

Identificarea sistemelor folosind rețele neuronale

November 30, 2019

1 Aspecte teoretice vizate

În lucrarea de față este prezentat un mecanism de identificare al sistemelor dinamice folosind rețele neuronale. Se va utiliza toolbox-ul *Deep Learning* din MATLAB.

2 Utilizarea rețelelor neuronale pentru identificare

2.1 Noțiuni introductive

Elementul de bază al unei rețele neuronale este *neuronul*. Structura unui neuron artificial este prezentată în figura 1. Elementele unui neuron artificial sunt:

- ponderile ce se aplică intrărilor x_1, x_2, \dots, x_n : w_k ;
- ponderea intrării de prag (engl. *threshold*, *bias*): b ;
- funcția de activare: φ .

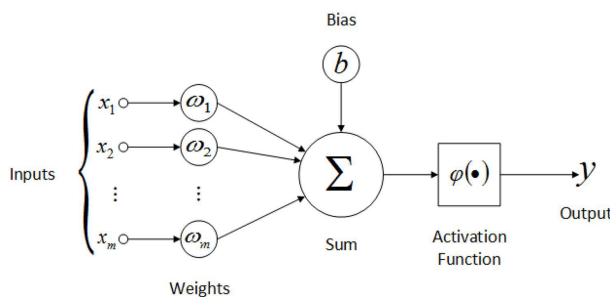


Figure 1: Structura unui neuron artificial

Valoarea semnalului de ieșire al unui neuron se calculează astfel:

$$y_k = \varphi \left(\sum_{k=1}^n w_k x_k + b \right). \quad (1)$$

Cele mai des utilizate funcții de activare sunt: funcția liniară ($\varphi_1(x) = x$), funcția sigmoidă ($\varphi_2(x)$) și funcția tangentă hiperbolică ($\varphi_3(x)$):

$$\varphi_2(x) = \frac{1}{1 + e^{-a \cdot x}}, \quad \varphi_3(x) = \frac{1 - e^{-a \cdot x}}{1 + e^{-a \cdot x}}. \quad (2)$$

O rețea neuronală este formată din mai mulți neuroni dispuși sub formă de straturi (engl. *layers*). Stratul de intrare (engl. *input layer*) nu este format din neuroni, ci din intrările aplicate rețelei. Stratul de ieșire

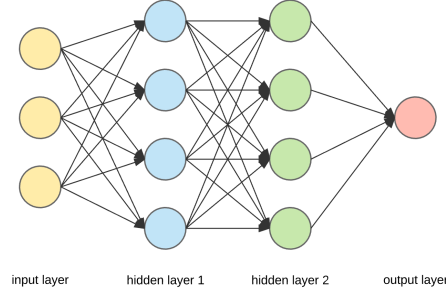


Figure 2: Exemplu de rețea neuronală

(engl. *output layer*) și straturile ascunse (engl. *hidden layers*) conțin unul sau mai mulți neuroni (straturile ascunse pot lipsi). În general, rețele neuronale sunt *total conectate*, adică fiecare neuron din fiecare strat este conectat cu fiecare neuron din stratul anterior.

Pentru ponderile fiecărui neuron avem doi indecși: unul care indică numărul stratului (k) și unul care indică poziția neuronului în stratul respectiv (j): w_{kj} . Scopul este de a găsi ponderile care minimizează indicele de performanță:

$$J = \sum_{i=1}^N \varepsilon_i^2 = \sum_{i=1}^N (y_i - \hat{y}_i)^2, \quad (3)$$

unde y_i este valoarea ieșirii pentru al i -lea set de intrări, iar \hat{y}_i este valoarea prezisă a ieșirii pentru același set de intrări. Aceste ponderi se găsesc în urma procesului de antrenare (engl. *training*) al rețelei.

