

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
INSTITUTO DE COMPUTAÇÃO
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

CAIO COHEN DE SOUZA

SITE INTERATIVO DE CÁLCULO NUMÉRICO
Desenvolvido em Angular

RIO DE JANEIRO
2024

CAIO COHEN DE SOUZA

SITE INTERATIVO DE CÁLCULO NUMÉRICO

Desenvolvido em Angular

Trabalho de conclusão de curso de graduação
apresentado ao Instituto de Computação da
Universidade Federal do Rio de Janeiro como
parte dos requisitos para obtenção do grau de
Bacharel em Ciência da Computação.

Orientador: Prof. Marcello Goulart Teixeira

RIO DE JANEIRO

2024

CIP - Catalogação na Publicação

S729s Souza, Caio Cohen de
Site interativo de cálculo numérico:
desenvolvido em angular / Caio Cohen de Souza. --
Rio de Janeiro, 2024.
76 f.

Orientador: Marcello Goulart Teixeira.
Trabalho de conclusão de curso (graduação) -
Universidade Federal do Rio de Janeiro, Instituto
de Computação, Bacharel em Ciência da Computação,
2024.

1. Teoria. 2. Implementação do site. 3. Iniciando
em Angular. I. Teixeira, Marcello Goulart, orient.
II. Título.

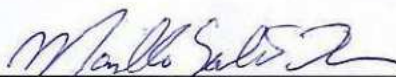
CAIO COHEN DE SOUZA

SITE INTERATIVO DE CÁLCULO NUMÉRICO
Desenvolvido em Angular

Trabalho de conclusão de curso de graduação
apresentado ao Instituto de Computação da
Universidade Federal do Rio de Janeiro como
parte dos requisitos para obtenção do grau de
Bacharel em Ciência da Computação.

Aprovado em 17 de SETEMBRO de 2024

BANCA EXAMINADORA:



Marcello Goulart Teixeira
Doutorado (UFRJ)

Documento assinado digitalmente



MARIA HELENA CAUTIERO HORTA JARDIM

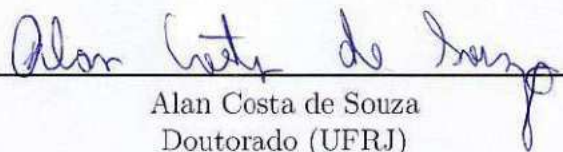
Data: 18/09/2024 15:18:15-0300

Verifique em <https://validar.iti.gov.br>

Maria Helena Cautiero Horta Jardim
Doutorado (UFRJ)



Mauro Antônio Rincon
Doutorado (UFRJ)



Alan Costa de Souza
Doutorado (UFRJ)

RESUMO

Este artigo apresenta uma nova aplicação desenvolvida para a comunidade acadêmica da UFRJ, com o objetivo de auxiliar os alunos de Cálculo Numérico a compreender melhor a disciplina, elevar o Coeficiente de Rendimento (CR) médio e reduzir a taxa de evasão. O site oferece explicações teóricas e ferramentas práticas interativas para reforçar o entendimento. O artigo inicia com um resumo do conteúdo abordado, seguido de uma explicação sobre a metodologia de desenvolvimento e implementação do site. Além disso, discute-se o potencial do site para futuras expansões, tanto para outras disciplinas do currículo de Ciências da Computação quanto para outros cursos e níveis de ensino. A aplicação encontra-se atualmente em estado Beta, representando um primeiro passo significativo para o desenvolvimento contínuo do projeto e sua contribuição para a comunidade acadêmica.

Palavras-chave: UFRJ; Cálculo Numérico

ABSTRACT

This article presents a new application developed for the academic community of UFRJ, aimed at helping Numerical Analysis students better understand the subject, improve the average Grade Point Average (GPA), and reduce dropout rates. The website offers theoretical explanations and interactive practical tools to reinforce understanding. The article begins with a summary of the content covered, followed by an explanation of the development and implementation methodology of the website. Additionally, it discusses the potential for future expansions of the site, both for other courses in the Computer Science curriculum and for other programs and educational levels. The application is currently in Beta status, representing a significant first step towards the ongoing development of the project and its contribution to the academic community.

Keywords: UFRJ; Numerical Analysis

LISTA DE ILUSTRAÇÕES

Figura 1 – Exempmplo gráfico do método de Newton-Raphson	14
Figura 2 – Exempmplo gráfico do método da Secante	15
Figura 3 – Discretização com $n = 9$	18
Figura 4 – Exemplo de interpolação de 8 pontos. Repare que o gráfico do polinômio passa por todos os pontos.	31
Figura 5 – Gráfico do polinômio interpolador do exemplo.	32
Figura 6 – Polinômios interpoladores de grau 2, em vermelho, e de grau 7 do exemplo.	39
Figura 7 – Polinômio de grau 4.	42
Figura 8 – Diferença, em módulo, entre os polinômios de grau 3 e 4 do exemplo.	42
Figura 9 – Gráfico de $f(x) = \frac{1}{1+25x^2}$ e polinômio interpolador de grau 4.	44
Figura 10 – Gráfico de $f(x) = \frac{1}{1+25x^2}$ e polinômio interpolador de grau 9.	44
Figura 11 – Gráfico de $f(x) = \frac{1}{1+25x^2}$ e polinômio interpolador de grau 19.	45
Figura 12 – Divisão do intervalo e polinômio interpolador em cada subintervalo: grau 5 em $[-5/9,5/9]$ e grau 2 em $[-1,-5/9]$ e em $[5/9,1]$. Repare a descontinuidade da derivada no quarto e no sétimo pontos.	46
Figura 13 – Concentrando pontos a serem interpolados próximos aos extremos do intervalo.	46
Figura 14 – Protótipo feito em Figma	56
Figura 15 – Funções de transformação dos intervalos	59
Figura 16 – Função que recupera os valores que serão mostrados nos eixos	60
Figura 17 – Avaliação quantitativa de 1 a 10 da interface do site.	71
Figura 18 – Avaliação quantitativa de 1 a 10 da usabilidade do site.	72
Figura 19 – Se a teoria do site abrange todo o conteúdo visto em sala	72
Figura 20 – Avaliação quantitativa da clareza e facilidade de entendimento da teoria	72
Figura 21 – Se o aluno teve dificuldade em utilizar a aplicação dos métodos	73
Figura 22 – Avaliação quantitativa do quão útil os alunos acreditam que o site teria sido quando eles cursavam a disciplina	73
Figura 23 – Avaliação quantitativa do quão útil os alunos acreditam que o site será para futuros alunos da disciplina	73

SUMÁRIO

1	INTRODUÇÃO	9
2	A TEORIA	10
2.1	FONTES DE ERRO	10
2.2	RAÍZES DE FUNÇÕES REAIS	11
2.2.1	Método da Bisseção	11
2.2.2	Método da Falsa Posição	12
2.2.3	Teorema do Ponto Fixo	12
2.2.4	Método de Newton-Raphson	13
2.2.5	Método da Secante	13
2.3	POLINÔMIO DE TAYLOR	14
2.4	PROBLEMAS DE VALOR INICIAL	15
2.4.1	Método de Euler	16
2.4.2	Método de Euler Melhorado	16
2.4.3	Métodos de Runge-Kutta	16
2.5	PROBLEMAS DE VALOR DE CONTORNO (PVC)	17
2.5.1	Problema de Dirichlet	18
2.6	SISTEMAS LINEARES	21
2.6.1	Método da eliminação Gaussiana	21
2.6.2	Algoritmo da Retro-Substituição	22
2.6.3	Fatoração LU	23
2.6.4	Como determinar LU	23
2.6.5	Métodos Iterativos	25
2.6.6	Método de Gauss-Jacobi	25
2.6.7	Critério de convergência do Método G-J	27
2.6.8	Método de Gauss-Seidel	27
2.6.9	Critério de convergência para o método G-S	28
2.6.10	Teste de parada para os métodos iterativos	29
2.6.11	Número de condicionamento	29
2.7	INTERPOLAÇÃO POLINOMIAL	30
2.7.1	Forma Direta	30
2.7.2	Forma de Lagrange	32
2.7.3	Forma de Newton	33
2.7.4	Interpolação inversa	42
2.7.5	Comparação entre as formas de interpolação	43

2.7.6	Fenômeno de Runge	44
2.8	AJUSTE DE CURVAS	47
2.8.1	Método dos mínimos quadrados	47
2.8.2	Ajuste não linear	49
2.8.3	Outras Linearizações	49
2.9	INTEGRAÇÃO NUMÉRICA	50
2.9.1	Regra dos trapézios	50
2.9.2	Regra dos trapézios Repetidos	51
2.9.3	1/3 de simpson	52
2.9.4	1/3 de simpson repetido	53
2.9.5	Quadratura Gaussiana	53
3	A IMPLEMENTAÇÃO DO SITE	55
3.1	A CRIAÇÃO DO PROTÓTIPO	55
3.2	IMPLEMENTANDO A TELA INICIAL	56
3.3	O PROBLEMA COM AS EQUAÇÕES	57
3.4	VÍDEOS DEMONSTRATIVOS	57
3.5	IMPLEMENTAÇÃO DOS GRÁFICOS	58
3.6	IMPLEMENTAÇÃO DOS SIMULADORES	61
3.6.1	Métodos de Runge Kutta	61
3.6.2	Raízes de funções reais	61
3.6.3	Ajustes de curvas	63
3.6.4	Interpolação	64
3.6.5	Integração Numérica	65
4	INSTALANDO E COMEÇANDO COM O ANGULAR	66
4.1	PRÉ-REQUISITOS	66
4.2	INSTALANDO O ANGULAR CLI	66
4.3	CLONANDO O PROJETO	66
4.4	ATUALIZANDO UM PROJETO ANGULAR E GERENCIAMENTO DE VERSÕES DO NODE COM NVM	66
4.4.1	Atualizando um Projeto Angular	67
4.4.2	Gerenciamento de Versões do Node.js com NVM	67
4.5	EXECUTANDO O PROJETO ANGULAR	68
4.6	ESTRUTURA DO PROJETO ANGULAR	69
4.7	CRIANDO COMPONENTES	70
4.8	RESUMO	70
5	EXPERIÊNCIA DOS ALUNOS	71
5.1	RESULTADOS	71

6	PLANOS FUTUROS E CONCLUSÃO	74
6.1	PLANOS FUTUROS	74
6.2	CONCLUSÃO	74
	REFERÊNCIAS	76

1 INTRODUÇÃO

Este relatório detalha o processo de criação e desenvolvimento de um site dedicado ao aprendizado de cálculo numérico. O objetivo principal deste projeto é fornecer uma plataforma interativa e abrangente para estudantes que buscam compreender os fundamentos e aplicações do cálculo numérico.

O estudo do cálculo numérico é essencial para diversas áreas, incluindo engenharia, ciência da computação, física e matemática aplicada. No entanto, muitos estudantes enfrentam desafios ao tentar dominar os conceitos e técnicas envolvidos nesse campo. Reconhecendo essa necessidade, este projeto foi concebido com a intenção de oferecer uma abordagem inovadora e acessível ao aprendizado do cálculo numérico.

Ao longo deste relatório, serão apresentados os principais aspectos do processo de criação do site, desde a concepção inicial até a implementação final. Serão discutidos os objetivos do projeto, as metodologias utilizadas, os recursos empregados e os desafios encontrados durante o desenvolvimento. Além disso, serão destacadas as características principais do site, incluindo a estrutura, os conteúdos disponíveis e as ferramentas interativas oferecidas aos usuários.

Espera-se que este relatório forneça uma visão abrangente do trabalho realizado, destacando não apenas os resultados alcançados, mas também o processo criativo e as decisões tomadas ao longo do caminho. Mais importante ainda, espera-se que o site resultante deste projeto seja uma ferramenta valiosa para estudantes interessados em aprimorar seus conhecimentos em cálculo numérico e, assim, contribuir para o seu sucesso acadêmico e profissional.

Existe um site parecido do Instituto Federal de São Paulo (IFSP) que também possui calculadoras numéricas. Porém, diferente do site apresentado neste relatório, o da IFSP é muito mais voltado para a utilidade prática do que pro ensino de novos estudantes, além da interface e usabilidade menos intuitiva e a falta de conteúdo teórico ou vídeos demonstrativos. A aplicação da IFSP é voltada para profissionais já formados.

Existem também outros como symbolab e wolframalpha, mas eles são mais genéricos, voltados pra matemática em geral, além de que o usuário precisa pagar uma mensalidade para ter acesso completo à plataforma.

2 A TEORIA

Este capítulo tem como propósito apresentar de forma resumida a teoria de cálculo numérico que foi incorporada ao site. Embora esta seção não esteja diretamente ligada à estrutura e funcionamento do site em si, é de suma importância para compreender os fundamentos subjacentes aos recursos e ferramentas disponíveis.

A teoria abordada neste capítulo foi elaborada pelo autor com base em suas anotações de aula e discussões com o orientador. Destaca-se que todo o conteúdo teórico é original, não tendo sido reproduzido de qualquer outro lugar.

Ao explorar esta seção, os leitores terão acesso a uma compreensão abrangente dos princípios fundamentais do cálculo numérico, incluindo conceitos-chave, métodos de resolução de problemas e aplicações práticas. Dessa forma, este capítulo serve como um alicerce para a compreensão e utilização eficaz do conteúdo apresentado no site, proporcionando uma base teórica para os usuários.

2.1 FONTES DE ERRO

No contexto do cálculo numérico, é fundamental compreender que seu objetivo principal é obter uma solução aproximada para problemas que são desafiadores ou mesmo impossíveis de resolver de forma analítica. Nesse sentido, é essencial reconhecer as fontes potenciais de erros que podem distanciar nossa solução do resultado ideal.

1-Dados: Para quem trabalha com Cálculo Numérico, muitas vezes ele precisa retornar um resultado com base em dados previamente obtidos e esses dados, muitas vezes, são aproximados devido a falhas na coleta dos dados ou devido ao uso de dispositivos de medição que, por si só, fornecem resultados aproximados. Esses erros podem ser mínimos e, em uma primeira impressão, serem inofensivos. Porém, ao aplicarmos métodos numéricos esse erro inicial pode influenciar negativamente o resultado final.

2-Representação numérica na máquina: A grande maioria dos problemas envolvem números decimais. Na máquina, esses números são representados na base 2. Como o sistema numérico dos computadores tem um quantidade finita, mesmo que grande, de bits para representar os números, normalmente é necessário efetuar arredondamentos para que a informação seja armazenada, fazendo com que o número na memória não seja exatamente o número original.

3-Operações Matemáticas: Considerando dois números com imprecisões, sejam elas causadas pela coleta de dados ou pelo armazenamento no computador, ao efetuarmos as quatro operações entre esses números a imprecisão pode aumentar. Assim, depois de diversas operações, o erro pode se acumular e tornar o resultado inviável.

4-Substituir o problema por um problema aproximado: A modelagem numérica consiste, de um modo geral, em rescrever o problema original de uma forma que seja viável a implementação computacional da solução deste problema aproximado. Dessa forma, ao aplicarmos um método numérico obtemos uma solução aproximada para o problema original. Para nossa sorte, ao aplicarmos a maioria dos métodos numéricos é possível também estimar o erro da solução aproximada ou, ao menos, controlar parâmetros do método de forma a obter uma solução aproximada com um erro aceitável.

A primeira fonte de erro citada acima, devido a coleta de dados, não é fonte de estudo do Cálculo Numérico. Assim, consideraremos sempre que os dados fornecidos são minimamente confiáveis. Para diminuir o erro da segunda fonte, podemos optar pela representação estendida dos números (double). Já os erros causados pela terceira fonte podem ser amenizados com estratégias que envolvem uma ordem apropriada das operações matemáticas como, por exemplo, priorizar divisões com o valor em módulo do divisor consideravelmente maior que o valor em módulo do dividendo.

Já os erros referentes a quarta fonte podem ser minimizados com a escolha apropriada do método numérico para resolver o problema. É nesta direção que podemos atuar de forma a obter resultados numéricos com considerável precisão.

2.2 RAÍZES DE FUNÇÕES REAIS

No primeiro tópico abordado na disciplina de Cálculo Numérico, exploramos métodos para encontrar as raízes de funções reais com variável real.

Na prática, há diversas abordagens para encontrar uma raiz de uma função. Por exemplo, se tivermos acesso ao gráfico da função em questão, podemos visualmente identificar os pontos onde ela cruza o eixo x . No entanto, essa é uma abordagem simplificada e bastante imprecisa, servindo apenas para termos uma ideia da localização das raízes.

Além disso, essa abordagem não é viável para computadores. Eles não têm capacidade para analisar infinitos pontos, sua memória é limitada e a representação numérica também possui suas restrições. Além disso, nem sempre é possível utilizar fórmulas fixas, como a fórmula de Bhaskara, para encontrar as raízes das funções reais.

Diante dessas limitações, neste contexto, são apresentados os quatro métodos numéricos com os quais é possível determinar um valor aproximado para as raízes de uma função real.

2.2.1 Método da Bisseção

Seja $f : [a, b] \rightarrow \mathbb{R}$ uma função contínua e possuindo uma e apenas uma raiz ξ em $[a, b]$ e $f(a)$ e $f(b)$ têm sinais opostos, seguindo o teorema do valor intermediário. Queremos determinar ξ .

O Método da Bisseção consiste em:

1. Fazemos $x_m = (a + b)/2$ e verificamos se a raiz está na metade esquerda, ou seja $[a, x_m]$, ou na metade direita do intervalo original, ou seja, $[x_m, b]$. Fazemos isto testando o sinal de $f(a) * f(x_m)$. Se for menor ou igual a zero então a raiz ξ está no intervalo $[a, x_m]$, caso contrário está no intervalo $[x_m, b]$.
2. Dado um ϵ tal que $0 < \epsilon \ll 1$ (muito próximo de 0), aplicamos o passo 1 até que $|f(\xi)| < \epsilon$ ou $|a - b| < \epsilon$.

Dessa forma, ao fim do processo teremos um valor x_m tão próximo da raiz ξ quanto menor for ϵ .

O método da bisseção é um método lento, mas tem convergência garantida, ou seja, sempre encontrará um valor aproximado para a raiz.

2.2.2 Método da Falsa Posição

Este método é similar ao método da bisseção. Seja $f : [a, b] \rightarrow \mathbb{R}$ uma função contínua e possuindo uma e apenas uma raiz ξ em $[a, b]$ e $f(a)$ e $f(b)$ têm sinais opostos. Queremos determinar ξ .

1. Geometricamente, traçamos uma secante entre a e b (segmento de reta com extremos em $(a, f(a))$ e $(b, f(b))$).
2. A equação desta reta secante é $f(x) = f(a) + (x - a) \frac{f(b) - f(a)}{b - a}$.
3. No eixo x , a reta passa pelo ponto $(x_m, 0)$, com $x_m = \frac{af(b) - bf(a)}{f(b) - f(a)}$.
4. Se $f(a) \times f(x_m) \leq 0$ faça $b = x_m$, ou seja, a raiz está no intervalo $[a, x_m]$, caso contrário a raiz está no intervalo $(x_m, b]$ e então faça $a = x_m$.
5. Dado um ϵ tal que $0 < \epsilon \ll 1$ (muito próximo de 0), continuamos a iteração 4 até que $f(\xi) < \epsilon$ ou $|a - b| < \epsilon$.

2.2.3 Teorema do Ponto Fixo

Este não é um método muito prático para se achar uma raiz, mas é importante que se entenda o teorema do ponto fixo para entender o próximo método.

Dada uma função $\varphi : [a, b] \rightarrow \mathbb{R}$ contínua, dizemos que ξ é ponto fixo de φ se ξ é tal que $\varphi(\xi) = \xi$.

Dada uma equação $f(x) = 0$, por algebrismo chegamos a $x = \varphi(x)$. O Método do Ponto Fixo consiste em gerar uma sequência x_k onde

$$x_{k+1} = \varphi(x_k), k = 0, 1, 2, \dots$$

sendo a função φ chamada de função de iteração.

