:
$$w_i(k + 1) = w_i(k) + \eta \cdot (t - o) \cdot x(k)$$

- Učenje perceptrona uz:

- $\eta = 0.02$

- $[w_2 \quad w_1 \quad w_0] = [1 \quad 1.3 \quad -5.85]$

- Postupak završava s:

$$[w_2 \quad w_1 \quad w_0] = [0.92 \quad 0.94 \quad -5.93]$$

<http://www.zemris.fer.hr/predmeti/nenr/simulacije/fsit/fsitprg.html>

2.

- Kako se klasificira novi uzorak [5 5]?

$$\text{Step}(0.92 \cdot 5 + 0.94 \cdot 5 - 5.93) = \text{Step}(3.37) = 1$$

- Kako se klasificira novi uzorak [1 1]?

$$\text{Step}(0.92 \cdot 1 + 0.94 \cdot 1 - 5.93) = \text{Step}(-4.07) = -1$$

- Vidimo da je perceptron NAUČIO klasificirati uzorke koje prethodno nije vidio!

- Problem s mrežama TLU perceptrona jest nemogućnost učenja konvencionalnim postupcima!
- Minsky-Papert (1969./1972.)
Perceptrons: An Introduction to Computational Geometry
 - Formalna analiza perceptrona; istaknuta dobra svojstva, ali ...
 - Istaknute nemogućnosti
 - Jedan perceptron ne može riješiti problem XOR-a
 - Troslojna unaprijedna mreža perceptrona (jedan skriveni sloj) ne može ostvariti sve "predikate" osim ako nema barem jedan neuron u prvom sloju koji je spojen na sve ulaze težinama različitim od nule (→ nema svojstvo lokalnosti neurona).
 - *Zima umjetne inteligencija*; kraj konektivizma do kasnih osamdesetih
- *Credit assignment problem* kod višeslojnih mreža → zasluge za pogrešku?
- Rješenje tražimo na drugom mjestu...

2.

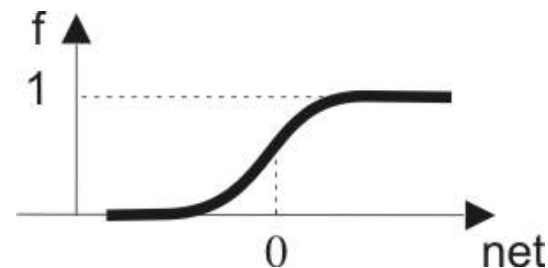
Nelinearna prijenosna funkcija?

Linearna kombinacija linearnih kombinacija opet je linearna kombinacija...

Trebamo nelinearne prijenosne funkcije!

→ Često se koristi sigmoidalna (logistička) prijenosna funkcija.

$$f(\text{net}) = \frac{1}{1 + e^{-a \cdot \text{net}}}$$



- Ovakve mreže su univerzalni aproksimatori – mogu aproksimirati proizvoljnu funkciju s proizvoljnom preciznošću.

1. Uvod u neuro-računarstvo

Povijesni pregled • Pravci AI • Definicija i vrste • Primjena • Učenje • Živčani sustav i biološki neuron • Model neurona i vrste • Arhitektura mreža

2. Procesni elementi

Perceptron • Pravilo perceptrona • Primjer • Učenje booleovih funkcija • Linearna odvojivost • XOR problem

3. Backpropagation algoritam

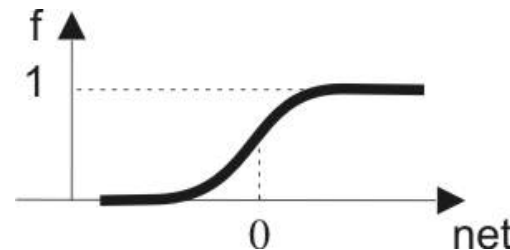
BACKPROPAGATION algoritam • Primjeri uporabe neuronskih mreža • Primjer sustava za raspoznavanje

3.

