

CURS 2: pierderea empirică = pe setul de antrenare $\frac{1}{N} \sum_{i=1}^N L(y_i, f(x_i)) = \frac{1}{N} \sum_{i=1}^N (y_i - f(x_i))^2$

MODELUL GENERAL DE INV. DIN EXEMPLU: ① mediul → stationar, independent din id. distribuit i.i.d = necesar ca învățarea să se facă cu succes; presupunere des întâlnită în ML și clarificare funcțională fără uid (Bayes mark); $p(y_i | x_i)$ cunoscând uid supervizorul → return. pt. fiecare x_i în concord cu prob. distribuție $F(x_i | z)$

hipoteză / ML → produce o funcție a.i. outputul funcție verifică $y_i = h(x_i), S = \{x_1, \dots, x_m\}$

$H = \text{hipoteză space} = X \times U \rightarrow \text{outputuri}$

$$|H| = \binom{k^2}{2} \rightarrow \text{inputuri} \quad \text{inputuri}$$

ex: $f: \{0,1\}^2 \rightarrow \{0,1\}$
② learning task → căută $h \in H$ care aproximiază cel mai bine cehetele date de supervizor

$$\text{pt. fiecare } x \in S \text{ avem } cost(x, u) = L(u, h(x))$$

$$\text{COSTUL MENIU/REAL} = R_{\text{real}}(R) = \sum_u L(u, h(x))$$

③ principiul inducției → indică ce trb. să ob.

în setul de antrenare pt. a minimiza R_{real}
⇒ ap. de învățare care să pună efectiv cum să rezolve principiul inducției.

- pt. un principiu sind, și mai multi alg. de învățare ex: princ. ind. spune că trb. alează că mai simplu și pt. rest de antrenare în formă method spine cum se poate obține soluția optimă (suboptimale).

TEOREMA INFERENȚEI INDUCTIVE = nu există un principiu inducției ideal sau unic

① EMPIRICAL RISK MINIMIZATION (ERM)

$$\text{aprox. } R_{\text{emp}}(R) = \frac{1}{m} \sum_i L(u_i, h_i) - \text{ipoteza care approx. bn. } S, \text{ volumul (deacă } S \text{ reprezentativ).}$$

② CHOICE OF MOST PROBABLE HYPOT. (RAY. PRINCIPLE)

- def. o prob. a priori ocupă spațiul de ipoteze
- apoi, de căd set antrenare ⇒ prob. a posteriori și se alege cea mai probabilă ipoteză

③ PRINCIPIUL COMPRESIEI INFORMAȚIEI - elimină redundanța din training data, pt. a extinde capabilitatea aplicabilei înțelegerii lumeni

CONDITII DE VALIDITATE ERM

NOT: $h^* = \text{Arg Min } R_{\text{real}}(h) \rightarrow$ ipoteza optimă

$h^* = \text{Arg Min } R_{\text{emp}}(h) \rightarrow$ ipoteza care se comportă cel mai bine pe mt. antrenare (principiu)

ERM principle relevant $\Leftrightarrow R_{\text{real}} \leq R_{\text{emp}}$ (val. f. apropiată)

CORELATIA $\rightarrow R_{\text{real}}(h^*) - R_{\text{real}}(h) > 0$

\rightarrow prob. co dif. $> \epsilon$ (given bound)

- depinde de căt de reprezentativ este multimea de antrenare S

LEGEA NR. MARII: media aritmetică a unei v.a. converge spre medie când nr. de exemplă crește

- incertitudinea creșterea nr. de ex. până se garantează $\forall \epsilon < \delta, \exists n \text{ a.i. } P(|R_{\text{real}}(h^*) - R_{\text{emp}}(h^*)| > \epsilon) < \delta$

ERM consistent $\Leftrightarrow R_{\text{real}}(h^*) \leq R_{\text{emp}}(h^*)$ converg la inf. așa $R_{\text{real}}(h^*) \leq R_{\text{emp}}(h^*)$

- legătura nr. mare nu e suficientă, trb. generalizată

CURS 3: BIAS - VARIANCE TRADEOFF

- bias = exces din cauza unei presupuneri eronate (bias mare \Rightarrow underfitting) → ratează caracteristici importante

- varianță = set de variabile mediu diversificate (varianță mare \Rightarrow overfitting)

- ca să reducem biasul - H mai mare \rightarrow varianță IDEAL: null noise, H redus și bine informat \Leftrightarrow cunoștințe și priori deosebite natură

METODE DE REGULARIZARE:

- inducția supraveghetă trb. să facă față riscului de overfitting. H prea mare \rightarrow sunt sante mai multe să alegă sănătos să alegă minimizarea. $R_{\text{emp}} \leq R_{\text{real}}$ f. măre $L(u_i, h(x_i, w)) = \sum_{i=1}^m (u_i - h(x_i, w))^2 + \lambda \|w\|^2$, $\|w\|^2 = w_0^2 + \dots + w_n^2$

$H = \sum_i h(x_i, w) = w_0 + w_1 x_1 + \dots + w_n x_n$, λ - importanța regularizării

SELECTIA IPOTEZEI (MODELULUI)

$H_1 \subseteq H_2 \subseteq \dots \subseteq H_d$; $h^* =$ ipoteza optimă a lui H_d

$\{R_{\text{real}}(h^*)\}_{1 \leq d \leq \infty} \Rightarrow$ capacitatea funcției deținător H

PROBLEMA SELECTIEI IPOTEZEI = pe bază S , $|S| = m$ să se alegă H^* în $H_d \in H$ astfel $R_{\text{real}}(h^*)$ minimal

$$d^* = \text{Arg Min}_{H_d} \{R_{\text{real}}(h^*)\}_{H_d \in H_d}$$

CURS 4: ESTIMAREA RISCULUI REAL AL IPOTEZEI

- împărțim S în A - mt. de antrenare și T - mt. de testare a ipotezei h , $S = \text{AUT}$, $ANT = \emptyset$ măsurarea erorilor

Not: $R_{\text{real}}(h)$ = estimator al riscului real (pe T)

- matricea de confuzie $M(i,j) =$ nr. de exemplă din clasa i , clasificate ca fiind j din T ; $R_{\text{real}}(h) = \frac{1}{T} \sum_{i,j} M(i,j)$

$$= \frac{1}{T}, \text{ unde } T = t$$

- măsurarea erorilor pe A \rightarrow matricea de confuzie empirică / supra eroilor nu reprezintă o estimare a riscului real, ci este doar proporțională

CĂZ PARTICULAR - clasificare binară

		TRUE		\sum
+	-	+	-	
		True	False	
-	-	TP	FP	N_+
		FN	TN	N_-
		N_+	N_-	

POȚERI PT. CALITATEA CLASIFICARII BINARE

Not: $TPR = \text{true pos. ratio} = \frac{TP}{N_+} \approx p(\hat{y} = 1 | y = 1) = \text{sensitivity}$

$FPR = \text{false pos. ratio} = \frac{FP}{N_-} \approx p(\hat{y} = 1 | y = 0) = \text{false hit rate}$

- plot: $OY - TPR$ și $OX - FPR$, pt. un set de tip I erruri $\tilde{\epsilon}$; un sistem care separă perfect are $TPR = 1 \wedge FPR = 0$

- $TPR = FPR = 0$ dacă $\tilde{\epsilon} = 1$ (total e clasif. negativ)

- $TPR = FPR = 1$ dacă $\tilde{\epsilon} = 0$ (total e clasif. pozitiv)

- calitatea curbei ROC = aria sub grafic

② EQUAL ERR. RATE (EER) / CROSS OVER RATE

$$FPR = FNR \quad TPR = p(\hat{y} = 1 | y = 1)$$

$$FNR = 1 - TPR, \quad FNR = p(\hat{y} = 0 | y = 1)$$

- trasăm dr. din conf. matr. am dr. joi și redem unde intersectarea ROC

③ PRECISION-RECALL CURVE (când avem multe ex. nuf.

