Învățarea rețelelor neuronale pe bază de exemple

Conf. dr. Călin Enăchescu, Universitatea Petru Maior Târgu-Mureş

The supervised learning process of a neural network, or the approximation of a smooth function using a set of examples, called training set, is ill-posed. Usually the training set doesn't contain enough information, therefore the condition of uniqueness is not satisfied. In order to transform the ill-posed problem of approximating a function from sparse examples into a well-posed problem, we need to take into consideration some a priori hypothesis about the function to be approximated. What is the weakest a priori hypothesis that can be considered without affecting the general frame of function approximation? The learning process (function approximation) is efficient if we obtain good generalization properties. But the generalization properties are a result of a certain level of redundancy, more precisely we can say that generalization properties are a result of the property that small changes of the input parameters results in small changes of the output parameters. But this property is named smoothness. Concluding, we can say that the learning process of a neural network is equivalent to the approximation of a smooth function from examples (the training set).

Cea mai semnificativă proprietate a rețelelor neuronale este capacitatea de a *învăța* din mediul înconjurător şi de a-şi îmbunătăți performanțele pe baza acestui proces de învățare. Rețeaua neuronală învață pe baza unui proces iterativ de ajustare a tăriilor sinaptice şi eventual al nivelului de activare. Dacă procesul de învățare decurge bine, atunci rețeaua neuronală acumulează tot mai multe informații, la fiecare iterație.

Evident că atunci când folosim termenul de "proces de învățare" ne situăm într-o terminologie mult prea largă, care este dependentă de mai mulți factori. Pentru aceasta vom defini, în contextul calculului neuronal, "învățarea" conform cu [4], în felul următor:

> *Def.1: Învățarea este un proces prin care parametrii rețelei neuronale sunt adaptați permanent prin intermediul unor stimuli proveniți de la mediul înconjurător căruia îi aparține rețeaua neuronală. Tipul de învățare este determinat de forma de modificare a parametrilor rețelei neuronale.*

Definiția de mai sus conține următoarea secvență de evenimente[2]:

- *Evenimentul 1*: Rețeaua neuronală *primeşte stimuli* de la mediul înconjurător;
- *Evenimentul 2*: Rețeaua neuronală *se modifică* ca răspuns la *stimuli*;
- *Evenimentul 3*: Ca urmare a acestor modificări permanente, care afectează structura sa internă, rețeaua neuronală *răspunde* de fiecare dată într-un *nou mod* mediului de la care vin stimuli.

Să încercăm să dăm o formulare matematică acestui proces descris mai sus. Pe baza celor prezentate mai sus şi în capitolul anterior, am văzut că ceea ce se modifică în cadrul procesului de învățare este tăria sinaptică. De aceea avem formularea matematică cea mai generală a procesului de învățare, exprimat prin formula:

$$w_{ji}(t+1) = w_{ji}(t) + \Delta w_{ji}(t) \qquad (1)$$

♦ $w_{ji}(t+1)$ şi $w_{ji}(t)$ reprezintă noua şi vechea valoare a tăriei sinaptice w_{ji} care uneşte axonul neuronului i de o dendrită a neuronului j.

BDD-A23844 © 2005 Editura Universității „Petru Maior”
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 00:56:15 UTC)

♦ $\Delta w_{ji}(t)$ reprezintă ajustarea aplicată tăriei sinaptice $w_{ji}(t)$, la momentul t, obținându-se valoarea $w_{ji}(t+1)$ la momentul $t+1$, în urma procesului de ajustare.

Ecuația (1) conține în mod evident efectele Evenimentelor 1, 2 şi 3 prezentate mai sus. Ajustarea $\Delta w_{ji}(t)$ este obținută ca urmare a unor stimuli ai mediului înconjurător (Evenimentul 1), iar valoarea modificată a tăriei sinaptice $w_{ji}(t+1)$ defineşte schimbarea din rețeaua neuronală, ca un rezultat al stimulilor prezentați rețelei neuronale (Evenimentul 2). Din momentul $t+1$ rețeaua neuronală răspunde într-un mod nou mediului înconjurător deoarece tăria sinaptică s-a modificat, devenind $w_{ji}(t+1)$ (Evenimentul 3).

Def. 2: Vom numi *algoritm de învățare*, un set de reguli predefinite care soluționează problema "învățării".

Un alt factor important relativ la procesul de învățare este modul de raportare a unei rețele neuronale la mediul înconjurător. În acest context putem defini:

Def. 3: Vom numi *paradigmă de învățare*, un model al mediului înconjurător în care are loc procesul de învățare al rețelei neuronale.

Învățare supervizată

Modificarea tăriilor sinaptice este făcută pe baza comparației dintre vectorul de ieşire $\mathbf{y}^{\mu} = (y_1^{\mu}, y_2^{\mu}, \ldots, y_m^{\mu})$ obținut la stratul de ieşire şi vectorul țintă $\mathbf{z}^{\mu} = (z_1^{\mu}, z_2^{\mu}, \ldots, z_m^{\mu}), \mu = 1, \ldots, P$, ce reprezintă rezultatul dorit a se obține la stratul de ieşire, când la stratul de intrare s-a prezentat vectorul de intrare $\mathbf{x}^{\mu} = (x_0^{\mu}, x_1^{\mu}, \ldots, x_n^{\mu}), \mu = 1, \ldots, P$ din mulțimea de antrenament.

Vectorul țintă $\mathbf{z}^{\mu}$ este furnizat de un *profesor (antrenor)*, de unde şi denumirea de învățare supervizată. Învățarea supervizată presupune prezentarea de către un antrenor a unor perechi de date de forma $(\mathbf{x}^{\mu}, \mathbf{z}^{\mu}), \mu = 1, \ldots, P$, ce formează o mulțime de date, numită *mulțime de antrenament*:

$$T = \left\{ \left(\mathbf{x}^{\mu}, \mathbf{z}^{\mu} \right) \middle| \mu = 1,2,\ldots,P \right\} \qquad (2)$$

Diferența dintre răspunsul obținut **y** şi răspunsul dorit **z**, reprezintă *eroarea* şi este folosită pentru a modifica tăriile sinaptice, pe baza unui algoritm specific, numit *lege de învățare*.