BACKPROPAGATION algoritam

- Kako bi mreža mogla predstaviti visoko nelinearne funkcije, prijenosna funkcija procesnih elemenata mora i sama biti **nelinearna funkcija**
- Želimo li koristiti gradijentne metode, prijenosna funkcija mora biti i **derivabilna**
- Rješenje ▶ sigmoidalna funkcija

$$f(\text{net}) = \frac{1}{1 + e^{-a \cdot \text{net}}}$$

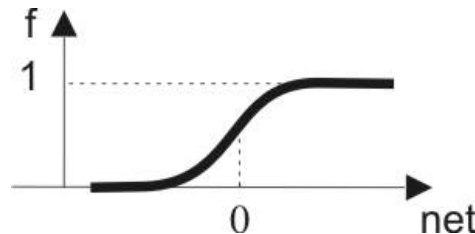


koja ima prvu derivaciju $df(x)/dx = f(x) [1-f(x)]$

3.

BACKPROPAGATION algoritam

- Koristimo neuron sa sigmoidalnom funkcijom



- Algoritam koristi metodu gradijentnog spusta kako bi minimizirao nastalu pogrešku $E(\vec{w})$ na izlazu mreže nad skupom primjera za učenje D

$$E(\vec{w}) = \frac{1}{2} \sum_{d \in D} (t_d - o_d)^2$$

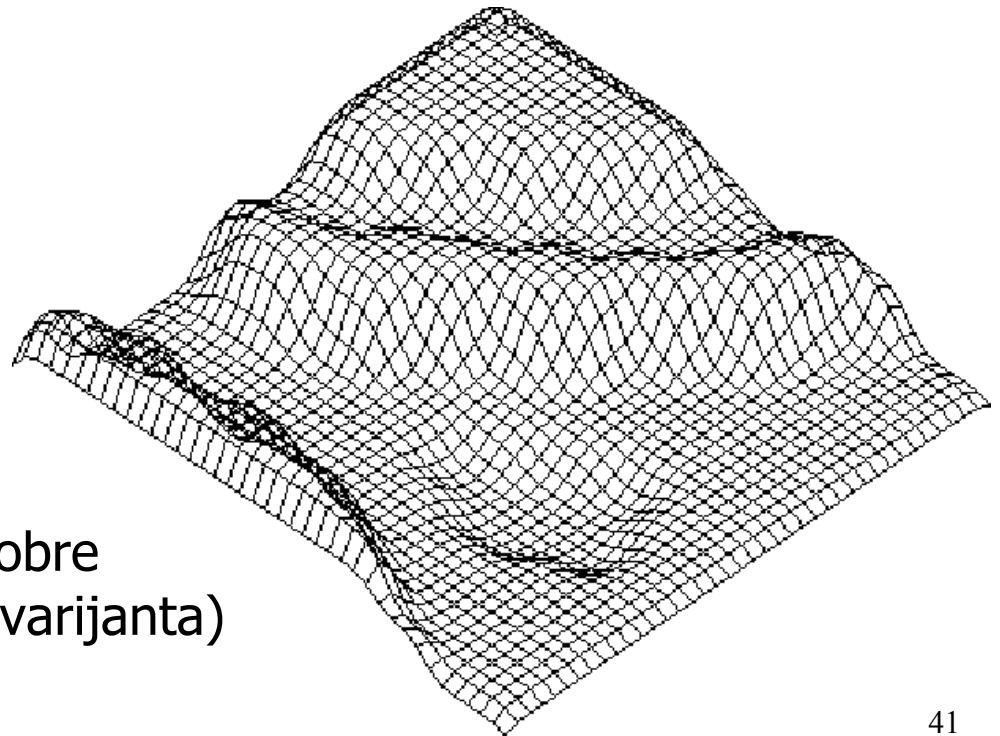
- Kod višeslojne mreže izlazni sloj može sačinjavati više neurona, pa definiramo pogrešku kao

$$E(\vec{w}) = \frac{1}{2} \sum_{d \in D} \sum_{k \in \text{outputs}} (t_{kd} - o_{kd})^2$$

3.

