Universidade de Brasília

DEPARTAMENTO DE ESTATÍSTICA

16 de junho de 2023

Lista 3: Otimização em RNA.

Prof. Guilherme Rodrigues

Redes Neurais Profundas

Tópicos especiais em Estatística 1

(A) **As questões deverão ser respondidas em um único relatório *PDF* ou *html*, produzido usando as funcionalidades do *Rmarkdown* ou outra ferramenta equivalente.**

(B) **O aluno poderá consultar materiais relevantes disponíveis na internet, tais como livros, *blogs* e artigos.**

(C) **O trabalho é individual. Suspeitas de plágio e compartilhamento de soluções serão tratadas com rigor.**

(D) **Os códigos *R* utilizados devem ser disponibilizados na integra, seja no corpo do texto ou como anexo.**

(E) **O aluno deverá enviar o trabalho até a data especificada na plataforma Aprender 3.**

(F) **O trabalho será avaliado considerando o nível de qualidade do relatório, o que inclui a precisão das respostas, a pertinência das soluções encontradas, a formatação adotada, dentre outros aspectos correlatos.**

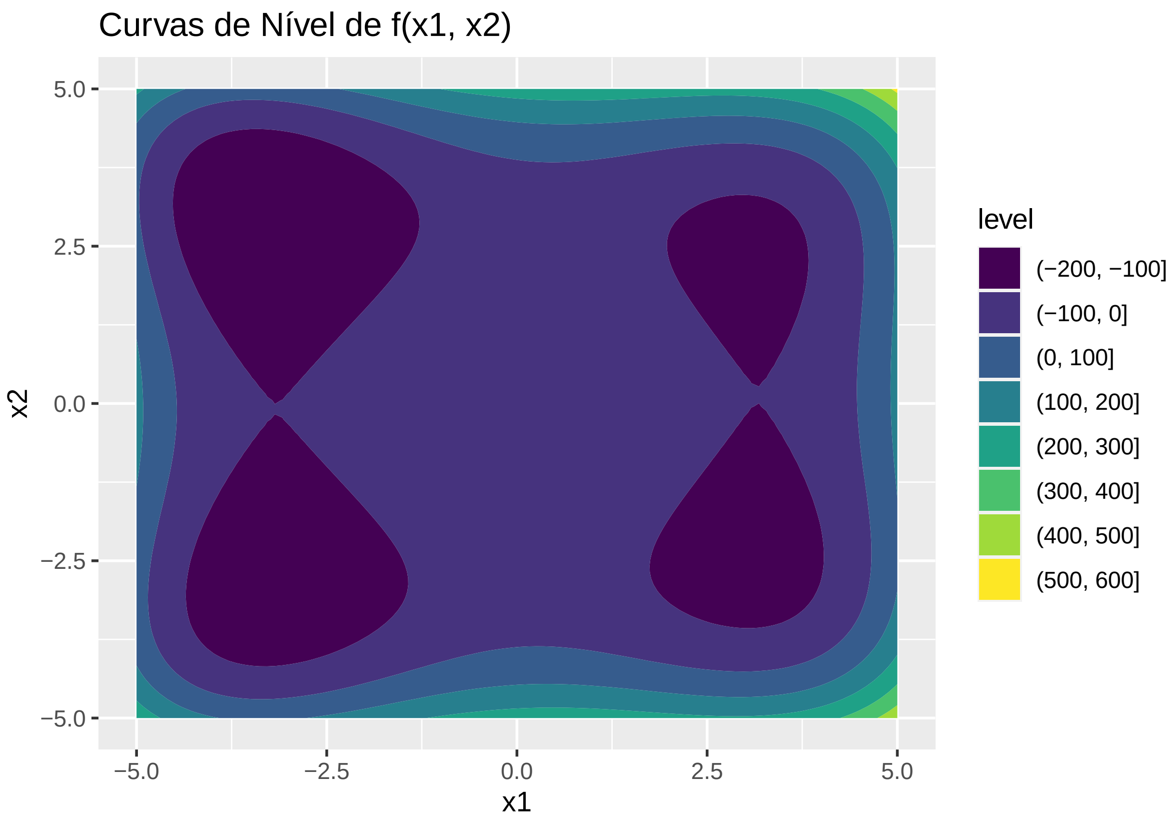
(G) **Escreva seu código com esmero, evitando operações redundantes, comentando os resultados e usando as melhores práticas em programação.**

Considere a função

$$f(x_1, x_2) = x_1^4 + x_2^4 + x_1^2 x_2 + x_1 x_2^2 - 20x_1^2 - 15x_2^2$$

para responder os itens a seguir.

a) Apresente um gráfico com as curvas de nível de $f(x_1, x_2)$. Quantos pontos críticos a função parece ter? Dica para usuários do `R`: use a função `geom_contour_filled()`.