Natura statistică a procesului de învățare

Să considerăm un fenomen descris printr-un vector $\mathbf{x} \in \mathbf{R}^n$ ce reprezintă o mulțime de variabile independente, şi un scalar real $z \in \mathbf{R}$ ce reprezintă o variabilă dependentă. Elementele vectorului **x** pot fi considerente ca vând interpretări fizice diferite.

Să presupunem de asemenea, că avem o mulțime de N măsurători (observații) ale variabilei **x**, şi anume:

$$\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \ldots, \mathbf{x}_N \qquad (3)$$

şi o mulțime corespunzătoare de scalari z, notată:

$$z_1, z_2, z_3, \ldots, z_N \qquad (4)$$

În mod obişnuit, nu posedăm informațiile necesare despre relația exactă dintre variabilele **x** şi z. De aceea vom nota această relație astfel:

$$z = f(\mathbf{x}) + \varepsilon \qquad (5)$$

unde f este o funcție de variabila **x**, iar ε este *eroarea* reprezentată sub forma unei variabile aleatoare. Eroarea ε semnifică eroarea pe care o facem în estimarea dependenței funcționale dintre variabilele **x** şi z. Ecuația (5) de mai sus este un model statistic [4], numit *model regresiv*.

Putem defini funcția f a modelului regresiv ca fiind:

$$f(\mathbf{x}) = E[z|\mathbf{x}] \qquad (6)$$

BDD-A23844 © 2005 Editura Universității „Petru Maior”
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 00:56:15 UTC)

unde E este operatorul de medie statistică. $E[z|\mathbf{x}]$ reprezintă media condiţională, semnificând faptul că vom obţine, în medie, valoarea z, dacă avem o realizare particulară a lui $\mathbf{x}$.

O reţea neuronală reprezintă de fapt un mecanism fizic pentru a implementa acest obiectiv: predicţionarea lui z pe baza lui $\mathbf{x}$. Acest lucru se realizează prin codificarea informaţiei conţinută în mulţimea de antrenament $T = \{(\mathbf{x}_i, z_i) | \, i = 1,2,\ldots,N\}$ în tăriile sinaptice. Este evidentă interpretarea din punct de vedere al calculului neuronal, dată celor două mărimi $\mathbf{x}$ şi z: $\mathbf{x}$ reprezintă vectorul (stimulul) de intrare în reţeaua neuronală, iar z reprezintă valoarea ţintă (dorită) a se obţine la stratul de ieşire al reţelei neuronale.

Să notăm cu $\mathbf{w}$, vectorul tăriilor sinaptice a reţelei neuronale, care va avea rolul de a aproxima modelul regresiv exprimat prin ecuaţia (6). Vom nota cu y valoarea de ieşire generată de reţeaua neuronală. Atunci, prin propagarea valorii de intrare $\mathbf{x}$, de la stratul de intrare, către stratul de ieşire, unde obţinem valoarea y, putem scrie corespondenţa [3]:

$$y = F(\mathbf{x}, \mathbf{w}) \tag{7}$$

De asemenea, datorită faptului că mulţimea de antrenament, conţine şi vectori ţintă, care sunt furnizaţi de un antrenor, este evidentă analogia cu paradigma învăţării supervizate. De aceea, modificarea vectorului tăriilor sinaptice se va face printr-un proces iterativ, ca răspuns la semnalul eroare:

$$e = z - y \tag{8}$$

Dacă ar fi să reprezentăm grafic modelul regresiv (6), sub noua sa interpretare dată de calculul neuronal, am obţine diagrama de mai jos:

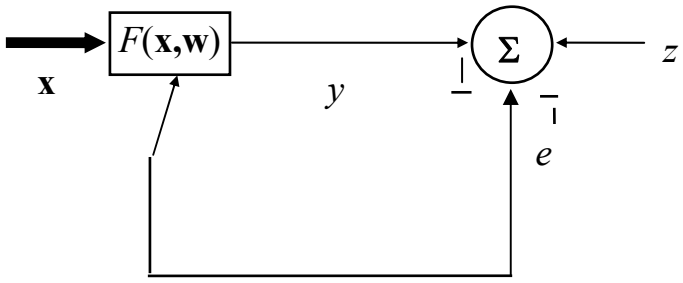


Fig. 1: Modelul corespunzător calculului neuronal.

Modificarea vectorului tăriilor sinaptice, se face folosind un algoritm de învăţare de tip corecţie a erorii MSE (3) sau (4). Putem atunci scrie [2]:

$$\mathbf{E}(\mathbf{w}) = \frac{1}{2} E[e^2] = \frac{1}{2} E[(z-y)^2] = \frac{1}{2} E[(z - F(\mathbf{x}, \mathbf{w}))^2] \tag{9}$$

Optimizarea reţelei neuronale înseamnă minimizarea funcţiei eroare. Pentru aceasta re scriem relaţia (9):

$$\begin{aligned} \mathbf{E}(\mathbf{w}) &= \frac{1}{2} E[(z - f(\mathbf{x}) + f(\mathbf{x}) - F(\mathbf{x}, \mathbf{w}))^2] = \\ &= \frac{1}{2} E[(z - f(\mathbf{x}))^2] + E[(z - f(\mathbf{x}))(f(\mathbf{x}) - F(\mathbf{x}, \mathbf{w}))] + \\ &+ \frac{1}{2} E[(f(\mathbf{x}) - F(\mathbf{x}, \mathbf{w}))^2] = \frac{1}{2} E[(z - f(\mathbf{x}))^2] + \frac{1}{2} E[(f(\mathbf{x}) - F(\mathbf{x}, \mathbf{w}))^2] \end{aligned} \tag{10}$$

Modelul general al procesului de învăţare

BDD-A23844 © 2005 Editura Universităţii „Petru Maior”
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 00:56:15 UTC)

Din studiul statistic al procesului de învățare am văzut echivalența dintre problema aproximării unei funcții descrise cu ajutorul unei mulțimi de antrenament $T = \{(\mathbf{x}_i, \mathbf{z}_i) | i = 1,2,\ldots,N\}$ cu procesul de învățare al unei rețele neuronale pe baza aceleaşi mulțimi de antrenament $T = \{(\mathbf{x}_i, \mathbf{z}_i) | i = 1,2,\ldots,N\}$. De asemenea modelele de aproximare prezentate corespundeau paradigmei de învățare supervizată. Un model de învățare supervizată are trei componente reprezentabile astfel [3]:

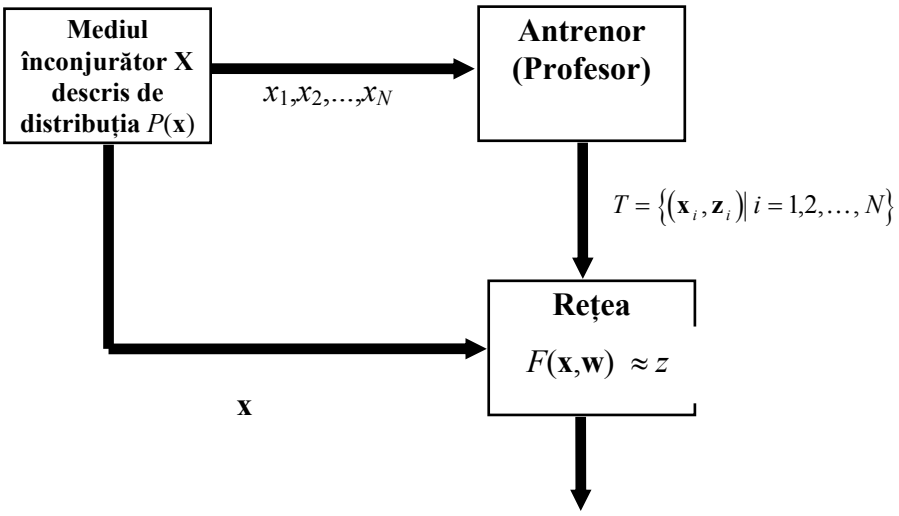


Fig. 2: Modelul învățării supervizate.

Cele trei componente sunt:

- *Mediul înconjurător* **X,** care transmite stimulul $\mathbf{x} \in \mathbf{X}$, generat de o distribuție probabilistică oarecare fixată $P(\mathbf{x})$;
- *Antrenorul,* care furnizează răspunsurile țintă **z**, ce se doresc a se obține la ieşirea rețelei neuronale, pentru orice vector de intrare **x**, în concordanță cu distribuția probabilistică fixă $P(z|\mathbf{x})$. Vectorii **x** şi **z** sunt legați prin relație funcțională necunoscută *f*:

$$\mathbf{z} = f(\mathbf{x}) \qquad (11)$$

- *Rețeaua neuronală* $F(\mathbf{x},\mathbf{w})$ - este capabilă să implementeze relația funcțională dintre **x** şi **z**, descrisă prin relația: $\mathbf{y} = F(\mathbf{x},\mathbf{w})$ (12)

Problema învățării constă în selectarea, pe baza unei mulțimi de antrenament $T = \{(\mathbf{x}_i, \mathbf{z}_i) | i = 1,2,\ldots,N\}$ cunoscută *a priori*, a funcției $F(\mathbf{x},\mathbf{w})$ ce aproximează vectorul țintă **z**, furnizat de antrenor. Selecția funcției $F(\mathbf{x},\mathbf{w})$ se bazează deci pe cele *N* elemente ale mulțimii de antrenament *T*, care sunt independent şi identic distribuite.

Problema învățării: *Problema fundamentală a învățării supervizate este dacă mulțimea de antrenament* $T = \{(\mathbf{x}_i, \mathbf{z}_i) | i = 1,2,\ldots,N\}$ *conține suficiente informații pentru a putea construi o funcție aproximantă* $F(\boldsymbol{x},\boldsymbol{w})$*, deci o rețea neuronală, capabilă să învețe cât mai bine datele de antrenament şi în plus să aibă capacitatea de* ***generalizare***.

Proprietatea de generalizare reprezintă capabilitatea unei rețele neuronale de a răspunde la date de intrare ce nu au făcut parte din mulțimea de antrenament. Este evident faptul că scopul învățării unei rețele neuronale trebuie să fie obținerea unei bune capacități de generalizare. Generalizarea poate fi privită, dacă considerăm rețeaua neuronală ca o aplicație între spațiul datelor de intrare şi spațiul datelor de ieşire (obținute la stratul de ieşire), ca fiind abilitatea de interpolare a aplicației respective.

BDD-A23844 © 2005 Editura Universității „Petru Maior”
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 00:56:15 UTC)

Să presupunem că după ce o rețea neuronală a efectuat faza de învățare, dorim să extragem o lege care să definească comportamentul ei. Vom reprezenta schematic modul de extragere a unei legi în Fig.3.

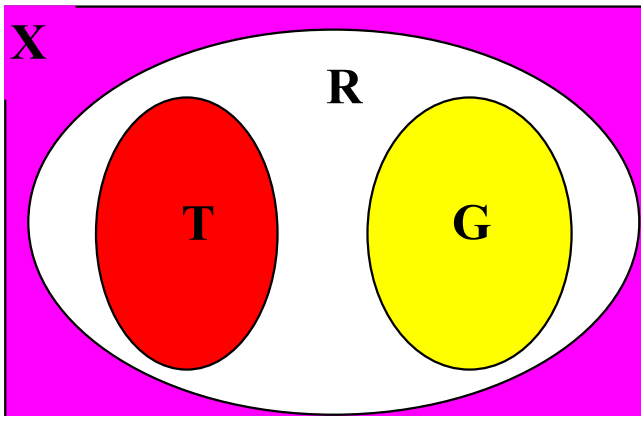


Fig.3: Reprezentarea schematică a modului de extragere a unei legi.

Să explicăm schema din figura de mai sus. **X** reprezintă spațiul tuturor datelor de intrare, perechi de forma (***vectori de intrare*, *vectori țintă***), date ce sunt consistente cu o anumită lege **R**. În procesul de învățare o submulțime a legii **R**, notată **T**, şi care reprezintă mulțimea de antrenament, este folosită pentru a învăța o rețea neuronală. După ce procesul de învățare s-a terminat, testăm capacitatea de generalizare a rețelei, cu ajutorul unei submulțimi $\mathbf{G} \subset \mathbf{R}$, disjunctă de **T**.