BACKPROPAGATION algoritam

- Učenje se svodi na pretraživanje u n -dimenzijskom prostoru hipoteza ($n = \text{ukupan broj težina}$)
- Površina pogreške $E(\mathbf{w})$ nije parabolička kao kod jednog procesnog elementa, već sadržava brojne **lokale minimume**

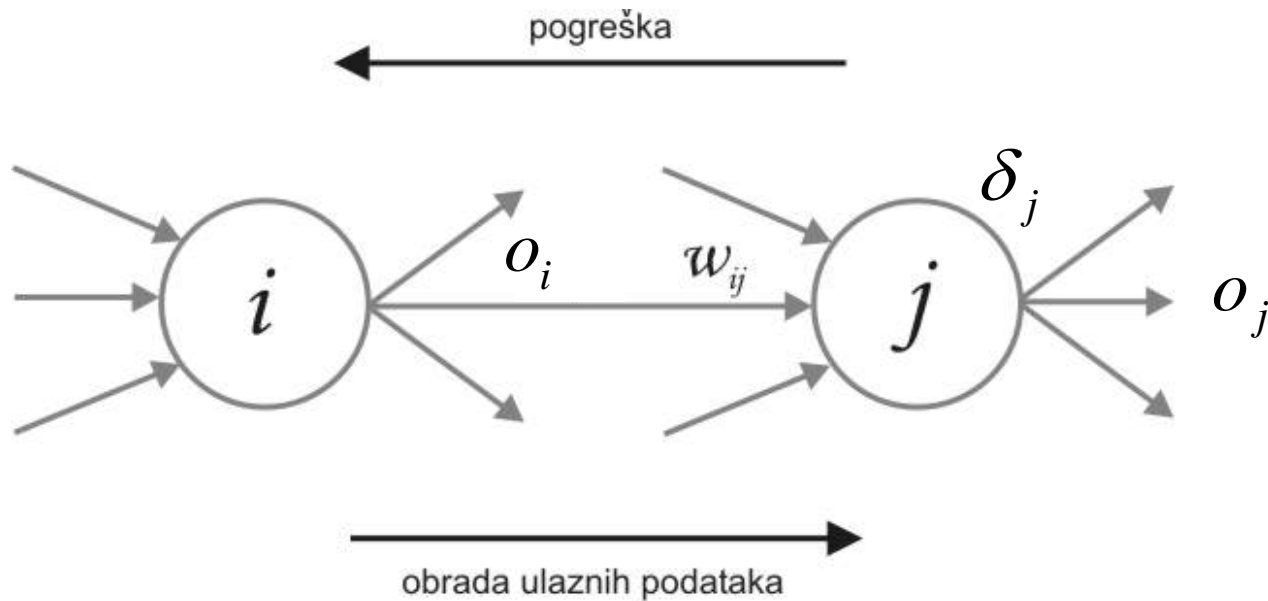


- Unatoč tome algoritam daje dobre rezultate (osobito *stohastička* varijanta)

3.

BACKPROPAGATION algoritam

- Notacija:



3.

BACKPROPAGATION algoritam

Inicijaliziraj težinske faktore na slučajne vrijednosti

Dok nije ispunjen uvjet zaustavljanja **čini**

Za svaki (\mathbf{x}, \mathbf{t}) iz D **čini**

Izračunaj izlaz o_u za svaku jedinicu u

Za svaku **izlaznu** jedinicu k izračunaj pogrešku δ_k

$$\delta_k \leftarrow o_k(1-o_k)(t_k - o_k)$$

Za svaku **skrivenu** jedinicu izračunaj pogrešku

$$\delta_h \leftarrow o_h(1-o_h) \sum_{s \in \text{Downstream}(h)} \omega_{hs} \delta_s$$

