Sztuczne sieci neuronowe

Krzysztof A. Cyran
POLITECHNIKA ŚLĄSKA
Instytut Informatyki, p. 311

Wykład 4

PLAN:

- Repetitio: backpropagation

-Inne gradientowe algorytmy uczenia:

algorytm zmiennej metryki

algorytm gradientów sprzeżonych

Backpropagation (podsumowanie)

- Obliczamy uogólnione delty dla neuronów wyjściowych

$$\delta_i^{(l)R} = \beta(d_i^{(l)} - y_i^{(l)}) y_i^{(l)} (1 - y_i^{(l)})$$

- Obliczamy uogólnione delty dla kolejnych (licząc od ostatniej do pierwszej) warstw ukrytych:

$$\delta_i^{(l)r} = \beta O_i^{(l)r} (1 - O_i^{(l)r}) \sum_{k=1}^{N_{r+1}} \delta_k^{(l)r+1} w_{ki}$$

- Dokonujemy modyfikacji wag według:

$$\Delta w_{ij}^{(l)} = \eta \delta_i^{(l)} O_j^{(l)}$$

Zalety algorytmu backpropagation

- metoda lokalna
- małe wymagania pamięciowe
- metoda uniwersalna

Wady algorytmu backpropagation

- Metoda wolnozbieżna (zwłaszcza przy płaskich funkcjach błędów – małe kroki, ale także przy zbyt stromych przy zbyt dużym współczynniku uczenia – oscylacje wokół minimum)
- Metoda znajduje minima lokalne funkcji błędu a nie globalne, których się poszukuje

Inercyjne modyfikacje Backpropagation – poprawa zbieżności

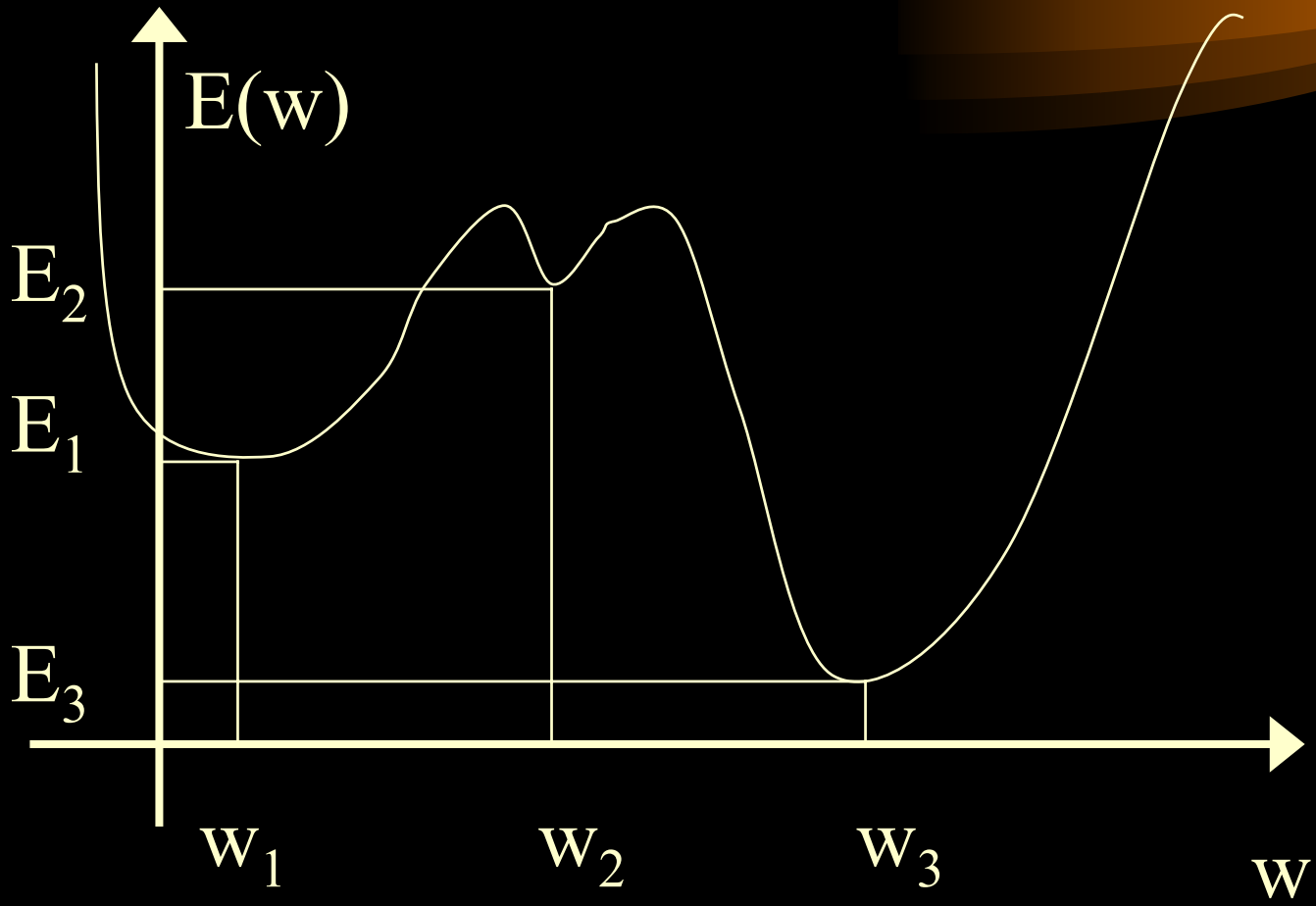
- Uczenie z momentum:

$$\Delta w_{ij}^{(l)} = \eta \delta_i^{(l)} O_j^{(l)} + \mu \Delta w_{ij}^{(l-1)}$$

- Uczenie z wykładniczym wygładzaniem:

$$\Delta w_{ij}^{(l)} = \eta((1-\mu)\delta_i^{(l)} O_j^{(l)} + \mu \Delta w_{ij}^{(l-1)})$$

Minima lokalne vs. minimum globalne (przypadek 1D)



Eliminacja wady typu: znajdowanie minimum lokalnego

- Rozpoczynanie uczenia od wielu różnych warunków początkowych (multistart) – wada: brak dobrego kryterium ustalającego ilość prób
- Stosowanie algorytmów miękkiej selekcji (symulowane wyżarzanie, algorytmy ewolucyjne) – wada: algorytmy te są nieefektywne czasowo

Ogólny wektorowy zapis algorytmu uczenia

$$\mathbf{W}(k+1) = \mathbf{W}(k) + \Delta\mathbf{W},$$
$$\Delta\mathbf{W} = \eta\,\mathbf{p}(\mathbf{W})$$

η- współczynnik uczenia

p (**W**) – kierunek poszukiwań w przestrzeni wielowymiarowej **W**

Algorytm uczenia w pseudokodzie

while (not ZnaleziononoMinimum) do

Wyznacz wektor kierunku w punkcie **W**

Minimalizuj Funkcję $E(\mathbf{W})$ na kierunku **p**

end while

Różnice w algorytmach uczenia

Różnice te polegają na różnych sposobach wyznaczania:

- kierunku poszukiwań **p**
- kroku η

Zasada działania algorytmów gradientowych

Wszystkie algorytmy gradientowe bazują na rozwinięciu funkcji celu $E(\mathbf{W})$ w szereg Taylora na kierunku $\mathbf{p}$ w najbliższym otoczeniu znanego rozwiązania $\mathbf{W}_0$

Rozwinięcie w szereg Taylora funkcji $E(\mathbf{W})$

Oznaczmy:

$$\mathbf{g}(\mathbf{W}) = \nabla E(\mathbf{W}) - \text{gradient } E(\mathbf{W}),$$

$$\mathbf{H}(\mathbf{W}) - \text{hesjan } E(\mathbf{W})$$

Wówczas:

$$E(\mathbf{W}+\mathbf{p}) = E(\mathbf{W}) + [\mathbf{g}(\mathbf{W})]^T \mathbf{p} + \frac{1}{2}\mathbf{p}^T \mathbf{H}(\mathbf{W})\mathbf{p} + \ldots$$

Punkt rozwiązania **W** jest minimum funkcji $E(\mathbf{W})$ gdy:

- $\mathbf{g}(\mathbf{W}) = \mathbf{0}$ - określa monotoniczność funkcji

Przypadek 1D:

pierwsza pochodna równa zero

- $\mathbf{H}(\mathbf{W})$ jest dodatnio określony – określa krzywiznę funkcji

Przypadek 1D:

druga pochodna dodatnia

Rodzaje metod minimalizacji gradientowej

- Metoda największego spadku (algorytm wstecznej propagacji błędów jest implementacją tej metody)
- Metoda zmiennej metryki (korzysta z informacji o krzywiźnie funkcji błędu)
- Metoda gradientów sprzężonych
- Inne