Putem deci concluziona că performanțele rețelei neuronale, relative la submulți-mea $\mathbf{T} \subset \mathbf{R}$, reprezintă capacitatea de memorare a rețelei, iar performanțele relative la submulțimea $\mathbf{G} \subset \mathbf{R}$, reprezintă capacitatea de generalizare a rețelei neuronale. De obicei **T** şi **G** sunt alese aleator din mulțimea **R**, ambele fiind generate de aceeaşi lege de distribuție.

De fapt, în procesul de învățare, rețeaua neuronală învață doar elementele sub-mulțimii **T**, fără a şti nimic despre **G** şi **R**. De aceea este natural ca această rețea neuronală, să fie capabilă de a generaliza orice mulțime de date de intrare care este consistentă cu **T**. Acest lucru este reprezentat în Fig.4.

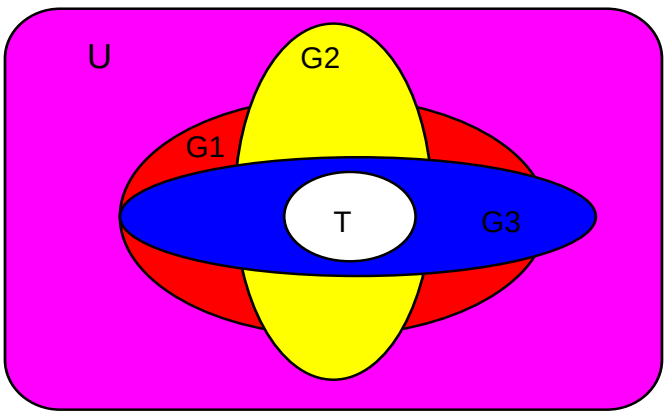


Fig.4. Reprezentarea schematică a capacității de generalizare a unei rețele neuronale.

Problema generalizării poate fi îngreunată dacă saturăm procesul de învățare a rețelei neuronale printr-un număr prea mare de date de antrenament. În această situație capacitatea de generalizare a rețelei neuronale este slabă. Ca un exemplu, în Fig.5., problema generalizării datorită supra-saturării procesului de învățare, privită prin prisma interpolării datelor de antrenament.

BDD-A23844 © 2005 Editura Universității „Petru Maior”
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 00:56:15 UTC)

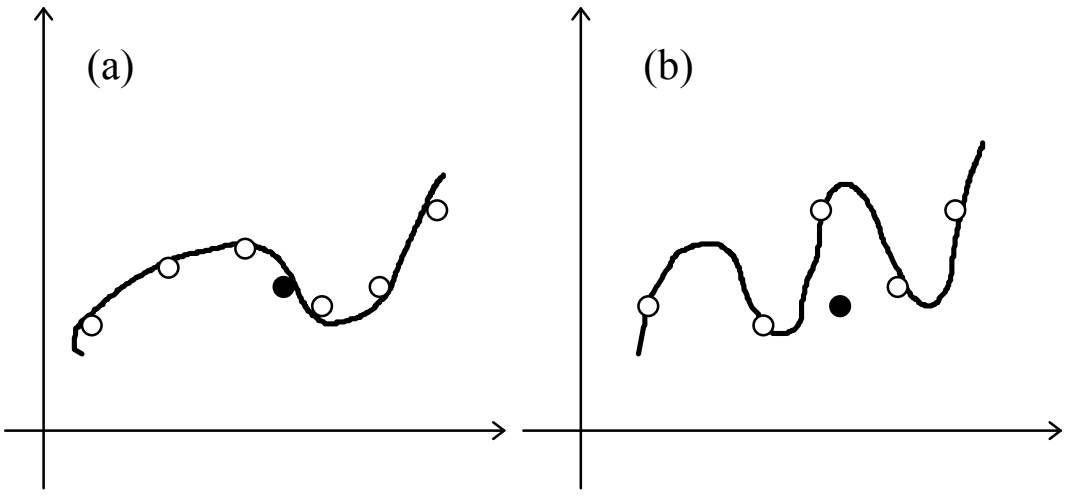


Fig.5. Reprezentarea schematică a problemei generalizării, unde avem:
o - date de antrenament; • - date pentru generalizare. (a). Învățare reuşită, generalizare bună. (b). Învățare saturată, generalizare slabă.

Aceste elemente referitoare la capacitatea de generalizare a rețelei neuronale, sugerează posibilitatea de a cuantifica estimativ capacitatea rețelei neuronale de a generaliza, în funcție de arhitectura sa şi de dimensiunea mulțimii de antrenament. Pentru aceasta, vom selecta din numeroasele posibilități de cuantificare a generalizării, următoarele:

- Numărul mediu de posibilități de generalizare în raport cu o mulțime de antrenament.
- Probabilitatea ca rețeaua neuronală antrenată să genereze, în medie, răspunsuri corecte pentru date de intrare alese aleator din spațiul datelor de intrare.
- Probabilitatea ca rețeaua neuronală antrenată să genereze, în medie, răspunsuri incorecte pentru date de intrare alese aleator din spațiul datelor de intrare.

Răspunsul la Problema învățării poate fi obținut dacă privim această problemă prin prisma teoriei aproximării, adică studiem învățarea unei rețele neuronale ca o ***problemă de aproximare***: s*ă găsim funcția* $F(\mathbf{x},\mathbf{w})$ *care aproximează cel mai bine funcția dorită* $f(\mathbf{x})$ [1] .

Să notăm cu *d* eroarea dintre vectorul țintă **z,** ce se doreşte a se obține pentru vectorul de intrare **x,** şi răspunsul generat de rețeaua neuronală, exprimat prin funcția aproximantă $F(\mathbf{x},\mathbf{w})$. Definită această eroare cu ajutorul distanței Euclidiene:

$$d\big(\mathbf{z};F(\mathbf{x},\mathbf{w})\big)=\big\|\mathbf{z}-F(\mathbf{x},\mathbf{w})\big\|^2 \qquad (13)$$

Vom defini funcționala ***risc*** [2] ca fiind media erorii definite mai sus:

$$R(\mathbf{w})=\int d\big(\mathbf{z};F(\mathbf{x},\mathbf{w})\big)dP(\mathbf{x},\mathbf{z}) \qquad (14)$$

integrala de mai sus este considerată în sens Riemann-Stieljes, iar $P(\mathbf{x},\mathbf{z})$ reprezintă distribuția probabilistică a vectorului de intrare **x** şi a vectorului țintă **z**.