Por exemplo, se $x^3 - x - 5 = 0$, podemos chegar a $x = (x + 5)^{1/3}$, ou seja, neste caso temos $\varphi(x) = (x + 5)^{1/3}$.

A ideia é, ao invés de determinar ξ com a função f , vamos determinar ξ como o ponto fixo de $x = \varphi(x)$. Isto é possível porque, considerando a forma geral da função de iteração como $\varphi(x) = x - A(x)f(x)$, com $A(\xi) \neq 0$, é possível mostrar que $f(\xi) = 0 \iff \varphi(\xi) = \xi$, ou seja, ξ é raiz da equação $f(x) = 0$ se e somente se ξ é ponto fixo de φ .

Se $\varphi(x) = (x + 5)^{1/3}$ e $x_0 = 0$, temos então $x_1 = \varphi(x_0) = 1.709975$, $x_2 = \varphi(x_1) = 1.886138$, $x_3 = \varphi(x_2) = 1.902502$ e $x_4 = \varphi(x_3) = 1.904008$, que coincide com a raiz da equação até a terceira casa decimal.

Mas nem toda função de iteração gera uma sequência convergente. Por exemplo, a partir da mesma equação podemos definir $\varphi(x) = x^3 - 5$. Considerando $x_0 = 0$, obtemos $x_1 = \varphi(x_0) = -5$, $x_2 = \varphi(x_1) = -130$ e $x_3 = \varphi(x_2) = -2197005.0$, indicando que a sequência não convergirá para a raiz $\xi = 1.904160$ (com 6 casas decimais).

É possível mostrar que, entre todos os métodos de ponto fixo possível para uma dada equação $f(x) = 0$, o método mais rápido será aquele em que $\varphi'(\xi) = 0$. Usaremos esta informação para estabelecer o próximo método.

2.2.4 Método de Newton-Raphson

O teorema do ponto fixo é tal que $\varphi(x) = x - A(x)f(x)$, com $A(\xi) \neq 0$, logo, $\varphi'(x) = 1 - A'(x)f(x) - A(x)f'(x)$. Fazendo $x = \xi$ e igualando a expressão a zero, obtemos $\varphi'(\xi) = 1 - A'(\xi)f(\xi) - A(\xi)f'(\xi) = 0 \Rightarrow A(\xi) = \frac{1}{f'(\xi)}$, pois $f(\xi) = 0$.

Assumindo que $A(x) = \frac{1}{f'(x)}$, obtemos a função de iteração $\varphi(x) = x - \frac{f(x)}{f'(x)}$, que é a função de iteração do Método de Newton-Raphson (MNR).

Assim, o MNR é dado por $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$, com x_0 sendo um valor inicial qualquer dentro das condições de convergência. É importante ressaltar que o MNR é ainda um Método do Ponto Fixo e, por isso, está sujeito a critérios de convergência.

Como teste de parada temos $|f(x_{k+1})| < \epsilon$ ou $|x_{k+1} - x_k| < \epsilon$, com $0 < \epsilon \ll 1$. Além disso, para garantir a parada do algoritmo, incluímos o teste $k \geq n$, com n sendo um número máximo de iterações definido.

Graficamente, este método funciona pegando a reta tangente no ponto $(x_0, f(x_0))$ e determina x_1 como o ponto de interseção desta reta com o eixo x . No passo seguinte, consideramos a reta tangente no ponto $(x_1, f(x_1))$ e x_2 será a abscissa do ponto onde esta reta corta o eixo x e assim por diante.

2.2.5 Método da Secante

Este método tem o propósito de evitar o cálculo da derivada da função $f(x)$, como é feito no MNR. Porém, ele é similar ao método da falsa posição, uma vez que esses dois métodos utilizam retas secantes para encontrar a raiz.

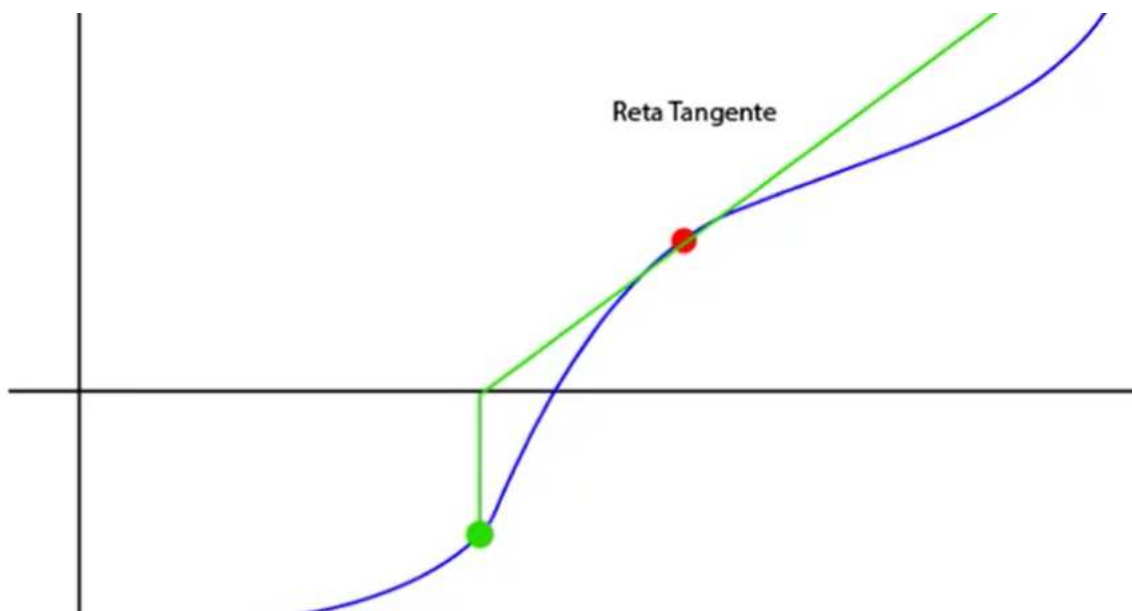


Figura 1 – Exemmplo gráfico do método de Newton-Raphson

Dados x_0 e x_1 iniciais, fazemos

$$x_{k+1} = x_k - \frac{f(x_k)}{\frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}}$$

onde a expressão $\frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}$ é a secante que aproxima a derivada de $f(x_k)$.

Como condição de parada temos $|f(x_{k+1})| < \epsilon$ ou $|x_{k+1} - x_k| < \epsilon$, com $0 < \epsilon \ll 1$. Ainda aqui é importante considerar um número máximo de iterações.

Gaficamente, este método traça a reta secante entre os pontos $(x_0, f(x_0))$ e $(x_1, f(x_1))$ e x_2 será a abcissa do ponto onde esta reta corta o eixo x . Em seguida, repetimos o processo para os pontos $(x_1, f(x_1))$ e $(x_2, f(x_2))$, sendo x_3 a abcissa do ponto onde esta nova reta corta o eixo x . Este processo é repetido até que algum critério de parada seja satisfeito.

2.3 POLINÔMIO DE TAYLOR

O polinômio de Taylor é uma forma de representar uma função por um polinômio. Claro, essa representação vem com um custo, já que se trata de uma aproximação. Porém, muitas vezes é preferível fazer essa transformação para resolver certos tipos de problemas. Além disso, há muitos métodos numéricos derivados do polinômio de Taylor.

Seja f uma função C^n em um determinado intervalo aberto $I = (b_1, b_2) \subseteq \mathbb{R}$ e $a \in I$. O polinômio de Taylor de grau n de f em torno de a será:

$$f(x) \approx f(a) + f'(a)(x-a) + \frac{f''(a)(x-a)^2}{2!} + \frac{f'''(a)(x-a)^3}{3!} + \dots + \frac{f^n(a)(x-a)^n}{n!}, \quad \forall x \in I.$$

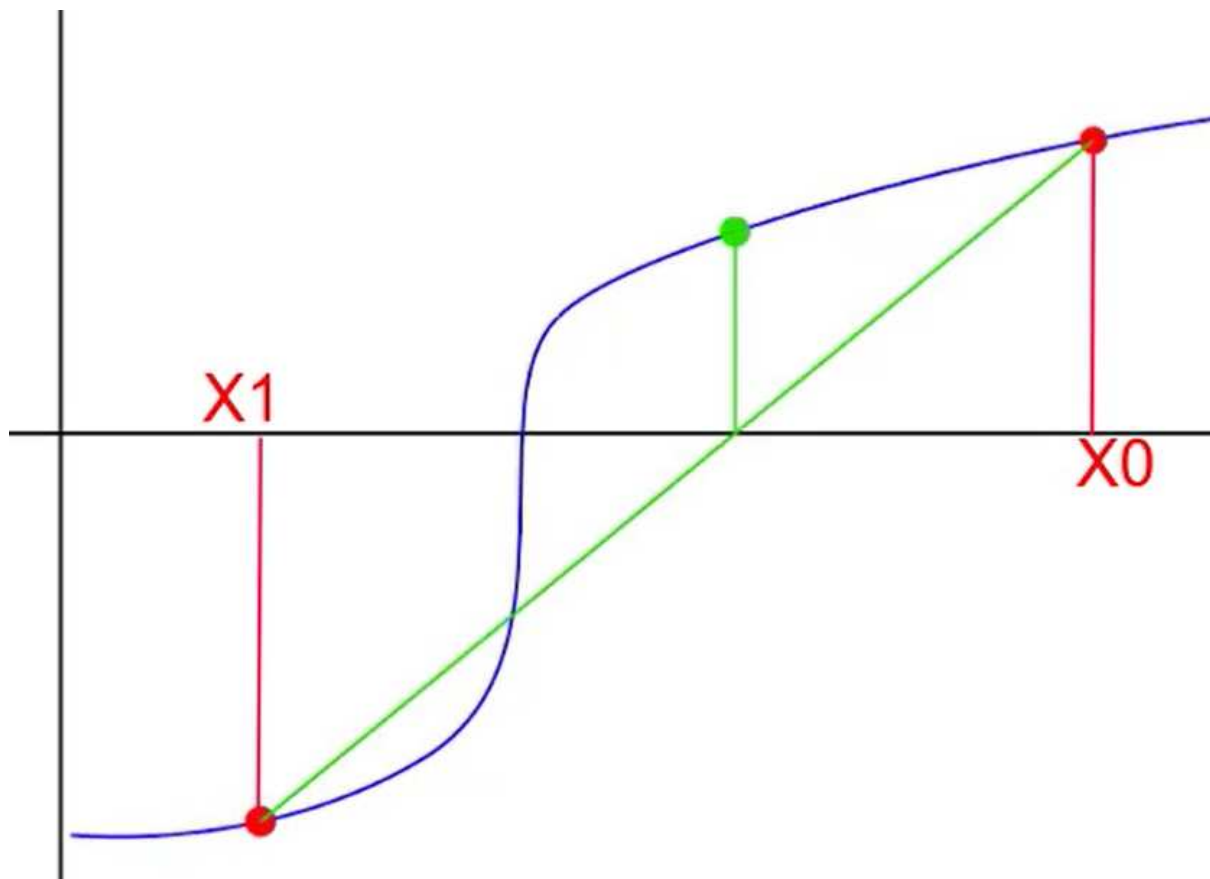


Figura 2 – Exemmplo gráfico do método da Secante

Se $f^{(n+1)}$ existe no intervalo I e $a \in I$, então, para $x \in I$ temos

$$f(x) = P_n(x) + R_n(x)$$

onde $P_n(x) = f(a) + f'(a)(x - a) + \frac{f''(a)(x-a)^2}{2!} + \dots + \frac{f^{(n)}(a)(x-a)^n}{n!}$, um polinômio de grau menor ou igual a n , e

$$R_n(x) = \frac{f^{(n+1)}(\xi)(x - a)^{n+1}}{(n + 1)!}$$

para algum ξ entre x e a . A função $R_n(x)$ é chamada de função erro.

Se o ponto a , chamado de centro da série, é igual a zero, a série é chamada de Série de MacLaurin.

2.4 PROBLEMAS DE VALOR INICIAL

Considere o Problema de Valor Inicial de primeira ordem $y'(t) = f(t, y(t))$ com o valor inicial $y(t_0) = y_0$

Queremos determinar valores da função $y(t)$ em determinados pontos. Devido a aplicação dos métodos numéricos, esses valores serão aproximados.

Dado h suficientemente pequeno, criamos a discretização $t_i = t_0 + ih$, com $i = 1, 2, 3, \dots$. Vamos então determinar valores $y_i \approx y(t_i)$.

2.4.1 Método de Euler

Este método consiste em, uma vez que se sabe derivada de y , dada pela função f , "caminhar" pela derivada da função utilizando um passo h . Quanto menor o passo, mais preciso será a estimativa. Em outras palavras, a cada passo dado de t_i para t_{i+1} , o valor da função varia de y_i para y_{i+1} , com este valor sendo determinado por meio da reta tangente ao gráfico de y no ponto (t_i, y_i) .

Assim, dado $y_i \approx y(t_i)$, determinamos o próximo valor y_{i+1} por

$$y_{i+1} = y_i + hy'_i \Rightarrow y_{i+1} = y_i + hf(t_i, y_i)$$

pois, pelo PVI, $y'_i = f(t_i, y_i)$.

Podemos chegar a esta mesma expressão se considerarmos o Polinômio de Taylor de grau 1 para a função y na vizinhança de um ponto \bar{t} , dado por $y(t) = y(\bar{t}) + (t - \bar{t})y'(\bar{t}) + R_1(t)$. Fazendo agora $t = t_{i+1}$ e $\bar{t} = t_i$, obtemos $y(t_{i+1}) = y(t_i) + (t_{i+1} - t_i)y'(t_i) + R_1(t)$ e então $y_{i+1} \approx y_i + hf(t_i, y_i)$, que é a mesma expressão acima.

Assim, o Método de Euler é dado por: para $i = 0, 1, 2, 3, \dots, n$, $y_{i+1} = y_i + hf(t_i, y_i)$.

2.4.2 Método de Euler Melhorado

O método de Euler Melhorado segue o mesmo princípio do anterior, mas em vez de estimar o valor da função seguindo a derivada, ele utiliza a reta que passa pela bissetriz do ângulo formado entre a reta tangente (aproximada) a y no ponto (t_i, y_i) e a reta tangente (aproximada) a y no ponto (t_{i+1}, \bar{y}_{i+1}) , com $\bar{y}_{i+1} = y_i + hf(t_i, y_i)$, ou seja, este é o valor de y_{i+1} que é obtido ao aplicarmos o Método de Euler.

Assim, o Método de Euler Melhorado é dado por

$$y_{i+1} = y_i + \frac{h}{2}[f(t_i, y_i) + f(t_{i+1}, \bar{y}_{i+1})], \quad i = 0, 1, 2, \dots, n-1,$$

onde $\bar{y}_{i+1} = y_i + hf(t_i, y_i)$.

2.4.3 Métodos de Runge-Kutta

Os métodos de Runge-Kutta são métodos tais que:

1. São de passos singulares, ou seja, para determinar y_{i+1} utilizamos apenas informações no passo anterior i .
2. Não precisam das derivadas da função f , mas precisam determinar o valor desta função várias vezes.

3. Após desenvolvermos a função f em série de Taylor, convenientemente, encontramos a expressão do método baseado na série de Taylor.

É fácil verificar que o Método de Euler é um Método de Runge-Kutta de ordem 1, já que ele pode ser obtido diretamente da Série de Taylor e a função erro é da ordem de h^2 .

Também é possível mostrar que o Método de Euler Melhorado é um Método de Runge-Kutta, mas de ordem 2. Para mostrar isso, basta desenvolver a parcela $f(t_{i+1}, \bar{y}_{i+1})$ em Polinômio de Taylor de grau 1 na direção t e na direção y , na vizinhança de (t_i, y_i) ,

Outros métodos de Runge-Kutta:

Ordem 3:

$$y_{n+1} = y_n + \frac{2}{9}K_1 + \frac{1}{3}K_2 + \frac{4}{9}K_3$$

$$K_1 = hf(x_n, y_n)$$

$$K_2 = hf\left(x_n + \frac{h}{2}, y_n + \frac{K_1}{2}\right)$$

$$K_3 = hf\left(x_n + \frac{3h}{4}, y_n + \frac{3K_2}{4}\right)$$

Ordem 4:

$$y_{n+1} = y_n + \frac{1}{6}[K_1 + 2K_2 + 2K_3 + K_4]$$

$$K_1 = hf(x_n, y_n)$$

$$K_2 = hf\left(x_n + \frac{h}{2}, y_n + \frac{K_1}{2}\right)$$

$$K_3 = hf\left(x_n + \frac{h}{2}, y_n + \frac{K_2}{2}\right)$$

$$K_4 = hf(x_n + h, y_n + K_3)$$

2.5 PROBLEMAS DE VALOR DE CONTORNO (PVC)

Vamos resolver numericamente uma EDO de segunda ordem do tipo:

$$y'' = f(x, y(x), y'(x))$$

dadas condições de contorno em dois pontos a e b .

Vamos estudar apenas os casos em que f é uma função linear em y e y' . Na verdade, vamos resolver apenas quando a EDO for do tipo:

$$y''(x) + Ay'(x) + By(x) = g(x)$$

com $n = 2$ condições de contorno.

$$\begin{cases} a_1 y(a) + b_1 y'(a) = \delta_1 \\ a_2 y(b) + b_2 y'(b) = \delta_2 \end{cases}$$

onde $a_1, a_2, b_1, b_2, \delta_1, \delta_2 \in \mathbb{R}$ e a_1 e b_1 não podem ser simultaneamente iguais a zero (o mesmo para a_2 e b_2).

Se $b_1 = b_2 = 0$, temos as condições de contorno $y(a) = \frac{\delta_1}{a_1}$ e $y(b) = \frac{\delta_2}{a_2}$, chamadas de condições de contorno de Dirichlet ou prescritas.

Se $a_1 = a_2 = 0$, temos as condições de contorno $y'(a) = \frac{\delta_1}{B_1}$ e $y'(b) = \frac{\delta_2}{B_2}$, chamadas de condições de contorno de Neumann ou de fluxo.

Qualquer outra condição de contorno é chamada de mista.

Para ilustrar a aplicação das diferenças finitas para resolver o PVC, vamos considerar o exemplo a seguir.

2.5.1 Problema de Dirichlet

Considere o problema de valor de contorno

$$\begin{cases} y''(x) + Ay'(x) + By(x) = g(x) \\ y(0) = \bar{y}_0 \\ y(1) = \bar{y}_1 \end{cases}$$

para $x \in [0, 1]$.

Vamos considerar a seguinte discretização do domínio $[0, 1]$: com $h = \frac{1}{n+1}$, seja $x_i = h \times i$, $i = 0, 1, 2, \dots, n+1$.

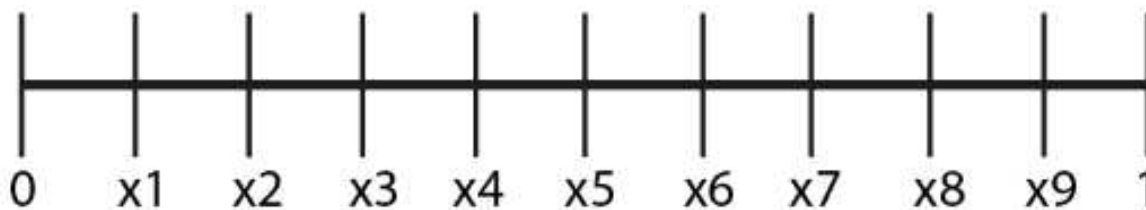


Figura 3 – Discretização com $n = 9$

O P.V.C na sua forma discreta é dado por

$$y_i'' + Ay_i' + By_i = g(x_i)$$

para $i = 1, 2, \dots, n$, pois \bar{y}_0 e \bar{y}_1 são dados do problema.