2.2 Utilizarea în scopul indentificării unui sistem liniar

În general, comportamentul dinamic al unui sistem liniar și invariant în timp se poate caracteriza prin:

$$y[n] = -a_1 y[n-1] - a_2 y[n-2] - \dots - a_{N_A} y[n-N_A] + b_1 u[n-n_d] + b_2 u[n-n_d-1] + \dots + b_{N_B} u[n-N_B-n_d]. \quad (4)$$

Scopul este de a determina coeficienții a_i și b_i . Rețele neuronale total conectate nu au memorie dinamică. Pentru a rezolva această problemă, se introduc $N_A + N_B$ blocuri de întârziere cu un tact (z^{-1}), iar rețeaua va avea N_B intrări provenite din intrare ($u[n-n_d], \dots, u[n-n_d-N_B]$) și N_A intrări provenite din reacție ($y[n-1], \dots, y[n-N_A]$).

Pentru identificare se folosește metoda ARX neliniară (NARX) cu rețele neuronale – **narxnet** (engl. *nonlinear autoregressive neural network with external input*). Structura acestei rețele este prezentată în figura 3. Rețeaua neuronală este formată dintr-un strat ascuns care conține între 7 și 15 neuroni (constatare de bună practică) și un strat de ieșire format dintr-un neuron. Pentru a identifica sisteme liniare, atât neuronul de ieșire, cât și neuronii din stratul ascuns au funcția de activare liniară.

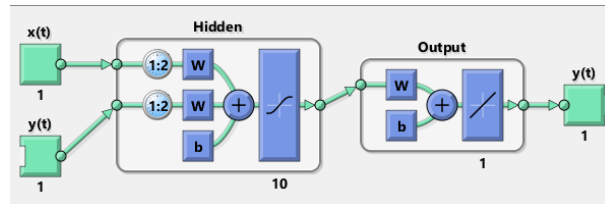


Figure 3: Structura NARXNET pentru identificare

Semnalele de intrare utilizate pentru identificarea unui sistem LTI asupra căruia nu există cunoștințe *a priori* sunt de tip zgomot alb (cel mai frecvent utilizat). În general, se utilizează semnale persistente, care să excite toate modurile sistemului supus identificării. Alte exemple de semnale persistente: SPAB, VCO.

Parametrii funcției **narxnet** sunt: numărul de tacti de întârziere pe intrare, numărul de tacti de întârziere pe reacție, dimensiunea stratului ascuns.

După ce parametrii modelului au fost identificați, se închide bucla folosind funcția `closeloop`. În figura 4 se prezintă structura sistemului identificat ce poate fi utilizată mai departe în aplicații. Se observă realizarea conexiunii de reacție pentru ca rețeaua să intre în modul de predicție.

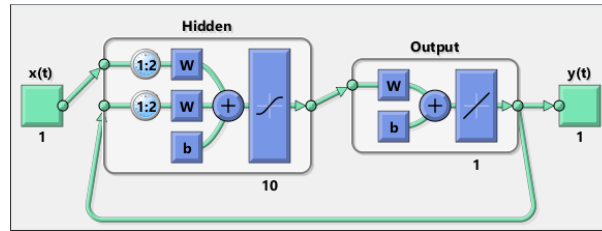


Figure 4: Structura în buclă închisă

3 Aplicații

Se dorește identificarea unor sisteme dinamice, după cum urmează:

1. Se consideră un sistem de ordin 1:
 - se realizează o simulare cu o perioadă de eșantionare care să genereze 1000 de puncte;
 - se aplică zgomot alb peste o intrare de tip treaptă;
 - se exportă datele de intrare-ieșire;
 - se identifică sistemul pe baza datelor culese.
2. Se consideră datele de la proiect:
 - importăm datele de intrare-ieșire;
 - se identifică sistemul intrare-viteză pe baza datelor primite.
3. Se consideră un sistem de ordin superior:
 - se realizează o simulare cu o perioadă de eșantionare care să genereze 1000 de puncte;
 - se aplică zgomot alb peste o intrare de tip treaptă;
 - se exportă datele de intrare-ieșire;
 - se identifică sistemul pe baza datelor culese.