În noua formulare, ***Problema învățării*** devine ***Problema minimizării***:

Problema minimizării: *Să se minimizeze funcționala risc* (14) *în raport cu clasa de funcții aproximante* $F(\mathbf{x},\mathbf{w})$*, când* $\mathbf{w}\in\mathbf{W}$.

Problema minimizării este complicată datorită faptului că distribuția probabilistică $P(\mathbf{x},\mathbf{z})$ este necunoscută, după cum se vede şi din relația de mai jos:

BDD-A23844 © 2005 Editura Universității „Petru Maior”
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 00:56:15 UTC)

$$P(\mathbf{x},\mathbf{z}) = P(\mathbf{z},\mathbf{x})P(\mathbf{x}) \quad (15)$$

Singura informație disponibilă este cea conținută în mulțimea de antrenament $T = \{(\mathbf{x}_i, \mathbf{z}_i) | i = 1,2,\ldots,N\}$. De aceea vom face apel la ***principiul inductiv al minimizării riscului empiric*** dezvoltat de Vapnik [4].

Ideea fundamentală a ***principiul inductiv al minimizării riscului empiric*** este de a utiliza un set independent de date de antrenament $T = \{(\mathbf{x}_i, \mathbf{z}_i) | i = 1,2,\ldots,N\}$ pentru funcția aproximantă $F(\mathbf{x},\mathbf{w})$, cu scopul de a defini ***funcționala risc empiric***:

$$R_{emp}(\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N} d(\mathbf{z}_i; F(\mathbf{x}_i, \mathbf{w})) \quad (16)$$

Funcționala risc empiric nu mai este în acest moment dependentă de distribuția probabilistică $P(\mathbf{x},\mathbf{z})$. Teoretic, la fel ca şi funcționala risc $R(\mathbf{w})$ (14), funcționala risc empiric $R_{emp}(\mathbf{w})$ (16) poate fi minimizată în raport cu parametrul $\mathbf{w}$, care corespunde tăriilor sinaptice ale rețelei neuronale.

Să facem notațiile:

- $\mathbf{w}_{emp}$ vectorul tăriilor sinaptice care minimizează funcționala risc empiric $R_{emp}(\mathbf{w})$;
- $F(\mathbf{x},\mathbf{w}_{emp})$ funcția aproximantă (rețeaua neuronală) corespunzătoare lui $\mathbf{w}_{emp}$.
- $\mathbf{w}_0$ vectorul tăriilor sinaptice care minimizează funcționala risc $R(\mathbf{w})$;
- $F(\mathbf{x},\mathbf{w}_0)$ funcția aproximantă (rețeaua neuronală) corespunzătoare lui $\mathbf{w}_0$.

Problema învățării, respectiv a minimizării devine în această abordare:

*în ce condiții funcția aproximantă $F(\mathbf{x},\mathbf{w}_{emp})$ este "**suficient de aproape**" de aproximanta dorită $F(\mathbf{x},\mathbf{w}_0)$? Condiția de apropiere va fi măsurată prin diferența dintre riscul empiric $R_{emp}(\mathbf{w})$ şi riscul $R(\mathbf{w})$.*

Pentru orice valoare fixată w* a vectorului tăriilor sinaptice, funcționala risc $R(\mathbf{w}^*)$ determină media următoarei variabile aleatoare:

$$A_{\mathbf{w}^*} = d(\mathbf{z}; F(\mathbf{x}, \mathbf{w}^*)) \quad (17)$$

Pe de altă parte, funcționala risc empiric $R_{emp}(\mathbf{w}^*)$ reprezintă media aritmetică a variabilei aleatoare $A_{\mathbf{w}^*}$. Pe baza unor elemente clasice de teoria probabilităților, dacă dimensionalitatea mulțimii de antrenament $T = \{(\mathbf{x}_i, \mathbf{z}_i) | i = 1,2,\ldots,N\}$ tinde la infinit, atunci media aritmetică a variabilei aleatoare $A_{\mathbf{w}^*}$ va converge către media sa. Această remarcă ne dă dreptul, din punct de vedere teoretic, să utilizăm în locul funcționala risc $R(\mathbf{w})$, funcționala risc empiric $R_{emp}(\mathbf{w})$.

Dar nu trebuie să ne aşteptăm ca vectorul tăriilor sinaptice ce minimizează funcționala risc empiric $R_{emp}(\mathbf{w})$ să minimizeze de asemenea şi funcționala risc $R(\mathbf{w})$.

Pentru aceasta vom aplica principiul minimizării riscului empiric, formulat astfel:

- în locul funcționala risc $R(\mathbf{w})$ vom construi funcționala risc empiric $R_{emp}(\mathbf{w})$ conform formulei (16), utilizând mulțimea dată de antrenament $T = \{(\mathbf{x}_i, \mathbf{z}_i) | i = 1,2,\ldots,N\}$;
- fie $\mathbf{w}_{emp}$ vectorul tăriilor sinaptice care minimizează funcționala risc $R(\mathbf{w})$ relativ la spațiul tăriilor sinaptice $\mathbf{W}$. Dacă dimensionalitatea N a mulțimii de antrenament tinde la infinit şi dacă funcționala risc empiric $R_{emp}(\mathbf{w})$ va converge uniform către funcționala risc $R(\mathbf{w})$, atunci funcționala risc empiric $R_{emp}(\mathbf{w})$ va converge în probabilitate către cea mai mică valoare posibilă a funcționalei risc $R(\mathbf{w})$, $\mathbf{w} \in \mathbf{W}$. Uniform convergența se defineşte astfel:

$$\text{Prob}\left\{\sup_{\mathbf{w}\in\mathbf{W}} |R(\mathbf{w}) - R_{emp}(\mathbf{w})| > \varepsilon\right\} \to 0, dacă\ N \to \infty \quad (18)$$

BDD-A23844 © 2005 Editura Universității „Petru Maior”
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 00:56:15 UTC)

ultima relație reprezintă condiția necesară şi suficientă pentru valabilitatea principiul minimizării riscului empiric.

Capacitatea de generalizare.

