

Основне академске студије  
Информациони инжењеринг

Основи рачунарске интелигенције

# Увод у неуронске мреже

(материјали за предавања)

1. Биолошки неурон
2. Вештачки неурон
3. Неуронска мрежа
4. Обучавање неуронске мреже
5. Извори и литература

## Неурон

### нервна ћелија

део нервног система организма

специјализована ћелија која прима и шаље хемијске или електричне сигнале

### основна структура

сома (тело)

наставци

дендрит

прима сигнале од других ћелија

аксон

преноси сигнале до других ћелија

## Неурон – основне врсте

униполарни

обухвата један наставак

биполарни

обухвата један аксон и један дендрит

мултиполарни

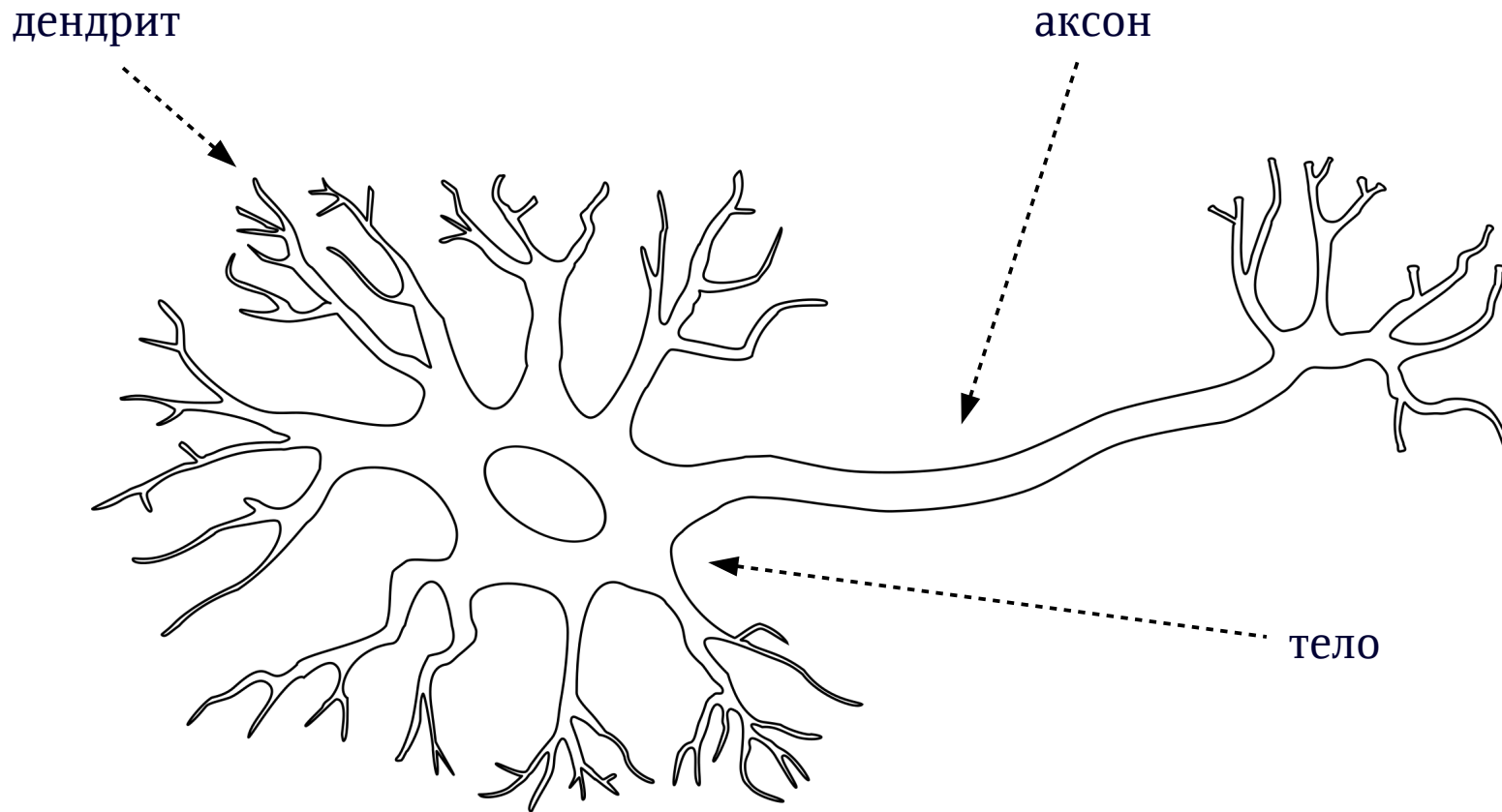
обухвата један аксон и више дендрита

псеудоуниполарни

обухвата један наставак који се даље грана на два

# Биолошки неурон

Неурон – слободна илустрација мултиполарног неурона



## Неурон – повезаност

неурони могу бити повезани међусобно

али и с другим врстама ћелија

синапса

веза између аксона једног и дендрита другог неурона

постоје и другачије изграђене везе

врсте синапси

хемијске

електричне

неурони примају и шаљу сигнале

ако је примљен довољно јак сигнал, шаље се сигнал другим ћелијама

пренос сигнала се спроводи преко хемијских једињења (неуротрансмитера)

## Неурон – пренос сигнала

### мембрански потенцијал

постоји разлика у наелектрисању између унутрашњости и спољашњости ћелије

ћелијска мембрана раздваја унутрашњу од спољашње средине  
мембрански потенцијал мировања је око  $-70 \text{ mV}$  (за одређене случајеве)

## Неурон – пренос сигнала

приликом преноса сигнала преко неурона долази до обртања мембранског потенцијала

при побуђивању неурона пријемом неуротрансмитера почиње деполаризација мембране

смањује се разлика потенцијала унутрашњости и спољашњости ћелије  
након достизања граничног потенцијала наступа потпуна деполаризација  
гранични потенцијал је  $-55 \text{ mV}$  (за одређене случајеве)

потпуна деполаризација води до вршног акционог потенцијала  
вршни акциони потенцијал је око  $40 \text{ mV}$  (за одређене случајеве)

почиње реполаризација

мембрански потенцијал постаје поново негативан

остварује се хиперполаризација

мембрански потенцијал се спушта испод нивоа потенцијала мировања

поново се успоставља мембрански потенцијал мировања

мембрански потенцијал долази до нивоа потенцијала мировања

## Неурон – пренос сигнала

настали акциони потенцијал преноси се преко аксона

преноси се као електрични импулс до завршетака аксона, где се дешава отпуштање неуротрансмитера и предаја сигнала другој ћелији

## Неурон – повезаност

људски мозак

сложен

број садржаних неурона је реда  $10^{11}$  (око 86 милијарди)

густо повезан

број веза по неурону је реда  $10^4$

1. Биолошки неурон
- 2. Вештачки неурон**
3. Неуронска мрежа
4. Обучавање неуронске мреже
5. Извори и литература

## Вештачки неурон

математички модел биолошког неурона

битно поједностављен у односу на биолошки неурон

задржани основни принципи понашања биолошког неурона

улазни сигнали

удруживање улазних сигнала

активација

излазни сигнали

намењен израчунавањима

користи у машинском учењу

зачеци у моделовању биолошког неурона

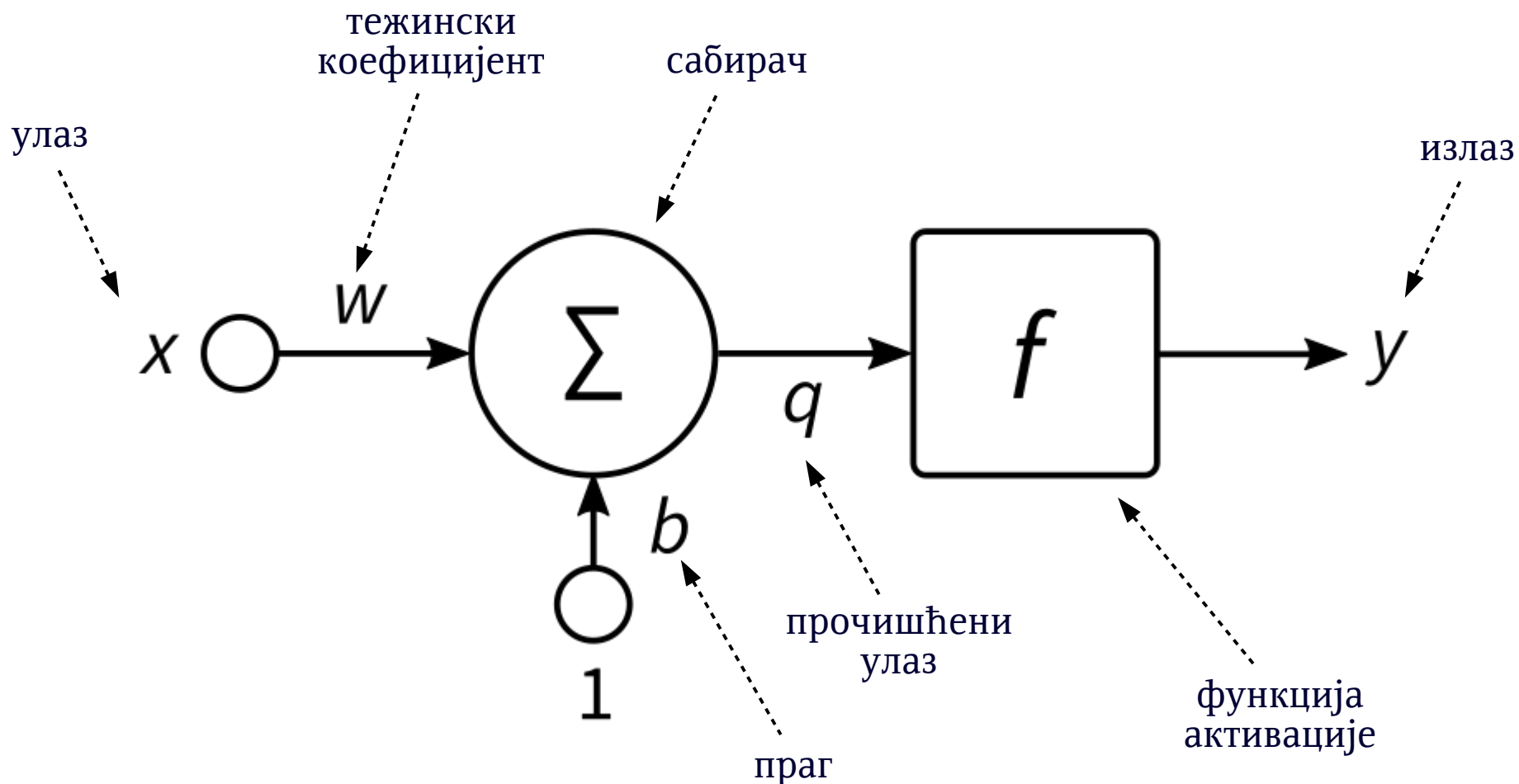
рад Ворена Макалока и Волтера Питса (1943)

