ADAM : Une méthode pour l'optimisation stochastique

Guillaume BERNARD-REYMOND, Guillaume BOULAND,
Camille MOTTIER, Abel SILLY

14 octobre 2024

L'article

Adam : A Method for Stochastic Optimization

Diederik P. Kingma
Université d'Amsterdam, OpenAI

Jimmy Lei Ba
Université de Toronto

Première année de publication : 2014

ICLR

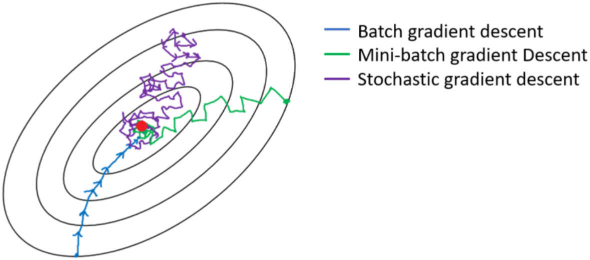
L'objectif

$f(\theta)$: fonction-objectif stochastique

Exemple : $f(\theta) = \sum_{i=1}^{n} L(x_i|\theta)$

Mini-batch : $f_t(\theta) = \sum_{i \in I_t} L(x_i|\theta)$

Objectif : minimiser $\mathbb{E}[f(\theta)]$



Méthode d'ordre 1

Rappel : Qu'est-ce qu'une méthode d'ordre 1 ? Évaluation de $f(\theta)$ et de $\nabla_\theta f(\theta)$

Algorithme 1 : Descente de gradient stochastique

Entrées : $f(\theta)$, α, θ_0

$t \longleftarrow 0$

tant que θ_t *ne converge pas* **faire**

  $t \longleftarrow t + 1$

  $g_t \longleftarrow \nabla_\theta f_t(\theta_{t-1})$

  $\theta_t \longleftarrow \theta_{t-1} - \alpha \cdot g_t(\theta_{t-1})$

fin

Sorties : θ_t

L'algorithme d'ADAM

Algorithme 2 : Adam

Entrées : $f(\theta)$, α, β_1, β_2, ε, θ_0

$m_0 \longleftarrow 0$

$v_0 \longleftarrow 0$

$t \longleftarrow 0$

tant que θ_t *ne converge pas* **faire**

$\quad t \longleftarrow t + 1$

$\quad g_t \longleftarrow \nabla_\theta f_t(\theta_{t-1})$

$\quad m_t \longleftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$ (Estimateur de $\mathbb{E}[g_t]$)

fin

Sorties : θ_t

Illustration Guibou !

L'algorithme d'ADAM

Algorithme 3 : Adam

Entrées : $f(\theta)$, α, β_1, β_2, ε, θ_0

$m_0 \longleftarrow 0$

$v_0 \longleftarrow 0$

$t \longleftarrow 0$

tant que θ_t *ne converge pas* **faire**

$\quad t \longleftarrow t + 1$

$\quad g_t \longleftarrow \nabla_\theta f_t(\theta_{t-1})$

$\quad m_t \longleftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$ (Estimateur de $\mathbb{E}[g_t]$)

$\quad v_t \longleftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$ (Estimateur de $\mathbb{E}[g_t^2]$)

fin

Sorties : θ_t

L'algorithme d'ADAM

Algorithme 4 : Adam

Entrées : $f(\theta)$, α, β_1, β_2, ε, θ_0

$m_0 \longleftarrow 0$

$v_0 \longleftarrow 0$

$t \longleftarrow 0$

tant que θ_t *ne converge pas* **faire**

$\quad t \longleftarrow t + 1$

$\quad g_t \longleftarrow \nabla_\theta f_t(\theta_{t-1})$

$\quad m_t \longleftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$ (Estimateur de $\mathbb{E}[g_t]$)

$\quad v_t \longleftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$ (Estimateur de $\mathbb{E}[g_t^2]$)

$\quad \widehat{m}_t \longleftarrow m_t / (1 - \beta_1^t)$ (« débiaisage »)