Elemente introductive referitoare la capacitatea de generalizare a rețelelor neuronale prezentate în acest capitol, sugerează posibilitatea de a cuantifica estimativ capacitatea rețelelor neuronale de a generaliza, în funcție de arhitectura sa și de dimensiunea mulțimii de antrenament. Pentru aceasta, vom selecta din numeroasele posibilități de cuantificare a generalizării, următoarele [3]:

- Numărul mediu de posibilități de generalizare în raport cu o mulțime de antrenament.
- Probabilitatea ca rețeaua neuronală antrenată să genereze, în medie, răspunsuri corecte pentru date de intrare alese aleator din spațiul datelor de intrare.
- Probabilitatea ca rețeaua neuronală antrenată să genereze, în medie, răspunsuri incorecte pentru date de intrare alese aleator din spațiul datelor de intrare.

Vom urma o idee prezentată în [4], pentru a studia prin prisma acestor elemente, capacitatea de generalizare a unei rețele neuronale .

Fie o mulțime de rețele neuronale cu o arhitectură dată fixată, specificată prin numărul de straturi, numărul de neuroni din fiecare strat, conexiuni sinaptice, funcții de activare. Fiecărei rețele neuronale îi corespunde o mulțime de tării sinaptice, pe care o vom nota **w**. O mulțime de tării sinaptice **w** poate fi interpretată ca un punct în spațiul tăriilor sinaptice posibile, spațiu pe care-l vom numi tot ***spațiul tăriilor sinaptice*** **W.**

Când vom considera media în raport cu mulțimea rețelelor neuronale, ea va reprezenta media în raport cu spațiul tăriilor sinaptice., medie calculată în raport cu o densitate probabilistică a priori, notată $\rho(\mathbf{w})$.

Putem defini ***volumul disponibil*** V_0 al spațiului tăriilor sinaptice:

$$V_0 = \int d\mathbf{w}\rho(\mathbf{w}) \tag{19}$$

Orice punct **w** din spațiul tăriilor sinaptice, reprezintă o rețea neuronală ce implementează funcția $F(\mathbf{x},\mathbf{w})$, funcție corespunzătoare valorilor generate de neuronii din stratul de ieşire, când la stratul de intrare se prezintă vectorul de intrare **x**. Astfel spațiul tăriilor sinaptice este partiționat într-o mulțime de submulțimi disjuncte, câte una pentru fiecare funcție $f(\mathbf{x})$, pe care mulțimea de rețele neuronale o poate implementa.

Volumul subspațiului care implementează o funcție particulară f, este:

$$V_0(f) = \int d\mathbf{w}\rho(\mathbf{w}) \cdot \theta_f(\mathbf{w}) \tag{20}$$

unde:

$$\theta_f(\mathbf{w}) = \begin{cases} 1, F(\mathbf{x},\mathbf{w}) = f(\mathbf{x}), & (\forall)\ \mathbf{x} \in \mathbf{X} \\ 0, & altfel \end{cases} \tag{21}$$

Fracția din spațiul tăriilor sinaptice care implementează o funcție dată f, sau probabilitatea de a obține funcția f, când alegem tării sinaptice aleatoare, conform distribuției $\rho(\mathbf{w})$ este:

$$R_0(f) = \frac{V_0(f)}{V_0} \tag{22}$$

Însumând în raport cu mulțimea tuturor funcțiilor, putem defini entropia informațională:

BDD-A23844 © 2005 Editura Universității „Petru Maior”
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 00:56:15 UTC)

$$S_0 = -\sum_f R_0(f) \cdot log_2\ R_0(f) \tag{23}$$

S_0 reprezintă diversitatea funcțională a arhitecturii rețelelor neuronale. Dacă S_0 are o valoare mare, avem nevoie de mai multă informație pentru a specifica o funcție particulară. În cazul în care avem K funcții posibile, de volum egal $V_0(f)$, obținem:

$$V_0(f) = \begin{cases} \dfrac{1}{K}, & \text{dacă } f \in \text{ celor K funcții de volum egal} \\ 0, & \text{altfel} \end{cases} \tag{24}$$

Atunci obținem: $S_0 = log_2\ K$ sau $2^{S_0} = K$ (25)

Să considerăm o paradigmă de învățare supervizată, în care se prezintă perechi de date $(\mathbf{x}_i, \mathbf{z}_i)$, ce corespund unei aplicații țintă:

$$\mathbf{z}_i = \overline{f}(\mathbf{x}_i), i = 1, \ldots, N \tag{26}$$

Presupunând că rețeaua neuronală a învățat cu succes (funcția eroare converge către zero), punctul **w** ce corespunde acestei rețele neuronală, va fi localizat într-un subspațiu al tăriilor sinaptice ce este compatibil cu datele de antrenament $(\mathbf{x}_i, \mathbf{z}_i)$. Presupunând că mulțimea de antrenament conține N perechi de date $(\mathbf{x}_i, \mathbf{z}_i)$, atunci volumul subspațiului rămas este:

$$V_N = \int d\mathbf{w} \rho(\mathbf{w}) \prod_{i=1}^{N} I(F, \mathbf{x}_i) \tag{27}$$

unde:
$$I(F, \mathbf{x}_i) = \begin{cases} 1, & F(\mathbf{x}_i, \mathbf{w}) = \overline{f}(\mathbf{x}_i) \\ 0, & altfel \end{cases} \tag{28}$$

V_N va conține subspațiul corespunzător funcției țintă $\overline{f}$, împreună cu alte subspații corespunzătoare altor funcții ce coincid cu $\overline{f}$ pe mulțimea datelor de antrenament. Evident, cu cât N este mai mare, mulțimea funcțiilor ce coincid cu $\overline{f}$ pe mulțimea datelor de antrenament este mai mică. De aici rezultă că procesul de învățare poate fi privit ca un proces de reducere continuă a spațiului admisibil al tăriilor sinaptice, adică:

$$V_0 \geq V_1 \geq V_2 \geq \ldots \geq V_N \tag{29}$$

Partea din spațiul tăriilor sinaptice ce corespunde unei funcții particulare f, se modifică după învățarea a N exemple, de la $R_0(f)$ la:

$$R_N(f) = \frac{V_N(f)}{V_N} \tag{30}$$

$V_N(f)$ reprezintă volumul spațiului tăriilor sinaptice consistent atât cu funcția f cât şi cu exemplele de învățat $(\mathbf{x}_i, \mathbf{z}_i)$. Avem:

$$V_N(f) = \int d\mathbf{w} \rho(\mathbf{w}) \theta_f(\mathbf{w}) \prod_{i=1}^{\mathbf{P}} I(F, \mathbf{x}_i) = V_0(f) \prod_{i=1}^{\mathbf{P}} I(F, \mathbf{x}_i) \tag{31}$$

Entropia corespunzătoare este:

$$S_N = -\sum_f R_P(f) \cdot \log_2\ R_N(f) \tag{32}$$

S_N reprezintă o măsură a numărului de funcții implementabile, ce sunt compatibile cu mulțimea de antrenament.