# Вештачки неурон

## Вештачки неурон

модел с једним улазом

$$y = f(wx + b)$$



# Вештачки неурон

## Вештачки неурон

модел с једним улазом – елементи

улаз  $x$  (енгл. *input*)

скалар

представља улазни сигнал

одговара сигналу који преко дендрита прима биолошки неурон

тежински коефицијент (тежина)  $w$  (енгл. *weight*)

скалар

тежински коефицијент придружен улазу  $x$

подесиви параметар

одговара снази синаптичке везе код биолошког неурона

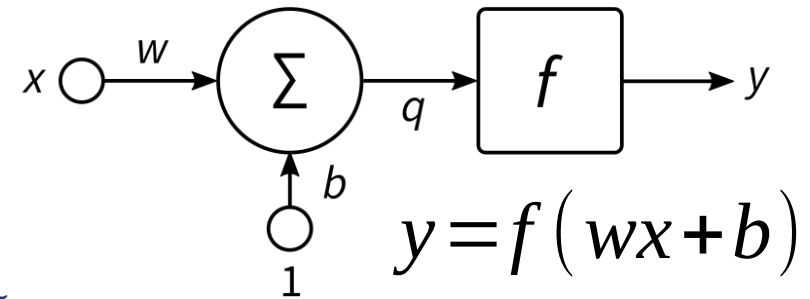
праг  $b$  (енгл. *bias*)

скалар

пријемчивост

подесиви параметар

одговара граничном потенцијалу код биолошког неурона



# Вештачки неурон

## Вештачки неурон

модел с једним улазом – елементи

сабирач  $\Sigma$  (енгл. *summer*)

функција која обједињује

улаз с тежинским коефицијентима и праг  
одговара акумулацији побуде код биолошког неурона  
у односу на гранични потенцијал

прочишћени улаз  $q$  (енгл. *net input*)

скалар

$$q = wx + b$$

одговара стању код биолошког неурона у погледу потенцијала

функција активације  $f$  (енгл. *activation function*)

функција која на основу прочишћеног улаза генерише излаз

функција преноса (енгл. *transfer function*)

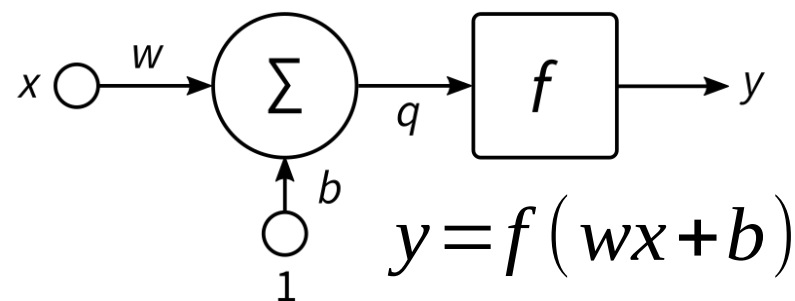
одговара понашању биолошког неурона у погледу генерисања сигнала

излаз  $y$  (енгл. *output*)

скалар

представља излазни сигнал

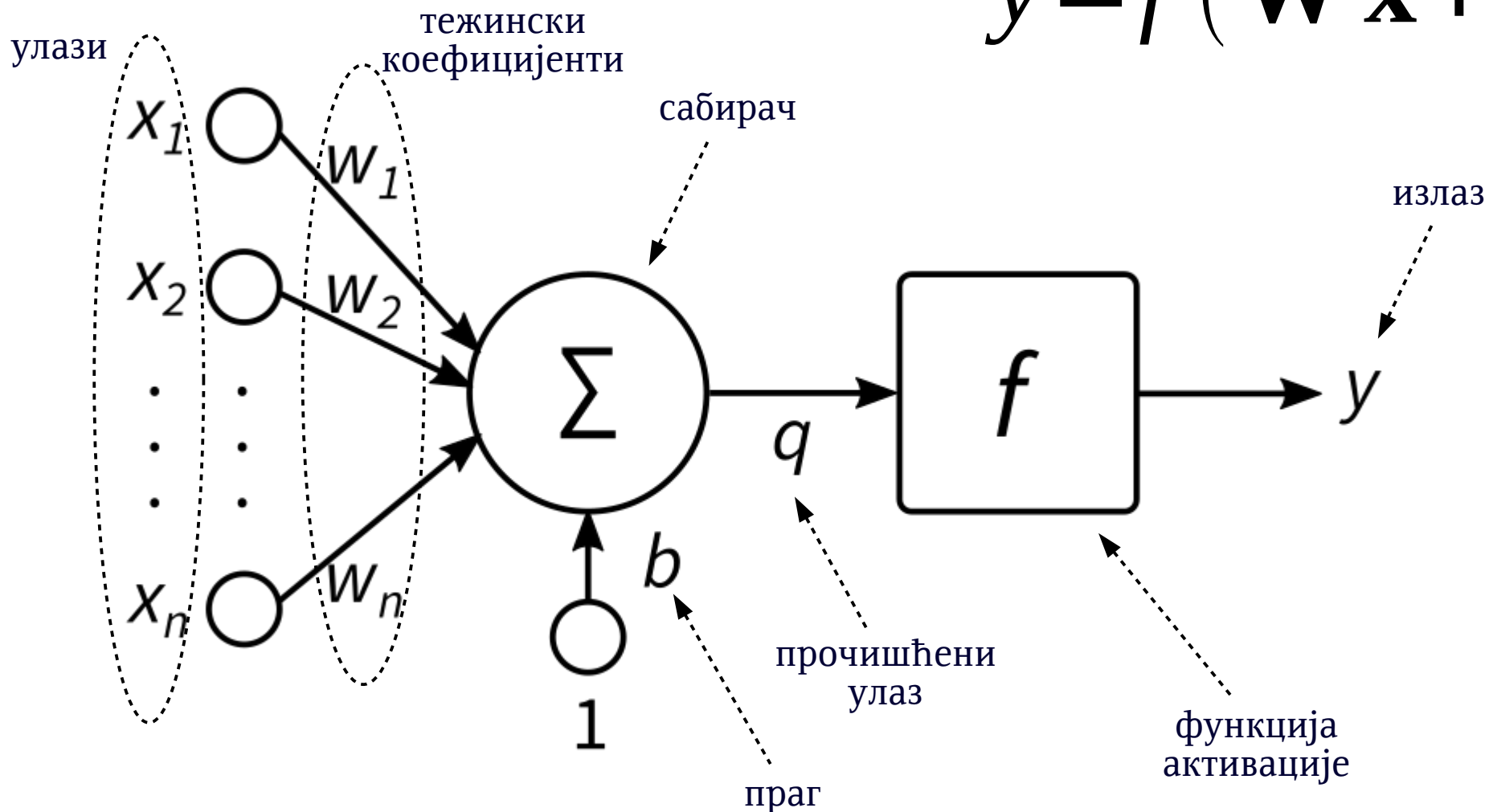
одговара сигналу који преко аксона шаље биолошки неурон



# Вештачки неурон

## Вештачки неурон

модел с више улаза



$$y = f(\mathbf{W}\mathbf{x} + b)$$

# Вештачки неурон

## Вештачки неурон

модел с више улаза – елементи

улази  $\mathbf{x}$

вектор дужине  $n$

представља улазне сигнале

тежински коефицијенти (тежине)  $\mathbf{W}$

матрица димензија  $1 \times n$

тежински коефицијенти придружени улазима  $\mathbf{x}$

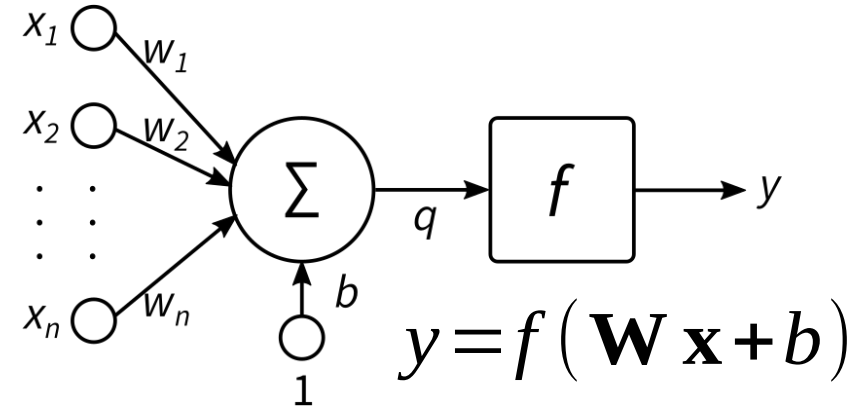
подесиви параметар

праг  $b$

скалар

пријемчивост

подесиви параметар



# Вештачки неурон

## Вештачки неурон

модел с више улаза – елементи

сабирач  $\Sigma$

функција која обједињује  
улазе с тежинским коефицијентима и праг

прочишћени улаз  $q$

скалар

$$q = \mathbf{W}\mathbf{x} + b$$

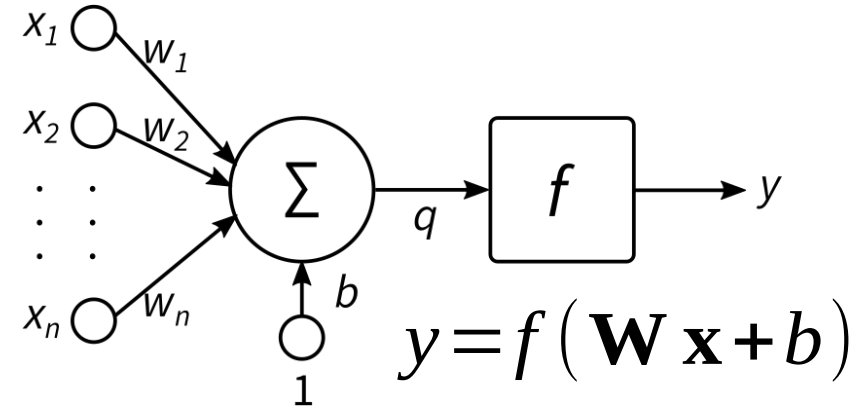
функција активације  $f$

функција која на основу прочишћеног улаза генерише излаз

излаз  $y$

скалар

представља излазни сигнал



## Функција активације – примери

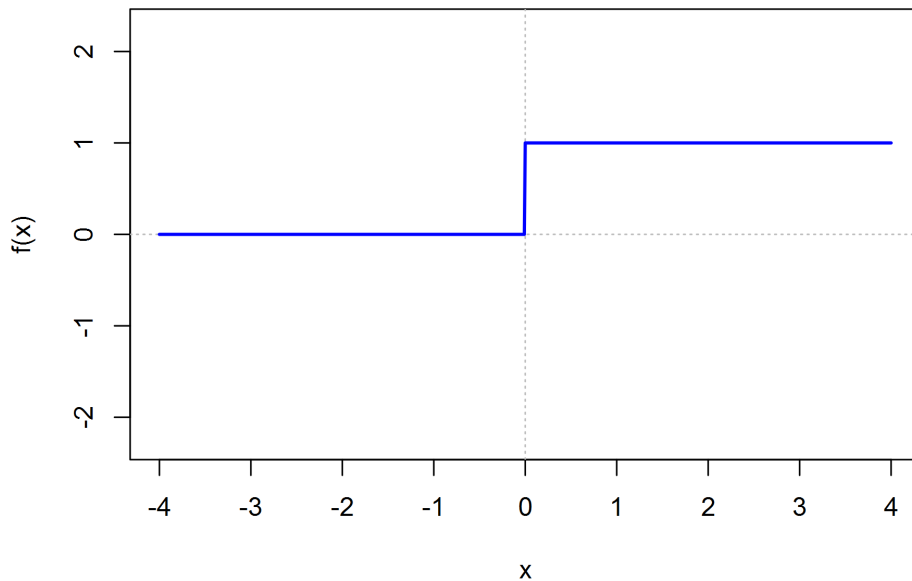
*степенаста – бинарна*

$$f(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases}$$

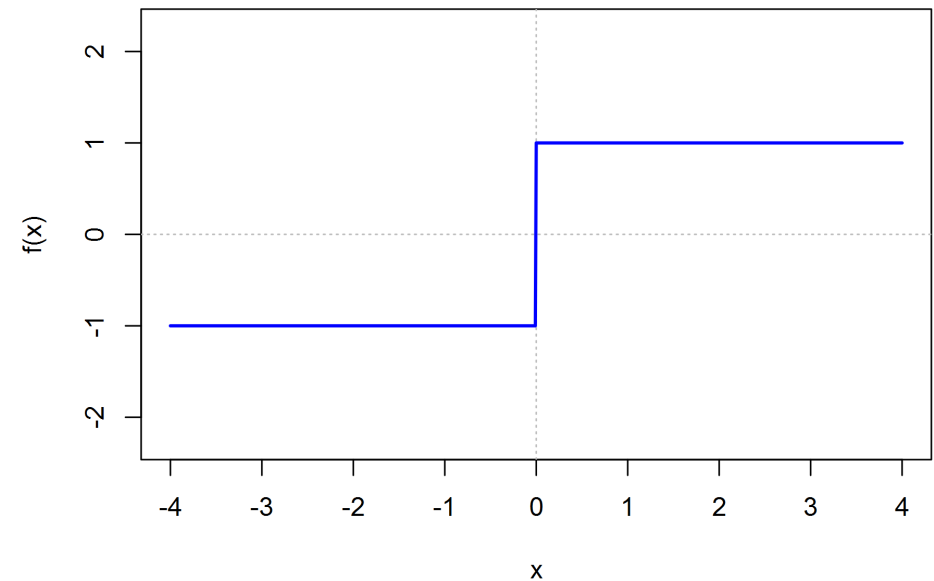
*степенаста – симетрична бинарна*

$$f(x) = \begin{cases} -1, & x < 0 \\ 1, & x \geq 0 \end{cases}$$

**Степенаста – бинарна**

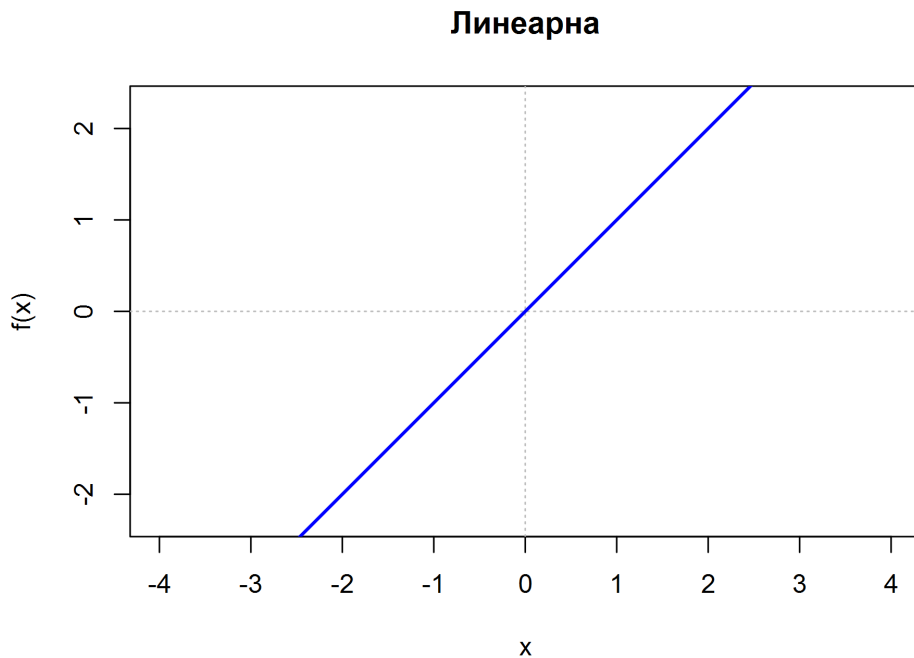


**Степенаста – симетрична бинарна**



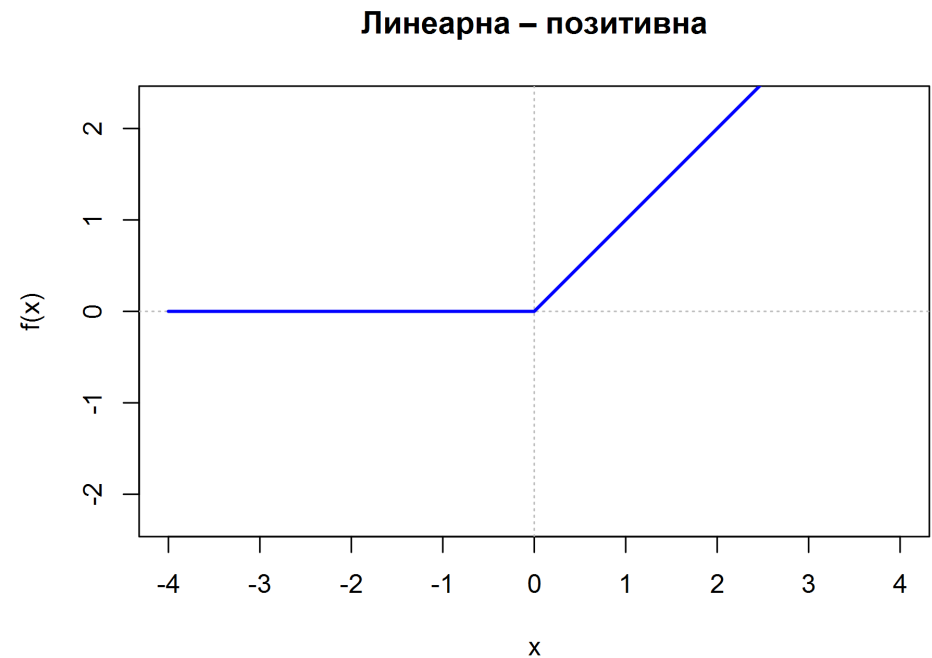
## Функција активације – примери

*линеарна*  
 $f(x) = x$



*линеарна – позитивна*

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases}$$



## Функција активације – примери

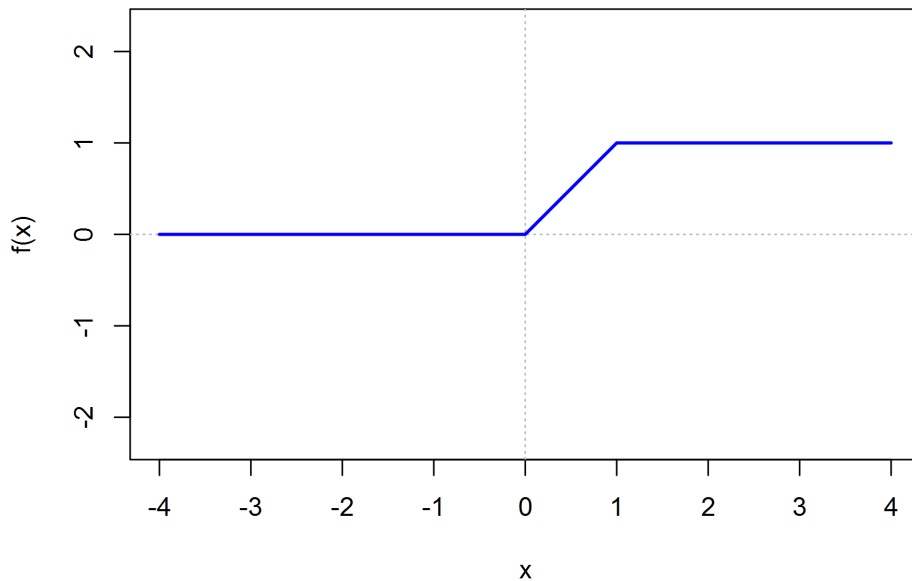
*линеарна – са засићењем*

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & 0 \leq x \leq 1 \\ 1, & x > 1 \end{cases}$$

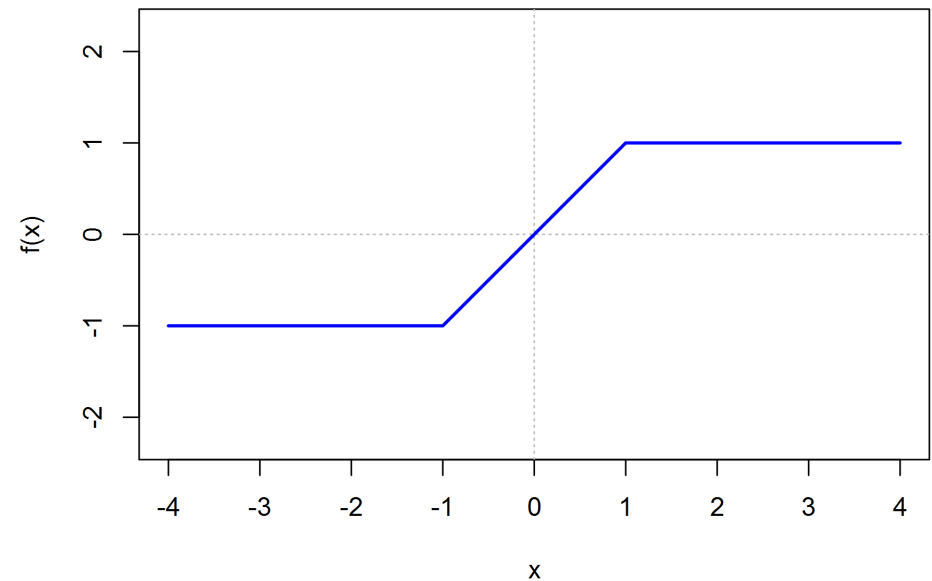
*линеарна – симетрична са засићењем*

$$f(x) = \begin{cases} -1, & x < -1 \\ x, & -1 \leq x \leq 1 \\ 1, & x > 1 \end{cases}$$