Nesse caso, pode-se perceber que dividimos o problema em 11 diferentes pontos. Desses 11 pontos, \bar{y}_0 e \bar{y}_1 já nos foram dados, mas y_1, y_2, \dots, y_n são incógnitas. Temos, neste caso, $n = 9$ incógnitas, então precisamos de n equações!

As n equações vêm da forma discreta do problema. Porém, aparecem ali y'_i e y''_i . Precisamos então reescrever a forma discreta da equação de forma a conter apenas as incógnitas $y_i, i = 1, 2, \dots, 9$ do problema.

Euler Avançado

A Série de Taylor para y em torno de $\bar{x} \in \mathbb{R}$ é dada por

$$y(x) = y(\bar{x}) + y'(\bar{x})(x - \bar{x}) + \frac{y''(\bar{x})(x - \bar{x})^2}{2!} + \frac{y'''(\bar{x})(x - \bar{x})^3}{3!} + o(h^4).$$

Com $\bar{x} = x_i$ e $x = x_{i+1}$, temos então

$$y(x_{i+1}) = y(x_i) + hy'(x_i) + \frac{h^2}{2}y''(x_i) + \frac{h^3}{6}y'''(x_i) + o(h^4)$$

Então:

$$y'(x_i) = \frac{y(x_{i+1}) - y(x_i)}{h} - \frac{o(h^2)}{h}.$$

Esta é uma aproximação de ordem 1 chamada de Euler Avançado.

Euler Atrasado

Com $\bar{x} = x_i$ e $x = x_{i-1}$, temos então

$$y(x_{i-1}) = y(x_i) - hy'(x_i) + o(h^2)$$

Assim,

$$y'(x_i) = \frac{y(x_i) - y(x_{i-1})}{h} + o(h)$$

e aí

$$y'_i \approx \frac{y_i - y_{i-1}}{h}$$

Esta é uma aproximação de ordem 1 chamada de Euler Atrasado.

Euler Centrado

Vimos que:

$$y(x_{i+1}) = y(x_i) + hy'(x_i) + \frac{h^2}{2}y''(x_i) + o(h^3)$$

e

$$y(x_{i-1}) = y(x_i) - hy'(x_i) + \frac{h^2}{2}y''(x_i) - o(h^3)$$

Subtraindo uma da outra, temos:

$$y(x_{i+1}) - y(x_{i-1}) = 2hy'(x_i) + o(h^3)$$

Logo,

$$y'(x_i) = \frac{y(x_{i+1}) - y(x_{i-1})}{2h} - o(h^2).$$

e aí

$$y'_i \approx \frac{y_{i+1} - y_{i-1}}{2h}.$$

Esta é uma aproximação de ordem 2 para a primeira derivada e é chamada de Euler Centrado.

Segunda Derivada

Para a derivada de ordem 2, vamos considerar as Séries de Taylor já apresentadas anteriormente

$$y(x_{i+1}) = y(x_i) + hy'(x_i) + \frac{h^2}{2}y''(x_i) + \frac{h}{3!}y'''(\bar{x}) + o(h^4)$$

e

$$y(x_{i-1}) = y(x_i) - hy'(x_i) + \frac{h^2}{2}y''(x_i) - \frac{h^3}{3!}y'''(\bar{x}) + o(h^4).$$

Somando as expressões acima termo a termo, obtemos

$$y(x_{i+1}) + y(x_{i-1}) = 2y(x_i) + h^2y''(x_i) + o(h^4).$$

Logo,

$$y''(x_i) = \frac{y(x_{i+1}) - 2y(x_i) + y(x_{i-1}))}{h^2} - o(h^2),$$

Assim,

$$y_i'' \approx \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2},$$

que é uma aproximação de ordem 2 para a segunda derivada.

Resolvendo o problema discreto

Substituindo a primeira derivada y_i' pela Diferença Centrada e a segunda derivada y_i'' por sua aproximação de ordem 2 na forma discreta do PVC, obtemos, considerando $y_i \approx y(x_i)$,

$$y_i'' + Ay_i' + By_i = g(x_i) \Rightarrow \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + A\frac{y_{i+1} - y_{i-1}}{2h} + By_i = g(x_i),$$

para $i = 1, 2, \dots, n$.

Simplificando, obtemos

$$(1 - A\frac{h}{2})y_{i-1} + (Bh^2 - 2)y_i + (1 + A\frac{h}{2})y_{i+1} = h^2g(x_i),$$

para $i = 1, 2, \dots, n$.

Na primeira equação, $i = 1$, aparece y_0 que, no caso do problema, é igual ao valor dado \bar{y}_0 . Assim, a primeira equação é dada por

$$(Bh^2 - 2)y_1 + (1 + A\frac{h}{2})y_2 = h^2g(x_1) - (1 - A\frac{h}{2})\bar{y}_0.$$

Na última equação, $i = n$, aparece y_{n+1} que, neste caso, é um valor dado e igual a \bar{y}_1 . Logo, a última equação é dada por

$$(1 - A\frac{h}{2})y_{n-1} + (Bh^2 - 2)y_n = h^2g(x_n) - (1 + A\frac{h}{2})\bar{y}_1.$$

Assim, obtemos o sistema linear com n equações, dado por

$$\begin{cases} (Bh^2 - 2)y_1 + (1 + A\frac{h}{2})y_2 = h^2g(x_1) - (1 - A\frac{h}{2})\bar{y}_0 \\ (1 - A\frac{h}{2})y_{i-1} + (Bh^2 - 2)y_i + (1 + A\frac{h}{2})y_{i+1} = h^2g(x_i), \quad i = 2, 3, \dots, n-1 \\ (1 - A\frac{h}{2})y_{n-1} + (Bh^2 - 2)y_n = h^2g(x_n) - (1 + A\frac{h}{2})\bar{y}_1 \end{cases}$$

Ao resolver o sistema linear acima, obtemos os valores y_i , que são as aproximações para $y(x_i)$.

2.6 SISTEMAS LINEARES

Seja \mathbf{A} uma matriz $n \times n$ e \mathbf{b} um vetor $n \times 1$. Queremos determinar \mathbf{x} , $n \times 1$, tal que $\mathbf{Ax} = \mathbf{b}$, se $\det(\mathbf{A}) \neq 0$.

2.6.1 Método da eliminação Gaussiana

O método da eliminação Gaussiana, também chamado de método da adição, consiste em:

1. Substitua a segunda linha pelo resultado da soma desta segunda linha com a primeira linha multiplicada por um fator conveniente, de forma que o primeiro elemento da segunda linha fique zerado. Atenção, tudo que for feito nas linhas da matriz \mathbf{A} tem que ser feito com os elementos do vetor \mathbf{b} .
2. Repetir o mesmo processo com a terceira linha, quarta linha, e assim por diante até a última linha, de forma que todos os elementos da primeira coluna abaixo da diagonal principal fiquem zerados.
3. Você fará o mesmo que fez com a primeira coluna, mas agora com a segunda coluna, ou seja, zerar todos os elementos da segunda coluna que estão abaixo da diagonal principal. Isso será feito substituindo cada linha a partir da terceira por ela mesma subtraída da segunda linha multiplicada por um fator conveniente.
4. Repetir todo esse processo até a penúltima coluna. O resultado final será uma matriz em que todos os elementos abaixo da diagonal principal são iguais a zero, ou seja, uma matriz triangular superior.

As operações entre as linhas que foram descritas acima são chamadas de operações válidas, uma vez que, quando realizadas entre as equações do sistema linear (ou seja, entre as linhas da matriz \mathbf{A} e do vetor \mathbf{b}) não alteram o resultado do sistema linear.

Resumindo, as operações válidas entre as equações de um sistema linear (ou entre as linhas da matriz aumentada $[\mathbf{A}|\mathbf{b}]$ são:

- Trocar duas linhas de posição, ou seja, $L_i \leftrightarrow L_j$.
- Multiplicar uma linha por um escalar diferente de zero, ou seja, $L_i \leftarrow \alpha L_i$, com $\alpha \neq 0$.
- Substituir uma linha pela soma desta linha com outra, ou seja, $L_i \leftarrow L_i + L_j$.

Essas operações são então aplicadas para transformar a matriz aumentada $[\mathbf{A}|\mathbf{b}]$ em uma matriz $[\overline{\mathbf{A}}|\overline{\mathbf{b}}]$ tal que $\overline{\mathbf{A}}$ é uma matriz triangular superior. Assim, obtemos o sistema linear equivalente ao original dado por $\overline{\mathbf{A}}\mathbf{x} = \overline{\mathbf{b}}$, cuja sol pode ser obtida pelo Método da Retro-substituição, descrito a seguir.

2.6.2 Algoritmo da Retro-Substituição

Uma vez que você já aplicou a eliminação Gaussiana, é possível resolver o Sistema Linear por meio da retro-substituição. Como a última linha da matriz \mathbf{A} (ou seja, a linha n , equivalente à n -ésima equação) possui um único elemento que não está zerado, o elemento a_{nn} , é possível descobrir o valor de x_n . Depois, passaremos para a linha superior, $n - 1$, onde há 2 elementos, $a_{n-1,n-1}$ e $a_{n-1,n}$, mas o elemento x_n nós já sabemos o valor, então podemos determinar x_{n-1} . Esse processo se repetirá até a primeira linha e, com isso, acharemos todas as incógnitas.

Abaixo segue o algoritmo da retro-substituição:

Algorithm 1 Resolução de Sistema Linear Triangular Superior

Entrada: Matriz \mathbf{A} de dimensão $n \times n$ e triangular superior, vetor b com n elementos.

Saída: Vetor x tal que $\mathbf{Ax} = \mathbf{b}$.

1. Calcule $x_n \leftarrow \frac{b_n}{a_{nn}}$
 2. **Para** $i \leftarrow n - 1$ **até** 1 **faça:**
 - a) Defina Soma $\leftarrow 0$
 - b) **Para** $j \leftarrow i + 1$ **até** n **faça:**
 - i. Atualize Soma \leftarrow Soma $-a_{ij} \times x_j$
 - c) Calcule $x_i \leftarrow \frac{b_i - \text{Soma}}{a_{ii}}$
-

O método da substituição é idêntico ao da retro-substituição, mas a entrada é uma matriz triangular inferior e as substituições começam a partir da primeira linha e vão descendo, ao invés de começar da última linha e ir subindo.

2.6.3 Fatoração LU

Dado o sistema linear $\mathbf{Ax} = \mathbf{b}$, com $\det(\mathbf{A}) \neq 0$, fatoramos a matriz \mathbf{A} na forma $\mathbf{A} = \mathbf{LU}$, onde \mathbf{L} (do inglês *lower*) é uma matriz triangular inferior com 1 na diagonal e \mathbf{U} (do inglês *upper*) é triangular superior. Assim, $\mathbf{Ax} = \mathbf{b}$ se torna $\mathbf{LUx} = \mathbf{b}$. Se considerarmos $\mathbf{y} = \mathbf{Ux}$, temos $\mathbf{Ly} = \mathbf{b}$. Dessa forma, obtemos dois sistemas lineares:

$$\mathbf{Ly} = \mathbf{b}$$

$$\mathbf{Ux} = \mathbf{y}$$

Resolvemos o sistema linear $\mathbf{Ly} = \mathbf{b}$ usando o método da substituição, obtendo o vetor \mathbf{y} . Com este vetor, resolvemos o sistema linear $\mathbf{Ux} = \mathbf{y}$ com o método da retro-substituição.

Obs: A fatoração LU faz mais operações do que a eliminação Gaussiana. Porém, a fatoração LU tem mais vantagens quando precisamos resolver vários sistemas lineares, todos com a mesma matriz \mathbf{A} , pois a fatoração LU é feita apenas uma vez.

2.6.4 Como determinar LU

Para ilustrar a fatoração, vamos considerar \mathbf{A} e \mathbf{b} , respectivamente, matriz 3×3 e vetor 3×1 , dados por

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

e

$$\mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

Na eliminação Gaussiana, para zerar o elemento a_{21} fazemos $m_{21} = \frac{a_{21}}{a_{11}}$ e, em seguida, $L_2 \leftarrow L_2 - m_{21}L_1$, e para zerar o elemento a_{31} , fazemos $m_{31} = \frac{a_{31}}{a_{11}}$ e $L_3 \leftarrow L_3 - m_{31}L_1$, sendo L_1, L_2 e L_3 as linhas da matriz aumentada $[\mathbf{A}|\mathbf{b}]$.

Após essas operações, obtemos

$$\mathbf{A}_1 = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & \bar{a}_{22} & \bar{a}_{23} \\ 0 & \bar{a}_{32} & \bar{a}_{33} \end{bmatrix}$$

Se definirmos a matriz

$$\mathbf{M}_1 = \begin{bmatrix} 1 & 0 & 0 \\ -m_{31} & 1 & 0 \\ -m_{31} & 0 & 1 \end{bmatrix}$$

então $\mathbf{M}_1\mathbf{A} = \mathbf{A}_1$, ou seja, a aplicação da matriz \mathbf{M}_1 sobre a matriz \mathbf{A}_1 resulta na matriz obtida pela eliminação dos elementos da primeira coluna, da mesma forma que obtida pela Eliminação de Gauss.

Tendo \mathbf{A}_1 , determinamos $m_{32} = \frac{\bar{a}_{32}}{\bar{a}_{22}}$ (agora estamos utilizando os elementos de \mathbf{A}_1 , não mais de \mathbf{A}). Repetindo o procedimento anterior,

$$\mathbf{M}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -m_{32} & 1 \end{bmatrix}$$

e então

$$\mathbf{M}_2\mathbf{A}_1 = \mathbf{A}_2 = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & \bar{a}_{22} & \bar{a}_{23} \\ 0 & 0 & \bar{a}_{33} \end{bmatrix}$$

Assim, $\mathbf{M}_2\mathbf{M}_1\mathbf{A} = \mathbf{A}_2 \Rightarrow \mathbf{A} = \mathbf{M}_2^{-1}\mathbf{M}_1^{-1}\mathbf{A}_2$.

Mas

$$\mathbf{M}_1^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ m_{31} & 1 & 0 \\ m_{31} & 0 & 1 \end{bmatrix},$$

$$\mathbf{M}_2^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & m_{32} & 1 \end{bmatrix}$$

e

$$\mathbf{M}_1^{-1}\mathbf{M}_2^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ m_{31} & 1 & 0 \\ m_{31} & m_{32} & 1 \end{bmatrix}.$$

Como a matriz acima é uma matriz triangular inferior com 1 na diagonal, vamos chamá-la de \mathbf{L} . Como a matriz \mathbf{A}_2 é uma matriz triangular superior, vamos chamá-la de \mathbf{U} .

Assim, chegamos a $\mathbf{A} = \mathbf{M}_2^{-1}\mathbf{M}_1^{-1}\mathbf{A}_2 \Rightarrow \mathbf{A} = \mathbf{LU}$, com

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 \\ m_{31} & 1 & 0 \\ m_{31} & m_{32} & 1 \end{bmatrix}.$$

a matriz que armazena os fatores multiplicativos do Método de Eliminação de Gauss e

$$\mathbf{U} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & \bar{a}_{22} & \bar{a}_{23} \\ 0 & 0 & \bar{a}_{33} \end{bmatrix}$$

a matriz resultante da aplicação do Método de Eliminação de Gauss em \mathbf{A} .

Obs: $\det(\mathbf{A}) = \det(\mathbf{LU}) = \det(\mathbf{L})\det(\mathbf{U})$. Como $\det(\mathbf{L}) = 1$ (pois é triangular com 1 na diagonal), $\det(\mathbf{A}) = \det(\mathbf{U})$.

Os métodos da eliminação Gaussiana e fatoração LU são classificados como métodos diretos, ou seja, após uma quantidade finita de operações matemáticas (que depende da dimensão do sistema linear) o método fornece a solução exata. Na prática, só não obtemos o valor exato porque, ao usar uma máquina para realizar os cálculos, há erros de arredondamento.

2.6.5 Métodos Iterativos

Nesta classe de métodos, os algoritmos fornecem, a cada iteração, um vetor. A sequência de vetores formada, a partir de um vetor inicial $\mathbf{x}^{(0)}$, pode ou não convergir para a solução do sistema linear.

2.6.6 Método de Gauss-Jacobi

Vamos considerar um sistema linear 3×3 , dado por:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 = b_3 \end{cases}$$

com solução única, ou seja, $\det(\mathbf{A}) \neq 0$.

Da primeira equação, temos $x_1 = \frac{b_1 - (a_{12}x_2 + a_{13}x_3)}{a_{11}}$ se $a_{11} \neq 0$.

Da segunda equação, temos $x_2 = \frac{b_2 - (a_{21}x_1 + a_{23}x_3)}{a_{22}}$ se $a_{22} \neq 0$.

E da terceira equação, $x_3 = \frac{b_3 - (a_{31}x_1 + a_{32}x_2)}{a_{33}}$ se $a_{33} \neq 0$.

Assim, dado um vetor $\mathbf{x}^{(k)}$, determinamos o vetor $\mathbf{x}^{(k+1)}$ fazendo

$$\begin{cases} x_1^{(k+1)} = \frac{b_1 - (a_{12}x_2^{(k)} + a_{13}x_3^{(k)})}{a_{11}} \\ x_2^{(k+1)} = \frac{b_2 - (a_{21}x_1^{(k)} + a_{23}x_3^{(k)})}{a_{22}} \\ x_3^{(k+1)} = \frac{b_3 - (a_{31}x_1^{(k)} + a_{32}x_2^{(k)})}{a_{33}} \end{cases}$$

Assim, a partir de um vetor inicial qualquer $\mathbf{x}^{(0)}$, geramos uma sequência de vetores $\{\mathbf{x}^{(k)}\}$, $k = 0, 1, 2, \dots$ que, sob certas condições, converge para a solução do sistema linear.

Como exemplo, considere o sistema linear

$$\begin{cases} 2x_1 - x_2 = 1 \\ x_1 + 3x_2 = 4 \end{cases}$$

Temos então, para este sistema,

$$x_1^{k+1} = \frac{1 + x_2^k}{2}$$

$$x_2^{k+1} = \frac{4 - x_1^k}{3}$$

com $k = 0, 1, 2, \dots$

Considerando

$$\mathbf{x}^0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

temos

$$\mathbf{x}^1 = \begin{bmatrix} \frac{1}{2} \\ \frac{4}{3} \end{bmatrix},$$

$$\mathbf{x}^2 = \begin{bmatrix} \frac{1 + \frac{4}{3}}{2} \\ \frac{4 - \frac{1}{2}}{3} \end{bmatrix} = \begin{bmatrix} \frac{7}{6} \\ \frac{7}{6} \end{bmatrix},$$

$$\mathbf{x}^3 = \begin{bmatrix} \frac{1 + \frac{7}{6}}{2} \\ \frac{4 - \frac{7}{6}}{3} \end{bmatrix} = \begin{bmatrix} \frac{13}{12} \\ \frac{17}{18} \end{bmatrix}.$$

Aparentemente, a sequência está convergindo para $x_1 = 1$ e $x_2 = 1$.

Porém, se considerarmos agora o sistema linear equivalente ao anterior, obtido apenas com a troca de posição das equações, ou seja,

$$\begin{cases} x_1 + 3x_2 = 4 \\ 2x_1 - x_2 = 1 \end{cases},$$

o método nos fornece as expressões

$$x_1^{k+1} = 4 - 3x_2^k$$

$$x_2^{k+1} = 2x_1^k - 1$$

com $k = 0, 1, 2, \dots$

iniciando com o mesmo vetor inicial obtemos então

$$\mathbf{x}^0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

$$\mathbf{x}^1 = \begin{bmatrix} 4 \\ -1 \end{bmatrix},$$

$$\mathbf{x}^2 = \begin{bmatrix} 4 - 3(-1) \\ 2 \times 4 - 1 \end{bmatrix} = \begin{bmatrix} 7 \\ 7 \end{bmatrix},$$

$$\mathbf{x}^3 = \begin{bmatrix} 4 - 3 \times 7 \\ 2 \times 7 - 1 \end{bmatrix} = \begin{bmatrix} -17 \\ 13 \end{bmatrix}.$$

Dessa vez, a sequência gerada não parece convergir para a solução.