S_N - S_{N-1} reprezintă cantitatea de informație obținută prin învățarea datei $\mathbf{x}_N$. Dacă învățarea s-a desfăşurat cu succes, obținem:

$$S_N = S_0 - N \tag{33}$$

În acest fel putem să ne gândim la o limită a numărului necesar de date de antrenament pentru a învăța o aplicație particulară $\overline{f}$ sau putem să ne gândim la estimarea eficienței învățării [2].

BDD-A23844 © 2005 Editura Universității „Petru Maior”
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 00:56:15 UTC)

Să presupunem că avem o mulțime de antrenament $\mathbf{x}_1$, $\mathbf{x}_2$,..., $\mathbf{x}_N$ aleasă aleator cu ajutorul unei distribuții $P(\mathbf{x})$, fiecare $\mathbf{x}_i$.,$i=1,...,N$ fiind independent.

Atunci, fiecare factor $I(f,\mathbf{x}_i)$ este independent de ceilalți, ceea ce ne permite să considerăm o medie în raport cu mulțimea tuturor datelor de antrenament. Vom folosi pentru această medie notația **<…>**, obținând:

$$\langle V_N(f)\rangle = V_0(f)\cdot\left\langle\prod_{\mu=1}^{N} I(f,\mathbf{x}_i)\right\rangle = V_0(f)\cdot g^N(f) \tag{34}$$

Media este relativă la $\mathbf{x}_1$, $\mathbf{x}_2$,..., $\mathbf{x}_N$, cu tăriile sinaptice corespunzătoare $P(\mathbf{x}_i)$, şi avem:

$$g(f) = \langle I(f,\mathbf{x})\rangle = \text{Prob}(f(\mathbf{x}) = \overline{f}(\mathbf{x})) \tag{35}$$

reprezentând:

- probabilitatea ca o funcție particulară f să fie egală cu funcția țintă $\overline{f}$ în punctul $\mathbf{x}$, punct ales aleator de distribuția $P(\mathbf{x})$.
- $g(f)$ se numeşte ***abilitatea de generalizare*** a lui f, specificând de fapt cât de mult f se apropie de $\overline{f}$. $g(f) \in [0,1]$ fiind independentă de mulțimea de antrenament.

Să notăm cu $P_N(f)$ probabilitatea ca o funcție f să implementeze, după învățarea a N exemple de antrenament, funcția țintă $\overline{f}$. Atunci:

$$P_N(f) = \left\langle\frac{V_N(f)}{V_N}\right\rangle \approx \frac{\langle V_P(f)\rangle}{\langle V_P\rangle} \tag{36}$$

Aproximarea de mai sus se bazează pe ipoteză că V_N nu variază mult în raport cu o mulțime de antrenament, deci $V_N \approx \langle V_N\rangle$ pentru orice mulțime de antrenament.

Cu ajutorul formulei (36) putem calcula distribuția abilității de generalizare în raport cu toate funcțiile posibile f:

$$\begin{aligned}\rho_N(g) &\equiv \sum_f P_N(f)\cdot\delta(g-g(f)) \propto \sum_f \langle V_N(f)\rangle\cdot\delta(g-g(f)) = \\ &= g^N\sum_f V_0(f)\cdot\delta(g-g(f)) \propto g^N\rho_0(g)\end{aligned} \tag{37}$$

Prin normalizare obținem:

$$\rho_N(g) = \frac{g^N\cdot\rho_0(g)}{\int (g^*)^N\rho_0(g^*)\cdot dg^*} \tag{38}$$

Deoarece distribuția inițială $\rho_0(g) = V_0^{-1}\sum_f V_0(f)\cdot\delta(g-g(f))$ depinde doar de arhitectura rețelei neuronale şi de restricția *a priori* încorporată în $\rho(\mathbf{w})$, rezultă din (2.62) următorul rezultat remarcabil:

putem calcula distribuția $\rho_P(\mathbf{w})$ după N exemple de antrenament, dacă cunoaştem distribuția abilității de generalizare, înainte de faza de învățare.

Putem să considerăm şi valoarea medie a abilității de generalizare:

$$G(N) = \int_0^1 g\cdot\rho_N(g)dg = \frac{\int_0^1 g^{N+1}\rho_0(g)dg}{\int_0^1 g^N\rho_0(g)dg} \tag{39}$$

Reprezentând grafic $G(N)$ în raport cu N-numărul de date de antrenament, obținem ***curba de învățare***. $G(N)$ poate fi folosit pentru a determina N în scopul învățării rețelei neuronale la un nivel corespunzător de performanță.

Comportamentul asimptotic a lui $\rho_N(g)$ şi deci şi a lui $G(N)$ când $N\to\infty$, este determinat de forma distribuției inițiale $\rho_0(g)$ în jurul punctului $g = 1$. Avem două posibilități:

BDD-A23844 © 2005 Editura Universității „Petru Maior”
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 00:56:15 UTC)

(a). Există o tranziție abruptă de lungime ε între $g = 1$ şi următoarea valoare $g = g_0$, pentru care $\rho_0(g_0)$. Atunci avem:

$$1 - G(N) \propto e^{-\frac{N}{\varepsilon}} \tag{40}$$

(b). Dacă nu există tranziții abrupte la $\rho_0(g)$, atunci avem:

$$1 - G(N) \propto \frac{1}{N} \tag{41}$$

Aceste rezultate deosebite prezentate în acest paragraf au o mare importanță teoretică:

> *putem calcula media probabilistică a abilității de generalizare corectă, când rețeaua neuronală a fost antrenat utilizând o mulțime de antrenament cu N elemente, dacă cunoaştem în principiu o funcție ce poate fi calculată înainte de începerea fazei de antrenare.*

Practic însă e dificil să exploatăm aceste rezultate, deoarece un calcul analitic al distribuției a priori $\rho_0(g)$ este posibil doar pentru probleme simple.