Линеарна – са засићењем



Линеарна – симетрична са засићењем

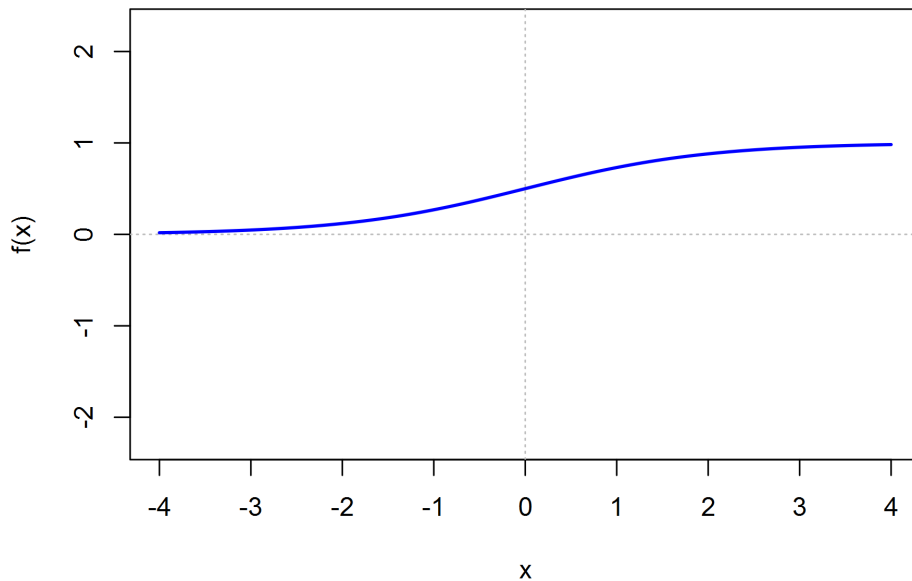


## Функција активације – примери

*сигмоидна – логистичка*

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

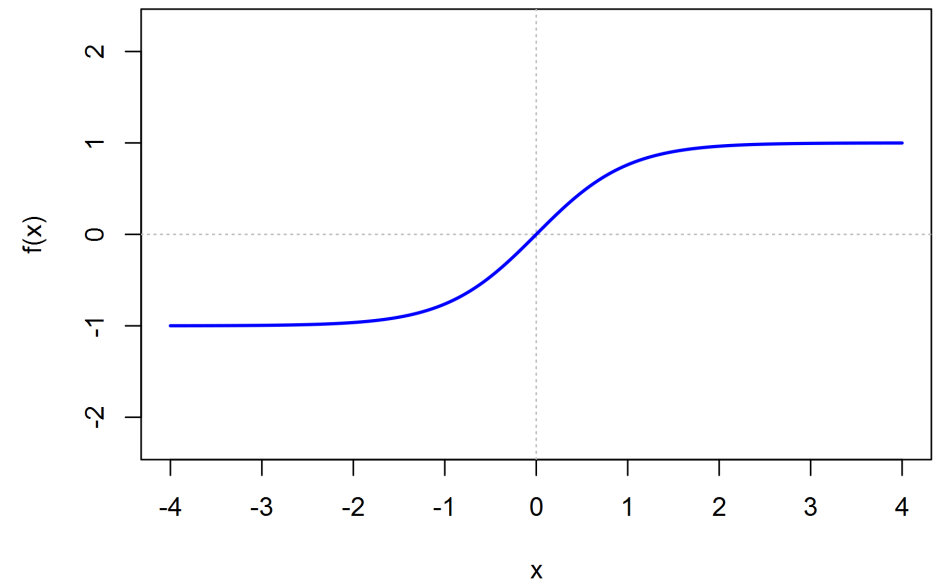
Сигмоидна – логистичка



*сигмоидна – хиперболички тангенс*

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

Сигмоидна – хиперболички тангенс



# Вештачки неурон

## Вештачки неурон

стохастички модел

увођење случајности у понашање неурона

пример

два могућа стања неурона

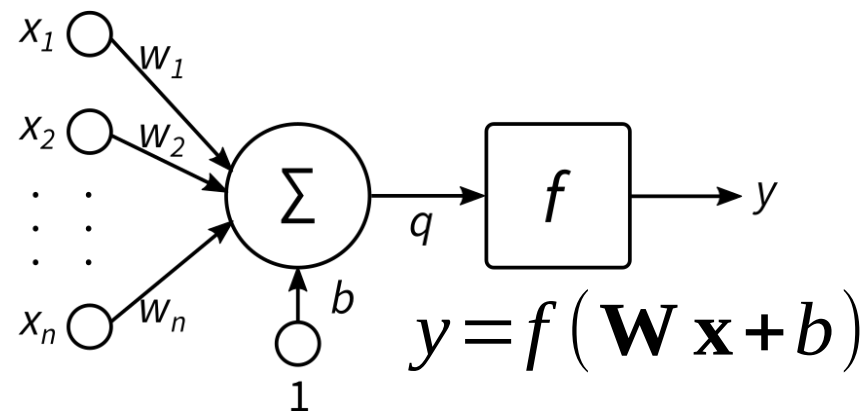
$$f(q) = \begin{cases} -1, & \text{с вероватноћом } 1 - P(q) \\ 1, & \text{с вероватноћом } P(q) \end{cases}$$

$$P(q) = \frac{1}{1 + e^{-\frac{q}{T}}}$$

псеудотемпература  $T$

елемент случајности

када  $T$  тежи 0, понашање приближава понашању класичног модела



1. Биолошки неурон
2. Вештачки неурон
- 3. Неуронска мрежа**
4. Обучавање неуронске мреже
5. Извори и литература

## Вештачка неуронска мрежа (неуронска мрежа)

неуронска мрежа као скуп повезаних вештачких неурона  
добија улаз и даје излаз

### основне архитектуре неуронских мрежа

неуронска мрежа организована кроз слојеве

слој као група неурона који активни у паралели и деле исте улазе

врсте неуронских мрежа по броју слојева и повезаности слојева

једнослојна

вишеслојна

рекурентна

## Једнослојна неуронска мрежа

обухвата један слој

сигнал бива преношен само унапред (од улаза ка излазима)

