

Neizrazito, evolucijsko i neuroračunarstvo

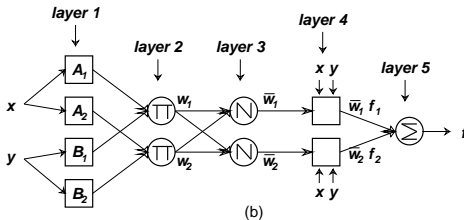
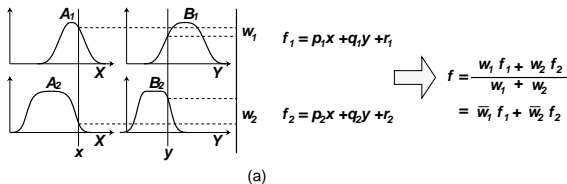
Neuro-fuzzy-evolucijski sustavi

dr.sc. Marko Čupić

Fakultet elektrotehnike i računarstva
Sveučilište u Zagrebu

23. siječnja 2014.

ANFIS uz zaključivanje tipa 3



Slika: ANFIS mreža (zaključivanje tipa 3)

Algoritam PSO za ANFIS

Sustav ANFIS za koji je razvijen algoritam treniranja opisan je sljedećim parametrima.

- I - broj ulaznih varijabli (na slici 1 $I = 2$).
- N^* - broj jezičnih izraza po jezičnoj varijabli (na slici 1 $N^* = 2$).
- N - ukupni broj jezičnih izraza $N = N^* \cdot I$ (na slici 1 $N = 4$).
- R - broj pravila kojima raspolaže sustav; pretpostavka je da vrijedi $R = N^*$; alternativa: $R = N^{*I}$.

Algoritam PSO za ANFIS

- neka su funkcije pripadnosti (antecedent dio pravila):

$$\mu_{A_i}(x) = e^{-\left[\left(\frac{x-c_i}{a_i}\right)^2\right]^{b_i}}$$

- neka je konsekvens oblika:

$$f_r(x_1, x_2, \dots, x_l) = \xi_{r,1}x_1 + \xi_{r,2}x_2 + \dots + \xi_{r,l}x_l + \xi_{r,l+1}$$

- sustav sadrži ukupno N parova parametara (a_i, b_i, c_i) te $R \cdot (l + 1)$ parametara u konsekvens dijelu koje treba naučiti

Algoritam PSO za ANFIS

- za učenje ćemo koristiti algoritam PSO
- čestica ima finiju strukturu:
 - svi parametri iste vrste su u zasebnoj kolekciji
 - čestica je kolekcija takvih kolekcija
- kolekcija ukupno ima:
 - jedna za sve a_i
 - jedna za sve b_i
 - jedna za sve c_i
 - jedna za sve $\xi_{r,1}$
 - ...
 - jedna za sve $\xi_{r,l+1}$
 -
 - total: $3 + (l + 1)$, svaki s po N elemenata

Koraci algoritma

- 1 inicijaliziraj sve čestice u roju $P_i \in P(t)$ na slučajno odabrane pozicije $\vec{x}_i(t)$
- 2 izračunaj dobrotu $F(\vec{x}_i(t))$ svih čestica koristeći njihovu trenutnu poziciju $\vec{x}_i(t)$
- 3 po potrebi ažuriraj $pbest$ svake čestice:
 $pbest_i = F(\vec{x}_i(t))$, $\vec{x}_{pbest_i} = \vec{x}_i(t)$
- 4 po potrebi ažuriraj $gbest$: $gbest = F(\vec{x}_i(t))$, $\vec{x}_{gbest} = \vec{x}_i(t)$
- 5 izračunaj brzinu svake čestice:

$$\vec{v}_i(t) = \vec{v}_i(t-1) + \vec{r}_1 \cdot C_1 \cdot (\vec{x}_{pbest_i} - \vec{x}_i(t)) + \vec{r}_2 \cdot C_2 \cdot (\vec{x}_{gbest} - \vec{x}_i(t))$$

- 6 Izračunaj nove pozicije svake čestice:

$$\vec{x}_i(t) = \vec{x}_i(t-1) + \vec{v}_i(t), \quad t = t + 1.$$

- 7 ponavlja postupak od koraka 2 do konvergencije

Algoritam PSO za ANFIS

- PSO se kombinira s GA
 - 1 nakon svake iteracije PSO-a odabere se čestica s najgorim $pbest_i$ koja će biti eliminirana i nadomještena novom
 - 2 iz populacije čestica se biraju dvije čestice i operatorom križanja nastaju dva djeteta
 - 3 lošije dijete se elimira a bolje nadomješta najgoru česticu eliminiranu u koraku 1

U izvornom radu algoritam se je pokazao boljim od klasičnih metoda temeljenih na gradijentnom spustu.

SAPSO za ANFIS

SAPSO: Simbiotski Adaptivni PSO

- Jedna čestica je kompletno rješenje, ali čestica kao takva u algoritmu ne postoji!
- Umjesto toga, algoritam radi s R podrojeva.
- Svaki podroj sastoji se od P podčestica.
- Podčestica odgovara jednom pravilu. Svaki podroj zadužen je razvoj optimalnog pravila.

SAPSO za ANFIS

Pretpostavke algoritma:

- Neizraziti skupovi koriste funkcije pripadnosti oblika:

$$\mu_{A_{ij}} = \exp \left(-\frac{(x_i - m_{ij})^2}{\sigma_{ij}^2} \right).$$

- Pravila su oblika

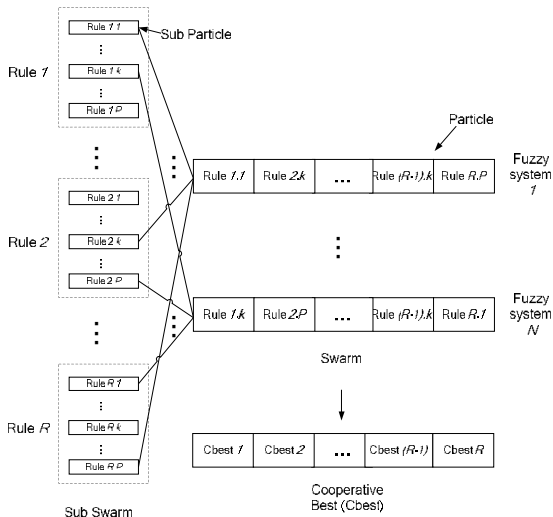
Ako x_1 je $A_{1j}(m_{1j}, \sigma_{1j})$ i ... i ako x_n je $A_{nj}(m_{nj}, \sigma_{nj})$
tada $y_j = w_{0j} + x_1 w_{1j} + \dots + x_n w_{nj}$.

- Podčestica tada ima $2n + n + 1 = 3n + 1$ parametara koje treba naučiti.

m_{1j}	σ_{1j}	m_{2j}	σ_{2j}	...	m_{nj}	σ_{nj}	...	w_{0j}	w_{1j}	...	w_{nj}
----------	---------------	----------	---------------	-----	----------	---------------	-----	----------	----------	-----	----------

Slika: Struktura podčestice koja odgovara jednom pravilu sustava ANFIS

SAPSO za ANFIS



SAPSO za ANFIS

O algoritmu:

- Algoritam treba N čestica za roj: gradi ih tako da nasumično bira po jednu podčesticu iz svakog podroja.
- Iznos funkcije dobrote može se definirati samo na razini čestice: koliko dobro sustav aproksimira zadane podatke.
- Problem: PSO treba dobrotu podčestica
 - Dobrota podčestice računa se kao prosječna dobrota svih čestica u koje je ta podčestica bila uključena.

SAPSO za ANFIS

O algoritmu:

- Dalje ažuriranje skoro pa uobičajeno: ažuriraju se podčestice u podrojevima
 - *pbest* i *gbest* su lokalni za podrojeve
 - definira se *cbest* – kooperativno najbolje rješenje (dobrota najbolje ikada pronađene izgrađene čestice)
 - prilikom ažuriranja koriste se *pbest*, *gbest* i *cbest*
- još jedna modifikacija: umjesto *pbest* koristiti *vbest* koji se određuje iz dinamičkog adaptivnog susjedstva.