Metoda największego spadku

- Liniowe przybliżenie funkcji $E(\mathbf{W})$
- Brak informacji o krzywiźnie funkcji błędu zawartej w Hesjanie (stąd dość wolna zbieżność algorytmu backpropagation, ale i prostota metody)
- Aby zapewnić warunek $E(\mathbf{W}_{k+1}) < E(\mathbf{W}_k)$ ustala się:

$$\mathbf{p} = -\mathbf{g}(\mathbf{W}),$$

zatem:

$$\Delta\mathbf{W} = -\eta\,\mathbf{g}(\mathbf{W})$$

Efekt zastosowania inercji do metody największego spadku

- Dla płaskich odcinków funkcji błędu zachodzi:

$$\Delta \mathbf{W}^{(l)} \approx \Delta \mathbf{W}^{(l+1)} = \Delta \mathbf{W}$$

zatem wówczas dla metody z momentum:

$$\Delta \mathbf{W} = \eta \, \mathbf{p} + \mu \, \Delta \mathbf{W} \quad \Rightarrow \quad \Delta \mathbf{W} = \frac{1}{1-\mu} \eta \, \mathbf{p}$$

co oznacza 1/(1-μ) krotne efektywne przyspieszenie uczenia (np. dla μ=0.9, przyśpieszenie jest 10 krotne)

- Dla stromych odcinków inercja zapobiega nadmiernym oscylacjom

Metoda zmiennej metryki

- algorytm oparty o Newtonowską metodę optymalizacji
- wykorzystuje kwadratowe przybliżenie funkcji $E(\mathbf{W})$

$$E(\mathbf{W}+\mathbf{p}) \approx E(\mathbf{W}) + [\mathbf{g}(\mathbf{W})]^T \mathbf{p} + \frac{1}{2}\mathbf{p}^T \mathbf{H}(\mathbf{W})\mathbf{p}$$

Metoda zmiennej metryki (cd.)

- Minimum funkcji błędu wymaga by:

$$\frac{\partial E(\mathbf{W}_k + \mathbf{p}_k)}{\partial \mathbf{p}_k} = \mathbf{0}$$

stąd: $$\mathbf{g}(\mathbf{W}_k) + \mathbf{H}(\mathbf{W}_k)\mathbf{p}_k = \mathbf{0}$$

oraz ostatecznie:

$$\mathbf{p}_k = -\mathbf{H}^{-1}(\mathbf{W}_k)\mathbf{g}(\mathbf{W}_k)$$

Metoda zmiennej metryki (cd.)

- Ponieważ bezpośrednie obliczanie hesjanu w każdym kroku jest bardzo czasochłonne, dlatego korzysta się z pojęcia przybliżenia hesjanu **G**(**W**) modyfikowanego w każdym kroku o pewną poprawkę, tak aby aktualna wartość **G**(**W**) przybliżała krzywiznę funkcji *E* zgodnie z zależnością:

$$\mathbf{G}(\mathbf{W}_k)(\mathbf{W}_k - \mathbf{W}_{k-1}) = \mathbf{g}(\mathbf{W}_k) - \mathbf{g}(\mathbf{W}_{k-1})$$

Metoda zmiennej metryki (cd.)

- Jednakże zamiast obliczać $\mathbf{G}(\mathbf{W}_k)$ w metodzie tej rekurencyjnie oblicza się $\mathbf{V}_k = \mathbf{G}^{-1}(\mathbf{W}_k)$ będące przybliżeniem odwrotności hesjanu
- Zgodnie z metodą BFGS jest ono równe:

$$\mathbf{V}_k = \mathbf{V}_{k-1} + \left(1 + \frac{\mathbf{r}_k^T \mathbf{V}_{k-1} \mathbf{r}_k}{\mathbf{s}_k^T \mathbf{r}_k}\right) \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{r}_k} - \frac{\mathbf{s}_k \mathbf{r}_k^T \mathbf{V}_{k-1} + \mathbf{V}_{k-1} \mathbf{r}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{r}_k}$$

$$\mathbf{V}_0 = \mathbf{1}$$

gdzie: $\mathbf{s}_k = \mathbf{W}_k - \mathbf{W}_{k-1}, \qquad \mathbf{r}_k = \mathbf{g}(\mathbf{W}_k) - \mathbf{g}(\mathbf{W}_{k-1})$

Metoda zmiennej metryki (cd.)

W innej odmianie metody ze zmienną metryką zwanej (od skrótów nazwisk twórców algorytmu) DFP, $\mathbf{V}_k$ wyznacza się według:

$$\mathbf{V}_k = \mathbf{V}_{k-1} + \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{r}_k} - \frac{\mathbf{V}_{k-1} \mathbf{r}_k \mathbf{r}_k^T \mathbf{V}_{k-1}}{\mathbf{r}_k^T \mathbf{V}_{\mathbf{k-1}} \mathbf{r}_k}$$

$$\mathbf{V}_0 = \mathbf{1}$$

Metoda zmiennej metryki (cd.)

Po wyznaczeniu $\mathbf{V}_k$ wyznacza się $\mathbf{p}_k$ ze wzoru:

$$\mathbf{p}_k = -\mathbf{V}_k \, \mathbf{g}(\mathbf{W}_k)$$

oraz sam wektor zmiany wag $\Delta\mathbf{W}_k$ jako:

$$\Delta\mathbf{W}_k = \eta \, \mathbf{p}_k$$

Metoda zmiennej metryki (uwagi)

- Przy pierwszej iteracji, korzysta się z metody największego spadku gdyż $\mathbf{V}_0=\mathbf{1}$
- Po wyznaczeniu kierunku $\mathbf{p}_k$ należy koniecznie przeprowadzić minimalizację kierunkową by odtworzona macierz hesjanu była zawsze dodatnio określona
- Metoda jest uważana za jedną z najlepszych metod optymalizacji funkcji wielu zmiennych
- Wadą jest duża złożoność obliczeniowa i pamięciowa (n^2 elementów hesjanu). Stąd stosuje się ją do niezbyt dużych sieci (liczba wag <1000)

Metoda gradientów sprzężonych

- W metodzie tej nie korzysta się z informacji o hesjanie
- Kolejne kierunki minimalizacji $\mathbf{p}_k$ wybiera się w ten sposób by były ortogonalne i sprzężone z wszystkimi poprzednimi kierunkami $\mathbf{p}_0$, ..., $\mathbf{p}_{k-1}$
- Zbiór wektorów $\mathbf{p}_i$ i=0, ..., k jest sprzężony względem macierzy $\mathbf{G}$, jeśli $\mathbf{p}_i^T \mathbf{G}\, \mathbf{p}_J = 0$ dla $i \neq j$.

Metoda gradientów sprzężonych (cd.)

Wektor $\mathbf{p}_k$ spełniający założenia ortogonalności i sprzężenia z pozostałymi da się przedstawić w postaci rekurencyjnej:

$$\mathbf{p}_k = -\mathbf{g}_k + \beta_{k-1}\mathbf{p}_{k-1}$$

Wartość $\mathbf{p}_k$ zależy tylko od wartości aktualnego gradientu $\mathbf{g}_k$, poprzedniego kierunku $\mathbf{p}_{k-1}$ oraz współczynnika sprzężenia β_{k-1} kumulującego informację o wszystkich poprzednich kierunkach

Metoda gradientów sprzężonych (cd.)

Wartość współczynnika sprzężenia może być obliczana według:

$$\beta_{k-1} = \frac{\mathbf{g}_k^T(\mathbf{g}_k - \mathbf{g}_{k-1})}{\mathbf{g}_{k-1}^T \mathbf{g}_{k-1}}$$

lub według:

$$\beta_{k-1} = \frac{\mathbf{g}_k^T(\mathbf{g}_k - \mathbf{g}_{k-1})}{-\mathbf{p}_{k-1} \mathbf{g}_{k-1}}$$

Metoda gradientów sprzężonych (cd.)

- Kierunek minimalizacji $\mathbf{p}_k$ w początkowym kroku ($k = 0$) wyznacza się metodą największego spadku, gdyż wówczas nie jest znana wartość współczynnika sprzężenia $\beta_{k\text{-}1}$

Metoda gradientów sprzężonych (cd.)

- Po wybraniu *n* kierunków (gdzie *n* jest liczbą minimalizowanych wag) historia jest „czyszczona” i kolejny kierunek ponownie wyznacza się w oparciu o metodę największego spadku
- Minimalizacja kierunkowa jest zalecana, ale nie jak w metodzie zmiennej metryki konieczna do poprawnego działania

Metoda gradientów sprzężonych (podsumowanie)

- Uczenie metodą gradientów sprzężonych nie wymaga pamiętania macierzy Hesjanu i z tego względu stosuje się ją zwłaszcza do uczenia sieci o kilku tysiącach połączeń lub więcej, jako alternatywę wobec metody o zmiennej metryce, która w tych warunkach nakłada bardzo duże wymagania co do wykorzystywanej pamięci.
- Dla tak dużych sieci jest jedną z najszybszych metod optymalizacji, jednakże przy relatywnie małych sieciach w porównaniu z metodą zmiennej metryki jest dużo wolniejsza (choć wciąż szybsza od klasycznego algorytmu propagacji wstecznej, tj. metody największego spadku).