енгл. *single-layer feedforward network*

слој садржи више неурона

улази се прослеђују целом слоју

сваки улаз до сваког неурона у слоју

излази се добијају из целог слоја

сваки неурон даје излаз

могућност подешавања

избор броја неурона

димензија улаза и број неурона у слоју не морају бити исти

избор функције активације може за сваки неурон посебно

# Неуронска мрежа

## Једнослојна неуронска мрежа

### један општи облик

постоји укупно  $n$  улаза

улази обједињени у вектор  $\mathbf{x}$  дужине  $n$

постоји укупно  $m$  неурона

сваки садржи праг, сабирач и функцију активације

сваки прихвата све улазе

сваки генерише по један излаз

сваки улаз повезан са сваким неуроном

за сваку везу улаз–неурон постоји једна тежина

тежине обједињене у матрици  $\mathbf{W}$

димензија  $m \times n$

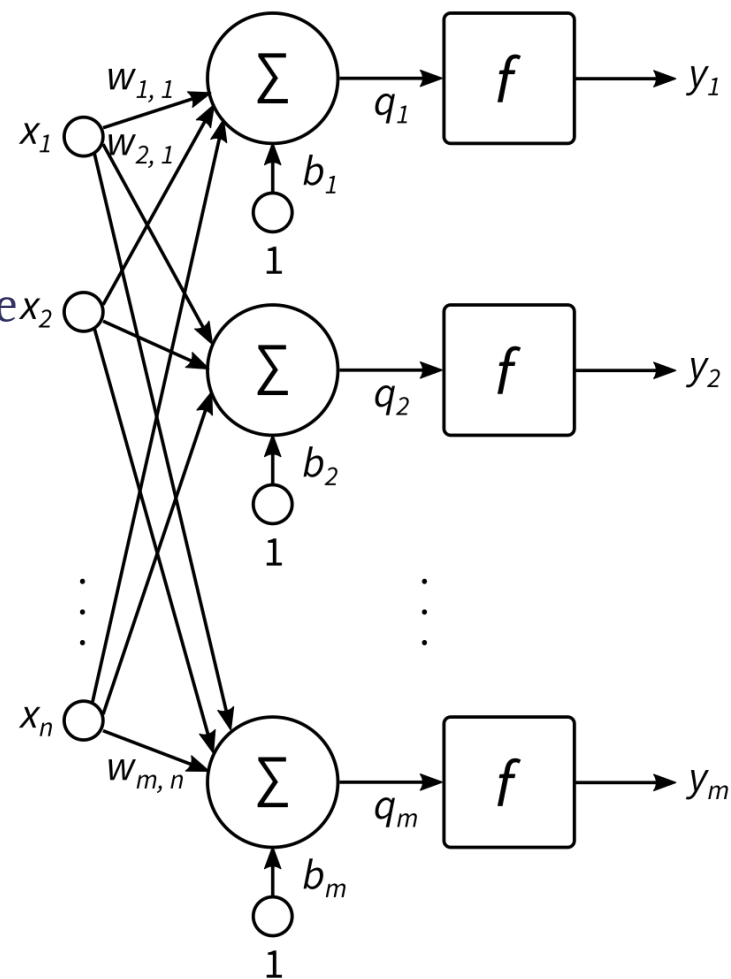
сваки неурон поседује сопствени праг

прагови обједињени у вектор  $\mathbf{b}$  дужине  $m$

постоји укупно  $m$  излаза

за сваки неурон постоји један излаз

излази обједињени у вектор  $\mathbf{y}$  дужине  $m$

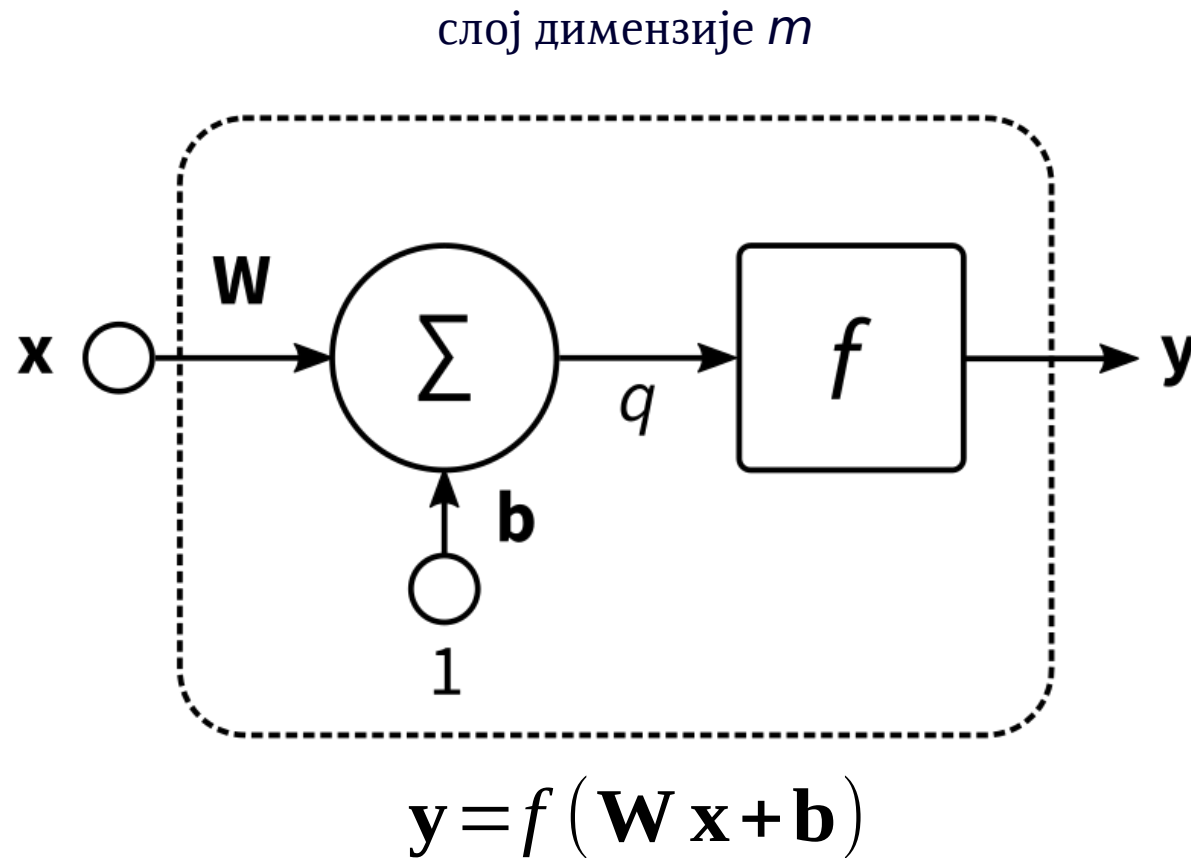


$$\mathbf{y} = f(\mathbf{W}\mathbf{x} + \mathbf{b})$$

# Неуронска мрежа

## Једнослојна неуронска мрежа

један општи облик – сажети приказ



## Вишеслојна неуронска мрежа

обухвата два или више слојева

сигнал бива преношен само унапред (од улаза ка излазима)

енгл. *multilayer feedforward network*

број слојева  $S$

слојеви не морају бити истих димензија

за сваки неурон и његове параметре уводи ознака којем слоју припадају

ознака у експоненту

слојеви секвенцијално поређани

излази једног слоја су улази за наредни слој

улази првог слоја су улази неуронске мреже

излази последњег слоја су излази неуронске мреже

врсте слојева по месту

излазни слој

последњи слој у неуронској мрежи

слој чији излази су излази неуронске мреже

скривени слој

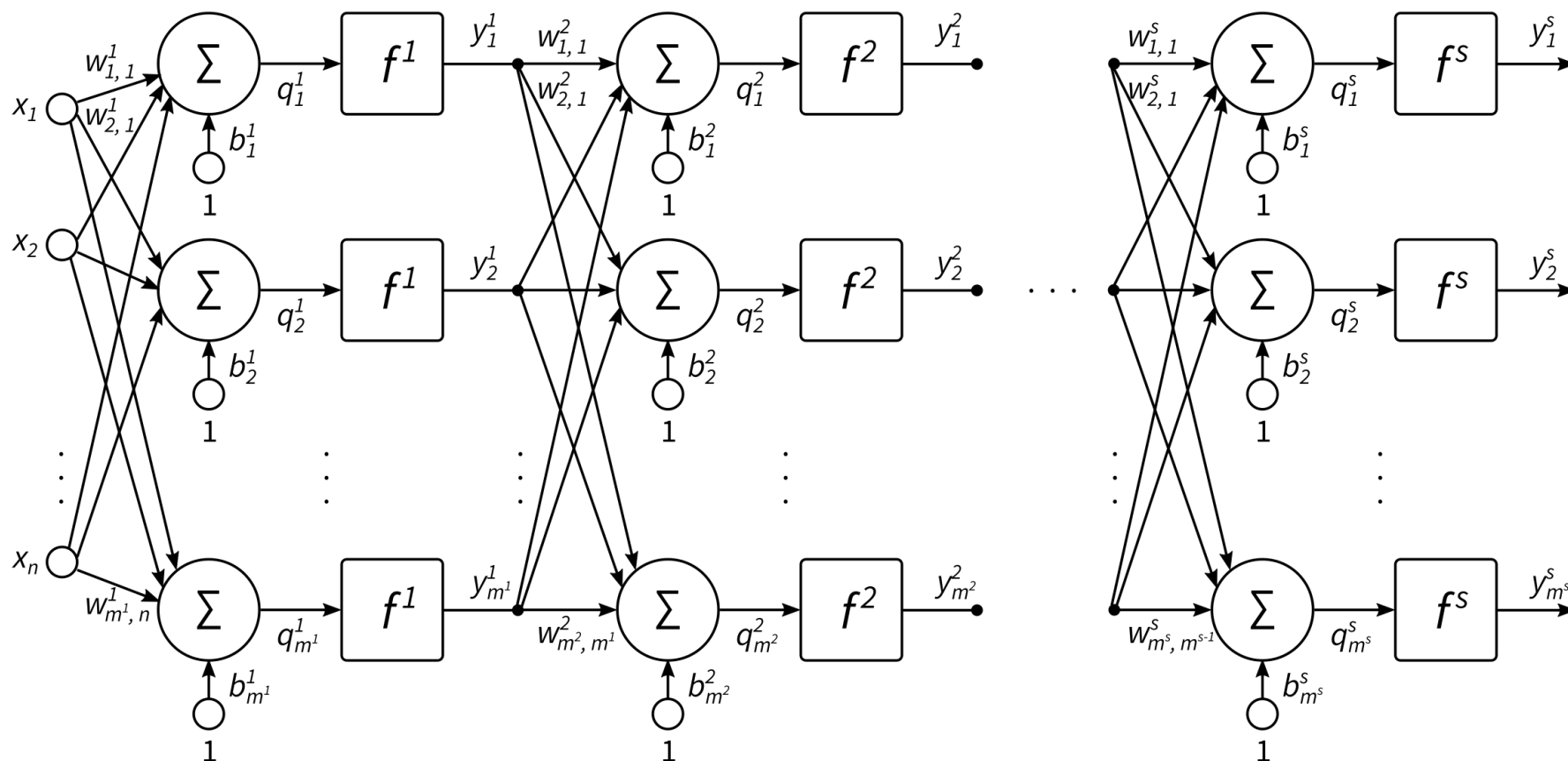
слој који није излазни

# Неуронска мрежа

## Вишеслојна неуронска мрежа

један општи облик

број слојева  $s$

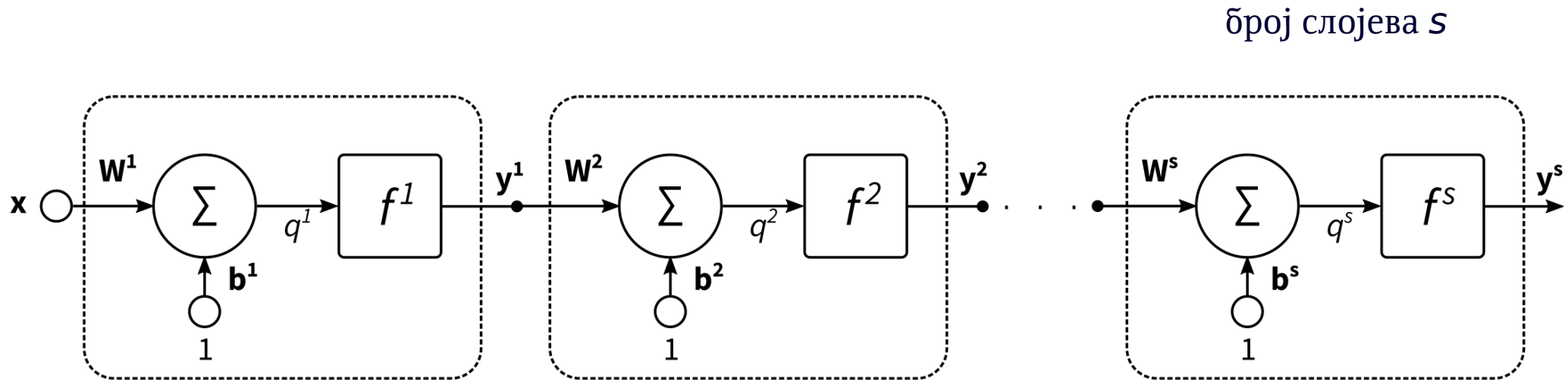


$$\mathbf{y}^r = f^r(\mathbf{W}^r \mathbf{y}^{r-1} + \mathbf{b}^r), \quad r = 1, 2, \dots, s; \quad \mathbf{y}^0 = \mathbf{x}$$

# Неуронска мрежа

## Вишеслојна неуронска мрежа

један општи облик – сажети приказ



$$\mathbf{y}^r = f^r(\mathbf{W}^r \mathbf{y}^{r-1} + \mathbf{b}^r), \quad r = 1, 2, \dots, S; \quad \mathbf{y}^0 = \mathbf{x}$$

## Рекурентна неуронска мрежа

карактерише је присуство повратне спреге

неки излази неуронске мреже могу служити и као њени улази

постоје и другачије врсте повратне спреге

постоје додатне компоненте

могућност увођења кашњења или интеграције у неуронску мрежу

1. Биолошки неурон
2. Вештачки неурон
3. Неуронска мрежа
- 4. Обучавање неуронске мреже**
5. Извори и литература

## Обучавање

поступак подешавања параметара модела како би модел био што успешнији у решавању постављеног задатка

врсте задатака (по Хејкину)

- асоцијација образаца

  - по узору на асоцијативно памћење код људи

- препознавање образаца

  - повезивање датог обрасца с неком од постављених категорија

- апроксимација функција

- управљање системима

- просторно филтрирање сигнала (енгл. *beamforming*)