$$\text{ROC nu e informative} - \text{PRECISION} = \frac{TP}{TP + FN}$$

$$p(y=1 | \hat{y}=1) \text{ și RECALL} = \frac{TP}{TP + FN}$$

(dăt pt. o mulțime de progruri \mathcal{T})

$$F^*-\text{score} - \text{pt. un progr fix} \rightarrow \text{putem calcula un}\text{sumpuj precision-recall value} = \frac{1}{2} = \frac{2PR}{P+R}$$

ESTIMAREA IPOTEZEI

$$\text{① cu intervale de încredere: } \left[\frac{\text{terr} \pm \sigma_{\text{terr}}}{x(1-x)} \right]$$

$$x \quad 50\% \quad 68\% \quad 80\% \quad 90\% \quad 95\% \quad 98\% \quad 99\%$$

$$y(x) \quad 0.67 \quad 1.00 \quad 1.28 \quad 1.64 \quad 1.96 \quad 2.33 \quad 2.58$$

t mai mare \Rightarrow încredere mai mică, interval mai mic
(alegeri mai empirice)/aleatorie

② cu cross-validation: împărțim S în N subșeturi, un subset testare, $n-1$ antrenare și repetăm de n ori; pt fiecare fold avem $R_{\text{real}}(h)$, iar eroarea finală $= \frac{1}{n} \sum_i R_{\text{real}}(h_i)$, $n = 5/10$ obicei

- la final, train pe tot S

③ leave-out method: extreme cross-validation; păstrăm căte un ex. pt. dețin, restul train; repetă de n ori

- pt. date puține; ipoteza variată mai puțin, dar eroarea mai variabilă; avem și rata de

④ bootstrap: alegem o mt. random de m elemente = BOOTSTRAP și antrenăm pe "bootstrap"; testăm ipoteza pe S ; BOOTSTRAP \Rightarrow P_1 și testăm pe tot $S \Rightarrow P_2$; repetăm procedul de K ori-obt. P_1 și P_2 ipoteze

$$R_{\text{real}}(h) = 0.636 P_1 + 0.368 P_2$$

⑤ JACK KNIFE - o metodă complexă, care reduce biasul ratei de eroare, folosind un același tip de date și pentru antrenare și pentru testare

TUNAREA ALGORITMILOR FOLOSIND UN SET DE VALIDARE: urm. evaluază perf. reală a metodei; și folosim pt. să determinăm mom. cnd. ipoteza converge

COMPARAREA DOUĂ IPOTEZE PRODUSE DE ACELAȘI ALGORITM PE 2 MT. TEST DIFERITE:

- h_1 și h_2 , T_1 și T_2 test set, $|T_1| = d_1$, $|T_2| = d_2$, T_1 și T_2 iid; $\delta_R(h_1, h_2) = \frac{\text{terr}_1 - \text{terr}_2}{\sqrt{\frac{d_1}{d_1-1}} + \sqrt{\frac{d_2}{d_2-1}}}$

INTERVALUL

$$\text{DE INCREDERE AL ESTIMATORULUI: } \left[\delta_R \pm \sigma_{\delta_R}(x) \right] \sqrt{\frac{\text{terr}_1(1-\text{terr}_1)}{d_1-1} + \frac{\text{terr}_2(1-\text{terr}_2)}{d_2-1}}$$

COMPARAREA DOI ALG. DE ANTRENARE: $D = \text{SUT}$

1) împărțim mt. mare în K părți: T_1, \dots, T_K și S_1, \dots, S_K

2) Fără $i=1, K$, $S_i = D \setminus T_i$, Antrenăm A_1 pe $S_i \Rightarrow h_i^1$

Antrenăm A_2 pe $S_i \Rightarrow h_i^2$, $d_i = R_i^1 - R_i^2$; $R_i^1 = R_{\text{real}}(h_i^1)$

$$3. \bar{d} = \frac{1}{K} \sum_{i=1}^K d_i$$

Interval de încredere \bar{d} :

$$\left[\bar{d} \pm \sigma_{\bar{d}}(x) \sqrt{\frac{1}{K(K-1)} \sum_{i=1}^K (d_i - \bar{d})^2} \right]$$

x	90%	95%	98%	99%
$\bar{d}_{(1,2)}$	2.92	4.3	6.96	9.92
$\bar{d}_{(2,5)}$	2.02	2.57	3.36	4.03
$\bar{d}_{(5,10)}$	1.81	2.23	2.76	3.17
$\bar{d}_{(10,30)}$	1.7	2.04	2.46	2.75
$\bar{d}_{(2, \infty)}$	1.64	1.96	2.33	2.58

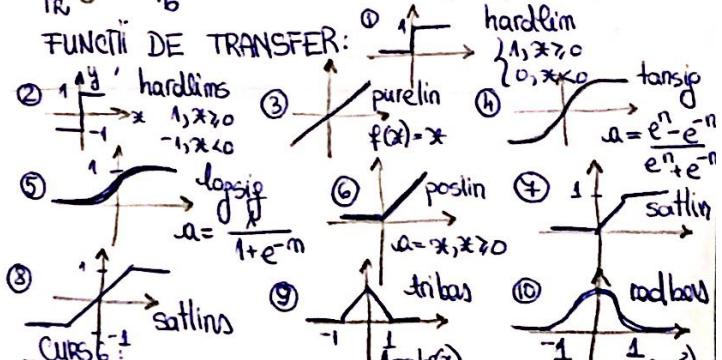
$z = 1/m_0 - m_1$, m_0 = ex. grădite de y_i în acurate de h_2

PAIRED TEST

- presupunerea că h_1 și h_2 au aceeași err. rate poate fi respinsă cu o prob. $> x\% \Leftrightarrow |z| > y(x)$

CURS 5: PERCEPTRONUL

$$P^0 \xrightarrow{w_0} \Sigma \xrightarrow{f} a \quad w = w_1, p_1, \dots, w_K, p_K, b \quad a = f(w_p + b) = f(m)$$



ROSENBLATT: ① initializează ponderile $w^{(0)}$
② calculează eticheta ex. current, modifică ponderi dacă eticheta e greșită: $w^{(n)} = w^{(n-1)} + p(d_i - y_i) \cdot x_i$
③ repetă până când de-a lungul unei epoci w stătă

ONLINE: ④ w_1 arbitrar

$$w^{(k+1)} = w^{(k)} + p(d^{(k)} - y^{(k)}) \cdot x^{(k)}$$

BATCH: ⑤ w_1 arbitrar

$$w^{(k+1)} = w^{(k)} + p \left(\sum_{i=1}^m [(d_i - y_i) \cdot x_i] \right)$$

! - în cazul perceptronului nu există o unică soluție
 \Leftrightarrow mt. este cliniar separabilă
(implementarea hardlim nu se fol. pt. probleme de clasificare binară)!

ALTE REGUINI DE ÎNVĂȚARE:

FUNCTIONA CRITERIU A PERCEPTRONULUI

$$J(w) = - \sum_{i=1}^K w_i^T \cdot z_i \rightarrow \text{suma ex. misclassificate}$$

$$z_i^i = d_i^i \cdot x_i^i = 1 \cdot x_i, \text{ d}_i = 1$$

$$-1 \cdot x_i, \text{ d}_i = -1$$

$$w^{(k+1)} = w^{(k)} - p \nabla J(w) = w^{(k)} + p \frac{\partial E}{\partial w_k} (\sum w_i^T \cdot z_i) \Rightarrow$$

$$\Rightarrow w^{(k+1)} = w^{(k)} + p \sum_{i \in \text{ex. gn.}} z_i = q_i$$

$$w^{(k+1)} = w^{(k)} + p \sum_{i=1}^m \frac{(d_i - y_i)}{2} z_i \quad // \text{dacă } d_i = 1 \Rightarrow 0 \\ y_i = 1 \quad \text{dacă } d_i = -1 \quad y_i = -1$$

BATCH UPDATE:

(2) MAY'S LEARNING RULE: $J(w) = \frac{1}{2} \sum_{z^T w \leq b} (z^T w - b)^2$

- core gradient continuu după dezvoltare de ①
- $\nabla J(w) = \sum_{z^T w \leq b} \frac{z^T w - b}{\|z\|^2} \cdot z$
- BATICHT: w^{k+1} arbitrar

$$w^{k+1} = w^k + p \sum_{z^T w \leq b} \frac{b - z^T w}{\|z\|^2} \cdot z$$

ONLINE: $w^{k+1} = w^k + p \cdot \frac{b - z^T w}{\|z\|^2} \cdot z$, dacă $z^T w \leq b$

ÎN PUNCTUL DE MINIM: $\nabla J(w^*) = 0$
 deci $w^* = (x^T)^{-1} \cdot x \cdot d$, unde $x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \\ x_{n+1} \\ \dots \\ x_m \end{pmatrix}$ și
 $d = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_m \end{pmatrix}$

CONDITII DE MINIM:

① $\frac{\partial E}{\partial w_1} = 0$ și ② Hessiana poz.
 $\frac{\partial^2 E(w)}{\partial w_1 \partial w_K} > 0$

③ $\frac{\partial E}{\partial w_K} = 0$ și $\frac{\partial^2 E(w)}{\partial w_K \partial w_1} > 0$

$$\left. \begin{array}{l} \frac{\partial^2 E(w)}{\partial w_1 \partial w_K} \\ \frac{\partial^2 E(w)}{\partial w_K \partial w_1} \\ \frac{\partial^2 E(w)}{\partial w_K^2} \end{array} \right\} > 0$$

NUME	FUNCȚIE CRITERIU	FCT. ACTIVARE
Butz rule (supervised)	$J(w) = - \sum_i (z^i)^T \cdot w$	$f(\text{net}) = \text{net} \cdot \text{sgn}(1)$
Widrow-Hoff (supervised) (α -LMS)	$J(w) = \frac{1}{2} \sum_i \frac{[d^i - (z^i)^T \cdot w]^2}{\ z^i\ ^2}$	$f(\text{net}) = \text{net}$
μ -LMS (supervised)	$J(w) = \frac{1}{2} \sum_i [d^i - (z^i)^T \cdot w]^2$	$f(\text{net}) = \text{net}$
Correlation rule	$J(w) = - \sum_i d^i (z^i)^T \cdot w$	$f(\text{net}) = \text{net}$
Delta rule	$J(w) = \frac{1}{2} \sum_i (d^i - y^i)^2$ $y^i = (z^i)^T \cdot w$	$f(\text{net}) = y$ $f - \text{sigmoid}$
Minkowski-n delta rule	$J(w) = \frac{1}{n} \sum d^i - y^i $	
Relative entropy delta rule	$J(w) = \frac{1}{2} \sum_i \left[(1+d^i) \ln \left(\frac{1+d^i}{1-y^i} \right) + (1-d^i) \ln \left(\frac{1-d^i}{1-y^i} \right) \right]$	
AHK	$J(w, b) = \frac{1}{2} \sum_i (z^i)^T \cdot w - b)^2$	

GRADIENT	CONDITII	EXPLICATII
$\begin{cases} \neq^K, \text{ dacă } (z^K)^T \cdot w^K \leq 0 \\ \neq^K \end{cases}$	$0 < \gamma < 1$	converge la soluție dacă este liniar dep.
$[d^K - (z^K)^T \cdot w^K] \cdot z^K$	$b > 0$	
$[d^K - (z^K)^T \cdot w^K] \cdot z^K$	$\frac{\partial}{\partial w_1} = \frac{\alpha}{\ z^K\ ^2}$	converge la min sse dacă $\ x_i\ = \ x_j\ , \forall i, j$
$d^K \cdot z^K$	$0 < \alpha < 2$	converge la SSE soluționă
$(d^K - y^K) f'(\text{net}) \cdot z^K$	$0 < \rho < \frac{2}{3\ z^K\ ^2}$	extinde μ -LMS pt. vectori care nu sunt ortonormalize
$\gamma \rho \eta (d^K - y^K) d^K - y^K ^{-1} f'(\text{net}) \cdot z^K$	$\rho > 0$	converge la SSE dacă vectorii sunt ortonormalize
$\rho \eta (d^K - y^K) \cdot z^K$	$0 < \rho < 1$	extinde μ -LMS pt. vectori care nu sunt ortonormalize
$\epsilon_i^K = (z^i)^T \cdot w^K - b^K$		

$$\Delta b = \begin{cases} p_1 \epsilon_i^K, \epsilon_i^K \geq 0 \\ 0, \text{ altfel} \end{cases} \quad b > 0$$

$$\begin{cases} p_2 (p_1 - 1) \epsilon_i^K z_i, \epsilon_i^K \geq 0 \\ -p_2 \epsilon_i^K z_i, \epsilon_i^K \leq 0 \end{cases} \quad 0 < p_2 < \frac{2}{\max_i \|z_i\|^2}$$

$$\epsilon_i^K = (z^i)^T \cdot w^K - b^K$$

④ $L(S) = \text{nr. de dihotomii în } S \setminus x_0 \in H \text{ și } x_0 \notin S$

Lema 1: $H \vdash S \setminus x_0 \vdash$ și $H \vdash S \setminus x_0 \vdash, S \dashv$ pot fi obținute de un perceptron $\Leftrightarrow L \vdash S \setminus x_0 \vdash$ din dep. în $x_0 \in H$

Lema 2: $L(S) = L(S \setminus \{x_0\}) + L(S \setminus \{x_0\})$

Lema 3: $L_{n-1}(S \setminus \{x_0\}) = L(P) \rightarrow P este un set de dim. n-1$

NOT: $\hat{L}(m, d) = \max L(S) = \text{nr. max. de partiții care pot fi formate cu } d \text{ parametri}$

$\hat{L}(m, d) \leq \hat{L}(m-1, d) + \hat{L}(m-1, d-1)$

$D(m, d) \leq D(m-1, d) + D(m-1, d-1)$

$D(m, d) = \text{nr. de mult. liniar separabile cu } m \text{ puncte într-o poziție generală cu ext. de dim. } d \text{ care pot fi învățate de un perceptron}$

UPPER BOUND: $m! / \text{nr. seturi de date}^{\text{max}}$

$$D(m, d) = \begin{cases} 2 \sum_{i=0}^d C_{m-1}^i, & m > d+1 \\ 2^m, & m \leq d+1 \end{cases}$$

② CURS 8: Separabilitatea liniară \rightarrow avem o fct. care ia val. $\in \{-1, 1\}$

- pt. un perceptron distinge 2 semiplane:

$H^+ = \{x | y = 1\} = \{x : w^T x - b \geq 0\}$ - marginie
 $H^- = \{x | y = -1\} = \{x : w^T x - b \leq 0\}$ - superplan

- $b = 0 \Rightarrow$ separare liniară orizontală

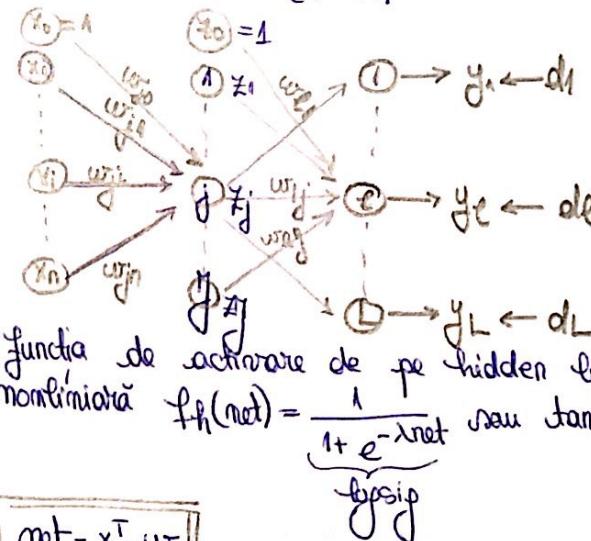
- în 2D - superplanul de separare este o dreaptă

$S^+ = \{x : z_i, d_i = 1 \text{ și } z_i = x_i\}$
 $S^- = \{x : z_i, d_i = -1 \text{ și } z_i = x_i\}$

