

Calcul neuronal și evolutiv.

Lab 3: Rețele neuronale cu învățare nesupervizată. Aplicații în gruparea datelor, maparea datelor și optimizare combinatorială.

1. Specificul învățării nesupervizate

Setul de antrenare conține doar date de intrare, rolul algoritmului de antrenare fiind de a identifica corelații între acestea care să permită extragerea de informații privind structura lor (de exemplu identificarea de grupuri de date similare).

Arhitectura suport pentru algoritmi de antrenare supervizată este de regulă constituită dintr-un nivel de unități de intrare și un nivel de unități funcționale. Cele mai cunoscute modele de rețele cu antrenare nesupervizată sunt:

- Rețelele bazate pe procese de competiție între unitățile funcționale (de tip WTA și ART) – pot fi utilizate în rezolvarea problemelor de grupare a datelor.
- Rețelele de tip Kohonen (utilizate pentru mapare topologică).

2. Gruparea datelor.

Gruparea datelor (clustering) are ca scop identificarea într-un set de date a unor subseturi de date similare între ele. Algoritmii partiționali de grupare permit identificarea câte unui prototip pentru fiecare dintre clase iar datele sunt asignate unei clase pe criteriul distanței minime față de prototip. Procesul de grupare poate fi ilustrat utilizând o rețea cu un nivel de unități liniare antrenată cu un algoritm de tipul WTA.

- Arhitectura.* Pentru gruparea unui set de date N-dimensionale în K clustere se consideră o rețea cu N unități de intrare și K unități de ieșire. Matricea W a ponderilor va avea K linii și N coloane. Fiecare dintre liniile matricii poate fi interpretată ca un vector prototip corespunzător cluster-ului identificat de unitatea funcțională corespunzătoare.
- Funcționare.* Pentru un vector de intrare X se determină unitatea funcțională k^* (numită unitate învingătoare) caracterizată prin faptul că $d(X, W^{k^*}) < d(X, W^k)$ pentru orice $k=1..K$.
- Antrenare.* Pentru fiecare exemplu X din setul de antrenare se determină unitatea învingătoare k^* și se ajustează ponderile corespunzătoare acesteia folosind o regulă de forma: $W^{k^*} = W^{k^*} + \eta(X - W^{k^*})$.

Aplicație 1. Să se implementeze o rețea WTA pentru gruparea unor date bidimensionale.

Indicație. Vezi fișierul WTA.sci

3. Rețele Kohonen.

3.1. Particularități ale rețelelor Kohonen

Principalele particularități ale rețelelor Kohonen (în raport cu rețelele de tip WTA) sunt:

- unitățile de ieșire sunt plasate în nodurile unei grile uni, bi sau tri-dimensionale și fiecare dintre unitățile de ieșire are asociată o mulțime de unități vecini (vecinătatea este determinată de tipul de distanță utilizat și de parametrul care controlează dimensiunea vecinătății).

- pentru fiecare element din setul de antrenare se ajustează atât ponderile unității învingătoare cât și cele ale unităților vecine.

Rețelele Kohonen pot fi utilizate pentru maparea topologică a unor date multi-dimensionale într-un spațiu de dimensiune mai mică.

Aplicație 2. Implementarea unei rețele Kohonen cu grilă pătratică și maparea acesteia pe un set de date generate uniform aleator în $[0,1] \times [0,1]$.

Indicație: fișier SOM.sci

Exercițiu. Reantrenați și testați rețeaua în cazul unor date generate în interiorul unei alte figuri geometrice (cerc, coroana circulară etc).

3.2. Problema comis-voiajorului

Este una dintre cele mai cunoscute probleme de optimizare combinatorială putând fi enunțată după cum urmează: “se consideră o mulțime de n orașe și un comis-voiajor care trebuie să viziteze toate orașele, trecând o singură dată prin fiecare și să se întoarcă în orașul de pornire astfel încât costul total al circuitului să fie cât mai mic”. Din punct de vedere formal, în varianta clasică problema este echivalentă cu cea a găsirii unui circuit Hamiltonian de cost minim într-un graf complet.

Problema poate fi rezolvată exact (prin analiza tuturor circuitelor posibile) pentru valori mici ale numărului de orașe, însă cum numărul de circuite este $(n-1)!/2$, pentru valori mari ale lui n nu există metode exacte eficiente. Dealtfel, este cunoscut că TSP face parte din clasa problemelor NP-complete.

Problema este importantă atât din punct de vedere teoretic cât și din punct de vedere practic întrucât o serie de probleme concrete pot fi formulate ca TSP:

- Identificarea rutei optime pentru mijloace de transport (persoane, mărfuri)
- Generarea traseelor urmate de dispozitivele de producere a circuitelor integrate

Diferite aplicații pot fi găsite la [<http://www.tsp.gatech.edu/apps/index.html>]

Există diferite variante ale problemei:

- Varianta asimetrică: costul trecerii de la un nod la altul depinde de sensul circuitului (Assymetric TSP)
- Variante cu restricții de precedență; restricțiile specifică faptul că un anumit nod trebuie vizitat înaintea altuia (Sequential Ordering Problem - SOP)
- Variante cu mai multe mijloace de transport care asigură aprovizionarea (Capacitated vehicle routing problem - CVRP)
- Variante generalizate în care există mai multe trasee directe între două noduri sau nodurile sunt înlocuite cu clustere de noduri (Generalized TSP)

Una dintre euristicile cel mai frecvent folosite este euristica Lin-Kernighan care se bazează pe ideea de a înlocui unele arce ale traseului cu altele în scopul reducerii costului. Cazul cel mai simplu este cel în care se înlocuiesc două arce ceea ce echivalează cu schimbarea ordinii de parcurgere a unei porțiuni din traseu (transformarea 2-opt).

Exemplu: In cazul a 6 orașe: A,B,C,D,E,F dacă traseul inițial este (A,C,B,E,F,D) prin inlocuirea arcului (A,C) cu arcul (A,F), a arcului (F,D) cu arcul (C,D) și inversarea ordinii de parcurgere a orașelor B și E se obține traseul (A,F,E,B,C,D). Se observă că acest traseu se poate obține din cel inițial prin inversarea ordinii în secvența (C,B,E,F).

3.3. Rezolvarea problemei comis voiajorului folosind rețele Kohonen

Principalele caracteristici ale acestei abordări:

- Rețelele de tip Kohonen au o arhitectură constituită din două unități de intrare și m unități de ieșire plasate în nodurile unei topologii de tip inel: vecinătatea de dimensiune $2s$ a unui neuron i conține mulțimea de neuroni $\{i-s, i-(s-1), \dots, i-1, i, i+1, \dots, i+(s-1), i+s\}$. Numărul de unități funcționale trebuie să fie mai mare decât numărul de orașe ale problemei (de exemplu $m=5n$)
- Fiecare unitate funcțională are asociate două ponderi inițializate aleator cu valori din domeniul coordonatelor orașelor.
- Procesul de antrenare constă în parcurgerea de mai multe ori a listei cu coordonate ale orașelor și ajustarea ponderilor folosind regula specifică. Pentru fiecare vector de intrare x (pereche de două coordonate) se efectuează următoarele prelucrări:
 - Se determină unitatea învingătoare, i^* , (cea având vectorul de ponderi cel mai apropiat de x)
 - Se ajustează ponderile tuturor unităților folosind regula:
 - $W^i(k+1) = W^i(k) + \eta(X - W^i(k))\Lambda(i, i^*)$
unde funcția de vecinătate este $\Lambda(i, i^*) = \exp(-(i - i^*)^2 / (2s^2))$

Aplicatie 3. Să se implementeze rețeaua Kohonen descrisă mai sus.

Indicatie. Vezi fisierul Kohonen_TSP.sci

Exercitiu: Scrieți o funcție care să calculeze costul traseului generat.

Temă. Implementați o rețea de tip ART (detalii în curs 4) pentru identificarea de clustere într-un set de date bidimensionale.

Indicație. Se modifică algoritmul de antrenare din implementarea WTA.sci