Métodos Numéricos - Implementação Gradiente Descendente

Gustavo Aparecido de Souza Viana

***Abstract*— Métodos númericos são utilizados a fim de aproximar a solução de um determinado problema, como por exemplo resolução de integral, derivada e entre outros. De contra partida, esse método de resolução é mais custoso do que soluções fechadas. Neste trabalho será aboradado a implementação do Gradiente Descendente para achar o "zero" ou mínimo da função. Os resultados mostraram que é possível encontrar o mínimo da função aproximada, mas dependendo do intervalo que é utilizado pode ser que o mínimo da função foi pulado e consequentemente não encontrado.**

I. INTRODUÇÃO

Métodos númericos são utilizados em diversas áreas como por exemplo em redes neurais para saber se a rede converge ou não, e para resolver problemas computacionais mas não de forma exata, ou seja, métodos númericos chegam a solução desejada com uma taxa de erro, sendo que quem está utilizando minimizará o erro para que seja irrelevante.

Métodos númericos pelo fato de serem métodos de otimização, ou seja, necessitam de diversas iterações e a partir de cada iteração vai se aproximando da solução final e otimizando a mesma, são métodos que possuem um alto custo computacional.

Neste trabalho foi abordado a implementação o método do Gradiente Descendente com objetivo de encontrar o mínimo da função ou "zero da função".

II. CONCEITOS FUNDAMENTAIS

Nesta seção, será apresentado os conceito básico de Gradiente Descedente.

O método do Gradiente Descendente utiliza a Equação 1, onde n representa cada iteração do método.

$$x(n+1) = x_n - f'(x_n) \tag{1}$$

III. METODOLOGIA

Nesta seção será apresentado a metodologia utilizada para implementar o Gradiente Descendente em C++. O código fonte pode ser encontrado em https://github.com/apparecidoo/ProgramacaoCientifica.

Foi criada a classe *NumericMethod* onde os métodos para as equações Eq. 2 e Eq. 3 foram implementados seguindo o pseudocódigo da Figura 1. Esses métodos continham como argumentos, o x_0 que é o valor inicial de x, o *beta* que é o espaçamento, e o por fim, o *epsilon* que é o valor mínimo tolerável da função. Para executarmos os testes, foi criado um método executando ambas equações mas variando o beta de 0.1 à 1 e assim, analisados os resultados.

x_0 = valor inicial

$f_0 = f(x_0)$ // avalie f em x_0

while $\left(f'_n \neq 0\right)$ {

$s_i = \dfrac{df}{dx}(x_i)$ // compute inclinação

$x_{i+1} = x_i - \beta \cdot s_i$ // ande em x

$f'_{i+1} = f'(x_{i+1})$ // avalie f no novo x_{i+1}

}

Fig. 1: Representação do pseudocódigo do Gradiente Descendente

$$f(x) = x^2 \tag{2}$$

$$f(x) = x^3 - 2x^2 + 2 \tag{3}$$

IV. EXPERIMENTOS E RESULTADOS

Nesta Seção será apresentado os experimentos e seus respectivos resultados. Foram feitos dois métodos onde cada um apresentava a solução da descida do gradiente, foi utilizado 0.1 para o valor de beta e valor do epsilon. Os resultados das duas equações estão na Figura 2 e Figura 3.

Os resultados demonstram que no primeiro método (relacionado a Eq. 2) quaisquer valores de beta foram encontrados um mínimo da função no ponto x = 0.

O mesmo ocorreu no segundo método (relacionado a Eq. 3), mas como podemos ver, o segundo método não convergiu com os valores de beta menores que 0.5 devido ao valor do x ser uma dizima, mesmo assim encontrou o mínimo da função no ponto x = 1.3333. Com valores maiores que 0.5 o método não encontrou o mínimo da função.

```
f(x) = x^2
Beta = 0.1
Converge - Function_i: 8.06953e-13 X: 4.03477e-13

Beta = 0.2
Converge - Function_i: 9.05068e-13 X: 4.52534e-13

Beta = 0.3
Converge - Function_i: 7.3787e-13 X: 3.68935e-13

Beta = 0.4
Converge - Function_i: 2.09715e-13 X: 1.04858e-13

Beta = 0.5
Converge - Function_i: 0 X: 0

Beta = 0.6
Cannot Converge - Function_i: -0.8 X: -0.4

Beta = 0.7
Cannot Converge - Function_i: -1.6 X: -0.8

Beta = 0.8
Cannot Converge - Function_i: -2.4 X: -1.2

Beta = 0.9
Cannot Converge - Function_i: -3.2 X: -1.6

Beta = 1
Cannot Converge - Function_i: -4 X: -2
```

Fig. 2: Representação dos resultados da Equação 2

```
f(x) = x^3 - 2x^2 + 2
Beta = 0.1
Cannot Converge - Function_i: 2.66667 X: 1.33333

Beta = 0.2
Cannot Converge - Function_i: 2.66667 X: 1.33333

Beta = 0.3
Cannot Converge - Function_i: 2.66667 X: 1.33333

Beta = 0.4
Cannot Converge - Function_i: 2.66667 X: 1.33333

Beta = 0.5
Converge - Function_i: 0 X: 0

Beta = 0.6
Cannot Converge - Function_i: -0.8 X: -0.4

Beta = 0.7
Cannot Converge - Function_i: -1.6 X: -0.8

Beta = 0.8
Cannot Converge - Function_i: -2.4 X: -1.2

Beta = 0.9
Cannot Converge - Function_i: -3.2 X: -1.6

Beta = 1
Cannot Converge - Function_i: -4 X: -2
```

Fig. 3: Representação dos resultados da Equação 3

V. CONCLUSÃO

Após a análise do conceito e resultados obtidos após a aplicação do método de Gradiente Descendente é possível apontar que método é preciso dependendo do beta utilizado, ou seja, dependendo da taxa de aprendizagem a solução pode ou não ser encontrada.

REFERENCES