Ugodi svaki težinski faktor w_{ij}

$$\omega_{ij} \leftarrow \omega_{ij} + \Delta\omega_{ij}$$

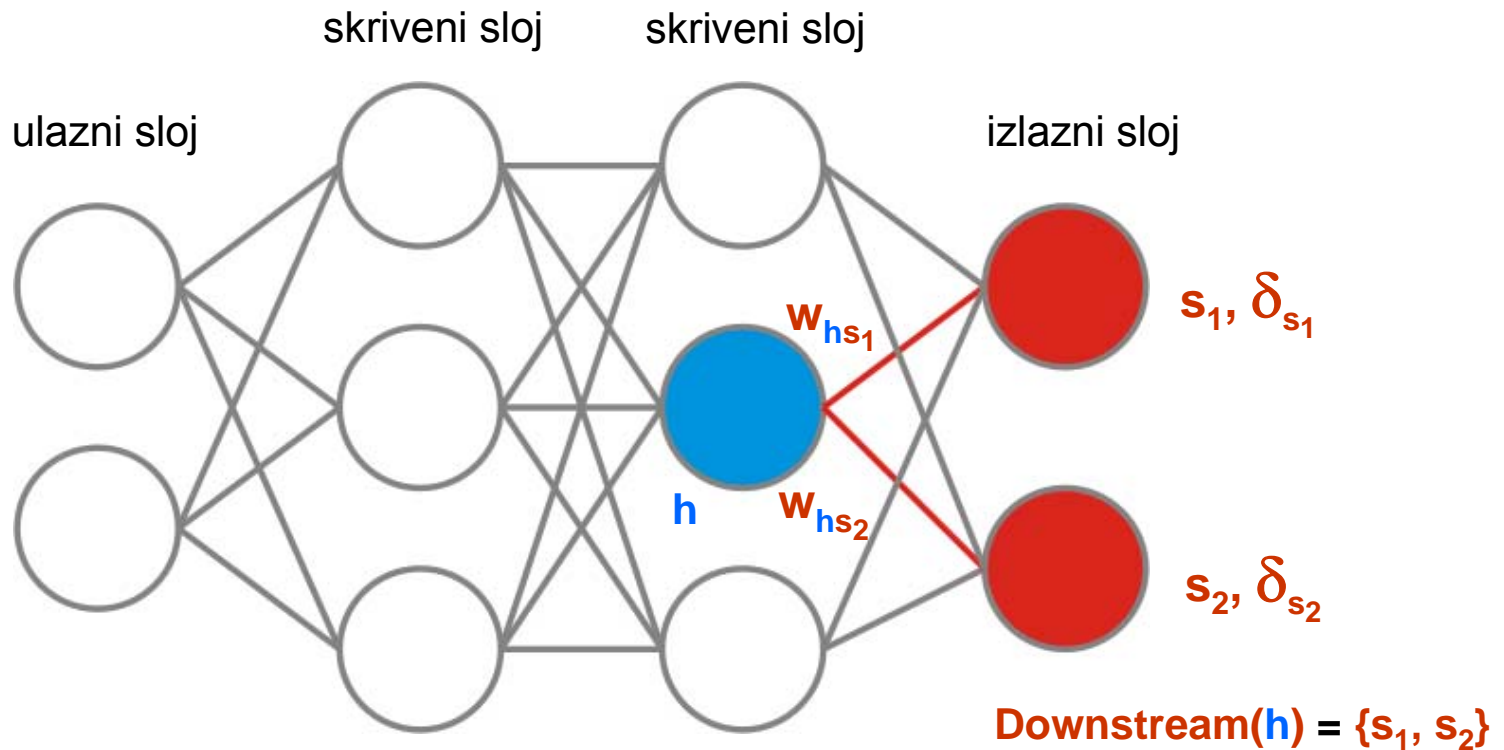
gdje je $\Delta\omega_{ij} = \eta\delta_j o_i$

Kraj

Kraj

3.