DEF: Spunem că S^+ și S^- sunt liniar separabile dacă $\exists H$ a.i. $S^+ \subset H^+$ și $S^- \subset H^-$ (în cazul 2D ex. $\begin{matrix} + \\ - \end{matrix} \rightarrow H^+ \cap H^- = \emptyset$) - o condiție suficientă pt. S^+ și S^- să fie liniar separabile \Leftrightarrow înfășurătoarea convexă a lui S^+ și a lui S^- să fie o linie liniar separabilă \Leftrightarrow $S^+ \cup S^- = \emptyset$

NUMARUL DE SETURI DE ANTRENARE INVATABILE: $L(S) = \text{nr. de dihotomii liniar separabile în } S$ implementate de un perceptron dihotomic aka partiții

CURS 10: RETELE PERCEPTRON MULTISTRAT (RPM)



- fundia de activare de pe hidden layer, sigmoid
 $f_h(\text{net}) = \frac{1}{1 + e^{-\lambda \text{net}}}$ sau tanh

$$\text{met} = x^T \cdot w$$

OBJECTIV: Să ajungăm $J(m+1) + L(y+1)$

SSE = sum. of squared errors

VREM SA: Minimizăm $E(w) = \frac{1}{2} \sum_{l=1}^L (d_l - y_l)^2$

varianta incrementală, pt. un exemplu

UPDATE PE STRAT DE IESIRE:

$$w_{lj}^{\text{nou}} = w_{lj}^{\text{vechi}} - p_0 \frac{\partial E}{\partial w_{lj}} = w_{lj}^{\text{vechi}} + p_0 (d_l - y_l) \cdot f'(net_l) \cdot z_j$$

$$w_{lj}^{\text{nou}} = (w_{lj}^{\text{vechi}} + p_0 (d_l - y_l) \cdot f'(net_l)) \cdot z_j$$

$$\text{met}_l = \sum_{j=0}^n w_{lj} \cdot z_j \rightarrow \text{suma pt. perceptronul de ieșire } l, y_l = f_l(\text{met}_l)$$

$$z_j = f_h \left(\sum_{i=0}^d w_{ji} \cdot x_i \right) = f_h(\text{met}_j)$$

BACKPROPAGATION = propagarea eroilor de la jumătate ($d_l - y_l$) ampre output layer și apoi către hidden layer pt. updateul ponderilor

UPDATE PE STRAT ASCUNS:

$$\Delta w_{ji} = -p_h \frac{\partial E}{\partial w_{ji}}$$

$$\frac{\partial E}{\partial w_{ji}} = \frac{\partial E}{\partial z_j} \cdot \frac{\partial z_j}{\partial \text{net}_j} \cdot \frac{\partial \text{net}_j}{\partial w_{ji}} \Rightarrow$$

$$\frac{\partial E}{\partial z_j} = f_h(\text{net}_j) = x_i$$

$$\frac{\partial \text{net}_j}{\partial w_{ji}} = \frac{1}{2} \sum_{l=1}^L (d_l - f_l(\text{met}_l))^2$$

$$\frac{\partial \text{net}_j}{\partial w_{ji}} = - \sum_{l=1}^L (d_l - y_l) \cdot f'_l(\text{met}_l) \cdot w_{lj}$$

$$w_{ji}^{\text{nou}} = w_{ji}^{\text{vechi}} + p_h \left[\sum_{l=1}^L (d_l - y_l) \cdot f'_l(\text{met}_l) \cdot w_{lj} \right] \cdot f_h(\text{net}_j) \cdot x_i$$

Puncte în poziție generală \Rightarrow m pct. într-un sp. d-dimensional sunt în poz. generală dacă nu există hiperplan (de dim d-1) care să contină d+1 puncte; prob. ca un set aleator T să poată fi clasif. de un perceptron Rosenblatt =

$$P(m, d) = \frac{D(m, d)}{2^m} = \left(\frac{1}{2}\right)^{m-1} \sum_{k=0}^{K-1} C_{m-1}^k$$

- Dacă S, S' seturi de m pct. în poz. gen. \Rightarrow $D(S) = D(S') = \hat{D}(m, d) = D(m, d)$

- Dacă S' nu are m pct. în poz. gen \rightarrow $D(S) < D(m, d)$

Inputuri cu valori binare - avem d variabile și alăptem training set doar cele 2^d inputuri $B(d) \leq D(2^d, d) < \frac{2^d}{(d-1)!}, \lim_{d \rightarrow \infty} \frac{\log_2 B(d)}{d^2} = 1$

$$\begin{array}{cccc} 1 & 2 & 3 & 4 \\ B(d) & 4 & 14 & 104 & 1882 \end{array}$$

- nu se atinge $\lim D(2^d, d)$ pt. că punctele nu sunt în poz. generală

COMPLEXITATEA - nr. minim de elemente dintr-un set care să separe m puncte, unde $x \in \mathbb{R}^d$, și la univătă setul de antrenare T fară eroare \forall

$$P_{d*}(d) = P(m, d) - P(m, d-1) = \sum_{i=0}^d C_{m-1}^i \left(\frac{1}{2}\right)^{m-1}, d=0, m-1$$

↪ probabilitatea ca un set T selectat random cu m el. poate fi corect clasif. de un percep. având $d* = d$ inputuri

$$P_{Nf}(m) = P(m, d) - P(m+1, d) = \frac{C_m^d}{2^n}, m > d$$

↪ cel mai mare m care are input d-dim și poate fi clasif. de un perceptron

Dim. max. set antrenare care pot fi corect clasif de un percep. cu d dimensiuni $\approx 2d$

MT. NESEPARABILĂ / ① $w_{ji} \neq b$ care clasif. cel mai mare nr. de ex. consecutive corect (ROCKET ALG.) ② PROBABILISTIC PERCEPTRON ALG. - celi imbaricării w dacă găsim un ex. greșit și o sumă totă probabilitate

PROBABILITATEA CLASIF. AMBIGUE (d 2 moduri, ambele clasif. corect training set, dar nu sunt de acord pe modul set)

$$A(m, d) = \frac{D(m, d-1)}{D(m, d)}$$

PT. LOGSIG: $\varphi'(net) = \lambda \varphi(net)[1 - \varphi(net)]$

PT. TANSIG: $\varphi'(net) = \beta(1 - \varphi^2(net))$

- varianta batch minimizată: $\frac{1}{2} \sum_{k=1}^m \sum_{l=1}^L (d_k^{(l)} - y_k^{(l)})^2$

Avantaje incremental: ① mai puțină memorie ocupată ② fiecare pas e random și explorată un spațiu mai mare de soluții; (mai calitativ)

STOCHASTIC GRADIENT DESCENT = INCREMENTAL / STANDARD GR. DESC = BATCH

BACKPROPAGATION CONVERGE CĂTRE UN MIN LOCAL CÂND SUPRAFAȚA NU E CONVEXĂ.

(se aplică pt. doar algoritm pe gradiențe)

Curs 11: Coborâre pe gradient - minibatchuri m
 $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}\}$

$$E(w) = \frac{1}{m} \sum_{j=1}^m \sum_{l=1}^L (d_j^{(l)} - y_j^{(l)})^2$$

$$w^{k+1} = w^k - \rho_{k+1} \cdot \nabla_{w^k} E(w^k)$$

↳ drb. să scădă la fiecare iteratie, altfel introduce zig-zag; pt. conv:

$$\sum_{k=1}^K p_k = \infty \text{ și } \sum_{k=1}^K p_k^2 < \infty$$

Dezavantaje BACKPROPAGATION:

- ① viteza lentoare de convergență
- ② oscilatii zig-zag
- ③ minimum local
- ④ eroare mică, dar overfitting
- ⑤ sensibil la initializare / la rata de învățare