Após o gráfico é percepitível que há 4 mínimos, sendo um o mínimo global.

b) Encontre (algericamente) o gradiente de f em relação ao vetor $x = (x_1, x_2)$. Isso é,

$$\nabla_x f(\boldsymbol{x}) = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}\right).$$

$\frac{\partial f}{\partial x_1} = 4x_1^3 + x_2^2 + 2x_1.x_2 - 40x_1$

$\frac{\partial f}{\partial x_2} = 4x_2^3 + x_1^2 + 2x_1.x_2 - 30x_2$

$\nabla f(x) = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}\right) = \left(4x_1^3 + x_2^2 + 2x_1.x_2 - 40x_1, 4x_2^3 + x_1^2 + 2x_1.x_2 - 30x_2\right)$

** c) ** Crie uma função computacional que implemente o método do gradiente para minimizar a função em estudo. Permita ao usuário definir a taxa de aprendizado, o número de passos e o ponto de partida.

```
## Mínimo local 1 =  2.903159 2.262064
## Valor =  -114.1802

## Mínimo local 2 =  3.284497 -2.624033
## Valor =  -160.9439

## Mínimo local 3 =  -3.040141 2.865981
## Valor =  -153.6491
```

```
## Mínimo global =  -3.510853 -3.196775
## Valor =  -218.7268
```

d) Use a função criada no item c) para encontrar o valor obtido pelo método do gradiente partindo-se do ponto inicial $(x_1^{(0)}, x_2^{(0)}) = (0, 5)$. Use taxa de aprendizado igual a 0.01 e execute 100 passos.

```
## Ponto inicial: (0.5, 0.5)

## Mínimo encontrado:  2.903159 2.262064

## Valor mínimo:  -114.1802
```

e) Repita o item d), agora com as seguintes taxas de aprendizado: 1, 0.1, 0.01, 0.001, 0.0001. Qual dessas opções lhe parece mais apropriada nesse caso? Justifique sua resposta.

```
##   Taxa de Aprendizado Mínimo x1 Mínimo x2 Valor Mínimo
## 1               1e+00       NaN       NaN          NaN
## 2               1e-01       NaN       NaN          NaN
## 3               1e-02 2.9031592 2.2620640    -114.1802
## 4               1e-03 2.9129912 2.0885358    -113.6805
## 5               1e-04 0.7221731 0.6536531     -15.7356
```

Com taxa = 1 e taxa = 0.1: Os valores divergem, ou seja, o gradiente não converge para nenhum ponto de mínimo, resultando em uma otimização mal-sucedida.