De asemenea, utilizarea abilității de generalizare medie, în raport cu subspațiile spațiului tăriilor sinaptice, consistente cu mulțimea de antrenament, nu este foarte potrivită, deoarece în practică legea de învățare poate favoriza unele subspații în raport cu altele. În fond, o procedură de învățare reprezintă un drum în spațiul tăriilor sinaptice, drum ce reprezintă ajustarea graduală a tăriilor sinaptice cu scopul minimizării funcției eroare şi nu o alegere aleatoare a tăriilor sinaptice restricționate de mulțimea de antrenament. Densitatea probabilistică inițială $\rho(\mathbf{w})$ încorporează într-un fel acest efect, dar nu în totalitate. De aceea vom încerca să studiem abilitatea de generalizare în cel mai rău caz şi nu în cel mediu.

Pentru a simplifica analiza noastră, vom considera problema clasificării binare, care corespunde unei rețele neuronale ce are în stratul de ieşire un singur neuron, cu funcția de activare $sgn(\mathbf{x})$.

Ne interesează $g(f)$ pentru funcția f pe care o implementează rețeaua neuronală, pentru a şti cât de bine aproximează funcția f, funcția țintă $\overline{f}$.

Să considerăm o mulțime de antrenament, constituită din P perechi de puncte $(\mathbf{x}_i, \mathbf{z}_i)$, $i = 1,\ldots,N$, cu $\mathbf{z}_i = \overline{f}(\mathbf{x}_i), i = 1,\ldots,N$.

Fie $g_N(F)$ numărul de mulțimi de antrenament, de dimensionalitate N, corect clasificate de funcția $F(\cdot,\mathbf{w})$, implementată de rețeaua neuronală. Scopul legii de învățare este de a ajusta tăriile sinaptice astfel încât să maximizăm $g_N(F)$, adică $g_N(F) = 1$, în condițiile unei învățări perfecte.

Diferența dintre $g(f)$ şi $g_N(f)$ este datorată faptului că $g(f)$ reprezintă cât de bine aproximează funcția f funcția țintă $\overline{f}$, în timp ce $g_N(f)$ reprezintă cât de bine aproximează funcția f funcția țintă $\overline{f}$, ca o medie relativă la o mulțime de antrenament cu N elemente.

Cu alte cuvinte $g_N(f)$ reprezintă o aproximantă a lui $g(f)$, în condiții ideale:

$$g_N(f) \to g(f), N \to \infty \tag{42}$$

In practică însă, avem relația:

$$g_N(F) > g(f) \tag{43}$$

pentru funcția $F(\cdot,\mathbf{w})$ obținută ca urmare a procesului de învățare.

Dacă însă vom considera o funcție arbitrară f din mulțimea funcțiilor pe care rețeaua neuronală le poate implementa şi o funcție $F(\cdot,\mathbf{w})$ asociată mulțimii de antrenament, vom fi în stare să estimăm cât de "proastă" poată fi aproximarea funcției țintă $\overline{f}$ de către f, în cel mai rău caz. Cum acest "***cel mai rău caz***" este aplicabil oricărei funcții f implementabile de rețeaua neuronală, obținem rezultatul:

BDD-A23844 © 2005 Editura Universității „Petru Maior"
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 00:56:15 UTC)

$$\text{Prob}(\max|g_N(f) - g(f)| > \varepsilon) \le 4 \cdot m(2N) \cdot e^{-\frac{\varepsilon^2 \cdot N}{8}} \quad (44)$$

unde $m(N)$ este o funcție ce depinde de dimensionalitatea N a mulțimii de antrenament, fiind numită ***funcție de creştere*** şi reprezintă numărul maxim de funcții diferite (binare în cazul nostru) care pot fi implementate de rețeaua neuronală pe baza unei mulțimi de antrenament cu N elemente.

Foarte importanta relație (44) a fost obținută de Vapnik şi Chervonenkis [4]. Membrul stâng al relației de mai sus reprezintă probabilitatea ca cea mai slabă aproximare să depăşească o limită ε, pentru orice funcție implementabilă de către rețeaua neuronală.

Dacă de exemplu $\varepsilon = 0.01$, vom şti cu probabilitatea de 99% că $g_N(f)$ şi $g(f)$ sunt la distanța de cel mult ε una de alta, pentru orice funcție f implementabilă de rețeaua neuronală.

Dacă procesul de învățare s-a desfăşurat cu succes, obținând un rezultat perfect, adică $g_N(F) = 1$, atunci vom şti cu o probabilitate foarte mare că:

$$g(f_w) > 1 - \varepsilon \quad (45)$$

Dacă funcția de activare este funcția $sgn(\mathbf{x})$ sau funcția treaptă, avem un număr total de 2^N funcții binare diferite, deci, în general:

$$m(N) \le 2^N \quad (46)$$

Limitările funcției de creştere pot fi generate şi de arhitectura rețelei neuronale. De exemplu dacă tăriile sinaptice pot lua valori doar într-o mulțime de valori cu k valori distincte, atunci:

$$m(N) \le k^{|w|} \quad (47)$$

unde $|\mathbf{w}|$ reprezintă numărul total de conexiuni sinaptice ale rețelei neuronale.

Bibliografie:

[1] Enăchescu, C., (1995) *Learning the Neural Networks from the Approximation Theory Perspective. Intelligent Computer Communication ICC'95 Proceedings,* 184-187, Technical University of Cluj-Napoca, Romania.

[2]. Enăchescu, C., Elemente de inteligență artificială. Calcul neuronal, Editura Universitatea Tehnică Cluj-Napoca, 1997, 175 pag., 1997.

[3]. Enăchescu, C., Fundamentelele rețelelor neuronale, Editura Casa Cărții de Ştiință, ISBN 973-9204-81-8, 200 pag., Cluj-Napoca, 1998.

[4] Haykin, S. (1994), Neural Networks. A Comprehensive Foundation. *IEEE Press*, MacMillian,.

BDD-A23844 © 2005 Editura Universității „Petru Maior”
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 00:56:15 UTC)