

O método Adam

Esta aula constitui um tópico adicional da disciplina. Trata-se de um conteúdo opcional. Sua atividade não valerá nota e não precisa ser enviada.



- Até agora, estudamos Redes Neurais imaginando que o treinamento dos seus parâmetros é feito usando uma aplicação tradicional do Método do Gradiente, tal como, por exemplo,

$$w_j = w_j - \alpha \frac{\partial}{\partial w_j} J(\vec{w}, b)$$

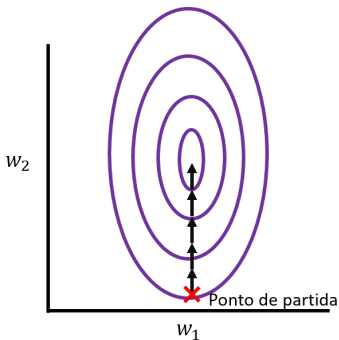
onde α é um valor fixo > 0 (taxa de aprendizado).

- Entretanto, hoje em dia existem métodos de otimização avançada aplicada à redes neurais que vão ajustando α automaticamente durante o processo de treinamento. Nesse contexto insere-se o **Método Adam**

Método do Gradiente versus Método Adam

Método do Gradiente *versus* Método Adam

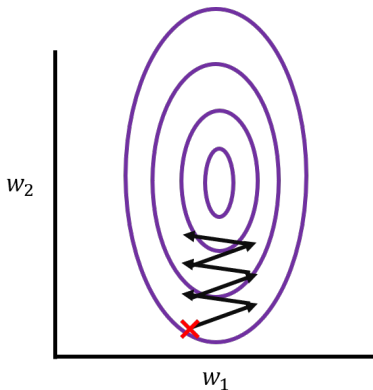
Primeiro caso: Utilizando um valor fixo e pequeno para α , o Método do Gradiente tenderá a dar vários pequenos passos em direção ao mínimo de $J(\vec{w}, b)$, conforme ilustrado abaixo:



- Para um exemplo como esse, o Método Adam perceberá que é possível **aumentar** o valor de α para agilizar o processo de chegada até o mínimo de $J(\vec{w}, b)$.

Método do Gradiente versus Método Adam

Segundo caso: utilizando um valor fixo e muito grande para α , o Método do Gradiente tenderá a dar passos maiores, porém poderá ocorrer o que está ilustrado abaixo:



- Nesse caso, o Método Adam perceberá que é melhor **reduzir** o valor de α para suavizar o processo de chegada até o mínimo de $J(\vec{w}, b)$.

Características do Método Adam

- A sigla 'Adam' abrevia a expressão **A**daptive **M**oment estimation
- Ele utiliza uma taxa de aprendizado α diferente para cada parâmetro que está sendo treinado, conforme exemplo abaixo:

$$\begin{aligned}w_1 &= w_1 - \alpha_1 \frac{\partial}{\partial w_1} J(\vec{w}, b) \\&\dots \\w_{12} &= w_{12} - \alpha_{12} \frac{\partial}{\partial w_{12}} J(\vec{w}, b)\end{aligned}$$

Características do Método Adam

- A sigla 'Adam' abrevia a expressão **Adaptive Moment estimation**
- Ele utiliza uma taxa de aprendizado α diferente para cada parâmetro que está sendo treinado, conforme exemplo abaixo:

$$\begin{aligned}w_1 &= w_1 - \alpha_1 \frac{\partial}{\partial w_1} J(\vec{w}, b) \\&\dots \\w_{12} &= w_{12} - \alpha_{12} \frac{\partial}{\partial w_{12}} J(\vec{w}, b)\end{aligned}$$

Ideia principal:

- Se w_j (ou b_j) está se movendo de forma constante, com pouca variação em termos de direção, aumentar α_j (Primeiro caso que vimos)
- Se w_j (ou b_j) está oscilando muito em termos de direção, diminuir α_j (Segundo caso que vimos)

Características do Método Adam

- A sigla 'Adam' abrevia a expressão **Adaptive Moment estimation**
- Ele utiliza uma taxa de aprendizado α diferente para cada parâmetro que está sendo treinado, conforme exemplo abaixo:

$$\begin{aligned}w_1 &= w_1 - \alpha_1 \frac{\partial}{\partial w_1} J(\vec{w}, b) \\&\dots \\w_{12} &= w_{12} - \alpha_{12} \frac{\partial}{\partial w_{12}} J(\vec{w}, b)\end{aligned}$$

Ideia principal:

- Se w_j (ou b_j) está se movendo de forma constante, com pouca variação em termos de direção, aumentar α_j (Primeiro caso que vimos)
- Se w_j (ou b_j) está oscilando muito em termos de direção, diminuir α_j (Segundo caso que vimos)

Observação:

- Nós já estamos utilizando o Método Adam nos nossos códigos, entregando a ele um palpite inicial acerca da taxa de aprendizado α .
- Ainda vale a pena **mantermos a estratégia** de testar diferentes valores para α para verificar se a convergência torna-se mais rápida ou não, apesar do método já ter uma estratégia automatizada para seleção desse parâmetro ao longo das iterações.

De olho no código!

De olho no código!

Iremos agora verificar como usar o Tensorflow para treinar um modelo de reconhecimento de dígitos de 0 a 9 escritos à mão.

Acesse o Python Notebook usando o QR code ou o link abaixo:

https://colab.research.google.com/github/xaximpvp2/master/blob/main/codigo_aula19_topico_adicional.ipynb



Acesse os dados necessários para rodar o código usando os links abaixo:

https://ufprbr0-my.sharepoint.com/:f:/g/personal/ricardo_schumacher_ufpr_br/E15c4hFwzLFNunFo17IRxuIB8_7_D29BISiKHsc8JY2ANQ?e=QCzPS4

OBS: Para adicionar os dados ao ambiente do Colab Notebook, no menu do canto esquerdo da tela do Colab clique em “Arquivos” e depois “Fazer upload para o armazenamento da sessão”. Então carregue os arquivos baixados.

Parte 1

Rode todo o código. Certifique-se que você o compreendeu.

Parte 2

- 1 Verifique se o aumento de número de épocas aumenta a taxa de acerto do modelo.