IMPUNATĂTIRI BACKPROPAGATION

- ① Strategii de initializare parametri (fără corespondență) ex: v.a. pe interval / distribuție gaussiană
- ② învățare cu moment: $\Delta w_i(t) = -\rho \frac{dE}{dw_i(t)} + \alpha' (w_i(t-1) - w_i(t-1))$
 - α' controlă că contribuția gradientelor din trecut; update și mai rapid multi gradienti în aceeași direcție
 - acceleră coborârea în direcția media; previne oscilații
- ③ rata de învățare variabilă
 - dacă $|\epsilon| \gg \rightarrow$ oscilații
 - dacă $|\epsilon| \ll \rightarrow$ converge prea lent
 - dacă $|\epsilon| \rightarrow \rightarrow \epsilon_+$ $\Rightarrow |\epsilon| \rightarrow \rightarrow \epsilon_+$
 - dacă $|\epsilon| \rightarrow \rightarrow \epsilon_- \Rightarrow |\epsilon| \rightarrow \rightarrow \epsilon_-$

Curs 12: Variable learning rate

- Adagrad: adaptive gr. desc - microscopă în funcție de toate ∇^2
- RMSProp - modif. Adagrad
- Adam - adaptive momentum

Curs 13: COBORÂRE PE GRADIENT FOLOSIND CAUTAREA PE DIRECȚIA LUI (STEEPST DESCEND)

- ① initializăm pt. de start \mathbf{z}_0
- ② cît timp nu se îndeplinește condiția de oprire
- ③ calculăm gradient în \mathbf{z}_k
- ④ căutăm α în direcția gradientului a.i. minimum (line search)
- ⑤ repetăm pas 2

DEZAVANTAJE:

- pe poz. convex prezintă zig-zag
- la fiecare pas alege o soluție de minimum local
- pe direcția coborârii pe gradient
- viteza lentoare de convergență

POATE FI IMPUNATĂTIT CU ALG. GR. CONJUGAT

- la fiecare pas alege o direcție care postrează o comp. comună de dir. anterioară

CURS 14.

① REUNIUNE FINITĂ DE POLIEDRE:

Dacă R^d este partitionat într-o reunire finită de poliedre, poate fi imp. de se trăce cu hidden layers 1 și 2 suficient de mult și un singur perceptron pe stratiul 3 (output).

② SANDWICH - $\forall m$ vectori în poz. generală în R^d pot fi împărțiți în S^+ și S^- de către feedforward net cu $\lceil \frac{m}{2} \rceil$ percep托on pe hidden layer și 1 perceptron pe output layer.

COMPLEX. UNEI RPM $p \leq m + (1 + \frac{2}{d}) + 1 \approx m$

REPREZENTAREA FCT. BOOLENE ARBITRARE

$$\varphi(\mathbf{x}_1, \dots, \mathbf{x}_n) = y$$

1) fct FND = $K_1 V \dots V K_n$, și fiecare $K_i = a_{i1} \dots a_{in}$

ai sau 1 ai

$$2) w_{ji} = \begin{cases} 2 & \varphi_{ji} = \mathbf{x}_i \\ -2 & \varphi_{ji} = \bar{\mathbf{x}}_i \end{cases}$$

$$- \Theta = m - 1 + \frac{1}{2} \sum_{j=1}^n w_{ji} \quad \boxed{- \Theta = m - 1 + \frac{1}{2} \sum_{j=1}^n w_{ji}} \quad \text{- bias}$$

3) mecanism de ieșire cu m întări

$$w_{(m_i)_K} = 2, i_1 = 1, \dots, m$$

$$- \Theta_{m+1} = 1$$

- hardlim pe percep托onii

SEMINAR: ① (Ω, \mathcal{F}, P) - spațiu de probabilitate

Ω - spațiu de selecție, finit / infinit (numărabil, nenumărabil)

\mathcal{F} - σ -algebră $\subseteq P(\Omega)$, $|P(\Omega)| = 2^{|\Omega|}$

$$\Omega \in \mathcal{F}$$

$A \in \mathcal{F} \Rightarrow \bar{A} \in \mathcal{F}$, A eveniment

$\bigcup_{i=1}^n A_i \in \mathcal{F}$, dacă fiecare $A_i \in \mathcal{F}$

$P: \mathcal{F} \rightarrow [0,1]$, a) $P(A) \geq 0, \forall A \in \mathcal{F}$ even.

$$b) P(\Omega) = 1$$

$$c) P(A \cup B) = P(A) + P(B)$$

dacă $A \cap B = \emptyset$

$$P(\emptyset) = 0, P(\bar{A}) = 1 - P(A)$$

$$P(A|B) = \frac{P(A \cap B)}{P(B)}, P(B) > 0 = \text{probabilitatea}$$

Def: Stând A_1, \dots, A_n partitie a lui Ω , A stând că B se devorează

- i) $A_i \cap A_j = \emptyset, \forall i, j, i \neq j$
- ii) $\bigcup_{i=1}^n A_i = \Omega$
- iii) $P(A_i) > 0$

FORMULA PROBABILITĂȚII TOTALE: A_1, \dots, A_n partitie ale lui Ω , $X \in \mathcal{F} = P(\Omega)$

$$X = X \cap \Omega = X \cap (\bigcup_{i=1}^n A_i) = \bigcup_{i=1}^n (X \cap A_i)$$

$$P(X) = P\left(\bigcup_{i=1}^n (X \cap A_i)\right) = \sum_{i=1}^n P(X \cap A_i) = \sum_{i=1}^n P(X|A_i) \cdot P(A_i)$$

FORMULA LUI BAYES | $P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$

A_1, \dots, A_n partitie a lui Ω , $X \in P(\Omega)$

$$P(A_i|X) = \frac{P(A_i) \cdot P(X|A_i)}{\sum_{j=1}^n P(A_j) \cdot P(X|A_j)}$$

② Clasificatorul Bayesian

$$d: \mathbb{R}^m \rightarrow \{0, 1, \dots, m-1\}$$

↳ clasificatorul ↳ clase posibile

CLAS. BAYESIAN: $d^*: \mathbb{R}^m \rightarrow \{0, 1, \dots, m-1\}$

ia decizia astfel:

$$d^*(x) = \underset{i=0, 1, \dots, m-1}{\operatorname{argmax}} (p(c_i|x))$$

$$p(c_i|x) = f(x|c_i) \cdot p(c_i) \rightarrow \text{prob. a priori}$$

$$\downarrow \text{prob. a posteriori}$$

↳ densitatea de prob. condiționată

③ CLASIF. BAYESIAN PT. DISTRIB. NORMALE MULTIDIM.



$$Y=0 \text{ femei } P(Y=0) = P \\ Y=1 \text{ bărbați } P(Y=1) = 1 - P$$

$x \in \mathbb{R}^m$; $y=0: f_{X|Y=0}(x) = f_0(x) =$

$$= \frac{1}{\sqrt{(2\pi)^m |\Sigma|}} \cdot e^{-\frac{1}{2} (x - \mu_0)^T \Sigma^{-1} (x - \mu_0)}$$

vector cu media

pt. fiecare proprietate

$f_0(x) \rightarrow$ densitatea în fiecare punct

$$\Sigma = \begin{pmatrix} \Sigma_{11} & \dots & \Sigma_{1m} \\ \vdots & \ddots & \vdots \\ \Sigma_{m1} & \dots & \Sigma_{mm} \end{pmatrix}$$

$$\sum_{ij} = \text{cov}(x_i, x_j) = E[(x_i - E(x_i))(x_j - E(x_j))]$$

// dacă avem datele date putem estimă media și putem calcula matricea de cov Σ_0

$$f_1(x) = f_{X|Y=1}(x) = \frac{1}{\sqrt{(2\pi)^m |\Sigma_1|}} e^{-\frac{1}{2} (x - \mu_1)^T \Sigma_1^{-1} (x - \mu_1)}$$