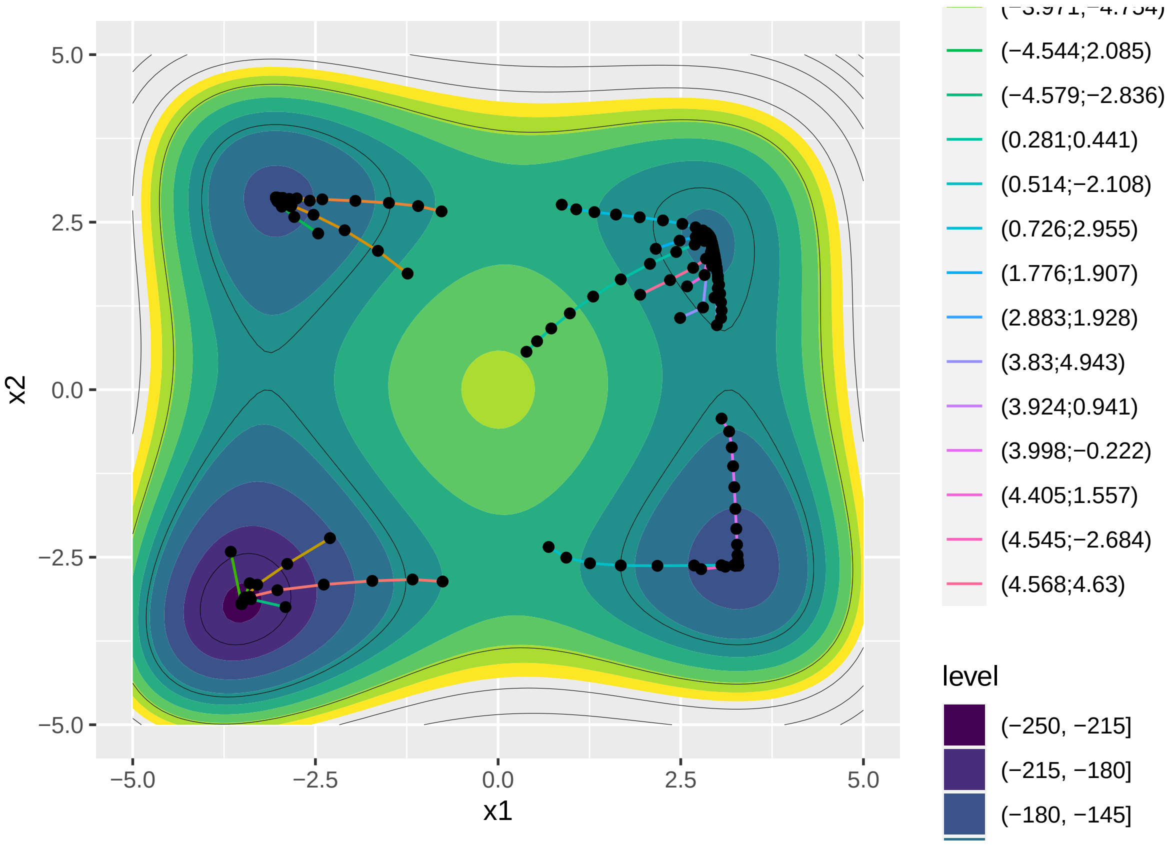
Com taxa = 0.01: O gradiente converge para o "Mínimo Local 3", que é um ponto de mínimo encontrado no processo de otimização.

Com taxa = 0.001: O gradiente converge para o "Mínimo Local 3", assim como no caso anterior, porém a taxa de aprendizagem menor faz com que o processo seja mais lento.

Com taxa = 0.0001: O gradiente não chega a nenhum ponto de mínimo, pois os passos são muito curtos e insuficientes para alcançar uma solução ótima.

Portanto, entre as opções testadas, a taxa de aprendizagem de taxa = 0.01 é a mais adequada para este caso, pois permitiu que o gradiente convergisse para um ponto de mínimo.

f) Fixe a semente do gerador de números aleatórios no valor 123 (se estiver usando o `R`, basta executar o código `set.seed(123)`). Repita novamente o item d), agora partindo de 20 pontos escolidos aleatoriamente (uniformimente) no quadrado $-5 < x_1, x_2 < 5$. Refaça o gráfico do item a) e adicione uma linha representando o caminho percorrido por cada uma das 20 otimizações. Qual foi o percentual de vezes em que o algoritmo encontrou o mínimo global da função (despresando um eventual desvio de menor importân-



cia)?

```
##   n Percentage
## 1 4         20
```

g) Repita o item d), substituindo o método do gradiente pelo método do gradiente com momento (veja a Seção 8.3.2 do livro *Deep Learning*). Use taxa de aprendizado $\epsilon = 0.01$, parâmetro de momento $\alpha = 0.9$ e velocidade inicial $v = 0$.

```
## Ponto de gradiente com momento =  3.275777 -2.617305
##  Valor =  -160.9395
```