$\quad \widehat{v}_t \longleftarrow v_t / (1 - \beta_2^t)$

fin

Sorties : θ_t

L'algorithme d'ADAM

Algorithme 5 : Adam

Entrées : $f(\theta)$, α, β_1, β_2, ε, θ_0

$m_0 \longleftarrow 0$

$v_0 \longleftarrow 0$

$t \longleftarrow 0$

tant que θ_t *ne converge pas* **faire**

$t \longleftarrow t + 1$

$g_t \longleftarrow \nabla_\theta f_t(\theta_{t-1})$

$m_t \longleftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$ (Estimateur de $\mathbb{E}[g_t]$)

$v_t \longleftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$ (Estimateur de $\mathbb{E}[g_t^2]$)

$\widehat{m}_t \longleftarrow m_t / (1 - \beta_1^t)$ (« débiaisage »)

$\widehat{v}_t \longleftarrow v_t / (1 - \beta_2^t)$

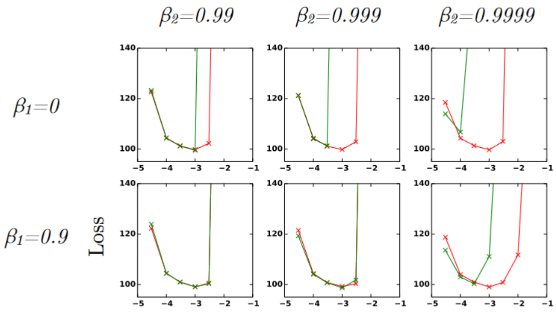
$\theta_t \longleftarrow \theta_{t-1} - \alpha \cdot \widehat{m}_t / (\sqrt{\widehat{v}_t} + \varepsilon)$ (Mise à jour)

fin

Sorties : θ_t

Correction de biais

$$
\begin{aligned}
& \mathbb{E}[m_t] \\
= \quad & (1-\beta_1)\sum_{i=1}^{t}\beta_1^{t-i}\mathbb{E}[g_i] \\
= \quad & \mathbb{E}[g_t]\underbrace{(1-\beta_1^t)}_{<1} + \underbrace{(1-\beta_1)\sum_{i=1}^{t}\beta_1^{t-i}(\mathbb{E}[g_i]-\mathbb{E}[g_t])}_{\zeta}
\end{aligned}
$$



after 100 epochs

Résultats de convergence

On définit le regret par :

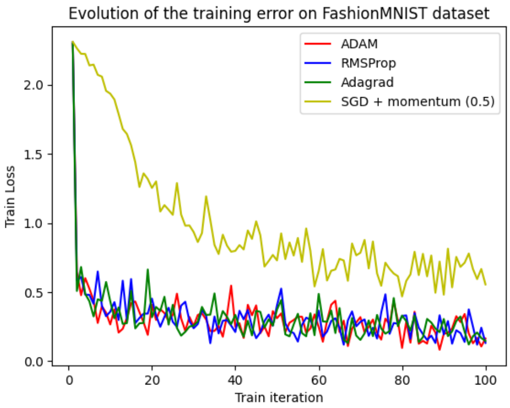
$$R(T) = \sum_{t=1}^{T} f_t(\theta_t) - \min_{\theta \in \mathcal{X}} \sum_{t=1}^{T} f_t(\theta)$$

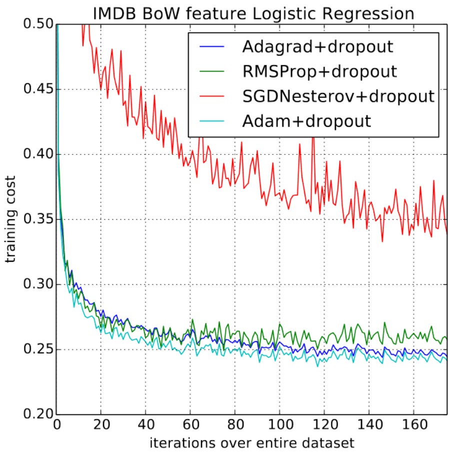
Théorème

Sous certaines hypothèses de majoration du gradient de f_t, et de l'écart entre les valeurs de θ_n on a :

$$\frac{R(T)}{T} = O\left(\frac{1}{\sqrt{T}}\right)$$

Simulations numériques





Conclusion

ADAM est un algorithme :

- facilement implémentable
- peu gourmand en mémoire
- efficace dans de nombreux cas

D'où son succès ! !

Limite de l'article : manque de support théorique