2.6.7 Critério de convergência do Método G-J

Para que o método de Gauss-Jacobi funcione, a matriz A do S.L precisa Estritamente Diagonalmente Dominante. O que é isso? O maior elemento, em módulo da linha precisa fazer parte da diagonal principal e, além disso, o somatório de todos os outros elementos da linha tem que ser menor, em módulo, que o elemento dominante.

Assim, temos que ter

$$|a_{ii}| > \sum_{j=1, j \neq i}^n |a_{ij}|$$

2.6.8 Método de Gauss-Seidel

O método de Gauss-Seidel é similar ao Método de Gauss-Jacobi, mas a cada iteração, depois de calcular o x_i^{k+1} , o novo valor já entra no cálculo de x_{i+1}^{k+1} .

Assim, ainda considerando como exemplo um sistema linear 3×3 , temos

$$\begin{cases} x_1^{(k+1)} = \frac{b_1 - (a_{12}x_2^{(k)} + a_{13}x_3^{(k)})}{a_{11}} \\ x_2^{(k+1)} = \frac{b_2 - (a_{21}x_1^{(k+1)} + a_{23}x_3^{(k)})}{a_{22}} \\ x_3^{(k+1)} = \frac{b_3 - (a_{31}x_1^{(k+1)} + a_{32}x_2^{(k+1)})}{a_{33}} \end{cases}$$

Repare que, ao calcular $x_2^{(k+1)}$ já temos o novo valor do primeiro elemento, ou seja, $x_1^{(k+1)}$. Este valor é então utilizado ainda na mesma iteração k .

O aproveitamento dos novos valores calculados na mesma interação tem como objetivo aumentar a velocidade de convergência, caso a convergência ocorra.

Vamos comnsiderar o mesmo exemplo anterior

$$\begin{cases} 2x_1 - x_2 = 1 \\ x_1 + 3x_2 = 4 \end{cases}$$

Temos então, para este sistema,

$$x_1^{k+1} = \frac{1 + x_2^k}{2}$$

$$x_2^{k+1} = \frac{4 - x_1^{k+1}}{3}$$

com $k = 0, 1, 2, \dots$

Considerando

$$\mathbf{x}^0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

temos

$$\mathbf{x}^1 = \begin{bmatrix} \frac{1}{2} \\ \frac{4}{3} \end{bmatrix},$$

que é o mesmo vetor determinado pelo método G-J,

$$\mathbf{x}^2 = \begin{bmatrix} \frac{1+\frac{4}{3}}{2} \\ \frac{4-x_1^2}{3} \end{bmatrix} = \begin{bmatrix} \frac{7}{6} \\ \frac{4-\frac{7}{6}}{3} \end{bmatrix} = \begin{bmatrix} \frac{7}{6} \\ \frac{17}{18} \end{bmatrix},$$

$$\mathbf{x}^3 = \begin{bmatrix} \frac{1+\frac{17}{18}}{2} \\ \frac{4-x_1^3}{3} \end{bmatrix} = \begin{bmatrix} \frac{35}{36} \\ \frac{4-\frac{35}{36}}{3} \end{bmatrix} = \begin{bmatrix} \frac{35}{36} \\ \frac{109}{105} \end{bmatrix}.$$

Aparentemente, a sequência está convergindo para a solução $x_1 = 1$ e $x_2 = 1$, porém mais rapidamente que o método G-J.

É fácil verificar que se considerarmos o sistema linear equivalente

$$\begin{cases} x_1 + 3x_2 = 4 \\ 2x_1 - x_2 = 1 \end{cases},$$

a sequência gerada pelo método G-S também não será convergente.

2.6.9 Critério de convergência para o método G-S

Agora, não basta apenas conferir se a matriz é diagonalmente dominante. No método de Gauss-Seidel, além da matriz ser diagonalmente dominante, também precisa atender ao critério de Sassenfeld:

O critério de Sassenfeld consiste em, uma vez que a matriz do sistema linear for diagonalmente dominante (pois se não for já nem passa do primeiro critério), calcular os $\beta_1, \beta_2, \dots, \beta_n$ da matriz, sendo β_1 a razão do somatório dos módulos de todos os elementos da linha que não fazem parte da diagonal principal dividido pelo elemento da diagonal em módulo, ou seja,

$$\beta_1 = \frac{\sum_{j=2}^n |a_{1j}|}{|a_{11}|}$$

Já β_2 é similar, porém o elemento da coluna 1 será multiplicado por β_1 , ou seja,

$$\beta_2 = \frac{\beta_1 |a_{21}| + \sum_{j=3}^n |a_{2j}|}{|a_{22}|}$$

Por sua vez, β_3 é determinado de forma análoga, com os dois elementos da matriz à esquerda da diagonal sendo multiplicados, respectivamente, por β_1 e β_2 , ou seja,

$$\beta_3 = \frac{\beta_1|a_{31}| + \beta_2|a_{32}| + \sum_{j=4}^n |a_{3j}|}{|a_{33}|}$$

Assim, de um modo geral, temos então

$$\beta_i = \frac{\sum_{j=1}^{i-1} \beta_j |a_{ij}| + \sum_{j=i+1}^n |a_{ij}|}{|a_{ii}|}, \quad i = 1, 2, \dots, n.$$

Se $\beta = \max_{i=1,2,\dots,n} \{\beta_i\} \leq 1$ então a sequência gerada pelo método G-S converge para a solução do sistema linear.

2.6.10 Teste de parada para os métodos iterativos

Vamos supor que um dos métodos acima, G-J ou G-S, gerem uma sequência $\{\mathbf{x}^{(k)}\}$ convergente para a solução de um sistema linear.

Assim, para cada k , temos

$$\mathbf{Ax}^{(k)} \neq \mathbf{b}$$

Seja então o vetor resíduo dado por $\mathbf{r}^k = \mathbf{b} - \mathbf{Ax}^k$. Assim, dado $0 < \epsilon \ll 1$, calculamos

$$\|\mathbf{r}^k\| = \|\mathbf{b} - \mathbf{Ax}^k\|$$

para alguma norma $\|\cdot\|$ de vetores. Se $\|\mathbf{r}^k\| < \epsilon$, encontramos uma solução \mathbf{x}^k aproximada para $\mathbf{Ax} = \mathbf{b}$.

Outro teste de parada bastante utilizado consiste em comparar duas soluções aproximadas consecutivas \mathbf{x}^k e \mathbf{x}^{k+1} , ou seja,

$$\frac{\|\mathbf{x}^{k+1} - \mathbf{x}^k\|}{\|\mathbf{x}^k\|} < \epsilon$$

Os critérios acima são válidos quando a sequência gerada converge. Para evitar cálculos desnecessários no caso da divergência, é conveniente incluir como teste de parada um número máximo de iterações, ou seja, o algoritmo para quando

$$k > N_{max}$$

, onde N_{max} é uma quantidade previamente definida.

2.6.11 Número de condicionamento

No começo do documento foi dito que erros na obtenção de dados a serem tratados numericamente podem comprometer todo o sistema, caso o sistema não tenha um bom número de condicionamento.

O número de condicionamento é uma medida que indica se um sistema linear tem boas condições para ser tratado numericamente. Como sabemos, os métodos numéricos têm muitas estimativas e aproximações e, quanto maior for o número de condicionamento, maior será a diferença de solução para qualquer pequena mudança nos valores dos elementos da matriz ou do vetor do sistema linear.

O número de condicionamento pode ser calculado da seguinte forma como

$$k(a) = \|A^{-1}\| \cdot \|A\|$$

, onde A é a matriz do sistema linear.

2.7 INTERPOLAÇÃO POLINOMIAL

Sejam $n + 1$ pontos $(x_i, f(x_i))$, $i = 1, 2, 3, \dots, n$ com $x_i < x_{i+1}$. Queremos encontrar um polinômio que interpole esses pontos, ou seja, o gráfico de polinômio deve passar pelos pontos dados. Obs: Com $n + 1$ pontos, o polinômio P_n tem grau menor ou igual a n .

2.7.1 Forma Direta

Estamos interessados em, dado um conjunto de pontos $(x_i, f(x_i))$, com $i = 0, 1, 2, \dots, n$ e $x_0 < x_1 < \dots < x_n$, determinar um polinômio que interpola esse conjunto de pontos, ou seja, determinar $p(x)$ tal que $p(x_i) = f(x_i)$ para todo i .

Como o conjunto acima possui $n + 1$ pontos, podemos determinar um polinômio de grau menor ou igual a n , uma vez que um polinômio de grau n possui $n + 1$ coeficientes a serem determinados.

Dessa forma, seja

$$p_n(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$$

Como queremos que este polinômio interpole os pontos dados, temos então

$$p_n(x_0) = a_0 + a_1x_0 + a_2x_0^2 + \dots + a_nx_0^n = f(x_0)$$

$$p_n(x_1) = a_0 + a_1x_1 + a_2x_1^2 + \dots + a_nx_1^n = f(x_1)$$

$$\vdots$$

$$p_n(x_n) = a_0 + a_1x_n + a_2x_n^2 + \dots + a_nx_n^n = f(x_n)$$

$$\begin{bmatrix} 1 & x_0 & x_0^2 & \dots & x_0^n \\ 1 & x_1 & x_1^2 & \dots & x_1^n \\ 1 & x_2 & x_2^2 & \dots & x_2^n \\ \dots & \dots & \dots & \dots & \dots \\ 1 & x_n & x_n^2 & \dots & x_n^n \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \dots \\ a_n \end{bmatrix} = \begin{bmatrix} f(x_0) \\ f(x_1) \\ f(x_2) \\ \dots \\ f(x_n) \end{bmatrix} \quad (2.1)$$

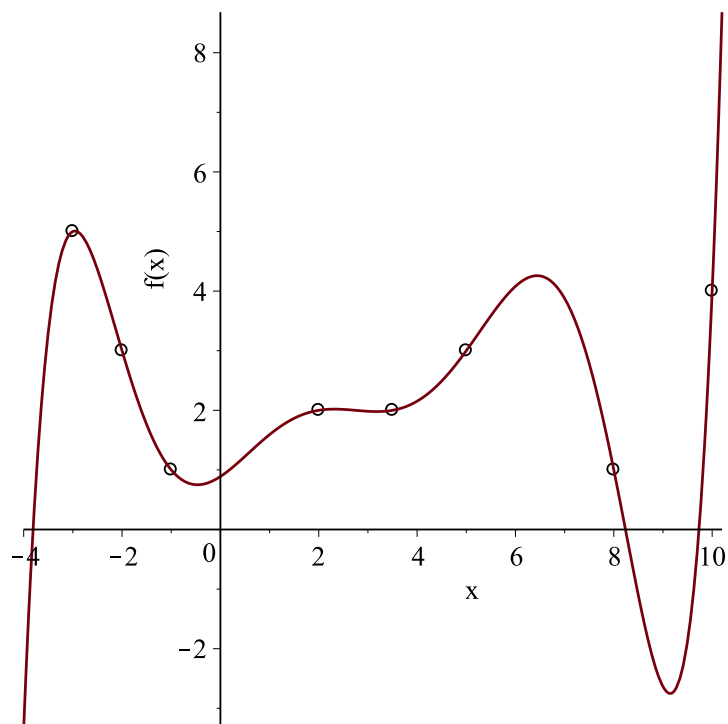


Figura 4 – Exemplo de interpolação de 8 pontos. Repare que o gráfico do polinômio passa por todos os pontos.

A matriz do sistema linear acima é uma matriz de Vandermonde e, como $x_i \neq x_j$ sempre que $i \neq j$, seu determinante é diferente de zero. Logo, o sistema linear tem solução única.

Assim, resolvendo o sistema linear determinam-se unicamente os coeficientes a_0, a_1, \dots, a_n do polinômio interpolador $p_n(x)$ que, dessa forma, também é único.

Esta forma de determinar o polinômio interpolador é chamada de forma direta.

Exemplo 2.1 : encontrar o polinômio interpolador dos pontos $(0, 0), (2, 3), (3, 1)$.

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 3 \\ 1 \end{bmatrix}$$

Resolvendo, temos $a_0 = 0, a_1 = 23/6$ e $a_2 = -7/6$. Logo, $p_2(x) = -7/6x^2 + 23/6x$

Outra solução: sabemos que o gráfico do polinômio interpolador passa pela origem, logo $a_0 = 0$. Utiliza-se os dois outros pontos para determinar o polinômio $p_2(x) = a_1x + a_2x^2$. Assim,

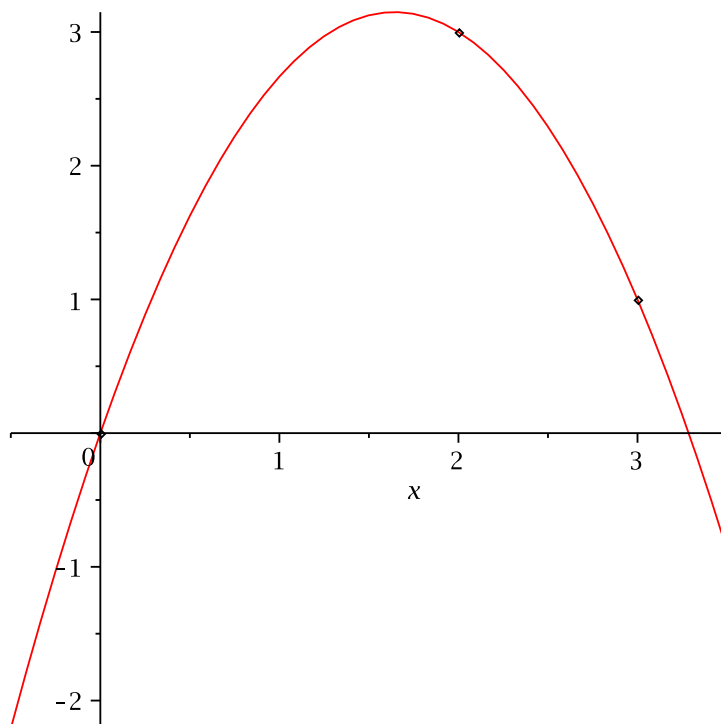


Figura 5 – Gráfico do polinômio interpolador do exemplo.

$$\begin{bmatrix} 2 & 4 \\ 3 & 9 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$$

cuja solução fornece os coeficientes $a_1 = 23/6$ e $a_2 = -7/6$, ou seja, obtemos o mesmo polinômio $p_2(x) = -7/6x^2 + 23/6x$

2.7.2 Forma de Lagrange

Dado um conjunto de pontos $(x_i, f(x_i))$, com $i = 0, 1, 2, \dots, n$ e $x_0 < x_1 < \dots < x_n$, a ideia é representar o polinômio interpolador $p_n(x)$ na forma

$$p_n(x) = f(x_0)L_0(x) + f(x_1)L_1(x) + f(x_2)L_2(x) + \dots + f(x_n)L_n(x)$$

onde $L_i(x)$, $i = 0, 1, 2, \dots, n$ são polinômios de grau menor ou igual a n .

Nota-se que, como $p_n(x)$ é o polinômio interpolador (único, como visto em 2.1), então $p_n(x_0) = f(x_0)$ e aí $L_0(x_0) = 1$ e $L_j(x_0) = 0$, $j = 1, 2, \dots, n$.

Além disso, $p_n(x_1) = f(x_1)$ e aí $L_1(x_1) = 1$ e $L_j(x_1) = 0$, $j = 0, 2, \dots, n$.

Generalizando para todos os pontos do conjunto dado, temos então

$$\begin{cases} L_i(x_i) = 1 \\ L_j(x_i) = 0 \quad \forall j \neq i \end{cases} \quad (2.2)$$

para $i = 0, 1, 2, \dots, n$.

Definem-se então os polinômios de Lagrange como

$$L_i(x) = \frac{(x - x_0)(x - x_1)\dots(x - x_{i-1})(x - x_{i+1})\dots(x - x_n)}{(x_i - x_0)(x_i - x_1)\dots(x_i - x_{i-1})(x_i - x_{i+1})\dots(x_i - x_n)} \quad (2.3)$$

para $i = 0, 1, 2, \dots, n$.

Exemplo 2.2 : encontrar o polinômio interpolador dos pontos $(0, 0), (2, 3), (3, 1)$ pela Forma de Lagrange.

O polinômio interpolador é tal que $p_2(x) = f(x_0)L_0(x) + f(x_1)L_1(x) + f(x_2)L_2(x)$.

Substituindo, temos $p_2(x) = 3L_1(x) + L_2(x)$, com

$$L_1(x) = \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} = \frac{(x - 0)(x - 3)}{(2 - 0)(2 - 3)} = \frac{-x^2 + 3x}{2}$$

$$L_2(x) = \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)} = \frac{(x - 0)(x - 2)}{(3 - 0)(3 - 2)} = \frac{x^2 - 2x}{3}$$

Logo, $p_2(x) = 3\frac{-x^2+3x}{2} + \frac{x^2-2x}{3} = -7/6x^2 + 23/6x$, como era esperado.

2.7.3 Forma de Newton

Na forma de Newton o polinômio interpolador dos $n+1$ pontos $(x_i, f(x_i))$, com $i = 0, 1, 2, \dots, n$ e $x_0 < x_1 < \dots < x_n$, é dado por

$$p_n(x) = d_0 + d_1(x - x_0) + d_2(x - x_0)(x - x_1) + \dots + d_n(x - x_0)(x - x_1)\dots(x - x_{n-1})$$

Os coeficientes $d_i, i = 0, 1, \dots, n$ serão deduzidos com o auxílio das diferenças divididas.

Definição: Diferenças Divididas

As diferenças divididas são definidas de forma recursiva da seguinte forma:

Diferença dividida de ordem zero: $f[x_0] = f(x_0)$

Diferença dividida de ordem um: $f[x_0, x_1] = \frac{f[x_1] - f[x_0]}{x_1 - x_0} = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$

Diferença dividida de ordem dois: $f[x_0, x_1, x_2] = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0}$

Diferença dividida de ordem n : $f[x_0, x_1, x_2, \dots, x_{n-1}, x_n] = \frac{f[x_1, x_2, \dots, x_n] - f[x_0, x_1, \dots, x_{n-1}]}{x_n - x_0}$

Uma propriedade importante das diferenças divididas é a indiferença em relação a permutação dos pontos, ou seja, $f[P_1] = f[P_2]$, onde P_1 e P_2 são quaisquer permutações dos pontos (x_1, x_2, \dots, x_n) . Para ilustrar esta propriedade, a diferença dividida de ordem um é tal que

$$f[x_0, x_1] = \frac{f[x_1]-f[x_0]}{x_1-x_0} = \frac{f(x_1)-f(x_0)}{x_1-x_0} = \frac{f(x_0)-f(x_1)}{x_0-x_1} = \frac{f[x_0]-f[x_1]}{x_0-x_1} = f[x_1, x_0]$$

Deve ser citado também que a diferença dividida de ordem n é uma aproximação para a derivada de ordem n de $f(x)$. Ainda considerando a diferença dividida de ordem um, vemos que

$$f[x_0, x_1] = \frac{f[x_1]-f[x_0]}{x_1-x_0} = \frac{f(x_1)-f(x_0)}{x_1-x_0} \cong \frac{df(x)}{dx} \text{ para algum } x \in [x_0, x_1]$$

De outra forma,

$$\lim_{x_1 \rightarrow x_0} f[x_0, x_1] = \lim_{x_1 \rightarrow x_0} \frac{f[x_1]-f[x_0]}{x_1-x_0} = \lim_{x_1 \rightarrow x_0} \frac{f(x_1)-f(x_0)}{x_1-x_0} = \frac{df}{dx}(x_0)$$

Esse raciocínio pode ser estendido de forma que

$$f[x_0, x_1, x_2, \dots, x_{n-1}, x_n] \cong \frac{d^n f(x)}{dx^n} \text{ para algum } x \in [x_0, x_n]$$

De fato, temos o seguinte teorema:

Teorema 2.1 *sejam $x_0 < x_1 < x_2 < \dots < x_n$ $n+1$ pontos. Seja $f(x)$ com derivadas de ordem até $n+1$ para todo x em $[x_0, x_n]$. Então dado um $x \in (x_0, x_n)$ qualquer, existe $\xi \in (x_0, x_n)$ tal que*

$$f[x_0, x_1, x_2, \dots, x_n, x] = \frac{f^{(n+1)}(\xi_x)}{(n+1)!}, \quad x \in (x_0, x_n).$$

OBS.: o índice x para ξ se justifica pois, neste caso, ξ depende de x . Repare que o teorema nada fala sobre este ξ , apenas que ele existe.

Dado um conjunto de pontos $(x_i, f(x_i))$, com $i = 0, 1, 2, \dots, n$ e $x_0 < x_1 < \dots < x_n$, constrói-se uma tabela chamada tabela de diferenças divididas, como pode ser visto no exemplo 2.3.

Exemplo 2.3 : *Construir a tabela d ediferenças divididas para o conjunto de pontos $(0, 0), (2, 3), (3, 1)$*

x_i	$f[x_i] = f(x_i)$	ordem1	ordem2
0	$f[x_0] = 0$	$f[x_0, x_1] = 3/2$	$f[x_0, x_1, x_2] = -7/6$
2	$f[x_1] = 3$	$f[x_1, x_2] = -2$	—
3	$f[x_2] = 1$	—	—

Para determinar o polinômio interpolador, inicialmente consideramos o polinômio interpolador de grau zero para $f(x)$ que passa pelo ponto $(x_0, f(x_0))$. Este polinômio é facilmente definido por

$$p_0(x) = f(x_0) = f[x_0]$$

Para todo $x \in [x_0, x_n]$ temos

$$f[x_0, x] = \frac{f(x) - f(x_0)}{x - x_0},$$

ou seja,

$$f(x) - f(x_0) = (x - x_0)f[x_0, x]$$

e aí

$$f(x) - p_0(x) = (x - x_0)f[x_0, x].$$

Dessa forma, o erro obtido ao aproximar a função $f(x)$ pelo polinômio de grau zero que interpola o ponto $(x_0, f(x_0))$ é dado por

$$E_0(x) = f(x) - p_0(x) = (x - x_0)f[x_0, x], \quad \forall x \in [x_0, x_n].$$

A reta que interpola $f(x)$ pelos pontos $(x_0, f(x_0))$ e $(x_1, f(x_1))$ é dada por

$$p_1(x) = f(x_0) + (x - x_0) \frac{f(x_1) - f(x_0)}{x_1 - x_0},$$

ou seja,

$$p_1(x) = p_0(x) + (x - x_0)f[x_0, x_1] = f[x_0] + (x - x_0)f[x_0, x_1].$$

Queremos encontrar uma expressão para o erro, ou seja, para $E_1(x) = f(x) - p_1(x) = f(x) - f[x_0] - (x - x_0)f[x_0, x_1]$. De modo análogo ao que fizemos para o polinômio de grau zero acima, partiremos da definição de $f[x_0, x_1, x]$ e usaremos as propriedades das diferenças divididas.

Temos então,

$$\begin{aligned} \forall x \in [x_0, x_n], \quad f[x_0, x_1, x] &= f[x_1, x_0, x] = \frac{f[x_0, x] - f[x_1, x_0]}{x - x_1} = \frac{f[x_0, x] - f[x_0, x_1]}{x - x_1} \Rightarrow \\ (x - x_1)f[x_0, x_1, x] &= \frac{f(x) - f(x_0)}{x - x_0} - f[x_0, x_1] \Rightarrow (x - x_1)f[x_0, x_1, x] = \frac{f(x) - f(x_0) - (x - x_0)f[x_0, x_1]}{x - x_0} \Rightarrow \\ (x - x_0)(x - x_1)f[x_0, x_1, x] &= f(x) - f[x_0] - (x - x_0)f[x_0, x_1] \Rightarrow f(x) - [f[x_0] + (x - x_0)f[x_0, x_1]] = \\ &= (x - x_0)(x - x_1)f[x_0, x_1, x] \Rightarrow f(x) - p_1(x) = (x - x_0)(x - x_1)f[x_0, x_1, x] \end{aligned}$$

Logo,

$$E_1(x) = f(x) - p_1(x) = (x - x_0)(x - x_1)f[x_0, x_1, x], \quad \forall x \in [x_0, x_n].$$

Aplicando o mesmo raciocínio sucessivamente obtemos

$$p_n(x) = f[x_0] + f[x_0, x_1](x_0 - x) + f[x_0, x_1, x_2](x - x_0)(x - x_1) + \dots + f[x_0, x_1, \dots, x_n](x - x_0)(x - x_1) \dots (x - x_{n-1})$$

com o erro sendo dado por

$$E_n(x) = (x - x_0)(x - x_1) \dots (x - x_n)f[x_0, x_1, \dots, x_n, x], \quad \forall x \in [x_0, x_n]$$

Exemplo 2.4 Encontrar o polinômio interpolador dos pontos $(0, 0)$, $(2, 3)$, $(3, 1)$ pela Forma de Newton.

Dados os 3 pontos acima, o polinômio interpolador é tal que $p_2(x) = f[x_0] + f[x_0, x_1](x_0 - x) + f[x_0, x_1, x_2](x - x_0)(x - x_1)$. Logo, é preciso construir a tabela de diferenças divididas para os pontos dados. Como visto anteriormente, a tabela é dada por

x_i	$f[x_i] = f(x_i)$	ordem1	ordem2
0	$f[x_0] = 0$	$f[x_0, x_1] = 3/2$	$f[x_0, x_1, x_2] = -7/6$
2	$f[x_1] = 3$	$f[x_1, x_2] = -2$	—
3	$f[x_2] = 1$	—	—

Assim, tomando os valores da primeira linha da tabela, temos $p_2(x) = 0 + 3/2(x - 0) - 7/6(x - 0)(x - 2) = -7/6x^2 + 23/6x$ e o erro é dado por $E_2(x) = (x - 0)(x - 2)(x - 3)f[0, 2, 3, x], \forall x \in [0, 3]$.

Exemplo 2.5 Encontrar o polinômio interpolador dos pontos $(-1, -3/2)$, $(2, 3)$, $(4, 6)$ pela Forma de Newton.

Dados os 3 pontos acima, o polinômio interpolador terá grau igual ou menor do que 2, ou seja, temos que determinar os coeficientes a_0, a_1 e a_2 do polinômio $p_2(x) = a_0 + a_1x + a_2x^2$.

A tabela de diferenças divididas para estes pontos é dada por

x_i	$f[x_i] = f(x_i)$	ordem1	ordem2
-1	$-3/2$	$3/2$	0
2	3	$3/2$	—
4	6	—	—

Dessa forma, o polinômio interpolador é dado por $p_2(x) = f[x_0] + f[x_0, x_1](x_0 - x) + f[x_0, x_1, x_2](x - x_0)(x - x_1) = -3/2 + 3/2(x - (-1)) + 0(x - (-1))(x - 2) = -3/2 + 3/2(x + 1) = 3/2x$.

Por sua vez, o erro é dado por $E_2(x) = (x - (-1))(x - 2)(x - 4)f[-1, 2, 4, x] = (x + 1)(x - 2)(x - 4)f[-1, 2, 4, x]$, $\forall x \in [-1, 4]$. Supondo, por exemplo, $f(0) = 1$, temos $E_2(0) = (0 + 1)(0 - 2)(0 - 4)f[-1, 2, 4, 0] = 8f[-1, 2, 4, 0]$. Incluindo o ponto $(0, 1)$ na tabela de diferenças divididas, temos

x_i	$f(x_i)$	ordem1	ordem2	ordem3
-1	-3/2	3/2	0	1/8
2	3	3/2	1/8	-
4	6	5/4	-	-
0	1	-	-	-

Logo, o erro no ponto $x = 0$ é dado por $E_2(0) = 8f[-1, 2, 4, 0] = 8 * 1/8 = 1$, como de fato era esperado pois $p_2(0) = 0$.

Exemplo 2.6 Encontrar o polinômio interpolador dos pontos $(-3, 5), (-2, 3), (-1, 1), (2, 2), (3.5, 2), (5, 3), (8, 1), (10, 4)$ pela Forma de Newton.

Neste caso, a tabela de diferenças divididas é dada por

x_i	$f[x_i]$	ordem1	ordem2	ordem3	ordem4	ordem5	ordem6	ordem7
-3	5	-2	0	7/60	-1403/38610	1751/231660	-544/521235	872/6776055
-2	3	-2	7/12	-71/594	43/1782	-629/160380	328/521235	-
-1	1	1/3	-2/27	4/81	-11/729	7567/2084940	-	-
2	2	0	2/9	-7/81	523/21060	-	-	-
7/2	2	2/3	-8/27	197/1755	-	-	-	-
5	3	-2/3	13/30	-	-	-	-	-
8	1	3/2	-	-	-	-	-	-
10	4	-	-	-	-	-	-	-

Com os valores representados com 4 casas decimais e truncamento, a tabela é dada por

x_i	$f[x_i] = f(x_i)$	<i>ordem1</i>	<i>ordem2</i>	<i>ordem3</i>	<i>ordem4</i>	<i>ordem5</i>	<i>ordem6</i>	<i>ordem7</i>
-3	5	-2	0	0.1166	-0.0363	0.0075	-0.0010	0.0001
-2	3	-2	0.5833	-0.1195	0.0241	-0.0039	0.0006	-
-1	1	0.3333	-0.0740	0.493	-0.0150	0.0036	-	-
2	2	0	0.2222	-0.0864	0.0248	-	-	-
3.5	2	0.6666	-0.2962	0.1122	-	-	-	-
5	3	-0.6666	0.4333	-	-	-	-	-
8	1	1.5	-	-	-	-	-	-
10	4	-	-	-	-	-	-	-

Assim, o polinômio interpolador é dado por $p_7(x) = 5 - 2(x + 3) + 0(x + 3)(x + 2) + 0.1166(x + 3)(x + 2)(x + 1) - 0.0363(x + 3)(x + 2)(x + 1)(x - 2) + 0.0075(x + 3)(x + 2)(x + 1)(x - 2)(x - 3.5) - 0.001(x + 3)(x + 2)(x + 1)(x - 2)(x - 3.5)(x - 5) + 0.0001(x + 3)(x + 2)(x + 1)(x - 2)(x - 3.5)(x - 5)(x - 8)$.

Exemplo 2.7 : Dados os pontos $(-3, 5), (-2, 3), (-1, 1), (2, 2), (3.5, 2), (5, 3), (8, 1), (10, 4)$, encontre um valor aproximado para $f(0)$ utilizando um polinômio interpolador de grau 2.

Neste caso, há mais pontos na tabela do que o necessário, já que para determinarmos um polinômio interpolador de grau 2 bastam 3 pontos. Neste caso, os 3 valores de x dados na tabela mais próximos do valor $x = 0$ são $x = -2, x = -1$ e $x = 2$. Dessa forma, vamos considerar os 3 pontos $(-2, 3), (-1, 1)$ e $(2, 2)$ para determinar $p_2(x)$. Na tabela já determinada, destacamos os valores de interesse:

x_i	$f[x_i]$	<i>ordem1</i>	<i>ordem2</i>	<i>ordem3</i>	<i>ordem4</i>	<i>ordem5</i>	<i>ordem6</i>	<i>ordem7</i>
-3	5	-2	0	7/60	-1403/38610	1751/231660	-544/521235	872/6776055
-2	3	-2	7/12	-71/594	43/1782	-629/160380	328/521235	-
-1	1	1/3	-2/27	4/81	-11/729	7567/2084940	-	-
2	2	0	2/9	-7/81	523/21060	-	-	-
7/2	2	2/3	-8/27	197/1755	-	-	-	-
5	3	-2/3	13/30	-	-	-	-	-
8	1	3/2	-	-	-	-	-	-
10	4	-	-	-	-	-	-	-

Logo, utilizando os valores acima destacados temos $p_2(x) = 3 - 2(x + 2) + 7/12(x + 2)(x + 1)$, cujo gráfico pode ser visto na figura a seguir, juntamente com o gráfico do polinômio interpolador $p_7(x)$. Logo, $p_2(0) = 1/6$.

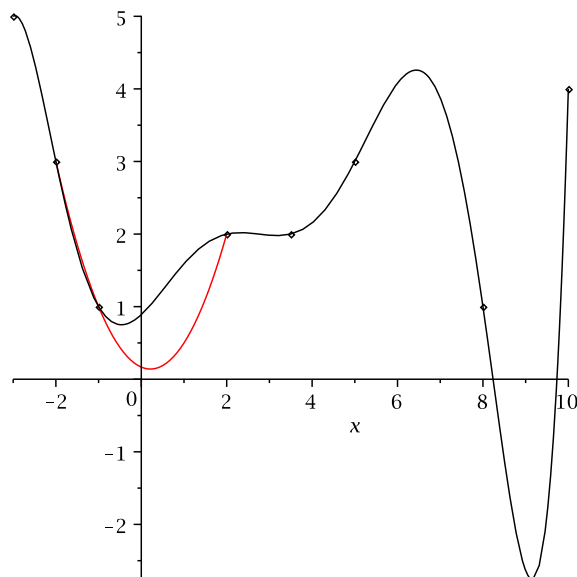


Figura 6 – Polinômios interpoladores de grau 2, em vermelho, e de grau 7 do exemplo.

Quando temos a função f que gerou as ordenadas dos pontos dados fica fácil determinar o erro, bastando para isso incluir o ponto $(\bar{x}, f(\bar{x}))$ na tabela de diferenças divididas e utilizar a definição do erro, como vimos no Exemplo 2.5. Porém, nas aplicações práticas isso normalmente não ocorre. Nesses casos, podemos usar o teorema que associa a derivada com a diferença dividida para estimar o erro.

Logo, com as condições desse teorema o erro pode ser escrito como

$$E_n(x) = (x - x_0)(x - x_1)\dots(x - x_n) \frac{f^{(n+1)}(\xi_x)}{(n+1)!}, \quad \forall x \in [x_0, x_n]$$

para algum $\xi_x \in (x_0, x_n)$.

Novamente considerando que não temos a função f (muito menos suas derivadas), podemos considerar a seguinte estimativa:

$$|E_n(x)| \leq |(x - x_0)(x - x_1)\dots(x - x_n)| \frac{M}{(n+1)!}$$

onde

$$M = \max\{f^{(n+1)}(x)\}, \quad \forall x \in [x_0, x_n]$$

Quando temos apenas os pontos discretos usamos o maior valor em módulo da diferença dividida de ordem $n+1$ para aproximar $\frac{M}{(n+1)!}$ no intervalo $[x_0, x_n]$.

Exemplo 2.8 *Dados os pontos $(-3, 5)$, $(-2, 3)$, $(-1, 1)$, $(2, 2)$, $(3.5, 2)$, $(5, 3)$, $(8, 1)$ e $(10, 4)$, encontre um valor aproximado para $f(0)$ utilizando um polinômio interpolador de grau 2 e estime o erro.*

Já vimos que, considerando os pontos $(-2, 3)$, $(-1, 1)$ e $(2, 2)$, obtemos $p_2(x) = 3 - 2(x + 2) + 7/12(x + 2)(x + 1)$ e então $p_2(0) = 1/6$.

Para estimar o erro, como não temos f e nem mesmo $f(0)$, vamos utilizar a tabela de diferenças divididas considerando todos os pontos mas apenas até a ordem 3 (já que o polinômio é de grau 2):

x_i	$f[x_i]$	ordem1	ordem2	ordem3
-3	5	-2	0	7/60
-2	3	-2	7/12	-71/594
-1	1	1/3	-2/27	4/81
2	2	0	2/9	-7/81
7/2	2	2/3	-8/27	197/1755
5	3	-2/3	13/30	-
8	1	3/2	-	-
10	4	-	-	-

Com 4 casas decimais, a tabela acima é dada por

x_i	$f[x_i] = f(x_i)$	ordem1	ordem2	ordem3
-3	5	-2	0	0.1166
-2	3	-2	0.5833	-0.1195
-1	1	0.3333	-0.0740	0.0493
2	2	0	0.2222	-0.0864
3.5	2	0.6666	-0.2962	0.1122
5	3	-0.6666	0.4333	-
8	1	1.5	-	-
10	4	-	-	-

Assim, a estimativa do erro é dada por

$$|E_n(x)| \leq |(x - (-2))(x - (-1))(x - 2)| \times (\text{maior valor, em módulo, da diferença dividida de ordem 3})$$

Logo,

$$|E_n(0)| \leq |(0 + 2)(0 + 1)(0 - 2)| \times 0.1195 = 4 \times 0.1195 = 0.4780$$

A tabela de diferenças divididas também é muito útil para determinar o grau do polinômio interpolador. Como visto anteriormente, a diferença dividida de ordem m é

uma aproximação para a derivada de ordem m da função $f(x)$. Assim, se em uma tabela obtivermos uma coluna j cujos valores são zero ou próximos desse valor, isso significa que a derivada de ordem j de $f(x)$ é zero ou próximo de zero. Dessa forma, usamos a tabela até a coluna $j - 1$, ou seja, o polinômio interpolador terá grau $j - 1$, já que, pelo menos aproximadamente, a inclusão de parcelas de grau maior ou igual a j não trará uma contribuição significativa ao polinômio, já que os valores dos coeficientes são iguais ou muito próximos a zero. Os exemplos a seguir ilustram este fato.

Exemplo 2.9 : Com os pontos $(-3, 9)$, $(-2, 4)$, $(-1, 1)$, $(0, 0)$, $(1, 1)$, $(2, 4)$, $(3, 9)$ e $(4, 16)$ obtemos um polinômio de grau 2 dado por $p_2(x) = 9 + 5(x - 3) + (x - 3)(x - 2)$. Veja que, na tabela de diferenças divididas, a coluna referente a ordem tem todos os valores iguais a zero.

x_i	$f[x_i]$	ordem1	ordem2	ordem3	ordem4	ordem5	ordem6	ordem7
-3	9	-5	1	0	0	0	0	0
-2	4	-3	1	0	0	0	0	-
-1	1	-1	1	0	0	0	-	-
0	0	1	1	0	0	-	-	-
1	1	3	1	0	-	-	-	-
2	4	5	1	-	-	-	-	-
3	9	7	-	-	-	-	-	-
4	16	-	-	-	-	-	-	-

Exemplo 2.10 : Considerando os pontos $(-2, 3.0016)$, $(-1, 3.0001)$, $(0, 1)$, $(1, 3.0001)$, $(2, 15.0016)$ e $(3, 43.0081)$, a tabela de diferenças finitas é dada por

x_i	$f[x_i]$	ordem1	ordem2	ordem3	ordem4	ordem5
-2	3.0016	-0.0015	-0.9993	0.9998	0.0001	0
-1	3.0001	-2.0001	2.0001	1.0002	0.0001	-
0	1	2.0001	5.0007	1.0006	-	-
1	3.0001	12.0015	8.0025	-	-	-
2	15.0016	28.0065	-	-	-	-
3	43.0081	-	-	-	-	-

Podemos ver então que o polinômio que interpola os 6 pontos dados é

$$p_4(x) = 0.0001x^4 + x^3 + 2x^2 - x + 1$$

uma vez que a diferença dividida de ordem 5 é igual a zero.

Porém, vemos que o coeficiente de grau 4 é 0,0001 e podemos assumi-lo pequeno o suficiente quando comparado com os outros coeficientes. Assim, podemos considerar o polinômio $p_3(x) = x^3 + 2x^2 - x + 1$ como uma aproximação para o polinômio de grau 4.

A figura a seguir mostra os dois polinômios, de grau 3 e de grau 4. Em sequência, a próxima figura mostra o gráfico de $|p_3(x) - p_4(x)|$.

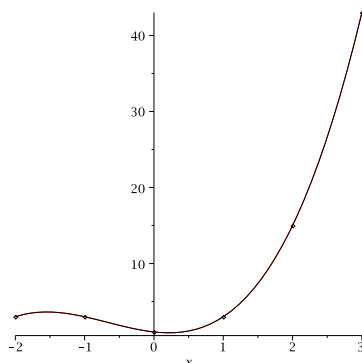


Figura 7 – Polinômio de grau 4.

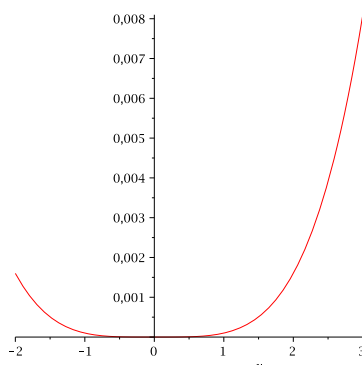


Figura 8 – Diferença, em módulo, entre os polinômios de grau 3 e 4 do exemplo.

2.7.4 Interpolação inversa

Em algumas aplicações, dado um conjunto de pontos $\{(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)\}$, com $x_1 < x_2 < \dots < x_n$ e $y_i = f(x_i)$ sendo $f(x)$ uma função, e dado também um valor \bar{y} , estamos interessados em determinar \bar{x} tal que $f(\bar{x}) \cong \bar{y}$.

Se os valores y_i formam uma sequência estritamente monótona, ou seja, $y_0 < y_1 < \dots < y_n$ ou $y_0 > y_1 > \dots > y_n$, então podemos supor que existe uma função $g(y)$ tal que $x_i = g(y_i)$. Neste caso, basta considerar o conjunto de pontos $\{(y_0, x_0), (y_1, x_1), \dots, (y_n, x_n)\}$ e

determinarmos o polinômio interpolador $p_n(y)$ desse conjunto. Em seguida, calculamos \bar{x} como $\bar{x} = g(\bar{y})$.

Caso a condição acima não seja satisfeita determinamos o polinômio interpolador $p_n(x)$ do conjunto de pontos original e, em seguida, determinamos algebricamente ou numericamente \bar{x} tal que $p_n(\bar{x}) = \bar{y}$. Os exemplos a seguir ilustram esse tipo de problema.

Exemplo 1.14: dados os pontos $(-3, 9), (-2, 4), (-1, 2), (0, 1), (1, 0)$, estimar \bar{x} tal que $f(\bar{x}) = 3$.

Como a sequência de valores de y_i é uma sequência estritamente decrescente, passamos a considerar o conjunto de pontos $\{(0, 1), (1, 0), (2, -1), (4, -2), (9, -3)\}$, cujo polinômio interpolador é dado, com 6 casas decimais, por $p_4(y) = -0.006349y^4 + 0.086111y^3 - 0.213888y^2 - 0.865873y + 1$. Logo, $\bar{x} = p(3) = -1.711905$.

Apesar da condição de monotocidade ter sido satisfeita, vamos resolver o mesmo problema pela segunda solução proposta, ou seja, utilizamos o conjunto de pontos originais para determinar $p_4(x) = 0.041667x^4 - 0.083334x^3 - 0.041666x^2 - .916666665x + 1$. Em seguida, com o auxílio de algum método numérico para determinação de raízes (ver capítulo sobre isso), determinamos x que satisfaz a equação $p_4(x) = 3$ e que pertença ao intervalo $[-3, 1]$, obtendo $\bar{x} = -1.612086$.

A diferença entre os valores calculados pelas duas propostas acima deve-se ao fato de que os polinômios interpoladores determinados acima são, na verdade, aproximações para as funções $f(x)$ e $g(x)$.

2.7.5 Comparação entre as formas de interpolação

Vamos inicialmente citar algumas características de cada uma das formas de interpolação apresentadas.

- i. Forma direta: bastante intuitiva teoricamente; a matriz do sistema é mal condicionada, o que acarreta em erros numéricos na solução do sistema; fornece a demonstração de unicidade do polinômio interpolador. a complexidade computacional depende do método de resolução de sistema linear adotado.
- ii. Forma de Lagrange: de fácil implementação; pouca utilização de memória (não necessita de armazenamento de matrizes); poucas operações numéricas; não fornece a expressão do polinômio na forma $p(x) = a_0 + a_1x + a_2x^2 + \dots a_nx^n$; útil quando precisamos determinar o valor do polinômios para poucos valores de x .
- iii. Forma de Newton: implementação um pouco mais difícil que a Forma de Lagrange; muitas operações de divisão; fornece expressão para o erro cometido na interpolação; fácil determinar polinômios de grau maior quando mais pontos são acrescentados ao conjunto..

2.7.6 Fenômeno de Runge

Este é o nome dado à oscilação não desejada que os polinômios apresentam em alguns casos de interpolação. A função clássica que ilustra este fenômeno é dada por $f(x) = \frac{1}{1+25x^2}$. As figuras 9, 10 e 11 ilustram tal fato.

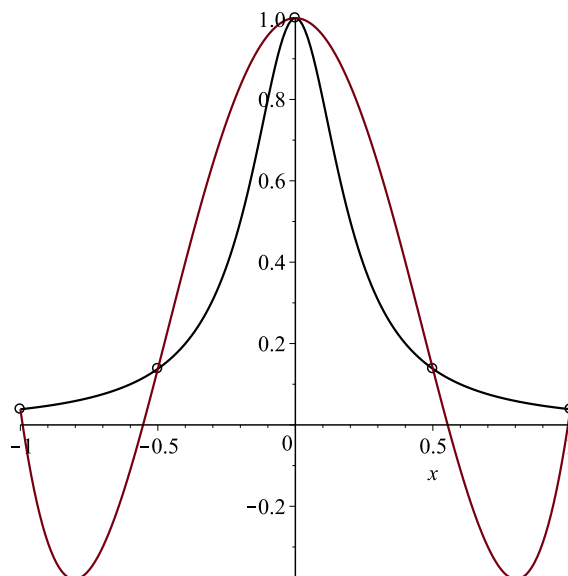


Figura 9 – Gráfico de $f(x) = \frac{1}{1+25x^2}$ e polinômio interpolador de grau 4.

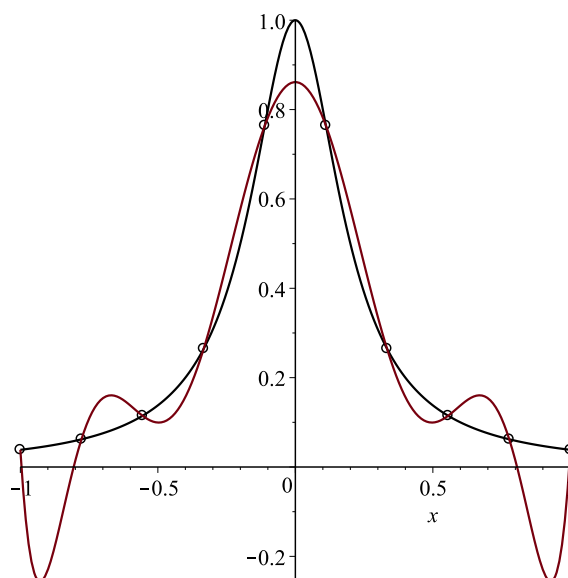


Figura 10 – Gráfico de $f(x) = \frac{1}{1+25x^2}$ e polinômio interpolador de grau 9.

Como pode ser visto nos gráficos acima, o aumento do grau do polinômio interpolador fez com que surgisse oscilações próximas aos pontos $x = -1$ e $x = 1$. Em outras palavras,

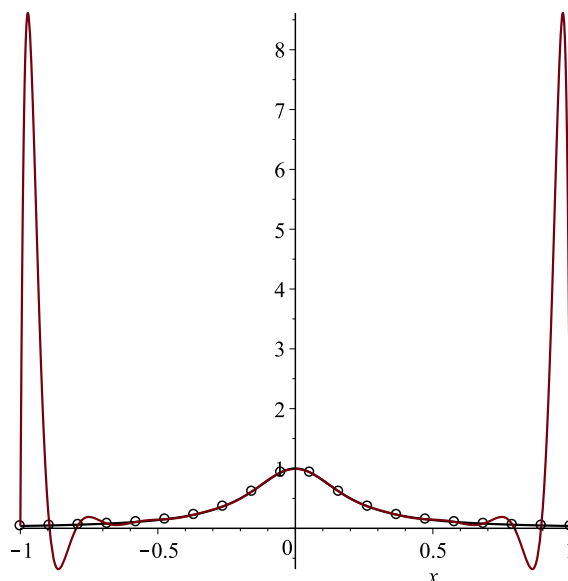


Figura 11 – Gráfico de $f(x) = \frac{1}{1+25x^2}$ e polinômio interpolador de grau 19.

o aumento do grau do polinômio interpolador faz com que o erro $E_n(x) = f(x) - p_n(x)$ aumente próximos a esses valores.

Há algumas formas de resolver este tipo de problema. Nelas serão considerados sempre 10 pontos.

- i.** dividir o intervalo original $[x_0, x_n]$ em subintervalos e, em seguida, determinar polinômios interpoladores em cada intervalo. Esta solução tem a desvantagem de não garantir a continuidade da derivada da função interpoladora obtida, como pode ser visto na figura 12. Neste exemplo consideramos uma partição regular do intervalo $[-1, 1]$ com 10 pontos e dividimos este intervalo em três subintervalos, dados por $[-1, -5/9]$ com os 3 primeiros pontos da partição, $[-5/9, 5/9]$ com 6 pontos e $[5/9, 1]$ com os 3 últimos pontos da partição. Em seguida obtivemos polinômios interpoladores de grau 2, 5 e 2, respectivamente, em cada subintervalo.
- ii.** redefinir os pontos da interpolação de modo a concentrá-los próximos aos pontos onde ocorre a oscilação e, em seguida, determinar o polinômio interpolador, neste caso de grau 9. A figura 13 apresenta esta solução concentrando os pontos a serem interpolados próximos aos extremos do intervalo. Como pode ser visto, isso resolve o problema do fenômeno de runge, porém há uma perda de precisão na parte central do intervalo. Além disso, nas aplicações práticas nem sempre podemos escolher livremente os pontos da interpolação.
- iii** uma outra solução consiste em combinar as duas soluções acima: concentra-se os pontos da partição nas regiões de maior oscilação e, além disso, determinam-se

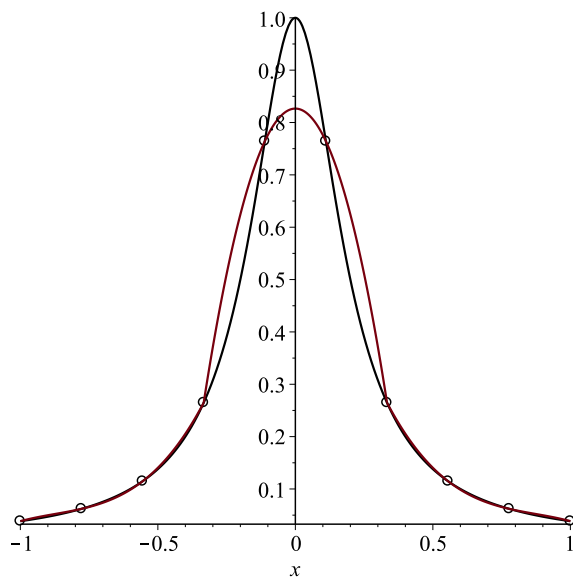


Figura 12 – Divisão do intervalo e polinômio interpolador em cada subintervalo: grau 5 em $[-5/9, 5/9]$ e grau 2 em $[-1, -5/9]$ e em $[5/9, 1]$. Repare a descontinuidade da derivada no quarto e no sétimo pontos.

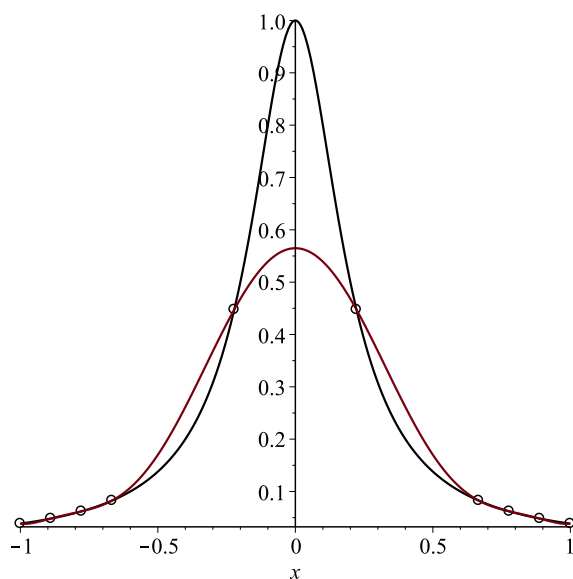


Figura 13 – Concentrando pontos a serem interpolados próximos aos extremos do intervalo.

vários polinômios de grau menor, considerando-se subintervalos.

Um leitor atento percebe que nas soluções apresentadas acima o polinômio interpolador não representa bem a função $f(x)$ na região central do intervalo $[-1, 1]$, apesar de terem solucionado o problema do fenômeno de Runge próximo dos extremos. Para resolver este outro problema é necessário aumentar a quantidade de pontos no subintervalo do

meio, porém corre-se o risco de novas oscilações indesejáveis. É preciso então, nos casos de funções como $f(x) = \frac{1}{1+25x^2}$, decidir cuidadosamente quantos e quais serão os pontos utilizados, bem como se serão determinados mais de um polinômio interpolador o intervalo dado.

Por fim, vale mencionar que o polinômio interpolador não é muito útil quando o objetivo é extrapolar, ou seja, encontrar um valor estimado para $f(x)$ quando $x \notin [x_0, x_n]$. Na verdade, em aplicações práticas a interpolação pode não fornecer bons resultados. Para ilustrar estes fatos, suponha que o gráfico do exemplo 3, reproduzido na figura 23, represente valores de alguma grandeza física que deve ser sempre positiva. Para $x > 10$ o polinômio interpolador cresce de forma bastante rápida, comportamento este que nem sempre está relacionado com os dados reais de uma aplicação, o que mostra que o polinômio interpolador não é muito útil para estimar valores de $f(x)$ com $x > 10$. Se analisarmos para $x < -4$ vemos que o polinômio assume valores negativos, que no caso não teriam significado físico. Além disso, reparamos que entre $x = 8$ e $x = 10$ o polinômio também assume valores negativos, ilustrando que até mesmo para regiões do interior do intervalo dado o polinômio interpolador pode assumir valores indesejáveis.

Para contornar o problema de valores indesejáveis ou sem significado físico no interior do intervalo $[x_0, x_n]$, algumas vezes causados pelo fenômeno de Runge, podemos lançar mão da interpolação spline.

2.8 AJUSTE DE CURVAS

O método da interpolação é um ótimo método para estipular o valor de uma função em um ponto dentro do intervalo, mas não é confiável caso o ponto esteja fora das extremidades. Para isso, existe o Ajuste de Curvas. Diferente do método da interpolação, ele não irá retornar uma função que passa exatamente pelos pontos, mas ele retornará uma função que representa o comportamento do conjunto de pontos, sem necessariamente passar pelos pontos. |

2.8.1 Método dos mínimos quadrados

Temos m pontos $(x_i, f(x_i)), i = 1, 2, \dots, m$ e queremos determinar uma função φ que represente o conjunto de pontos. Vamos considerar φ tal que $\varphi(x) = a_1g_1(x) + a_2g_2(x) + \dots + a_n g_n(x)$, onde $a_1 \in \mathbb{R}$ e $g(x)$ são funções pré-determinadas.

Como a função φ é linear nas incógnitas a_i , o método é chamado de linear. Para avaliar a proximidade de φ com os pontos, vamos considerar $d = \sum_{i=1}^n d_i$, onde $d_i = (\varphi(x_i) - f(x_i))^2$ é chamado de desvio ao quadrado no ponto i e, conseqüentemente, d é o desvio total. Queremos determinar os coeficientes a_1, a_2, \dots, a_n de forma que o desvio total seja o menor possível.

Vamos definir uma função $F : \mathbb{R}^n \rightarrow \mathbb{R}$ dada por

$$F[\alpha_1, \alpha_2, \dots, \alpha_n] = \sum_{i=1}^m (\varphi(x_i) - f(x_i))^2 = \sum_{i=1}^m [\alpha_1 g_1(x_i) + \alpha_2 g_2(x_i) + \dots + \alpha_n g_n(x_i) - f(x_i)]^2,$$

Temos que determinar $\alpha_1, \alpha_2, \dots, \alpha_n$ de modo que F seja o mínimo possível. Como a função F é quadrática, basta calcular o ponto crítico que, neste caso, é um ponto de mínimo. Queremos então encontrar onde o gradiente de F seja igual a zero. Para isso, com $F(\alpha_1, \alpha_2, \dots, \alpha_n)$, iremos calcular a derivada direcional.

$$\frac{\delta F}{\delta \alpha_1} = 0 \Rightarrow \sum_{i=1}^m [\alpha_1 g_1(x_i) g_1(x_i) + \alpha_2 g_2(x_i) g_1(x_i) + \dots + \alpha_n g_n(x_i) g_1(x_i)] = \sum_{i=1}^m f(x_i) g_1(x_i)$$

$$\frac{\delta F}{\delta \alpha_2} = 0 \Rightarrow \sum_{i=1}^m [\alpha_1 g_1(x_i) g_2(x_i) + \alpha_2 g_2(x_i) g_2(x_i) + \dots + \alpha_n g_n(x_i) g_2(x_i)] = \sum_{i=1}^m f(x_i) g_2(x_i)$$

...

$$\frac{\delta F}{\delta \alpha_n} = 0 \Rightarrow \sum_{i=1}^m [\alpha_1 g_1(x_i) g_n(x_i) + \alpha_2 g_2(x_i) g_n(x_i) + \dots + \alpha_n g_n(x_i) g_n(x_i)] = \sum_{i=1}^m f(x_i) g_n(x_i)$$

Obtemos então n equações. Como são n incógnitas $\alpha_1, \alpha_2, \dots, \alpha_n$, temos um sistema linear $n \times n$ que pode ser representado do seguinte modo:

$$\mathbf{A} = \begin{bmatrix} \sum_{i=1}^m g_1(x_i) g_1(x_i) & \sum_{i=1}^m g_1(x_i) g_2(x_i) & \dots & \sum_{i=1}^m g_1(x_i) g_n(x_i) \\ \sum_{i=1}^m g_2(x_i) g_1(x_i) & \sum_{i=1}^m g_2(x_i) g_2(x_i) & \dots & \sum_{i=1}^m g_2(x_i) g_n(x_i) \\ \sum_{i=1}^m g_3(x_i) g_1(x_i) & \sum_{i=1}^m g_3(x_i) g_2(x_i) & \dots & \sum_{i=1}^m g_3(x_i) g_n(x_i) \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^m g_{n-1}(x_i) g_1(x_i) & \sum_{i=1}^m g_{n-1}(x_i) g_2(x_i) & \dots & \sum_{i=1}^m g_{n-1}(x_i) g_n(x_i) \\ \sum_{i=1}^m g_n(x_i) g_1(x_i) & \sum_{i=1}^m g_n(x_i) g_2(x_i) & \dots & \sum_{i=1}^m g_n(x_i) g_n(x_i) \end{bmatrix},$$

$$\mathbf{b} = \begin{bmatrix} \sum_{i=1}^m f(x_i) g_1(x_i) \\ \sum_{i=1}^m f(x_i) g_2(x_i) \\ \sum_{i=1}^m f(x_i) g_3(x_i) \\ \vdots \\ \sum_{i=1}^m f(x_i) g_{n-1}(x_i) \\ \sum_{i=1}^m f(x_i) g_n(x_i) \end{bmatrix}$$

e o vetor incógnita

$$\mathbf{x} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_{n-1} \\ \alpha_n \end{bmatrix}.$$

Ao resolver o sistema linear acima obtemos os coeficientes α_i e, com eles, determinamos a função de ajuste $\varphi(x)$.

2.8.2 Ajuste não linear

$\varphi(x) = \alpha_1 + \alpha_2 x + \alpha_3 x^2$ é linear nas variáveis α_j , $j = 1, 2, \dots, n$. Vamos supor agora que os pontos aparentam seguir um padrão dado, por exemplo, por $\varphi(x) = a_1 e^{a_2 x}$. Vamos linearizar o problema.

Considere $y = a_1 e^{a_2 x}$. Logo $y = a_1 e^{a_2 x} \Rightarrow \ln(y) = \ln(a_1) + \ln(e^{a_2 x}) \Rightarrow \ln(y) = \ln(a_1) + a_2 x$.

Sejam $\ln(y) = z$, $\ln(a_1) = \beta_1$ e $a_2 = \beta_2$. Então obtemos $z = \beta_1 + \beta_2 x$.

Vamos considerar $\bar{\varphi}(x) = \beta_1 + \beta_2 x$, que é a função de ajuste dos pontos linearizados, ou seja, $\{(x_i, \ln(y_i)), i = 1, 2, \dots, n\}$.

Aplicamos o ajuste de curva linear a este conjunto para determinar β_1 e β_2 . Em seguida, determinamos os parâmetros originais $a_1 = e^{\beta_1}$ e $a_2 = \beta_2$.

Por fim, determinamos $\varphi(x) = a_1 e^{a_2 x}$.

Há várias formas diferentes de linearizar os pontos originais, a apresentada acima é apenas uma delas. Para verificar se a linearização é adequada, plotamos os pontos linearizados. Se eles estiverem seguindo o padrão de uma reta, é sinal que a linearização foi bem feita.

2.8.3 Outras Linearizações

1. $y \approx \frac{1}{a_1 + a_2 x} = \varphi(x)$

Logo, $y = \frac{1}{a_1 + a_2 x} = a_1 + a_2 x = \frac{1}{y} = z$ e aí $\bar{\varphi}(x) = a_1 + a_2 x$ para os pontos $\{(x_i, \frac{1}{y_i})\}$ se $y_i \neq 0$, $\forall i = 1, 2, \dots, n$.

2. $y \approx a_1 x^{a_2} = \varphi(x)$.

Logo, $y = a_1 x^{a_2}$, $\ln(y) = \ln(a_1 x^{a_2}) = \ln(a_1) + \ln(x^{a_2}) = \ln(a_1) + a_2 \ln(x) = z$. Fazendo $\ln(a_1) = \beta_1$, $a_2 = \beta_2$ e $\ln(x) = t$, temos $z = \beta_1 + \beta_2 t$.

Assim, $\bar{\varphi}(x) = \beta_1 + \beta_2 t$ com $\beta_1 = \ln(a_1)$ e $\beta_2 = a_2$, para os pontos $\{(\ln(x_i), \ln(y_i))\}$.

2.9 INTEGRAÇÃO NUMÉRICA

Inicialmente, vamos estudar uma família de métodos chamada de Newton-Cotes. Dada uma função f integrável em $[a, b]$, queremos determinar $\int_a^b f(x) dx$. Nesta família de métodos, vamos discretizar $[a, b]$ em sub-intervalos de mesmo tamanho e determinar um polinômio P_n que interpole esses pontos. Em seguida, fazemos:

$$\int_a^b f(x) dx \approx \int_a^b P_n(x) dx$$

Obs: quando a e b fazem parte da discretização, temos métodos de Newton-Cotes fechados. Caso contrário, são ditos abertos.

2.9.1 Regra dos trapézios

Vamos considerar a seguinte discretização: $x_0 = a$, $x_1 = b$, apenas. Nesse caso,

$$\int_a^b f(x) dx \approx \int_{x_0}^{x_1} P_1(x) dx.$$

Escolhendo a forma de Lagrange para encontrar P_n :

$$\int_{x_0}^{x_1} [f(x_0)L_0(x) + f(x_1)L_1(x)] dx = \int_{x_0}^{x_1} \left[f(x_0) \frac{x - x_1}{x_0 - x_1} + f(x_1) \frac{x - x_0}{x_1 - x_0} \right] dx =$$

$$\begin{aligned} & f(x_0) \int_{x_0}^{x_1} \frac{x - x_1}{-h} dx + f(x_1) \int_{x_0}^{x_1} \frac{x - x_0}{h} dx = \\ & \frac{f(x_0)}{-h} \int_{x_0}^{x_1} (x - x_1) dx + \frac{f(x_1)}{h} \int_{x_0}^{x_1} (x - x_0) dx \end{aligned}$$

Realizando o método da substituição:

$$z = x - x_1 \quad \text{e} \quad z = x - x_0$$

para as duas integrais, respectivamente

$$\begin{aligned} & \frac{f(x_0)}{-h} \int_0^h (z - h) dz + \frac{f(x_1)}{h} \int_0^h (z) dz = \\ & \frac{f(x_0)}{-h} \left[\frac{z^2}{2} - hz \right]_0^h + \frac{f(x_1)}{h} \left[\frac{z^2}{2} \right]_0^h = \\ & \frac{f(x_0)}{-h} \left[\frac{h^2}{2} - h^2 \right] + \frac{f(x_1)}{2h} (h^2) = \\ & \frac{h}{2} f(x_0) + \frac{h}{2} f(x_1) = \\ & \frac{h}{2} f(x_0) + \frac{h}{2} f(x_1) = \frac{h}{2} [f(x_0) + f(x_1)]. \end{aligned}$$

Logo :

$$\int_a^b f(x) dx \approx \frac{h}{2}[f(x_0) + f(x_1)].$$

Mas olha só, essa é a fórmula da área de um trapézio. Por isso, é chamado de Regra dos trapézios. E o erro? Sabemos que $f(x) = P_1(x) + E_1(x)$. Assim:

$$\int_a^b f(x) dx = \int_{x_0}^{x_1} P_1(x) dx + \int_{x_0}^{x_1} E_1(x) dx$$

Então o erro na integração é

$$\int_{x_0}^{x_1} E_1(x) dx = \int_{x_0}^{x_1} (x - x_0)(x - x_1) \frac{f''(\xi)}{2!} dx$$

Onde $\xi \in (x_0, x_1)$ e $\xi = \xi(x)$ Vamos supor que $f \in C^2([a, b])$. Além disso, vemos que $g(x) = (x - x_0)(x - x_1)$ é tal que $g(x) \leq 0$ em $[x_0, x_1]$

Obs: Teorema do valor médio para integrais Se f é contínua em $[a, b]$, então existe $c \in (a, b)$ tal que

$$\int_z^b f(x) dx = f(c) \cdot (b - a)$$

Se f é contínua em $[a, b]$ e g não muda de sinal neste intervalo, então existe $c \in (a, b)$ tal que

$$\int_a^b g(x) f(x) dx = f(c) \int_a^b g(x) dx$$

Logo, aplicando o T.V.M para integrais, existe $c \in (a, b)$ tal que

$$\int_{x_0}^{x_1} E_1(x) dx = \frac{f''(c)}{2} \int_z^b (x - x_0)(x - x_1) dx$$

realizando o método da substituição visto anteriormente:

$$\begin{aligned} \frac{f''(c)}{2} \int_a^b z(z - h) dz &= \frac{f''(c)}{2} \left[\frac{z^3}{3} - h \frac{z^2}{2} \right] = \frac{f''(c)}{2} \left[\frac{h^3}{3} - \frac{h^3}{2} \right] = \\ &= \frac{f''(c)}{2} \frac{-h^3}{6} = \frac{-f''(c)h^3}{12} \end{aligned}$$

Assim, $|E| = \left| -\frac{f''(c)h^3}{12} \right| = \left| -\frac{|f''(c)|h^3}{12} \right| \leq \frac{h^3}{12} M$ Onde $M = \max\{|f''(x)|\}$ em $[a, b]$

Mas nesse caso foi utilizado apenas um único intervalo, nos dando uma faixa de erro muito imprecisa.

2.9.2 Regra dos trapézios Repetidos

$x_{i+1} - x_i = h, i = 1, 2, \dots, n - 1$ (discretização regular)

Temos então n sub-intervalos. Aplicando a Regra do Trapézio a cada um deles, temos:

$$\int_a^b f(x) dx = \int_{x_0}^{x_1} f(x) dx + \int_{x_1}^{x_2} f(x) dx + \dots + \int_{x_{n-1}}^{x_n} f(x) dx$$

$$= \frac{h}{2} \left[f(x_0) + 2 \sum_{i=2}^{n-1} f(x_i) + f(x_n) \right]$$

E o erro nesse caso?

Para cada sub-intervalo $i = 1, 2, \dots, n$, temos $E_i = -\frac{f''(c_i)}{12}h^3$, com $c_i \in (x_{i-1}, x_i)$. O erro total será a soma de todos os E_i :

$$\sum_{i=1}^n E_i = \sum_{i=1}^n -\frac{f''(c_i)}{12}h^3$$

Como assumimos $f \in C^2([a, b])$, existe $c \in (a, b)$ tal que $f''(c) = \frac{1}{n} \sum_{i=1}^n f''(c_i)$.

Logo:

$$E_t = -\frac{nf''(c)}{12}h^3,$$

e aí

$$E_t = \left| -\frac{nf''(c)}{12}h^3 \right|, \text{ e aí } |E_t| \leq \frac{nh^3}{12}M,$$

com $M = \max\{|f''(x)| \mid x \in [a, b]\}$.

2.9.3 1/3 de simpson

A partir do ponto a e b , traçar x no ponto médio entre eles e encontrar um polinômio de segundo grau que passe pelos 3 pontos.

$$\int_a^b f(x) dx = \int_{x_0}^{x_2} P_2(x) dx$$

Usando o método de Lagrange para encontrar P_2 fica:

$$\begin{aligned} & \int_{x_0}^{x_2} [f(x_0)L_0(x) + f(x_1)L_1(x) + f(x_2)L_2(x)] dx \\ & \int_{x_0}^{x_2} f(x_0) \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} dx + \int_{x_0}^{x_2} f(x_1) \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} dx + \int_{x_0}^{x_2} f(x_2) \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} dx \\ & = \frac{f(x_0)}{2h^2} \int_{x_0}^{x_2} (x-x_1)(x-x_2) dx - \frac{f(x_1)}{h^2} \int_{x_0}^{x_2} (x-x_0)(x-x_2) dx + \frac{f(x_2)}{2h^2} \int_{x_0}^{x_2} (x-x_0)(x-x_1) dx \end{aligned}$$

Realizando o mesmo método das substituições para resolver as integrais:

$$= f(x_0) \frac{h}{3} + 4f(x_1) \frac{h}{3} + f(x_2) \frac{h}{3} = \frac{h}{3} [f(x_0) + 4f(x_1) + f(x_2)]$$

E o erro dado por:

$$E = -\frac{h^5}{90} f^{(4)}(c)$$

2.9.4 1/3 de simpson repetido

Assim como na Regra do Trapézio repetido, você divide o intervalo em sub-intervalos e aplica a Regra do Trapézio em cada um dos sub-intervalos, no Simpson Repetido você fará o mesmo. Mas fique atento, diferente do M.T, o M.S precisa de 3 pontos, 2 sub-intervalos, em outras palavras, o número de sub-intervalos que você pode usar na Regra de Simpson Repetido precisa ser sempre par.

$$\int_a^b f(x) dx \approx \int_{x_0}^{x_n} f(x) dx = \frac{h}{3} [f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + 2f(x_4) + \dots + 2f(x_{n-2}) + 4f(x_{n-1}) + f(x_n)].$$

Erro:

$$E = -\frac{nh^5}{180} f^{(4)}(c) \quad \text{com } c \in (x_0, x_n)$$

$$|E| \leq \frac{nh^5}{180} M, \quad \text{onde } M = \max\{|f^{(4)}(x)|\}$$

2.9.5 Quadratura Gaussiana

De um modo geral, os métodos de Newton-Cotes são capazes de integrar, sem erro, polinômios de grau menor ou igual a n , onde n é a quantidade de pontos que definem P_n .

RT: $n = 2$

RS: $n = 3$

Com a quadratura Gaussiana, com n pontos, somos capazes de integrar, sem erro, P_{2n-1} , pagando o preço de não saber quais os pontos da discretização.

Por exemplo, seja $n = 2$. Temos então que determinar os dois pontos da discretização e seus pesos associados, ou seja:

$$\int_{-1}^1 g(x) dx \approx W_1 g(\xi_1) + W_2 g(\xi_2),$$

onde os pesos de Gauss W_1, W_2 e os pontos de Gauss ξ_1, ξ_2 têm que ser determinados. São 4 incógnitas. Mas com esses 2 pontos queremos ser capazes de integrar P_3 sem erro. Como $P_3(x) = a_0 + a_1x + a_2x^2 + a_3x^3$, para integrar sem erro este polinômio, basta integrar sem erro as funções $1, x, x^2, x^3$.

$$\int_{-1}^1 1 dx = 2 = W_1 \cdot 1 + W_2 \cdot 1 \quad \Rightarrow \quad W_1 + W_2 = 2$$

$$\int_{-1}^1 x dx = 0 = W_1 \cdot \xi_1 + W_2 \cdot \xi_2$$

$$\int_{-1}^1 x^2 dx = \frac{2}{3} = W_1 \cdot \xi_1^2 + W_2 \cdot \xi_2^2$$

$$\int_{-1}^1 x^3 dx = 0 = W_1 \cdot \xi_1^3 + W_2 \cdot \xi_2^3$$

Temos então 4 equações:

$$W_1 + W_2 = 2$$

$$W_1 \xi_1 + W_2 \xi_2 = 0$$

$$W_1 \xi_1^2 + W_2 \xi_2^2 = \frac{2}{3}$$

$$W_1 \xi_1^3 + W_2 \xi_2^3 = 0$$

Isso nem sistema linear é, então não precisa se preocupar em resolvê-lo, a solução é:
 $\xi_1 = -\frac{\sqrt{3}}{3}$, $\xi_2 = \frac{\sqrt{3}}{3}$, $W_1 = W_2 = 1$.

Logo, a quadratura gaussiana com 2 pontos é:

$$\int_{-1}^1 g(\xi) d\xi \approx 1 \cdot g\left(-\frac{\sqrt{3}}{3}\right) + 1 \cdot g\left(\frac{\sqrt{3}}{3}\right)$$

3 A IMPLEMENTAÇÃO DO SITE

Este capítulo tem como propósito detalhar o processo completo de implementação do site dedicado ao aprendizado de cálculo numérico. Desde a concepção inicial até a sua realização final, serão abordados aspectos cruciais, como a definição da ideia, a seleção e aplicação de métodos e ferramentas, bem como os desafios enfrentados ao longo do caminho.

Exploraremos cada etapa do desenvolvimento do site, desde a definição dos requisitos e a estruturação da arquitetura até a implementação das funcionalidades e o design da interface. Serão discutidas as escolhas de tecnologias e plataformas, os processos de desenvolvimento utilizados e as estratégias adotadas para garantir a qualidade e a eficiência do produto final.

Ao longo deste capítulo, será possível compreender não apenas os aspectos técnicos e práticos envolvidos na implementação do site, mas também a visão e o esforço por trás da transformação de uma ideia em realidade.

Esse capítulo será narrado em primeira pessoa, pois ele será contado sob a perspectiva do desenvolvedor do site.

3.1 A CRIAÇÃO DO PROTÓTIPO

Inicialmente, optei por seguir uma abordagem semelhante à metodologia adotada pela empresa em que estagiei, começando pela criação de um protótipo da tela inicial do site (landing page) para submetê-lo à avaliação do meu orientador. Este método baseia-se em um processo iterativo, onde o projeto é discutido, um protótipo é desenvolvido e apresentado ao cliente (neste caso, o orientador), e somente após sua aprovação é que o desenvolvimento completo do site é iniciado.

Este primeiro passo, no entanto, trouxe consigo um desafio significativo: a parte visual do site. Reconhecendo que o design não é minha especialidade, levei um tempo considerável apenas refletindo sobre como o site deveria ser estruturado e apresentado. Dado o caráter matemático do site, busquei referências em plataformas semelhantes, como Geogebra, Symbolab e Wolfram, além de outros sites com estilos genéricos.

Após realizar essa pesquisa de referências visuais, decidi que o site deveria ter um design simplista, porém objetivo, com o uso de cores fortes e contrastantes para facilitar a navegação e a compreensão do conteúdo.

No entanto, surgiram novos desafios relacionados à obtenção de imagens para o site. Consciente da importância da originalidade e da ética no trabalho acadêmico, decidi não utilizar imagens encontradas na internet, mas sim criar minhas próprias ilustrações. Para

isso, instalei a ferramenta de desenho Adobe Illustrator e dediquei-me a criar ilustrações personalizadas para cada tópico de cálculo numérico que seria abordado no site.

O resultado desse processo inicial pode ser visualizado na tela inicial do site, composta por diversos cards, cada um representando um tópico específico de cálculo numérico, ilustrados com desenhos criados por mim mesmo.



Figura 14 – Protótipo feito em Figma

Uma barra azul em cima que é o topo do site, a logo da UFRJ e como eu ainda não tinha decidido um nome para o site, simplesmente coloquei “cálculo numérico”. Como a ideia é de que depois de clicar em qualquer um dos cards, o usuário entraria em uma outra página, a logo e o título do site serviriam como uma âncora para ele voltar para a página inicial. Eu também fiz questão de deixar os outros cards da fileira de baixo cortados na imagem, mostrando que para baixo tem mais itens. O protótipo foi feito usando a ferramenta Figma.

3.2 IMPLEMENTANDO A TELA INICIAL

Após a aprovação do protótipo pelo meu orientador, dei início à fase de implementação do projeto. Optei por utilizar o framework Angular para o desenvolvimento do site, uma escolha motivada pela minha familiaridade com essa tecnologia, adquirida durante meu estágio profissional.

O Angular é um framework que se baseia no conceito de componentes, onde cada componente representa uma seção distinta do site. Essa abordagem modular facilita a organização e a manutenção do código, além de proporcionar uma experiência de desenvolvimento mais fluida.

Além disso, para auxiliar no aspecto visual do site, integrei a biblioteca Bootstrap ao projeto para garantir uma interface limpa e moderna.

3.3 O PROBLEMA COM AS EQUAÇÕES

Agora chegou o momento de implementar as páginas para cada teoria. Quando o usuário clicar em qualquer card, a URL será alterada dinamicamente. Por exemplo, se a tela inicial possui a URL "www.tcccaiocohen.com.br" e o usuário clicar no card do polinômio de Taylor, a URL se tornará "www.tcccaiocohen.com.br/taylor". Isso acionará o carregamento de uma nova página, juntamente com um novo componente do framework.

No entanto, surgiu uma nova dificuldade: o HTML não oferece suporte nativo para equações matemáticas. Dado que a teoria do site é rica em equações, tabelas e gráficos, este se tornou um desafio significativo.

Após uma extensa pesquisa na internet sobre ferramentas de interpretação de LaTeX para HTML, não encontrei nenhuma solução que atendesse às minhas necessidades, especialmente considerando o uso de um framework. Inicialmente, tentei criar mini componentes usando CSS para estilizar o texto e simular o LaTeX, mas isso se mostrou complexo e ineficiente.