BACKPROPAGATION algoritam



$$\delta_{s_1} = o_{s_1} (1 - o_{s_1}) (t_{s_1} - o_{s_1})$$

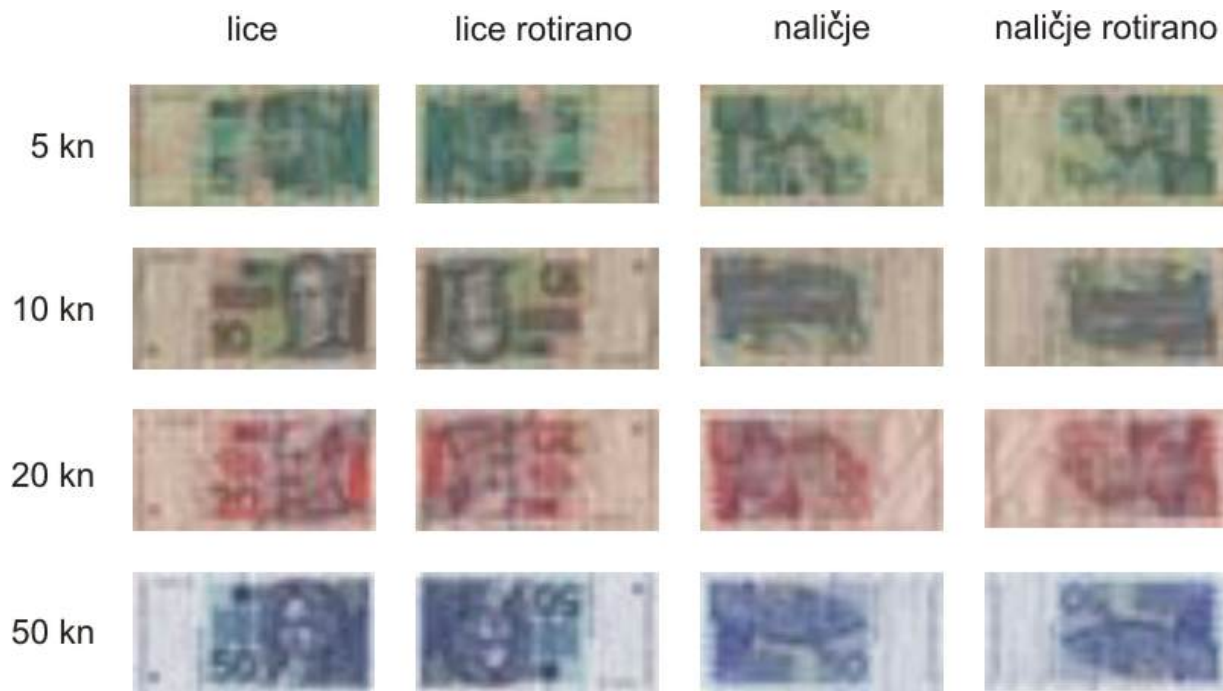
$$\delta_{s_2} = o_{s_2} (1 - o_{s_2}) (t_{s_2} - o_{s_2})$$

$$\delta_h = o_h (1 - o_h) (w_{hs_1} \delta_{s_1} + w_{hs_2} \delta_{s_2}) \quad \delta_h \text{ će služiti za ugađanje ulaza u } h$$

3.

