EP3 – Relatório

Interpolação e

diferenciação numérica

Pedro Leyton Pereira – NUSP 9778794

Raphael dos Reis Gusmão – NUSP 9778561

Introdução

Este relatório desenvolve a solução do **EP3** da matéria **MAC0210 - Laboratório de Métodos Numéricos**. O enunciado era, resumidamente, deduzir uma formula que aproxima as derivadas de uma função (por exemplo as cores de cada *pixel* em uma imagem); implementar um programa que aproxime as derivadas e, em seguida, interpolar a imagem comprimida; e criar funções para serem aproximadas e verificar o erro .

Parte I: Aproximação das derivadas

Utilizando polinômios de Taylor para funções de 2 variáveis, podemos encontrar aproximações para as derivadas de .

■ Função a ser aproximada

■ Aproximação

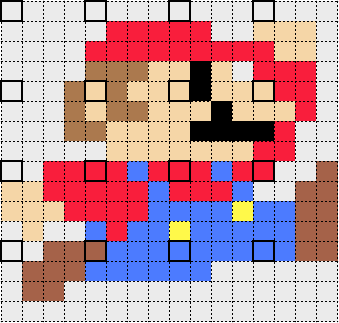
■ Erro

Para encontrar aproximações para as derivadas de nos pontos que estão nas extremidades tal que ou , podemos usar o método *backward* para os polinômios de Taylor.

Parte II: Programa

O programa pode ser encontrado em ***ep3.m*** na pasta. Aqui estão as explicações de funções não imediatamente óbvias.

compress(img, scale): Comprime a imagem na escala (scale) desejada. O resultado final é uma imagem com vezes menos *pixels*, pegos em uma malha uniforme da imagem original. Um exemplo seria uma imagem com 200x200 *pixels*, e com fator de escala 5. O resultado final seria uma imagem 40x40, composta por 1 *pixel* a cada 5 *pixels* da imagem original, para cada direção.

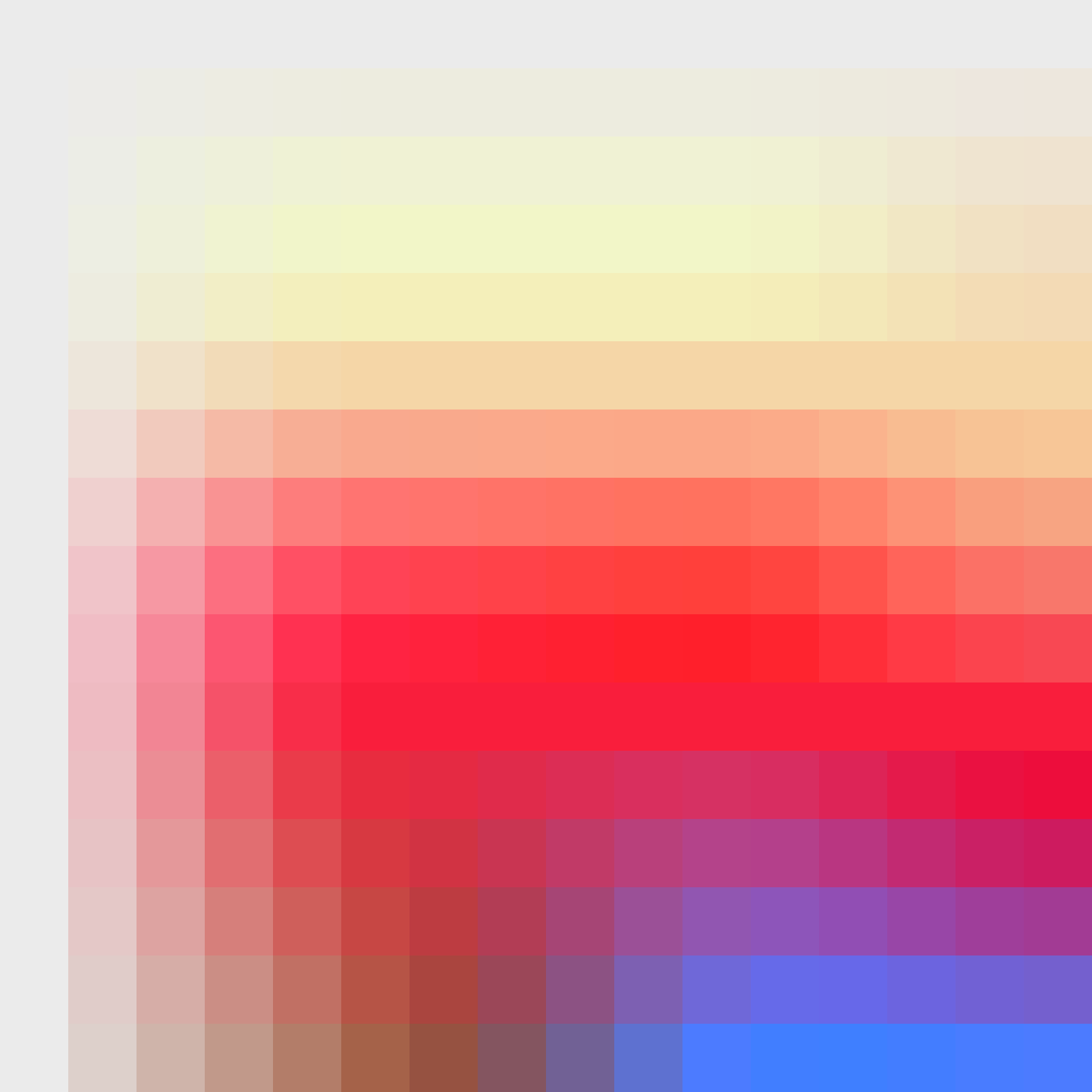


*Visualização de uma imagem 16x16 sendo comprimida em escala 4 para uma imagem 4x4.*



decompress(img, scale): Recebe a imagem comprimida pela compress() e a escala (scale) desejada e interpola a imagem para que ela tenha um tamanho vezes maior. Essa parte utiliza a interpolação bicúbica para estimar os valores originais dos *pixels*.

*Visualização da compressão da imagem anterior (4x4) sendo descomprimida para o tamanho original (16x16). A imagem não se parece com a original pois havia pouca informação contida na imagem comprimida.*





Parte III: Funções e avaliações delas

Nosso programa já contém funções *built-in* caso queira testar como funciona. Para isso, basta executar interpolaFuncao() sem nenhum argumento, e então escolha uma das funções disponíveis.

Para utilizar o programa com sua própria função, execute da seguinte maneira: interpolaFuncao(f, ax, bx, ay, by, hx, hy, scale), no qual f é uma função anônima que recebe @(x, y) e devolve um valor único, ou seja, uma função .

Também é possível carregar a função de um arquivo, usando o ***handler*** interpolaFuncao(). Este simplesmente carrega o arquivo e interpola a função desejada.

A interpolação de funções e feita da seguinte maneira: o programa avalia a função em poucos pontos e depois estima, por interpolação, o resto da função. A quantidade de pontos que a função avaliará depende da scale utilizada.

Aqui estão algumas das imagens feitas pelo nosso programa: