

brmiversity: Umělá inteligence a teoretická informatika

Přednáška č. 11

Petr Baudiš <pasky@ucw.cz>

brmlab 2011

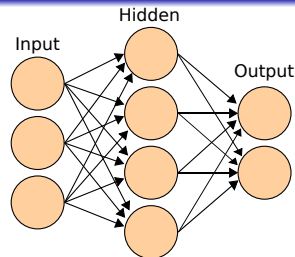


Outline

- 1 Neuronové sítě
- 2 Adaptivní agenti
- 3 Evoluční algoritmy

Samoorganizace pomocí ANN

- Umělé neurony (“výpočetní krabičky”) dostávají vstupy (čísla) a na jejich základě generují výstup (číslo)
- Dnes: “Jedna” vrstva, chceme asociovat různé vstupy s různými neurony
- Učení bez učitele — redukcí (s určitými parametry) najdi sám
- Učení s učitelem — redukuj podle předložených tříd



Kompetiční učení

- Neurony soutěží o vzory; váhový vektor \vec{w} odpovídá pozici v prostoru vzorů
- **Vítěz bere vše:** Vítěz je právě jeden nejbližší neuron
- **Adaptace:** Vítězný neuron posuneme směrem k vzoru

$$\vec{w}' = \vec{w} + \alpha \cdot (\vec{x} - \vec{w})$$

- **(Laterální inhibice:** Okolní neurony odsuneme směrem *od* vzoru)
- Připomíná vám to něco?

Kompetiční učení

- Neurony soutěží o vzory; váhový vektor \vec{w} odpovídá pozici v prostoru vzorů
- **Vítěz bere vše:** Vítěz je právě jeden nejbližší neuron
- **Adaptace:** Vítězný neuron posuneme směrem k vzoru

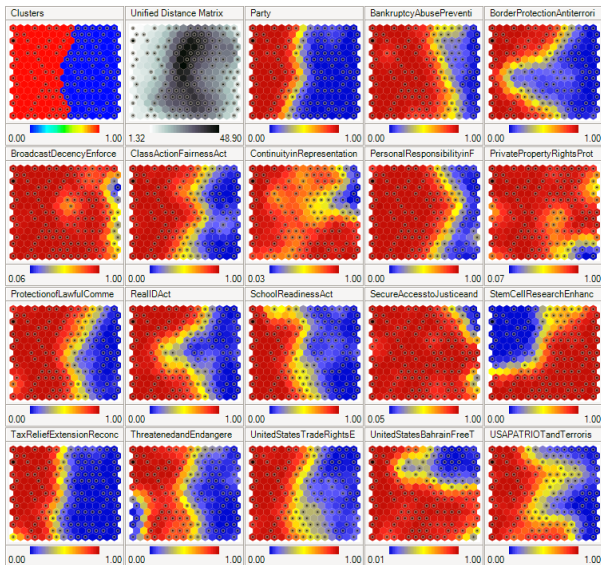
$$\vec{w}' = \vec{w} + \alpha \cdot (\vec{x} - \vec{w})$$

- (**Laterální inhibice:** Okolní neurony odsuneme směrem *od* vzoru)
- Připomíná vám to něco? *k*-means shlukování!
- Lze dokázat, že konverguje ke *k*-means a při dobře ohraničených shlucích je stabilní
- Tvar shluku je omezen lineárním klasifikátorem
- Lepší stabilita: dávková aktualizace

Kohonenova mapa

- Uspořádání výstupních neuronů je důležité (třeba 2D mřížka)
- Tedy: n -rozměrný vstupní prostor (kombinace n vstupních neuronů) redukuje se do k -rozměrného výstupního prostoru
- Spatiální koherence: Vzory blízko ve vstupním prostoru by měly zůstat blízko ve výstupním (a naopak?)
- Fonetický “psací stroj”, ekonomická data, politická data

Kohonenova mapa



Kohonenova mapa

- Váhový vektor \vec{w} neuronu odpovídá jeho pozici ve vstupním prostoru
- Vzdálenost vstupu k neuronu j : $d_j = \sum_{i=1}^k (x_i - w_{ij})^2$
- Vybavení: Vyber nejbližší neuron ($\operatorname{argmin}_j d_j$)

Kohonenova mapa

- Váhový vektor \vec{w} neuronu odpovídá jeho pozici ve vstupním prostoru
- Vzdálenost vstupu k neuronu j : $d_j = \sum_{i=1}^k (x_i - w_{ij})^2$
- Vybavení: Vyber nejbližší neuron ($\operatorname{argmin}_j d_j$)
- Neurony jsou rozmístěny v euklidovském prostoru, každý neuron má okolí dané poloměrem r
- Funkce laterální interakce $\Phi(a, b)$ — síla propojení neuronů a a b , obvykle klesá se vzdáleností
- Učení: Vyber nejbližší neuron c , neurony v okolí r posuň

$$w'_{ij} = w_{ij} + \alpha(t) \cdot \Phi(c, j) \cdot (x_i - w_{ij})$$

$\alpha(t)$ klesá s časem

- Lze dokázat konvergenci

Lateral Vector Quantization

Učení s učitelem — máme definované *třídy* $c(\vec{x})$
 \vec{x} by měl patřit ke stejné třídě jako nejbližší \vec{w}

Adaptační pravidlo LVQ1

- Adaptujeme pouze vítězný (nejbližší) neuron:
- $c(\vec{x}) = c(\vec{w}_j)$: $w'_{ij} = w_{ij} + \alpha(t) \cdot (x_i - w_{ij})$
- $c(\vec{x}) \neq c(\vec{w}_j)$: $w'_{ij} = w_{ij} - \alpha(t) \cdot (x_i - w_{ij})$

Adaptační pravidlo LVQ2.1

- Adaptujeme dva nejbližší neurony; soustředíme se na *posouvání hranic oblastí*
- Buď j, k nejbližší neurony; učíme se pouze pokud j má správnou třídu, k špatnou třídu
- Zároveň x je z “okénka”, na rozhraní i a j ve vstupním prostoru (okolí dělící nadroviny \vec{w}_i a \vec{w}_j)
- $w'_{ij} = w_{ij} + \alpha(t) \cdot (x_i - w_{ij})$
- $w'_{ik} = w_{ik} - \alpha(t) \cdot (x_i - w_{ik})$
- LVQ3: $w'_{ik} = w_{ik} + \varepsilon \cdot \alpha(t) \cdot (x_i - w_{ik})$,
je-li k také správná třída (stabilizace řešení)

Další metody samoorganizace

- Sítě se vstřícným šířením
- Adaptive Resonance Theory (ART): Laterální inhibice, zpětné hrany
- PCA pomocí posilovaného učení lineárního klasifikátoru (neuronu) — Ojův algoritmus

Otázky?

Příště: Modulární, hierarchické a hybridní modely.

Outline

- 1 Neuronové sítě
- 2 Adaptivní agenti
- 3 Evoluční algoritmy

Etologické modely

- Analytické vs. simulační modely
- Emergentní chování
- Akční patterny; apetitivní chování; konzumační chování
- Lorenzův psychohydraulický model
- V umělých bytostech Tyrrellova free-flow hierarchie (už bylo)

Non-free artwork omitted. See <http://terpconnect.umd.edu/~wrstrick/secu/ansc455/model.gif>

Populační dynamika

Rozmnožování

- Fibonacciho posloupnost $F_{i+2} = F_{i+1} + F_i$ (množení králíků)
- Logistická funkce: $dN/dt = rN(1 - N/K)$
- r je rychlost množení, K je kapacita prostředí;
 r -stratég vs. K -stratég
- Řešení $\frac{1}{1+e^{-t}}$ resp. $\frac{KP_0e^{rt}}{K+P_0(e^{rt}-1)}$

Model predátor-kořist

- Dynamický systém Lotka-Volterra
- $dx/dt = x(\alpha - \beta y)$ $dy/dt = -y(\gamma - \delta x)$
- y jsou predátoři, x je kořist
- Rovnovážné stavy nebo cykly, např. $\{y = \alpha/\beta, x = \gamma/\delta\}$
- Obecněji Kolmogorovské modely

Otázky?

Příště: Reprezentace znalostí.

Outline

- 1 Neuronové sítě
- 2 Adaptivní agenti
- 3 Evoluční algoritmy

Koevoluce

- Populace problémů a populace řešitelů
- Paralelní vývoj problémů i řešení
- Fitness řešitelů: Procento vyřešených problémů
- Fitness problému: Procento selhávajících řešitelů

- Analogie s parazity (červená královna)
- Třídící sítě
- Knihovna zahájení v Go
- Evoluční teorie her
- Umělý život

Otevřená evoluce

- Strop fitness je (skoro?) nekonečno!
- Typicky umělý život (fact-free science)
- Fitness je endogenní — obvykle přežití (rozmnožení) nebo smrt

- Core Wars (evoluce v principu neprobíhá)
- Tierra, Avida (soutěž i o zdroje jako CPU, samozměny)
- Evolve (celulární automat)
- DigiHive (částice, konzistentní fyzikální model)
- Framsticks, breve (3D model světa)
- Brmlife

- Skutečně otevřená evoluce — otevřený problém

Pravděpodobnostní model GA

- Opět trocha suché teorie, jak GA funguje
- Vzorečky: Pravděpodobnost vzniku schématu, sledujeme vývoj fitness jako dynamický systém
- Jak GA funguje? Stejně nevíme!

Otázky?

Příště: Hrst aplikací.

Děkuji vám

pasky@ucw.cz

Příště: Umělá inteligence a adaptivní agenti (reprezentace znalostí).
Neuronové sítě. Evoluční algoritmy.
Vyčísitelnost (algoritmicky nerozhodnutelné problémy).