Primjer: Raspoznavanje novčanica

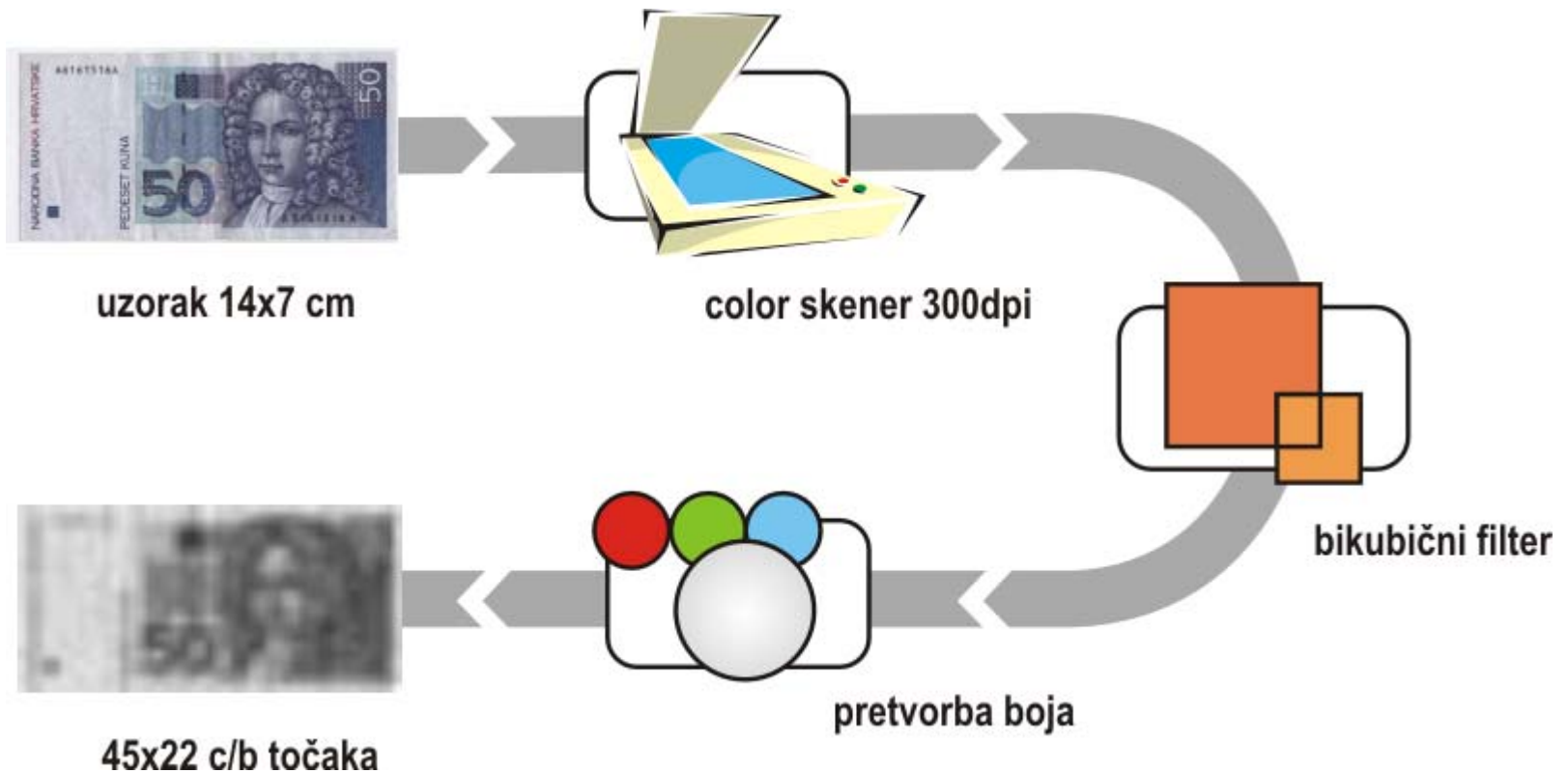
- **Zadatak** ▶ klasificiranje četiri vrste papirnatih novčanica neovisno o orijentaciji
- Skup primjera za učenje sačinjava **16** različitih uzoraka uzorkovanih s **45x22** slikovna elementa:



3.

Primjer: Raspoznavanje novčanica

- Uzorci se prije dovođenja na ulaze ANN predprocesiraju:



3.