  - издвајање просторних карактеристика сигнала

## Обучавање

обучавање се спроводи према алгоритму обучавања  
за различите врсте обучавања постоје различити алгоритми

### основне врсте обучавања

надгледано обучавање

енгл. *supervised learning*

обучавање се спроводи на основу датих улаза и њима одговарајућих излаза  
излаз модела за посматрани улаз треба да буде што ближи жељеном излазу

потпомогнуто обучавање

енгл. *reinforcement learning*

за дате улазе не постоје унапред дати жељени излази већ се добија повратна  
информација о понашању модела током низа корака

ненадгледано обучавање

енгл. *unsupervised learning*

искоришћава се само улаз и модел треба да обухвати карактеристике улаза

# Обучавање неуронске мреже

## Перцептрон

концепт неуронске мреже развијен током 50-их година XX века

рад Френка Розенблата и сарадника

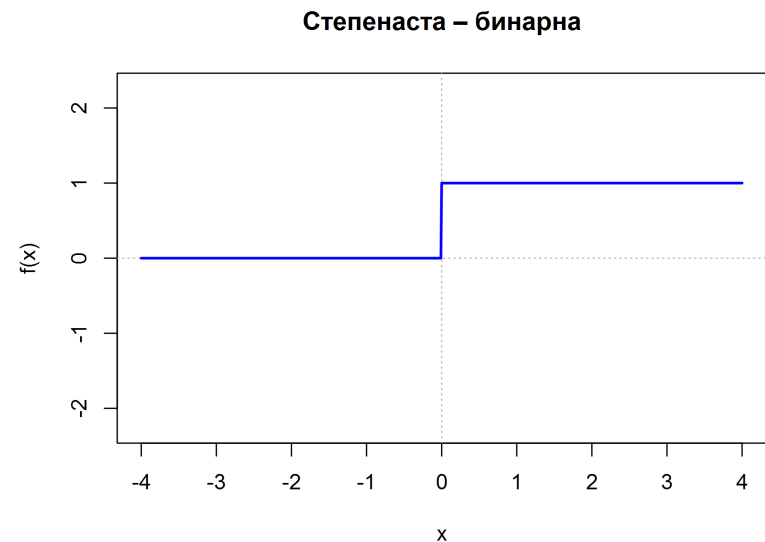
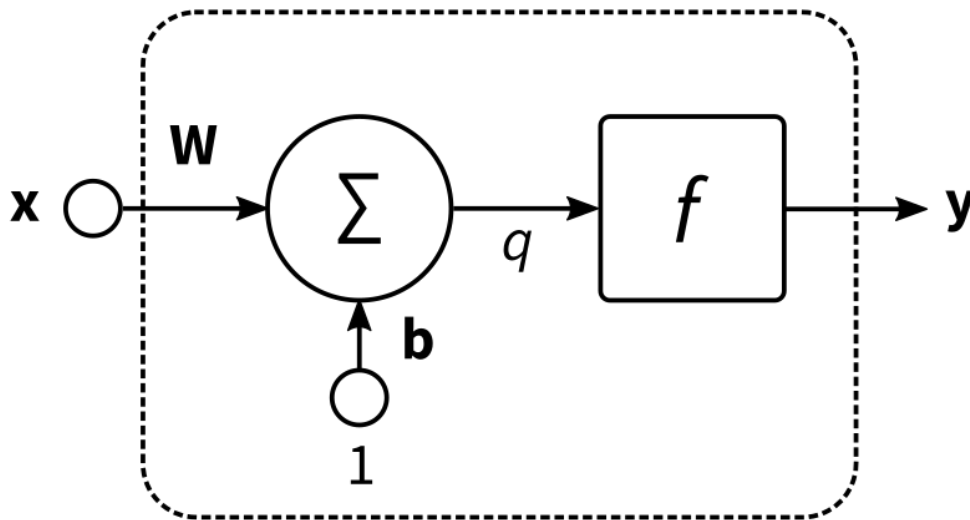
имплементиран и софтверски и хардверски

структура перцептрона

једнослојна неуронска мрежа

функција активације

степенаста функција с два нивоа (0 и 1) и скоком с 0 на 1 у вредности 0



# Обучавање неуронске мреже

## Обучавање перцептрона

### алгоритам за обучавање перцептрона

дати су улази с очекиваним излазима (укупно  $h$  таквих парова)

$\{(\mathbf{x}_i, \mathbf{z}_i), i \in \{1, 2, \dots, h\}\}$

$\mathbf{x}_i$  – задати улаз

$\mathbf{z}_i$  – очекивани излаз за задати улаз

циљ обучавања је добити вредности параметара за неуронску мрежу (**W** и **b**) тако да за дати улаз неуронска мрежа даје очекивани излаз  
обучена неуронска мрежа може бити примењена и за улазе који нису коришћени током обучавања

# Обучавање неуронске мреже

## Обучавање перцептрона

### алгоритам за обучавање перцептрона

почетне вредности параметара су мале случајне вредности  
неуронска мрежа добија дате улазе један по један

за сваки пар који чине улаз  $\mathbf{x}$  и њему одговарајући излаз  $\mathbf{z}$  примењује се правило за подешавање вредности параметара перцептрона

$$\mathbf{W}'' = \mathbf{W}' + \mathbf{e} \mathbf{x}^T$$

$$\mathbf{b}'' = \mathbf{b}' + \mathbf{e}$$

$\mathbf{e}$  – грешка перцептрона

представља разлику између очекиваног излаза  $\mathbf{z}$  и генерисаног излаза  $\mathbf{y}$

$$\mathbf{e} = \mathbf{z} - \mathbf{y}$$

$\mathbf{W}''$  – ново стање матрице тежина

$\mathbf{W}'$  – претходно стање матрице тежина

$\mathbf{b}''$  – ново стање вектора прагова

$\mathbf{b}'$  – претходно стање вектора прагова

## Обучавање перцептрона

поступак обучавања перцептрона ће гарантовано конвергирати према решењу у коначном броју корака ако решење постоји  
основни перцептрон је применљив код проблема који припадају класи линеарно раздвојивих

перцептрон може бити коришћен у класификацији улазних вектора који су линеарно раздвојиви

основни перцептрон може бити неприкладан за друге проблеме

пример је операција ексклузивна дисјункција

за дате парове операнада треба направити раздвајање у зависности од резултата операције

Марвин Мински и Симор Паперт су крајем 60-их година XX века указали на ограничења основног перцептрона

# Обучавање неуронске мреже

## Обучавање вишеслојне неуронске мреже

обучавање пропацијом грешке уназад (по Хејгану и сарадницима)

бројни истраживачи независно развијали и примењивали алгоритам пропације уназад у разним областима

алгоритам пропације уназад у контексту неуронских мрежа

Пол Вербос 70-их година XX века

Дејвид Рамелхарт, Џефри Хинтон и Роналд Вилијамс 80-их година XX века

...

# Обучавање неуронске мреже

## Обучавање вишеслојне неуронске мреже

обучавање пропагацијом грешке уназад (по Хејгану и сарадницима)

обучавање се спроводи на основу датог скупа за обучавање

скуп за обучавање садржи примере

сваки пример обухвата улаз и очекивани излаз

једна епоха одговара примени свих примера из скупа за обучавање током обучавања

обучавање се спроводи по епохама

варијанте обучавања пропагацијом грешке уназад

поступно (инкрементално, секвенцијално, онлајн)

параметри неуронске мреже бивају кориговани након сваке појединачне примене примера из скупа за обучавање

скупно (пакетно)

параметри неуронске мреже бивају кориговани након што су сви примери из скупа за обучавање примењени

# Обучавање неуронске мреже

## Обучавање пропагацијом грешке уназад

### полазне поставке

дата је неуронска мрежа са  $S$  слојева

параметри неуронске мреже

тежине  $\mathbf{W}^r$  и прагови  $\mathbf{b}^r, r = 1, 2, \dots, s$

почетне вредности параметара за неуронску мрежу су мале случајне вредности

дати су улази с очекиваним излазима (укупно  $h$  таквих парова)

$\{(\mathbf{x}_i, \mathbf{z}_i), i \in \{1, 2, \dots, h\}\}$

$\mathbf{x}_i$  – задати улаз

$\mathbf{z}_i$  – очекивани излаз за задати улаз

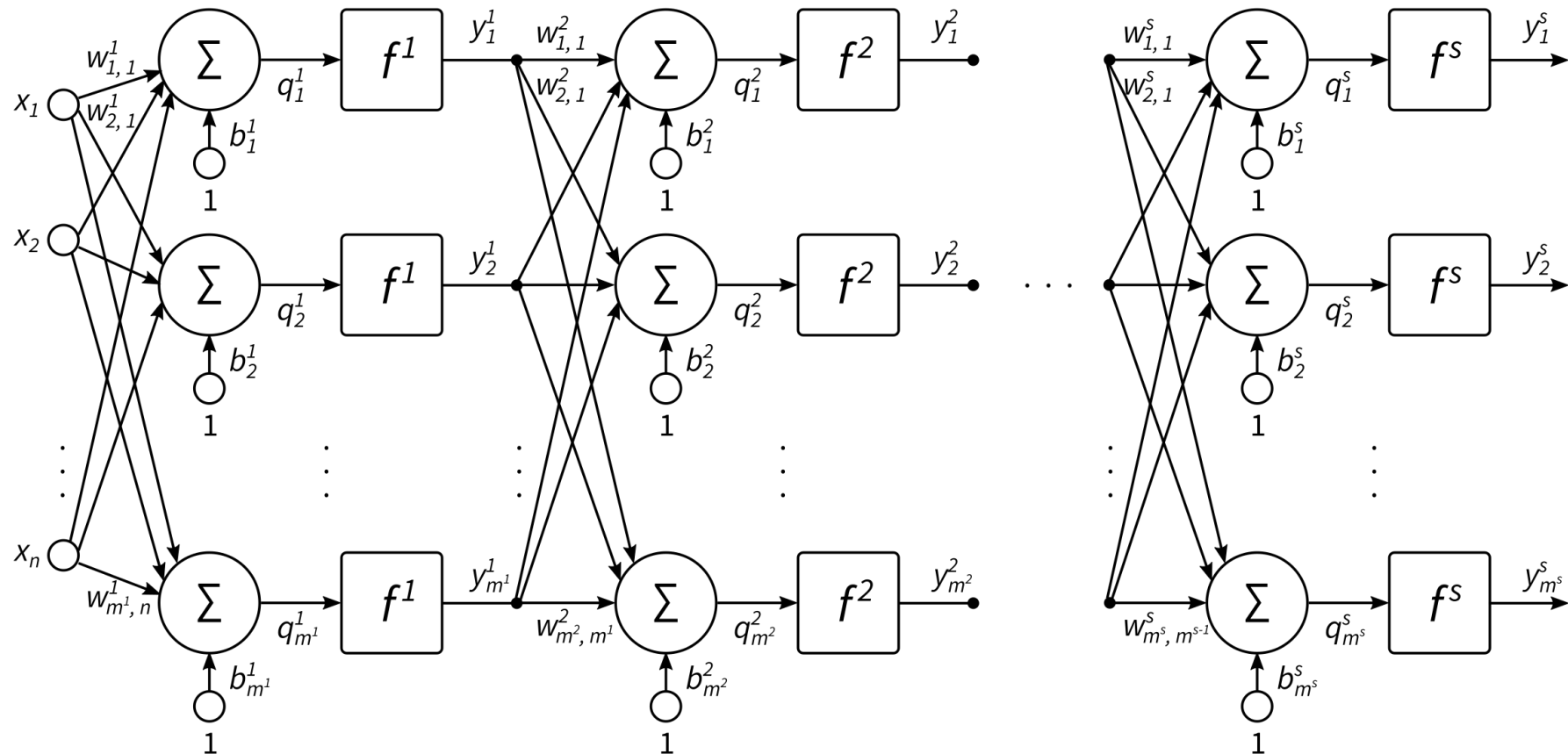
циљ обучавања је добити вредности параметара за неуронску мрежу тако да перформансе неуронске мреже буду што боље