④ REGRESIE SIMPLĂ

$$S = \{(x_1, y_1), \dots, (x_m, y_m)\} - \text{mt. de antrenare}$$

Vrem să approximăm $y_i = ax_i + b$

$$\text{Considerăm } E(a, b) = \sum_{i=1}^m (y_i - (ax_i + b))^2$$

$(a^*, b^*) = ?$ u.i. $E(a^*, b^*) = \text{minimum}$
pt. de minimum global

$$\text{CONDITIE 1: } \frac{\partial E}{\partial a} (a^*, b^*) = 0, \frac{\partial E}{\partial b} (a^*, b^*) = 0$$

$$\text{CONDITIE 2: } H_E(a^*, b^*) = \begin{pmatrix} \frac{\partial^2 E}{\partial a^2} (a^*, b^*) & \frac{\partial^2 E}{\partial a \partial b} (a^*, b^*) \\ \frac{\partial^2 E}{\partial b \partial a} (a^*, b^*) & \frac{\partial^2 E}{\partial b^2} (a^*, b^*) \end{pmatrix}$$

$$\frac{\partial E}{\partial a} = \frac{\partial E}{\partial a} (E(a, b)) = 2 \sum_{i=1}^m (y_i - (ax_i + b)) \cdot x_i$$

$$\frac{\partial E}{\partial b} = 2 \sum_{i=1}^m (y_i - (ax_i + b))(-1)$$

$$\frac{\partial^2 E}{\partial a^2} = 2 \sum_{i=1}^m x_i^2, \frac{\partial^2 E}{\partial a \partial b} = \frac{\partial E}{\partial a} (2 \sum_{i=1}^m (y_i - (ax_i + b))(-1)) =$$

$$\frac{\partial^2 E}{\partial b^2} = 2 \sum_{i=1}^m 1 = 2m, \frac{\partial^2 E}{\partial b \partial a} = \sum_{i=1}^m x_i^2$$

$$a^* = \frac{\bar{xy} - \bar{x} \cdot \bar{y}}{\bar{x}^2 - \bar{x}^2} = \frac{\text{cov}(x, y)}{\text{var}(x)}$$

$$b^* = \bar{y} - a^* \bar{x}$$

$$\text{NOT: } \bar{x} = \frac{1}{m} \sum x_i, \bar{y} = \frac{1}{m} \sum y_i$$

$$\bar{x}^2 = \frac{1}{m} \sum x_i^2, \bar{xy} = \frac{1}{m} \sum x_i y_i$$

REGRESIE MULTIPLEX:

$$S = \{(\mathbf{x}_1, \dots, \mathbf{x}_m, \mathbf{y}_1), \dots, (\mathbf{x}_m, \dots, \mathbf{x}_m, \mathbf{y}_m)\}$$

$$\mathbf{y} \sim \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$$

$$\beta = (\beta_0, \dots, \beta_p)$$

$$E(\beta) = \sum_{i=1}^m (y_i - (\beta_0 + \beta_1 x_{1i} + \dots + \beta_p x_{pi}))^2$$

ACELEAASI CONDIȚII:

$$\left\{ \begin{array}{l} \frac{\partial E(\beta)}{\partial \beta_0} = 0 \\ \vdots \\ \frac{\partial E(\beta)}{\partial \beta_p} = 0 \end{array} \right. \Rightarrow E(\beta^*) \geq 0$$

$$E(\beta^*) = \sum \varepsilon_i^2 = \sum \varepsilon_i \cdot \varepsilon_i^T = \| \varepsilon \|_2^2 = \varepsilon^T \cdot \varepsilon = (\mathbf{y} - \mathbf{x}^T \beta)^T (\mathbf{y} - \mathbf{x}^T \beta)$$

$$E(\beta^*) = \mathbf{y}^T \mathbf{y} - \mathbf{y}^T \mathbf{x}^T \beta - \beta^T \mathbf{x} \mathbf{y} + \beta^T \mathbf{x} \mathbf{x}^T \beta$$

- REGUΛI:**
- ① $\nabla_{\beta} (\beta^T \mathbf{x}) = \mathbf{x}$
 - ② $\nabla_{\beta} (\beta^T A \beta) = (A + A^T) \beta$

$$\boxed{\beta^* = (\mathbf{x} \mathbf{x}^T)^{-1} \mathbf{x}^T \mathbf{y}}$$

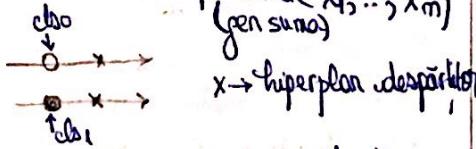
CU REGULARIZARE - adică $+ \lambda \|w\|_2^2$

$$\Rightarrow \beta^* = (\mathbf{x}^T \mathbf{x} + \lambda I_p)^{-1} \mathbf{x}^T \mathbf{y}$$

⑤ DEF. FORMALĂ A UNUI PERCEPTRON

$P = (w, P, G_w, f_P)$ → funcția de transfer
 ↓ ponderi, bias $P \rightarrow$ parametrii lui f_P
 $G_w \rightarrow$ funcția de
 diferențiere (cum sunt
 procesate x_1, \dots, x_m)
 (gen sumă)
 $x \rightarrow$ hiperplan despărțitor

⑥ $|d|=1$ $m=1$



$m=2$

$$\begin{aligned} & \text{---} \xrightarrow{000000} \text{---} \xrightarrow{000000} 2 \left(\sum_{j=0}^1 C_{m-1}^j \right) = \\ & \text{---} \xrightarrow{000000} \text{---} \xrightarrow{000000} 2(1+3) = 8 \\ & \left\{ \begin{array}{l} \text{---} \xrightarrow{000000} \text{---} \xrightarrow{000000} \text{pot fi} \\ \text{---} \xrightarrow{000000} \text{---} \xrightarrow{000000} \text{invariante} \end{array} \right. \\ & \text{dim } W_1 \text{ total } \end{aligned}$$

$|d|=2$ $D(3,2) = 2^3 = 8 \Rightarrow$ date multimedie liniar separabile

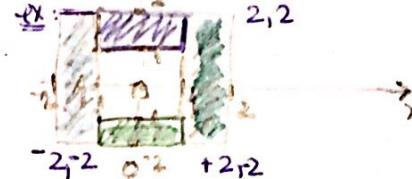
$|d|=3$ $D(4,3) = 2^4$

$m=5$, $D(5,3) = 2(C_4^0 + \dots + C_4^3) = 30$ (din max 32)
 $\Rightarrow \exists 2$ multimedii în poz. generală care nu sunt liniar separabile

E_0 O F G
 D_0 C
 A B

B, D, E, F, G în poz.
 generală, D, F eticheta 0
 B, E, G eticheta 1

\Rightarrow nu sunt liniar separabile (sunt invers/simetric)

