ADAM optimalizációs eljárás

Kormányos Andor

Komplex Rendszerek Fizikája Tanszék

2023. november 14.

Legmeredekebb ereszkedés gépi tanulásban

Emlékeztető:

- iteratív algoritmus
- válasszunk egy tetszőleges kezdőértéket az **a** paramétervektor elemeinek
- `repeat until convergence` {
  $a_j := a_j - \alpha \frac{\partial}{\partial a_j} J(\mathbf{a}; \mathbf{x}^{(i)}, y^{(i)})$
  }

Alapesetben

- az α learning rate állandó
- minden a_j esetén ugyanazt az α-t használjuk

Legmeredekebb ereszkedés gépi tanulásban

Emlékeztető:

- iteratív algoritmus
- válasszunk egy tetszőleges kezdőértéket az **a** paramétervektor elemeinek
- `repeat until convergence {`
  $a_j := a_j - \alpha \frac{\partial}{\partial a_j} J(\mathbf{a}; \mathbf{x}^{(i)}, y^{(i)})$
  }

Alapesetben

- az α learning rate állandó
- minden a_j esetén ugyanazt az α-t használjuk

Továbblépés:

- α változtatása ahogy minimumkeresés előrehalad (pl. kezdetben nagyobb, majd ahogy közeledünk a minimumhoz egyre kisebb)
- különböző a_j esetén különböző α_j használata

ADAM módszer

A gradient descent módszer kiterjesztése: Adaptive Movement Estimation Algorithm: ADAM

- a gradiens vektor átlagával kapcsolatos mennyiséget, és
- gradiens vektor szórásával kapcsolatos mennyiség átlagát számoljuk

ADAM módszer

A gradient descent módszer kiterjesztése: Adaptive Movement Estimation Algorithm: ADAM

- a gradiens vektor átlagával kapcsolatos mennyiséget, és
- gradiens vektor szórásával kapcsolatos mennyiség átlagát számoljuk

Bevezetünk három új "hyperparamétert"

- $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$
- tapasztalat szerint ezek az értékek problémák széles körére működnek

Emellett persze inicializálni kell az α értékét is, pl $\alpha = 0.01$

ADAM módszer

Előkészület:

- $\mathbf{a}$ paramétervektor elemeit egy `a[j]` vektorban tárolhatjuk
- hasonlóan, a $\frac{\partial}{\partial a_j} J(\mathbf{a})$ értékét egy `derivJ[j]` vektor elemeiként
- $\mathtt{a}^{(p)}[j]$ és $\mathtt{derivJ}^{(p)}[j]$ jelöli ezek értékét a p-ik iterációs lépésben

ADAM módszer

Előkészület:

- **a** paramétervektor elemeit egy `a[j]` vektorban tárolhatjuk
- hasonlóan, a $\frac{\partial}{\partial a_j} J(\mathbf{a})$ értékét egy `derivJ[j]` vektor elemeiként
- $\mathtt{a}^{(p)}[j]$ és $\mathtt{derivJ}^{(p)}[j]$ jelöli ezek értékét a p-ik iterációs lépésben
- $\mathtt{m}^{(p=0)}[j] = 0$ és $\mathtt{v}^{(p=0)}[j] = 0$
- **m** és **v** lényegében az *exponenciális mozgóátlagot* és az *exponenciális mozgó szórást* fogjál tartalmazni
- az iteráció során a korábbi értékek egyre kevésbé számítanak a pillanatnyi átlag/szórás számításakor

ADAM módszer

Előkészület:

- **a** paramétervektor elemeit egy `a[j]` vektorban tárolhatjuk
- hasonlóan, a $\frac{\partial}{\partial a_j} J(\mathbf{a})$ értékét egy `derivJ[j]` vektor elemeiként
- $\mathtt{a}^{(p)}[j]$ és $\mathtt{derivJ}^{(p)}[j]$ jelöli ezek értékét a p-ik iterációs lépésben
- $\mathtt{m}^{(p=0)}[j] = 0$ és $\mathtt{v}^{(p=0)}[j] = 0$
- **m** és **v** lényegében az *exponenciális mozgóátlagot* és az *exponenciális mozgó szórást* fogjál tartalmazni
- az iteráció során a korábbi értékek egyre kevésbé számítanak a pillanatnyi átlag/szórás számításakor

Szükség lesz még két segédváltozó vektorra: $\hat{m}[j]$, $\hat{v}[j]$

ADAM módszer

$\texttt{repeat until convergence}\ \{$

$\texttt{calculate derivJ}^{(p)}[j]$

$$\mathtt{m}^{(p)}[j] = \beta_1 \mathtt{m}^{(p-1)}[j] + (1.0 - \beta_1)\mathtt{derivJ}^{(p)}[j]$$
$$\mathtt{v}^{(p)}[j] = \beta_2 \mathtt{v}^{(p-1)}[j] + (1.0 - \beta_2)(\mathtt{derivJ}^{(p)}[j])^2$$

$$\hat{m}[j] = \frac{\mathtt{m}^{(p)}[j]}{1.0-(\beta_1)^p}$$
$$\hat{v}[j] = \frac{\mathtt{v}^{(p)}[j]}{1.0-(\beta_2)^p}$$

$$a^{(p)}[j] := a^{(p-1)}[j] - \alpha \frac{\hat{m}[j]}{\sqrt{\hat{v}[j]}+\epsilon}$$

$\}$

Látható, hogy a paramétertérben az effektív lépéshossz nagysága a p-ik lépésben $\Delta^{(p)}[j] = \alpha \frac{\hat{m}[j]}{\sqrt{\hat{v}[j]}+\epsilon}$

ADAM módszer

- megmutatható, hogy a legtöbb esetben $|\Delta^{(p)}[j]| \leq \alpha$
- a $\frac{\hat{m}[j]}{\sqrt{\hat{v}[j]}}$ felfogható egyfajta "jel-zaj aránynak" (signal-to-noise ratio)
- pl ha deriválté előjele gyakran változik, akkor $\frac{\hat{m}[j]}{\sqrt{\hat{v}[j]}} < 1$
- ha a jel-zaj arány kicsi, akkor kisebb lépésekkel megy előre a keresés

ADAM módszer

- megmutatható, hogy a legtöbb esetben $|\Delta^{(p)}[j]| \leq \alpha$
- a $\frac{\hat{m}[j]}{\sqrt{\hat{v}[j]}}$ felfogható egyfajta "jel-zaj aránynak" (signal-to-noise ratio)
- pl ha deriált előjele gyakran változik, akkor $\frac{\hat{m}[j]}{\sqrt{\hat{v}[j]}} < 1$
- ha a jel-zaj arány kicsi, akkor kisebb lépésekkel megy előre a keresés

Az ADAM módszert ismertető eredeti kézirat:
https://arxiv.org/abs/1412.6980