перформансе се оцењују у односу на одабрану меру перформанси

# Обучавање неуронске мреже

## Обучавање пропацијом грешке уназад

један општи облик вишеслојне неуронске мреже број слојева  $s$



$$\mathbf{y}^r = f^r(\mathbf{W}^r \mathbf{y}^{r-1} + \mathbf{b}^r), \quad r = 1, 2, \dots, s; \quad \mathbf{y}^0 = \mathbf{x}$$

# Обучавање неуронске мреже

## Обучавање пропагацијом грешке уназад (поступно)

### основна идеја

за дати улаз познат је очекивани излаз

дати улаз бива прослеђен неуронској мрежи која онда генерише излаз

генерисани излаз настаје тако што улаз бива пропагиран унапред кроз неуронску мрежу од почетног слоја према излазном

разлика између очекиваног и генерисаног излаза користи се приликом подешавања параметара неуронске мреже (тежина и прагова)

грешка је разлика између очекиваног и генерисаног излаза

грешка се пропагира уназад од излазног слоја према почетном

користи се средња квадратна грешка која се апроксимира квадратном грешком на основу пропагираних вредности грешке коригују се вредности тежина и прагова у неуронској мрежи

користи се метода стохастичког градијентног спуста

енгл. *stochastic gradient descent*

циљ кориговања је минимизација грешке

претходни кораци представљају једну итерацију

изводе се итерације с паровима улаз – очекивани излаз док понашање неуронске мреже не постане довољно добро у односу на задати критеријум

# Обучавање неуронске мреже

## Обучавање пропагацијом грешке уназад (поступно)

извођење поступка – одређивање грешке

средња квадратна грешка као мера перформанси  
очекивање квадратне грешке

$$\begin{aligned} F(\mathbf{c}) &= E[\mathbf{e}^T \mathbf{e}] \\ &= E[(\mathbf{z} - \mathbf{y})^T (\mathbf{z} - \mathbf{y})] \end{aligned}$$

$\mathbf{c}$  –  $d$ -димензионални вектор који обухвата тежине и прагове у неуронској мрежи  
апроксимација средње квадратне грешке

користи се квадрат грешке у итерацији  $k$

$$\begin{aligned} \hat{F}(\mathbf{c}) &= E[(\mathbf{z}(k) - \mathbf{y}(k))^T (\mathbf{z}(k) - \mathbf{y}(k))] \\ &= \mathbf{e}^T(k) \mathbf{e}(k) \end{aligned}$$

# Обучавање неуронске мреже

## Обучавање пропагацијом грешке уназад (поступно)

извођење поступка – минимизација грешке

примена методе стохастичког градијентног спуста

примена методе по итерацијама

$$\mathbf{c}(k+1) = \mathbf{c}(k) - \lambda(k) \nabla \hat{F}(\mathbf{c}) \Big|_{\mathbf{c}=\mathbf{c}(k)}$$

$$\nabla \hat{F}(\mathbf{c}) = \begin{bmatrix} \frac{\partial}{\partial c_1} \hat{F}(\mathbf{c}) \\ \frac{\partial}{\partial c_2} \hat{F}(\mathbf{c}) \\ \vdots \\ \frac{\partial}{\partial c_d} \hat{F}(\mathbf{c}) \end{bmatrix}$$

$\lambda$  – брзина обучавања

подесива позитивна скаларна вредност

може бити константна током обучавања

може бити променљива током обучавања

# Обучавање неуронске мреже

## Обучавање пропагацијом грешке уназад (поступно)

извођење поступка – минимизација грешке  
примена методе стохастичког градијентног спуста  
правила подешавања параметара

$$w_{i,j}^r(k+1) = w_{i,j}^r(k) - \lambda \frac{\partial \hat{F}}{\partial w_{i,j}^r}$$

$$b_i^r(k+1) = b_i^r(k) - \lambda \frac{\partial \hat{F}}{\partial b_i^r}$$

# Обучавање неуронске мреже

## Обучавање пропагацијом грешке уназад (поступно)

извођење поступка – увођење осетљивости

сређивање правила подешавања

$$\begin{aligned}w_{i,j}^r(k+1) &= w_{i,j}^r(k) - \lambda \frac{\partial \hat{F}}{\partial w_{i,j}^r} \\ &= w_{i,j}^r(k) - \lambda \frac{\partial \hat{F}}{\partial q_i^r} \frac{\partial q_i^r}{\partial w_{i,j}^r} \\ &= w_{i,j}^r(k) - \lambda v_i^r y_j^{r-1}\end{aligned}$$

$$\begin{aligned}b_i^r(k+1) &= b_i^r(k) - \lambda \frac{\partial \hat{F}}{\partial b_i^r} \\ &= b_i^r(k) - \lambda \frac{\partial \hat{F}}{\partial q_i^r} \frac{\partial q_i^r}{\partial b_i^r} \\ &= b_i^r(k) - \lambda v_i^r\end{aligned}$$

прочишћени улаз

$i$ -ти члан прочишћеног улаза у  $r$ -том слоју

$$q_i^r = \sum_{j=1}^{m^{r-1}} w_{i,j}^r y_j^{r-1} + b_i^r$$

осетљивост

осетљивост апроксимације на промене у  $i$ -том члану прочишћеног улаза у  $r$ -том слоју

$$v_i^r = \frac{\partial \hat{F}}{\partial q_i^r}$$

# Обучавање неуронске мреже

## Обучавање пропагацијом грешке уназад (поступно)

извођење поступка – увођење осетљивости

правила подешавања – скаларно

$$w_{i,j}^r(k+1) = w_{i,j}^r(k) - \lambda v_i^r y_j^{r-1}$$

$$b_i^r(k+1) = b_i^r(k) - \lambda v_i^r$$

правила подешавања – матрично и векторски

$$\mathbf{W}^r(k+1) = \mathbf{W}^r(k) - \lambda \mathbf{v}^r (\mathbf{y}^{r-1})^T$$

$$\mathbf{b}^r(k+1) = \mathbf{b}^r(k) - \lambda \mathbf{v}^r$$

осетљивост – векторски  
осетљивост апроксимације на промене  
прочишћених улаза у  $r$ -том слоју

$$\mathbf{v}^r = \frac{\partial \hat{F}}{\partial \mathbf{q}^r} = \begin{bmatrix} \frac{\partial \hat{F}}{\partial q_1^r} \\ \frac{\partial \hat{F}}{\partial q_2^r} \\ \vdots \\ \frac{\partial \hat{F}}{\partial q_{m^r}^r} \end{bmatrix}$$

# Обучавање неуронске мреже

## Обучавање пропагацијом грешке уназад (поступно)

извођење поступка – одређивање осетљивости за скривене слојеве

$$\begin{aligned}\mathbf{v}^r &= \frac{\partial \hat{F}}{\partial \mathbf{q}^r} \\ &= \left( \frac{\partial \mathbf{q}^{r+1}}{\partial \mathbf{q}^r} \right)^T \frac{\partial \hat{F}}{\partial \mathbf{q}^{r+1}}\end{aligned}$$

# Обучавање неуронске мреже

## Обучавање пропагацијом грешке уназад (поступно)

извођење поступка – одређивање осетљивости за скривене слојеве

Јакобијева матрица

$$\frac{\partial \mathbf{q}^{r+1}}{\partial \mathbf{q}^r} = \begin{bmatrix} \frac{\partial q_1^{r+1}}{\partial q_1^r} & \frac{\partial q_1^{r+1}}{\partial q_2^r} & \dots & \frac{\partial q_1^{r+1}}{\partial q_{m^r}^r} \\ \frac{\partial q_2^{r+1}}{\partial q_1^r} & \frac{\partial q_2^{r+1}}{\partial q_2^r} & \dots & \frac{\partial q_2^{r+1}}{\partial q_{m^r}^r} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial q_{m^{r+1}}^{r+1}}{\partial q_1^r} & \frac{\partial q_{m^{r+1}}^{r+1}}{\partial q_2^r} & \dots & \frac{\partial q_{m^{r+1}}^{r+1}}{\partial q_{m^r}^r} \end{bmatrix}$$

# Обучавање неуронске мреже

## Обучавање пропагацијом грешке уназад (поступно)

извођење поступка – одређивање осетљивости за скривене слојеве

Јакобијева матрица

сређивање појединачног члана

$$\begin{aligned}\frac{\partial q_i^{r+1}}{\partial q_j^r} &= \frac{\partial \left( \sum_{l=1}^{m^r} w_{i,l}^{r+1} y_l^r + b_i^{r+1} \right)}{\partial q_j^r} \\ &= w_{i,j}^{r+1} \frac{\partial y_j^r}{\partial q_j^r} \\ &= w_{i,j}^{r+1} \frac{\partial f^r(q_j^r)}{\partial q_j^r} \\ &= w_{i,j}^{r+1} \dot{f}^r(q_j^r)\end{aligned}$$

# Обучавање неуронске мреже

## Обучавање пропагацијом грешке уназад (поступно)

извођење поступка – одређивање осетљивости за скривене слојеве

Јакобијева матрица

сређени облик

$$\frac{\partial \mathbf{q}^{r+1}}{\partial \mathbf{q}^r} = \mathbf{W}^{r+1} \dot{\mathbf{F}}^r(\mathbf{q}^r)$$

$$\dot{\mathbf{F}}^r(\mathbf{q}^r) = \begin{bmatrix} \dot{f}^r(q_1^r) & 0 & \cdots & 0 \\ 0 & \dot{f}^r(q_2^r) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \dot{f}^r(q_{m^r}^r) \end{bmatrix}$$