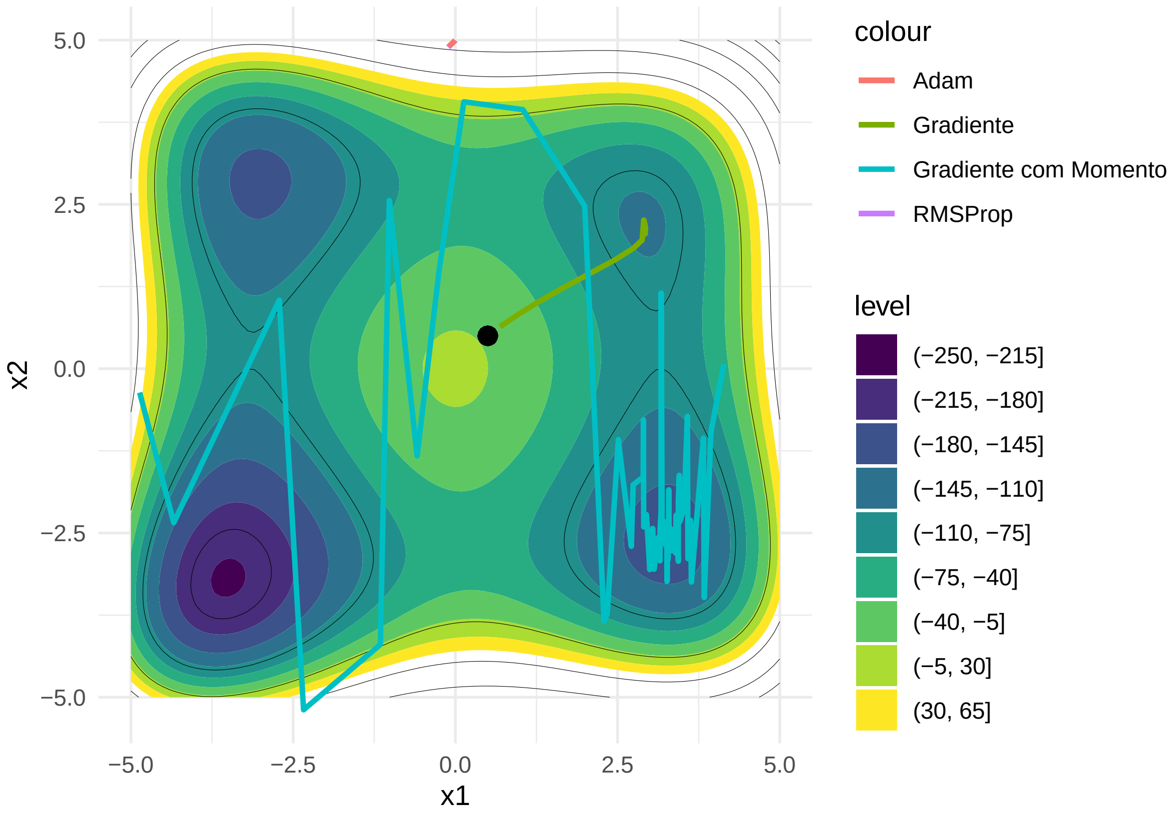
h) Repita o item d), substituindo o método do gradiente pelo método RMSProp (veja a Seção 8.5.2 do livro *Deep Learning*). Use taxa de aprendizado $\epsilon = 0.001$, taxa de decaimento $\rho = 0.9$ e constante $\delta = 10^{-6}$.

```
## Ponto RMSProp =  -0.1097923 4.891533
##  Valor =  210.7905
```

i) Repita o item d), substituindo o método do gradiente pelo método ADAM (veja a Seção 8.5.3 do livro *Deep Learning*). Use taxa de aprendizado $\epsilon = 0.001$ e taxas de decaimento $\rho_1 = 0.9$ e $\rho_2 = 0.999$.

```
## Ponto Adam =  -0.1013468 4.901361
##  Valor =  214.181
```

j) Apresente graficamente, em uma única figura, os caminhos percorridos pelas otimizações executadas nos itens d), g), h) e i).


colour
Adam
Gradiente
Gradiente com Momento
RMSProp
level
(−250, −215]
(−215, −180]
(−180, −145]
(−145, −110]
(−110, −75]
(−75, −40]
(−40, −5]
(−5, 30]
(30, 65]
x2
5.0
2.5
0.0
−2.5
−5.0
−5.0
−2.5
0.0
2.5
5.0
x1