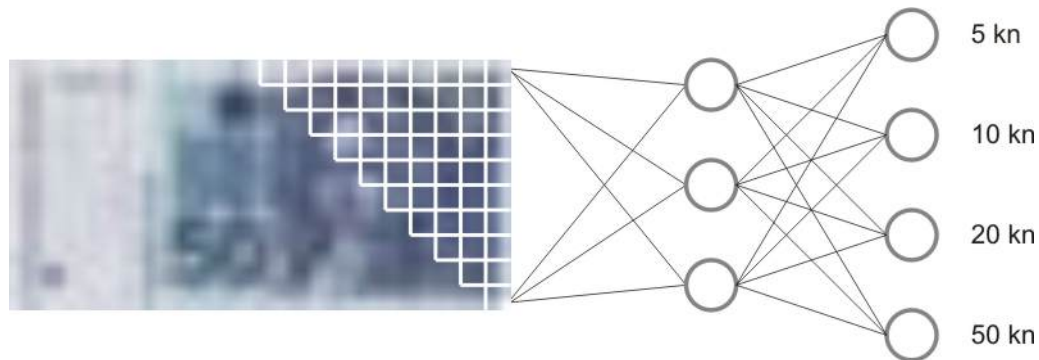
Primjer: Raspoznavanje novčanica

- Uzorci su u stvari digitalizirane slike, pa će se međusobno razlikovati u intenzitetu odgovarajućih slikovnih elemenata (posljedica razlike u stupnju istrošenosti novčanice, izgužvanosti papira, oštećenosti).
- U primjeru je korišten **generator umjetnih uzoraka** koji generira uzorke različitog stupnja oštećenja u svrhu provjere rada mreže i ugađanja njezinih parametara.
- Prednosti:
 - dostupnost velikog broja uzoraka za provjeru,
 - mogućnost promatranja odziva mreže u ovisnosti o promjeni samo jednog parametra uzorka,
 - mogućnost definiranja praga tolerancije oštećenih uzoraka.
- Nedostaci:
 - generirani uzorci nisu u potpunosti stohastičke prirode,
 - generatorom se rijetko mogu obuhvatiti sva obilježja pravih uzoraka.

3.

Primjer: Raspoznavanje novčanica

- **Parametri mreže** ▶ aciklička potpuno spojena uniformno-slojevita mreža strukture **990x3x4**, stopa učenja=**0.02**, moment=**0.02**, provjera nad skupom za testiranje svakih **2500** epoha



- Rezultat klasifikacije odgovara neuronu čiji je izlaz najveći
- Mreža nije učena s anti-primjerima, no međusoban odnos izlaznih vrijednosti može poslužiti kao **mjera pouzdanosti** klasifikacije
- Moguće je utvrditi neki prag tolerancije oštećenja uzorka

- Neuroračunarstvo: vrlo živo i aktivno područje zanimljivo inženjerima, biolozima, neurolozima, psiholozima, medicini, ...
- Puno različitih vrsta neuronskih mreža: unaprijedne, samoorganizirajuće, rekurentne, ...
- Omogućavaju učenje iz samih podataka, s učiteljem ili bez njega
- Rješavaju zadatke koje mi ne znamo algoritamski opisati i riješiti
- Izuzetno robusne
- Primjena: modeliranje složenih procesa, regresija, predikcija trendova, raspoznavanje uzoraka, klasifikacijski problemi, ...

- T. M. Mitchell, *Machine Learning*. The McGraw-Hill Companies, Inc., 1997.
- R. S. Michalski, I. Bratko, M. Kubat, *Machine Learning And Data Mining*, John Wiley & Sons Ltd., 1998
- P. Picton, *Neural Networks*. PALGRAVE, 1994.
- B. Dalbelo Bašić, Bilješke s predavanja. Fakultet elektrotehnike i računarstva, Zagreb, 2001.
- K. Gurney, "Computers and Symbols versus Nets and Neurons". Dept. Human Sciences, Brunel University, Uxbridge, 2001.
- K. Gurney, "Drawing things together – some perspectives". Dept. Human Sciences, Brunel University, Uxbridge, 2001.
- D. Mišljenčević, I. Maršić, *Umjetna inteligencija*. Školska knjiga, Zagreb, 1991.
- *Automata Theory*. Encyclopaedia Britannica Inc., 2001 CD-ROM Edition.