# Обучавање неуронске мреже

## Обучавање пропагацијом грешке уназад (поступно)

извођење поступка – одређивање осетљивости за скривене слојеве

$$\begin{aligned}\mathbf{v}^r &= \frac{\partial \hat{F}}{\partial \mathbf{q}^r} \\ &= \left( \frac{\partial \mathbf{q}^{r+1}}{\partial \mathbf{q}^r} \right)^T \frac{\partial \hat{F}}{\partial \mathbf{q}^{r+1}} \\ &= \dot{\mathbf{F}}^r(\mathbf{q}^r) (\mathbf{W}^{r+1})^T \frac{\partial \hat{F}}{\partial \mathbf{q}^{r+1}} \\ &= \dot{\mathbf{F}}^r(\mathbf{q}^r) (\mathbf{W}^{r+1})^T \mathbf{v}^{r+1}\end{aligned}$$

$$\frac{\partial \mathbf{q}^{r+1}}{\partial \mathbf{q}^r} = \mathbf{W}^{r+1} \dot{\mathbf{F}}^r(\mathbf{q}^r)$$

$$\dot{\mathbf{F}}^r(\mathbf{q}^r) = \begin{bmatrix} \dot{f}^r(q_1^r) & 0 & \dots & 0 \\ 0 & \dot{f}^r(q_2^r) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \dot{f}^r(q_{m^r}^r) \end{bmatrix}$$

# Обучавање неуронске мреже

## Обучавање пропацијом грешке уназад (поступно)

извођење поступка – одређивање осетљивости за излазни слој  
сређивање појединачног члана

$$\begin{aligned}v_i^s &= \frac{\partial \hat{F}}{\partial q_i^s} \\&= \frac{\partial ((\mathbf{z} - \mathbf{y})^T (\mathbf{z} - \mathbf{y}))}{\partial q_i^s} \\&= \frac{\partial \left( \sum_{j=1}^{m^s} (z_j - y_j)^2 \right)}{\partial q_i^s} \\&= -2(z_i - y_i) \frac{\partial y_i}{\partial q_i^s} \\&\dots\end{aligned}$$

## Обучавање пропагацијом грешке уназад (поступно)

извођење поступка – одређивање осетљивости за излазни слој  
сређивање појединачног члана

$$\begin{aligned}v_i^s &= -2(z_i - y_i) \frac{\partial y_i}{\partial q_i^s} \\ &= -2(z_i - y_i) \frac{\partial y_i^s}{\partial q_i^s} \\ &= -2(z_i - y_i) \frac{\partial f^s(q_i^s)}{\partial q_i^s} \\ &= -2(z_i - y_i) \dot{f}^s(q_i^s)\end{aligned}$$

сређени облик

$$\mathbf{v}^s = -2 \dot{\mathbf{F}}^s(\mathbf{q}^s)(\mathbf{z} - \mathbf{y})$$

# Обучавање неуронске мреже

## Обучавање пропагацијом грешке уназад (поступно)

преглед поступка за неуронску мрежу са  $S$  слојева

израчунавање тежина и прагова

$$\mathbf{W}^r(k+1) = \mathbf{W}^r(k) - \lambda \mathbf{v}^r (\mathbf{y}^{r-1})^T, \quad r=1, 2, \dots, s$$

$$\mathbf{b}^r(k+1) = \mathbf{b}^r(k) - \lambda \mathbf{v}^r, \quad r=1, 2, \dots, s$$

израчунавање осетљивости

$$\mathbf{v}^r = \dot{\mathbf{F}}^r(\mathbf{q}^r) (\mathbf{W}^{r+1})^T \mathbf{v}^{r+1}, \quad r=1, 2, \dots, s-1$$

$$\mathbf{v}^s = -2 \dot{\mathbf{F}}^s(\mathbf{q}^s) (\mathbf{z} - \mathbf{y})$$

израчунавање излаза

$$\mathbf{y} = \mathbf{y}^s$$

$$\mathbf{y}^r = f^r(\mathbf{W}^r \mathbf{y}^{r-1} + \mathbf{b}^r), \quad r=1, 2, \dots, s$$

$$\mathbf{y}^0 = \mathbf{x}$$

# Обучавање неуронске мреже

## Обучавање пропагацијом грешке уназад (скупно)

извођење поступка слично као за поступну варијанту али уз нека кључна одступања

средња квадратна грешка као мера перформанси

$$\begin{aligned} F(\mathbf{c}) &= E[\mathbf{e}^T \mathbf{e}] \\ &= E[(\mathbf{z} - \mathbf{y})^T (\mathbf{z} - \mathbf{y})] \\ &= \frac{1}{h} \sum_{g=1}^h (\mathbf{z}_g - \mathbf{y}_g)^T (\mathbf{z}_g - \mathbf{y}_g) \end{aligned}$$

$\mathbf{c}$  – вектор који обухвата тежине и прагове у неуронској мрежи  
тотални градијент за средњу квадратну грешку

$$\begin{aligned} \nabla F(\mathbf{c}) &= \nabla \left( \frac{1}{h} \sum_{g=1}^h (\mathbf{z}_g - \mathbf{y}_g)^T (\mathbf{z}_g - \mathbf{y}_g) \right) \\ &= \frac{1}{h} \sum_{g=1}^h \nabla \left( (\mathbf{z}_g - \mathbf{y}_g)^T (\mathbf{z}_g - \mathbf{y}_g) \right) \end{aligned}$$

# Обучавање неуронске мреже

## Обучавање пропагацијом грешке уназад (скупно)

преглед поступка за неуронску мрежу са  $S$  слојева

израчунавање тежина и прагова

$$\mathbf{W}^r(k+1) = \mathbf{W}^r(k) - \frac{\lambda}{h} \sum_{g=1}^h \mathbf{v}_g^r (\mathbf{y}_g^{r-1})^T, \quad r=1, 2, \dots, s$$

$$\mathbf{b}^r(k+1) = \mathbf{b}^r(k) - \frac{\lambda}{h} \sum_{g=1}^h \mathbf{v}_g^r, \quad r=1, 2, \dots, s$$

израчунавање осетљивости где  $g = 1, 2, \dots, h$

$$\mathbf{v}_g^r = \dot{\mathbf{F}}^r(\mathbf{q}_g^r) (\mathbf{W}^{r+1})^T \mathbf{v}_g^{r+1}, \quad r=1, 2, \dots, s-1$$

$$\mathbf{v}_g^s = -2 \dot{\mathbf{F}}^s(\mathbf{q}_g^s) (\mathbf{z}_g - \mathbf{y}_g)$$

израчунавање излаза где  $g = 1, 2, \dots, h$

$$\mathbf{y}_g = \mathbf{y}_g^s$$

$$\mathbf{y}_g^r = f^r(\mathbf{W}^r \mathbf{y}_g^{r-1} + \mathbf{b}^r), \quad r=1, 2, \dots, s$$

$$\mathbf{y}_g^0 = \mathbf{x}_g$$

## Обучавање пропагацијом грешке уназад

### завршетак обучавања

услов за завршетак обучавања (критеријум заустављања)

постоје разни услови за завршетак обучавања

када апсолутна промена средње квадратне грешке на нивоу епохе постане довољно мала

примери 0,01% или 0,1% или 1%

када је способност генерализације задовољавајућа

# Обучавање неуронске мреже

## Обучавање пропагацијом грешке уназад

рачунски ефикасан поступак за обучавање вишеслојних неуронских мрежа

поступак обучавања може бити дуготрајан

не постоји гаранција да ће бити добијено оптимално решење

може се покушати с различитим почетним условима

може се покушати с насумичним одабиром примера из скупа за обучавање

## Обучавање неуронске мреже

### ризици у обучавању

недоученост (енгл. *underfitting*)

правилности из скупа за обучавање нису адекватно пренете на неуронску мрежу  
релативно висока грешка приликом обучавања

преученост (енгл. *overfitting*)

и правилности и шум из скупа за обучавање су пренети на неуронску мрежу  
релативно ниска грешка приликом обучавања али релативно висока грешка  
приликом примене неуронске мреже на новим примерима

### пожељне карактеристике обучене неуронске мреже

добре перформансе на скупу за обучавање

добре перформансе на скупу који садржи нове примере

неуронска мрежа је једноставна онолико колико је то могуће  
по број параметара или по броју садржаних неурона

## Обучавање неуронске мреже

### способност генерализације

обучена неуронска мрежа способна за прихватљиво понашање и када је примењена за нове примере

за примере који нису коришћени приликом обучавања

фактори који утичу на способност генерализације (по Хејкину)

величина и репрезентативност скупа за обучавање

архитектура неуронске мреже

комплексност постављеног задатка

помоћна правила о неопходној величини скупа за обучавање

број параметара у неуронској мрежи треба да буде мањи од броја примера у скупу за обучавање

у случају задатка класификације, величина скупа за обучавање по реду величине одговара односу броја параметара у неуронској мрежи и прихватљивог процента примера који нису исправно класификовани

## Обучавање неуронске мреже

### могућности

неуронске мреже као универзални апроксиматори

вишеслојна неуронска мрежа с једним скривеним слојем може се користити за униформну  $\epsilon$  апроксимацију за дати скуп обучавања

теорема о универзалној апроксимацији

1. Биолошки неурон
2. Вештачки неурон
3. Неуронска мрежа
4. Обучавање неуронске мреже
- 5. Извори и литература**

## Основни извори и литература

- ◆ Hagan MT, Demuth HB, Beale MH, De Jesús O. Neural Network Design. 2nd Edition. eBook. Internet: <https://hagan.okstate.edu/nnd.html>
- ◆ Haykin S. Neural Networks and Learning Machines. 3rd Edition. Pearson Education (Upper Saddle River, NJ, USA); 2009.
- ◆ Kecman V. Learning and Soft Computing: Support Vector Machines, Neural Networks, and Fuzzy Logic Models. MIT Press (Cambridge, MA, USA); 2001.
- ◆ Tan P-N, Steinbach M, Kumar V. Introduction to Data Mining. Pearson Education (Boston, MA, USA); 2006.
- ◆ Clark MA, Choi J, Douglas M. Biology 2e. OpenStax, Rice University (Houston, TX, USA); 2018. Internet: <https://openstax.org/details/books/biology-2e>

## Додатни извори и литература

- ◆ Shiffman D. The Nature of Code. 2012. Internet: <https://natureofcode.com/>
- ◆ Goodfellow I, Bengio Y, Courville A. Deep Learning. MIT Press (Cambridge, MA, USA); 2016. Internet: <https://www.deeplearningbook.org/>

Основне академске студије  
Информациони инжењеринг

Основи рачунарске интелигенције

# Увод у неуронске мреже

(материјали за предавања)