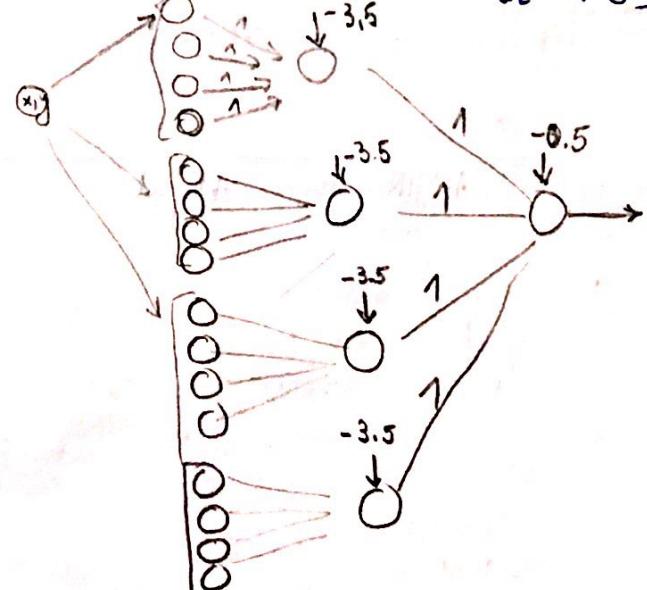


$$\text{pt. } \textcolor{blue}{\bullet} : \begin{cases} -y = -2 \Rightarrow -y + 2 = 0 \\ -x = 1 \Rightarrow -x - 1 = 0 \\ y = -2 \Rightarrow y + 2 = 0 \\ x = -2 \Rightarrow x + 2 = 0 \end{cases} \quad \mathbb{I}W = \begin{bmatrix} x & y \\ 0 & -1 \\ -1 & 0 \\ 0 & 1 \\ 1 & 0 \\ -1 & 1 \\ 2 & 1 \\ 2 & 0 \\ -1 & 0 \\ 1 & 1 \\ -1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 1 & -1 \\ 2 & 1 \\ 2 & 0 \\ -1 & 0 \\ 0 & -1 \\ 2 & 0 \\ 0 & 1 \\ 2 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$\text{pt. } \textcolor{black}{\bullet} : \begin{cases} -y = -2 \Rightarrow -y + 2 = 0 \\ x = -1 \Rightarrow x + 1 = 0 \\ -x = -1 \Rightarrow -x + 1 = 0 \\ y = 1 \Rightarrow y - 1 = 0 \end{cases} \quad b_1 = \begin{bmatrix} 2 \\ -1 \\ 1 \\ 2 \\ 2 \\ 2 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 2 \\ 2 \\ 2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\text{pt. } \textcolor{green}{\bullet} : \begin{cases} x = -1 \Rightarrow x + 1 = 0 \\ -x = -1 \Rightarrow -x + 1 = 0 \\ y = -2 \Rightarrow y + 2 = 0 \\ -y = 1 \Rightarrow -y - 1 = 0 \end{cases} \quad \mathbb{I}W = \begin{bmatrix} x & y \\ 1 & 0 \\ -1 & 0 \\ 1 & 1 \\ 1 & -1 \\ 1 & 0 \\ 1 & 1 \\ 2 & 0 \\ -1 & 0 \\ 0 & -1 \\ 2 & 0 \\ 0 & 1 \\ 2 & 1 \\ 0 & 1 \\ 2 & 0 \\ 0 & 1 \\ 2 & 1 \\ 2 & 0 \\ -1 & 0 \end{bmatrix}$$

$$\text{pt. } \textcolor{red}{\bullet} : \begin{cases} -y = -2 \Rightarrow -y + 2 = 0 \\ y = -2 \Rightarrow y + 2 = 0 \\ x = 1 \Rightarrow x - 1 = 0 \\ -x = -2 \Rightarrow -x + 2 = 0 \end{cases} \quad b_2 = \begin{bmatrix} 2 \\ 2 \\ 2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$



$$\mathbb{I}W_2 = \begin{bmatrix} 1 & 1 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & \dots & 0 \\ 0 & \dots \\ 0 & \dots \\ 0 & \dots \end{bmatrix} \quad b = \begin{bmatrix} -3,5 \\ 1 \end{bmatrix}$$

$$\mathbb{I}W_3 = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \quad b_{W_3} = [-3,5]$$

- 2 vectori sunt ortonormali dacă au produsul scalar 0

EQUAȚIE DREAPTA

2 puncte: $\frac{x - x_A}{x_B - x_A} = \frac{y - y_A}{y_B - y_A}$

pantă: $y - y_A = m(x - x_A)$

DERIVATE ① $(fg)' = f'g + fg$
 ② $\left(\frac{f}{g}\right)' = \frac{f'g - fg'}{g^2}$

③ $(ax)' = a \cdot \ln a$ ④ $(\cos x)' = -\sin x$

⑤ $(\sin x)' = \cos x$ ⑥ $(\ln x)' = \frac{1}{x}$

⑦ $(\log_a x)' = \left(\frac{\ln x}{\ln a}\right)' = \frac{1}{\ln a} \cdot \frac{1}{x}$

⑧ $f'_p(x) = \frac{1}{\cos^2 x} = 1 + f'_p^2 x$

⑨ $f'_p(x) = -\frac{1}{\sin^2 x} = -(1 + f'_p^2 x)$

⑩ $\arcsin' x = \frac{1}{\sqrt{1-x^2}}$

⑪ $\arccos' x = -\frac{1}{\sqrt{1-x^2}}$

⑫ $\arctg' x = \frac{1}{1+x^2}$

⑬ $\text{arcctg}' x = -\frac{1}{1+x^2}$

⑭ $(f \circ g)'(x) = f'(g(x)) \cdot g'(x)$

$\|x\| = (x^T \cdot x)^{\frac{1}{2}} = \sqrt{x_1^2 + \dots + x_m^2}$

BACKPROPAGATION ALGORITHM

① initializăm ponderile

② do for each training ex.

predict = $\text{dim}(\text{net}, \text{ex})$

actual = teacher-output(ex).

compute error(prediction - actual) at the output units

compute Δw_{hi} , + weights from hidden to output

compute Δw_i , + weights from input to hidden update weights

③ until all examples classif. correctly or another stopping condition

④ return network

ALTE CHESTII:

- rezumatele plătoare des întâlnite la logice (se dorește o rată de învățare mare scăzută)

SATURAREA NEURONILOR

- problema saturării neuronului are legătură cu învățarea parametrilor pe parcursul antrenării

- pt. backprop. se aplică funcții neliiniare care nu sunt diferențiale. Se folosesc des tanimp în logici, însă acestea nu sunt intervale compacte în care pot lua valori; astfel, trebuie să se compresze un interval foarte larg într-unul foarte mic

- un neuron este saturat când ia val. foarte apropiate

MULTIME ANTRENARE = un set de exemple folosit pt. învățarea ponderilor retelei perceptrone multiple optime c.r. să minимizăm eroare

MULTIME VALIDARE = un set de exemple folosit pt. ameliorarea arhitecturii retelei

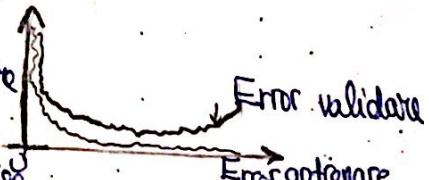
- pt. căutarea nr. optim de neuroni de pe hidden layer sau pt. early stopping

MULTIMEA DE TEST = exemple folosite pentru evaluarea performanței retelei

→ furnizează o evaluare obiectivă a modelului în cîmpul concrevenii

EARLY STOPPING:

- eroarea de antrenare scade, pe cînd eroarea de validare nu scade



- dacă un anumit număr de iterări, eroarea de validare nu scade, oprire antrenamentul și revenire la parametrii de la momentul anterior de minim (un fel de regulație - eficientă și simplă)

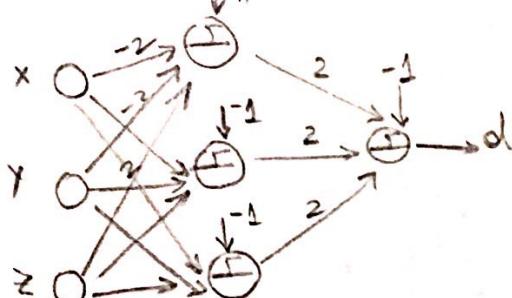
PUNCT STACIONAR ÎN ORIGINE: derivații și anlocuri de 0

2016 - subiecte

II. 2. reteaua se simplifică. $F: \{0,1\}^3 \rightarrow \{0,1\}$

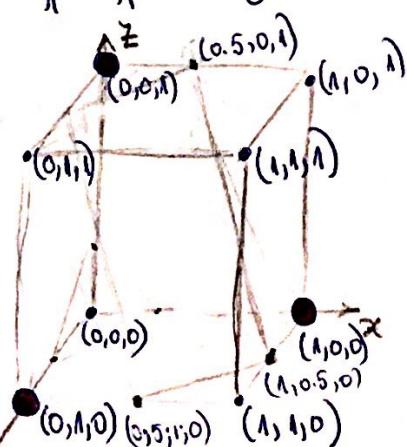
$$F(x,y,z) = (1-x)(1-y)(z) + (1-x)(y)(1-z) + x(1-y)(1-z) - \text{arhitectură ne-sandwich}$$

$$F(x,y,z) = (1-x \wedge y \wedge z) \vee (1-x \wedge y \wedge \neg z) \vee (x \wedge y \wedge \neg z)$$



3. Construiește reteaua aplicând teorema sandwich. Verif. funcționarea corectă a rețelei

	x	y	z	$F(x,y,z)$
①	0	0	0	0
②	0	0	1	1
③	0	1	0	1
④	0	1	1	0
⑤	1	0	0	1
⑥	1	0	1	0
⑦	1	1	0	0
⑧	1	1	1	0

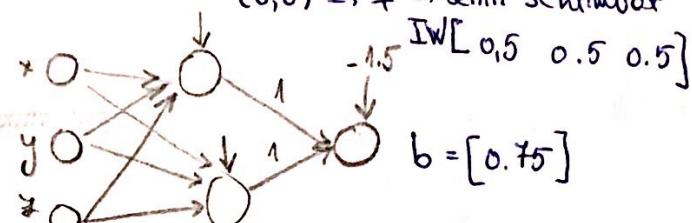


SEMIPLAN 1:

x	y	z	$F(x,y,z)$
0.5	1	0	1
1	0.5	0	1
0.5	0	1	1

// dezvoltăm pe coloane

$$\begin{array}{|cccc|} \hline x & y & z & F \\ \hline 0.5x & 1-y & -z & 0 \\ 1-x & 0.5y & -z & 0 \\ 0.5x & -y & 1-z & 0 \\ \hline \end{array} = (0.5x)(0.5-y)(1-z) + (1-x)(-y)(-z) + (0.5-x)(0.5y)(-z) + (-z)(-y)(0.5-z) - (1-y)(1-z)(1-x) \\ = (0.5^2 - 0.5y - 0.5z + xy)(1-z) + (y + zy)(-z) + (0.5 - 0.5y - z + 2yz)(-z) + (z \cdot 0.5 - zy)(0.5-z) - (zy)(0.5-z) - (1-y - z + 2yz)(1-x) = 0.5^2 - 0.5^2 z - 0.5y - 0.5yz - 0.5z + 0.5xz + 2yz - xyz + 2yz - 0.5z + 0.5yz + 2yz - 0.5^2 z - 0.5xz - 0.5yz + 2yz - 0.5yz + 2yz - 1 + x + y - 2yz + z - 2z - yz + 2yz \\ \Rightarrow -0.5x - 0.5y - 0.5z - 0.45 = 0 \\ (0,0) \rightarrow 0 \rightarrow \text{se în schimbă} \end{array}$$



// Analog semiplan 2

$$\text{III. } \begin{array}{ccc} x & \xrightarrow{\text{a}} & \textcircled{+} & \xrightarrow{\text{c}} & \textcircled{+} & \xrightarrow{\text{y}} \\ & & \uparrow \text{b} & & & \end{array} \quad x \xrightarrow{\text{a}} \textcircled{+} \xrightarrow{\text{b}} \textcircled{+} \xrightarrow{\text{c}}$$

a) spațiul H al ipotezelor purelin (domină $(x \cdot a+b) \cdot c$)

b) $f: \mathbb{R} \rightarrow [0,1], f(z) = \frac{1}{1+e^{-z}}$

$$\log \sinh(\log \sinh(xa) \cdot b + c) = \frac{1}{1+e^{-2x}} \Leftrightarrow$$

$$\Leftrightarrow \log \sinh(xa) \cdot b + c = \frac{1}{1+e^{-xa}} \cdot b + c = -2x$$

$$\lim_{x \rightarrow \infty} \frac{1}{1+e^{-xa}} \cdot b + c \rightarrow b + c \quad \infty$$

$$\lim_{x \rightarrow -\infty} -2x \rightarrow -\infty$$

NUME	FUNCTIE CRITERIU	FCT. ACTIVARE	GRADIENT	CONDITII	EXPLICATI
Stochastic μ -LMS rule	$J = \frac{1}{2} \sum_i (d_i - (x_i^T w)^2)$	$f(\text{net}) = \text{net}$	$[d^k = (x^k)^T w^k], x^k$	$\sum_i p_i^k \geq 0$ $\sum_{k=1}^K p_i^k = +\infty$ $\sum (p_i^k)^2 < \infty$	$\langle \cdot \rangle = \text{mean operator}$
AHK II	$J(w, b) = \frac{1}{2} \sum_i [(x_i^T w - b_i)^2]$ margin vect. $b > 0$	$f(\text{net}) = g(\text{net})$	$\Delta b_i^k = \begin{cases} p_1 \varepsilon_i^k, \varepsilon_i^k > \frac{-b_i^k}{p_1} \\ 0, \quad \text{otherwise} \end{cases}$	$0 < p_2 < \frac{2}{\max_i \ x_i\ ^2}$	
			$\Delta w = \begin{cases} p_2 (p_1 - 1) \varepsilon_i^k \cdot x_i^k, \varepsilon_i^k > \frac{-b_i^k}{p_1} \\ -p_2 \varepsilon_i^k \cdot x_i^k, -n < -n \end{cases}$		
AHK III	$J(w, b) = \frac{1}{2} \sum_i [(x_i^T w - b_i)^2]$ margin vect. $b > 0$	$f(\text{net}) = g(\text{net})$	$\Delta b_i^k = \begin{cases} p_1 \varepsilon_i^k, \varepsilon_i^k > \frac{-b_i^k}{p_1} \\ 0, \quad \text{otherwise} \end{cases}$	$b^k > 0$ $0 < p_1 < 2$ $0 < p_2 < \frac{2}{n}$	Converge, pt. costuri liniar si relativ separabile
			$\Delta w = \begin{cases} p_2 (p_1 - 1) \varepsilon_i^k x_i^k, \varepsilon_i^k > \frac{-b_i^k}{p_1} \\ 0, \quad \text{otherwise} \end{cases}$		
Delta rule for stochastic units	$J(w) = \frac{1}{2} \sum_i (d_i - \langle y_i \rangle)^2$	Stoch activ.: $y = \begin{cases} +1, & P(y=1) \\ -1, & P(y=-1) \end{cases}$ $P(y=1) = \frac{1}{e + e^{-2p_{\text{net}}}} \cdot$	$f[d^k = \tanh(p_{\text{net}}^k)] \cdot$ $[1 - \tanh^2(p_{\text{net}}^k)] x^k$	$0 < p < 1$	Performance in the average is \Leftrightarrow to the data rule applied to unit with deterministic character.
Seminar 4:	$E(w_j^{(1)}, w_j^{(2)}) = \frac{1}{2} \sum_i (y_k^i - d_k^i)^2$	Regula de update coborare pe gradient, vom imlementa:	$y^L = \begin{cases} y_1 \\ y_2 \\ \vdots \\ y_b \\ \vdots \\ y_L \end{cases} \Rightarrow \frac{\partial E}{\partial w_{ba}^{(2)}}(w_j^{(1)}, w_j^{(2)}) = (y^b - d^b) \cdot$ $\varphi'(\sum_{j=1}^L x_j \cdot w_{bj}^{(2)}) \cdot x_a$		
Stat	$w_j^{(1)(\text{new})} = w_j^{(1)(\text{old})} - \beta_1 \nabla w_j^{(1)} E(w_j^{(1)}, w_j^{(2)})$ $w_j^{(2)(\text{new})} = w_j^{(2)(\text{old})} - \beta_2 \nabla w_j^{(2)} E(w_j^{(1)}, w_j^{(2)})$	$y^L = \sum_{j=0}^L y_j \cdot w_j^{(2)}$			
	$y^L = \varphi(\sum_{j=0}^L x_j \cdot w_j^{(2)}) = \varphi(\sum_{j=0}^L w_j^{(2)} \cdot f(\sum_{l=0}^m w_l^{(l)} x_l))$				
	Stratul de ieșire:				
	$w_{ba}^{(2)(\text{new})} = w_{ba}^{(2)(\text{old})} - \beta_0 \frac{\partial E}{\partial w_{ba}^{(2)}}(w_{ba}^{(2)}, w_{ba}^{(2)})$ $\frac{\partial E}{\partial w_{ba}^{(2)}}(w_{ba}^{(2)}, w_{ba}^{(2)}) = \frac{\partial E}{\partial w_{ba}^{(2)}} \left(\frac{1}{2} \sum_{l=1}^L (y_l^L - d^L)^2 \right) =$ $= \frac{1}{2} \cdot 2 \sum_{l=1}^L (y_l^L - d^L) \cdot (y_l^L)' = \frac{\partial E}{\partial w_{ba}^{(2)}}$				