Diante dessa dificuldade, tive uma ideia: decidi utilizar um documento temporário no Google Docs para escrever as equações necessárias e, em seguida, salvar as equações como imagens. Dessa forma, as equações seriam exibidas no site como imagens, contornando a limitação do HTML e garantindo uma representação precisa das fórmulas matemáticas.

3.4 VÍDEOS DEMONSTRATIVOS

Quando chegou o momento de explicar a teoria por trás dos métodos para encontrar raízes de funções reais, percebi que apenas equações escritas não ofereciam uma compreensão visual suficiente. Decidi então complementar a explicação teórica com vídeos demonstrativos, mostrando cada método sendo aplicado passo a passo na prática.

Inicialmente, considerei procurar na internet por vídeos que atendessem às minhas necessidades. No entanto, em respeito à ética profissional, optei por criar os vídeos eu mesmo. Utilizando o Adobe Illustrator, a mesma ferramenta que empreguei para desenhar os cards da página inicial, criei gráficos genéricos representando o estado inicial de cada método. A ideia era ilustrar, passo a passo, o funcionamento de cada método, salvando imagens de cada etapa. Posteriormente, compilei essas imagens em um editor de vídeos para criar as videoaulas correspondentes.

Durante esse processo, foi crucial selecionar cuidadosamente as funções e os pontos iniciais para evitar uma quantidade excessiva de etapas, o que prolongaria demasiadamente a produção dos vídeos. Além disso, era importante escolher funções genéricas para

não influenciar negativamente o aprendizado dos alunos. Meu orientador desempenhou um papel fundamental ao auxiliar nessas escolhas.

Ao abordar os problemas de valor inicial, também optei por criar vídeos demonstrativos. Nesse momento, meu orientador destacou um erro em minha abordagem: todos os vídeos iniciavam a demonstração a partir da origem, o que poderia induzir os alunos a acreditarem que essa era a única maneira correta de iniciar os métodos. Consequentemente, precisei refazer todos os vídeos para corrigir esse equívoco.

3.5 IMPLEMENTAÇÃO DOS GRÁFICOS

Tipicamente, ao realizar análises de dados e representá-los graficamente, recorro ao Python ou ao R. Entretanto, dado o contexto de um site em um navegador, precisava encontrar uma solução em JavaScript.

Embora já tivesse experiência anterior com gráficos em JavaScript, encontrei desafios com as bibliotecas disponíveis. Muitas delas eram complexas e pouco flexíveis, além de não estarem atualizadas para a versão do framework que estava utilizando. Além disso, não tinha uma ideia clara de como integrar essas bibliotecas aos métodos do site, correndo o risco de descobrir sua insuficiência durante a implementação.

Por esses motivos, optei por desenvolver manualmente meu próprio componente de gráfico, utilizando o canvas do HTML. Embora não me arrependa dessa escolha, reconheço que foi a parte mais desafiadora da implementação do site.

Tive que lidar com alguns conceitos específicos do canvas: ele utiliza um sistema de coordenadas em pixels, onde a origem está no canto superior esquerdo. Meu objetivo era passar uma lista de pontos no eixo X e no eixo Y e desenhar um gráfico com base neles. Para isso, criei uma função que traduzia o valor de um ponto (no eixo X ou no eixo Y) para sua posição em pixels no canvas.

Essa função calculava dinamicamente o valor de input multiplicado pelo tamanho do passo, que era calculado com base na largura do canvas. No entanto, o gráfico resultante ocupava todo o canvas e não havia espaço para os rótulos, então adicionei uma variável chamada "padding" à função para resolver esse problema.

Com isso, criei as funções LGX(pos) e LGY(pos) (LG refere-se a local-global), permitindo que o gráfico fosse desenhado corretamente.

```

LGX(pos: number) {
  let first = this.x1[0];
  let last = this.x1[this.x1.length - 1];
  let espaco = last-first;
  let passo = (640-(2*(this.configuracoes.padding)))/espaco;
  return (passo*(pos-first))+this.configuracoes.padding;
}

LGY(pos: number) {
  let first = Math.min(...this.y1);
  let last = Math.max(...this.y1);
  if(first == last){
    first = first-first;
    last = last + last;
  }
  let espaco = last-first;
  let passo = (360-(2*(this.configuracoes.padding)))/espaco;
  return (360-(2*(this.configuracoes.padding)))-(passo*(pos-first))+this.configuracoes.padding;
}

```

Figura 15 – Funções de transformação dos intervalos

Entretanto, ainda era necessário desenhar os eixos e representar os valores em cada ponto, o que também se mostrou uma tarefa trabalhosa.

Para determinar quais pontos seriam representados nos eixos, foi necessário definir um critério claro, já que não seria viável mostrar o valor de milhares de pontos. Optei por exibir cinco pontos distintos: os valores mínimo e máximo do intervalo, o ponto médio entre eles e os pontos médios entre o ponto médio e os extremos. Por exemplo, se a função estiver no intervalo de 0 a 10, os pontos exibidos no eixo X serão 0, 10, 5, 2.5 e 7.5.

Essa lógica foi implementada na função `getLabelPositions`:

```

getLabelPositions(array: number[]){
  let retorno = [0,0,0,0,0]
  retorno[0] = +Math.min(...array).toFixed(2)
  retorno[4] = +Math.max(...array).toFixed(2)
  if(retorno[0]==retorno[4]){
    retorno[0] -= retorno[0]
    retorno[4] += retorno[4]
  }
  retorno[2] = +( (retorno[0]+retorno[4])/2 ).toFixed(2)
  retorno[1] = +( (retorno[0]+retorno[2])/2 ).toFixed(2)
  retorno[3] = +( (retorno[2]+retorno[4])/2 ).toFixed(2)
  return retorno;
}

```

Figura 16 – Função que recupera os valores que serão mostrados nos eixos

O `.toFixed(2)` serve para arredondar o valor para duas casas decimais depois da vírgula. E depois, para desenhar os eixos Y e X eu usei as funções `LGX` e `LGY` para definir o ponto de origem no 0,0. Além disso, o tamanho do número nas labels interfere no tamanho do espaço que deve ser dado entre o gráfico e as bordas do canvas, com isso eu coloquei que o padding (em pixels) seria 10 vezes o valor do maior número das labels do eixo Y. O componente de gráfico que eu desenvolvi suporta apenas 2 funções de input (o que na verdade seriam 4 listas de números, 2 para cada eixo X e duas para cada eixo Y), pois era o suficiente para o que eu mostraria no site. Uma seria a função analítica e a outra a aproximada. Depois eu também fiz ele aceitar até dois pontos extras, que seriam desenhados como círculos no gráfico, para os simuladores dos métodos das raízes.

Durante o desenvolvimento dos simuladores de ajuste de curvas, decidi aumentar significativamente a quantidade de pontos que poderiam ser desenhados, tornando-a praticamente ilimitada. O componente de gráfico foi ajustado para receber uma lista de coordenadas e, para cada coordenada, desenhar um ponto, respeitando as relações entre os sistemas de coordenadas local e global.

Adicionalmente, foram implementados dois botões na lateral do gráfico (presentes em alguns dos gráficos) para permitir zoom-in e zoom-out. Esses botões ajustam o intervalo exibido no gráfico, mantendo o ponto médio constante, e redesenham o gráfico conforme necessário.

3.6 IMPLEMENTAÇÃO DOS SIMULADORES

Após concluir o desenvolvimento do componente gráfico, decidi implementar simuladores interativos para o usuário. Essa ideia foi concebida para que os alunos pudessem experimentar na prática como os métodos numéricos funcionam. Ao testar diferentes entradas em diversas situações, os alunos podem aprofundar seu entendimento dos métodos.

3.6.1 Métodos de Runge Kutta

O primeiro simulador que desenvolvi foi o dos métodos de Runge-Kutta. O objetivo desses métodos é prever o valor de um ponto em uma função desconhecida, utilizando o valor de sua derivada e um ponto inicial. O propósito do simulador é, além de demonstrar o comportamento do método ao longo da função, enfatizar a importância de escolher um valor adequado para o "passo".

Com isso em mente, foram feitos 4 diferentes simuladores. Os 3 primeiros são para fins puramente educativos, a função da EDO e o intervalo da amostra são fixos e o aluno pode definir a quantidade de passos que caberão dentro do intervalo (alterando, assim, o tamanho do passo)

O último é para fins experimentais, o aluno poderá colocar a $f'(x)$ que ele desejar, definir o ponto inicial, o tamanho e a quantidade de passos, e o gráfico a seguir mostrará o comportamento dos métodos RK1, RK2 e RK4.

Os algoritmos 1, 2 e 3 são os pseudo-códigos das implementações dos RK1, RK2 e RK4:

Algorithm 2 RK1

Entrada: h, x_0, y_0, n

Saída: Vetores x e y contendo as aproximações.

1. Defina $x[0] \leftarrow x_0$
 2. Defina $y[0] \leftarrow y_0$
 3. **Para** $i \leftarrow 1$ **até** n **faça:**
 - a) Defina $x[i] \leftarrow x[0] + i \times h$
 - b) Defina $y[i] \leftarrow y[i - 1] + h \times f'(x[i - 1])$
-

3.6.2 Raízes de funções reais

Os próximos simuladores desenvolvidos foram para os métodos de encontrar a raiz de uma função real. O objetivo desses simuladores é permitir que os alunos insiram qualquer função e pontos iniciais que desejarem, observando como cada método se comporta a cada passo.

Algorithm 3 RK2

Entrada: h, x_0, y_0, n **Saída:** Vetores x e y contendo as aproximações.

1. Defina $x[0] \leftarrow x_0$
 2. Defina $y[0] \leftarrow y_0$
 3. **Para** $i \leftarrow 1$ **até** n **faça:**
 - a) Defina $x[i] \leftarrow x[0] + i \times h$
 - b) Calcule $K_1 \leftarrow h \times f'(x[i - 1])$
 - c) Calcule $K_2 \leftarrow h \times f'(x[i])$
 - d) Defina $y[i] \leftarrow y[i - 1] + \frac{K_1 + K_2}{2}$
-

Algorithm 4 RK4

Entrada: h, x_0, y_0, n **Saída:** Vetores x e y contendo as aproximações.

1. Defina $x[0] \leftarrow x_0$
 2. Defina $y[0] \leftarrow y_0$
 3. **Para** $i \leftarrow 1$ **até** n **faça:**
 - a) Defina $x[i] \leftarrow x[0] + i \times h$
 - b) Calcule $K_1 \leftarrow h \times f'(x[i - 1])$
 - c) Calcule $K_2 \leftarrow h \times f'(x[i]) + \frac{h}{2}$
 - d) Calcule $K_3 \leftarrow h \times f'(x[i]) + \frac{h}{2}$
 - e) Calcule $K_4 \leftarrow h \times f'(x[i])$
 - f) Defina $y[i] \leftarrow y[i - 1] + \frac{K_1 + 2 \times K_2 + 2 \times K_3 + K_4}{6}$
-

É importante notar que o simulador permite ao usuário inserir funções ou valores iniciais inválidos. Isso é intencional, para que os alunos percebam e se lembrem dos requisitos específicos de cada método.

Não há um critério de parada predefinido; o aluno pode continuar avançando no método até ficar satisfeito com os resultados.

Exceto pelo método de Newton, todos os outros métodos iteram a partir de dois pontos distintos. Portanto, há dois campos para o usuário inserir os valores dos pontos. Ao selecionar o método de Newton, o segundo ponto será ignorado.

Ao lado do gráfico, a partir da primeira iteração, serão exibidos os novos valores que cada ponto está assumindo. Isso é útil para os alunos que estão implementando seu próprio método ou testando no papel, permitindo comparar o resultado de seu trabalho

com cada iteração no site e verificar a consistência.

Como o objetivo é apenas ilustrar o comportamento dos métodos, os valores dos pontos apresentados têm uma precisão de apenas duas casas decimais.

Algorithm 5 Método da Bisseção

Entrada: Função f , intervalo $[a, b]$.

Saída: Aproximação da raiz.

1. **Se** $f(a) \times f(b) \geq 0$ **então:**
 - a) **Retorne** "Os pontos a e b devem ter sinais diferentes".
 2. Calcule $c \leftarrow (b + a)/2$.
 3. **Se** $f(c) \times f(a) < 0$ **então:**
 - a) $b \leftarrow c$.
 4. **Senão:**
 - a) $a \leftarrow c$.
 5. **Retorne** $(a + b)/2$ (*Aproximação da raiz*).
-

Algorithm 6 Método da Falsa Posição

Entrada: Função f , intervalo $[a, b]$.

Saída: Aproximação da raiz.

1. **Se** $f(a) \times f(b) \geq 0$ **então:**
 - a) **Retorne** "Os pontos a e b devem ter sinais diferentes".
 2. Calcule $x_0 \leftarrow \frac{a \times f(b) - b \times f(a)}{f(b) - f(a)}$.
 3. **Se** $f(x_0) \times f(a) < 0$ **então:**
 - a) $b \leftarrow x_0$.
 4. **Senão:**
 - a) $a \leftarrow x_0$.
 5. **Retorne** x_0 (*Aproximação da raiz*).
-

3.6.3 Ajustes de curvas

O simulador de ajustes de curvas foi desenvolvido para auxiliar os alunos na compreensão de como a distribuição dos pontos de dados e o grau do polinômio de ajuste impactam a precisão das previsões.

Algorithm 7 Método de Newton

Entrada: Função f , aproximação inicial x_{n-1} .

Saída: Aproximação da raiz.

1. Calcule $x_n \leftarrow x_{n-1} - \frac{f(x_{n-1})}{f'(x_{n-1})}$.
 2. **Retorne** x_n (*Aproximação da raiz*).
-

Algorithm 8 Método da Secante

Entrada: Função f , intervalo $[a, b]$, contador de iterações.

Saída: Aproximação da raiz.

1. Calcule $x_0 \leftarrow \frac{a \times f(b) - b \times f(a)}{f(b) - f(a)}$.
 2. **Se** contador $\%2 = 0$ **então:**
 - a) $a \leftarrow x_0$.
 3. **Senão:**
 - a) $b \leftarrow x_0$.
 4. Atualize o contador: $contador \leftarrow contador + 1$.
 5. **Retorne** x_0 (*Aproximação da raiz*).
-

Inicialmente, o usuário pode definir o intervalo do gráfico que deseja visualizar. Em seguida, é possível inserir os pontos de dados manualmente, um a um. Embora este método não seja prático para grandes volumes de dados, o objetivo do simulador é didático, focando na ilustração do processo de ajuste de curvas, e não em aplicações práticas extensivas.

Após a inserção dos pontos, o usuário seleciona o grau do polinômio de ajuste desejado. O simulador então gera e exibe um gráfico da função estimada com base nos parâmetros fornecidos, permitindo aos alunos visualizarem como diferentes configurações afetam a precisão do ajuste.

3.6.4 Interpolação

O simulador de interpolação foi criado para ajudar os alunos a compreenderem o comportamento de uma função interpolada a partir dos pontos fornecidos.

Existem várias técnicas para obter a interpolação de uma função, como o método de Newton, de Lagrange e a forma direta. No entanto, todas essas técnicas levam ao mesmo resultado, pois a função interpolada é única para um conjunto dado de pontos. Por essa razão, não oferecemos a opção de escolher entre diferentes métodos de interpolação no simulador, a fim de evitar confusões. Permitir a escolha do método poderia levar o aluno

a pensar que o resultado da interpolação seria diferente dependendo do método escolhido, o que não é o caso.

Os alunos podem definir o intervalo do gráfico e adicionar ou remover pontos conforme desejarem. O simulador inclui um exemplo padrão para facilitar a compreensão inicial. Após ajustar os pontos, um botão permite gerar a interpolação, que é exibida no gráfico junto com a definição da função interpolada.

Além disso, há um campo onde o usuário pode inserir um valor de x para obter o valor correspondente da função interpolada naquele ponto específico.

3.6.5 Integração Numérica

O objetivo deste simulador é permitir que os alunos observem como a quantidade de subintervalos e o método escolhido influenciam o resultado da integração numérica.

O usuário pode selecionar a função que deseja integrar (um exemplo padrão é fornecido para facilitar a compreensão), definir o intervalo de integração e determinar o número de subintervalos (que deve ser par para o método de Simpson). Além disso, o usuário pode escolher entre dois métodos de integração disponíveis.

Os dois métodos disponíveis são a regra dos trapézios e a regra de $1/3$ de Simpson. Embora a quadratura gaussiana também tenha sido ensinada, sua representação gráfica é mais abstrata, pois envolve a troca de variáveis. Diferentemente das regras de Newton-Cotes, como os métodos dos trapézios e de Simpson, a quadratura gaussiana não é iterativa, tornando sua visualização gráfica menos intuitiva para fins de simulação. Portanto, a quadratura gaussiana não foi incluída no simulador.

O simulador ajuda os alunos a entenderem como a escolha do método e a divisão do intervalo em subintervalos afetam a precisão da integração, fornecendo uma visualização clara e interativa dos resultados.

4 INSTALANDO E COMEÇANDO COM O ANGULAR

O site foi desenvolvido utilizando o framework Angular. Neste capítulo estarei ensinando os passos iniciais de como começar a utilizar o framework. Porém, caso o objetivo seja adicionar novas implementações ao site, o ideal é que estude-se por fora pelo menos o básico do framework.

4.1 PRÉ-REQUISITOS

Antes de começar, certifique-se de ter os seguintes pré-requisitos instalados em seu sistema:

- **Node.js:** Angular requer Node.js. Você pode baixá-lo e instalá-lo a partir do site oficial (<https://nodejs.org/>).
- **NPM (Node Package Manager):** NPM geralmente é instalado junto com o Node.js. Você pode verificar se está instalado executando `npm -v` no terminal.

4.2 INSTALANDO O ANGULAR CLI

O Angular CLI (Command Line Interface) facilita a criação e o gerenciamento de projetos Angular. Para instalar o Angular CLI, execute o seguinte comando no terminal:

```
npm install -g @angular/cli
```

4.3 CLONANDO O PROJETO

Após instalar o Angular CLI, você pode clonar o projeto que está hospedado no repositório: (<https://github.com/CaioCohen/CalcNumerico>)

4.4 ATUALIZANDO UM PROJETO ANGULAR E GERENCIAMENTO DE VERSÕES DO NODE COM NVM

Conforme o tempo passa, novas versões do framework são liberadas, é importante que o desenvolvedor saiba lidar com a característica dinâmica do site. Aquele que estiver encarregado de desenvolver o site pode abordar a situação de duas formas: atualizar o framework do site (o recomendado) ou regredir o framework instalado em sua máquina para se acomodar à versão do site utilizando o Node Version Manager (NVM).

4.4.1 Atualizando um Projeto Angular

Para garantir que seu projeto Angular esteja utilizando a versão mais recente, siga os passos abaixo:

1. **Atualize o Angular CLI globalmente:** Primeiro, é necessário atualizar o Angular CLI instalado globalmente em seu sistema:

```
npm uninstall -g @angular/cli  
npm install -g @angular/cli@latest
```

2. **Atualize o Angular CLI no projeto:** Dentro da pasta do seu projeto, execute:

```
ng update @angular/cli @angular/core
```

Isso atualizará o Angular CLI e os pacotes principais para a versão mais recente.

3. **Verifique as dependências:** Se houver dependências específicas ou pacotes de terceiros em seu projeto, pode ser necessário atualizá-los também (atualmente, existe apenas um pacote externo sendo usado, o mathjs):

```
ng update <nome-do-pacote>
```

4. **Testes e correções:** Após a atualização, é essencial rodar os testes do projeto e verificar se há necessidade de ajustes no código devido a possíveis mudanças ou descontinuações de APIs.

4.4.2 Gerenciamento de Versões do Node.js com NVM

Para garantir que você esteja utilizando a versão correta do Node.js para o seu projeto, é recomendável usar o Node Version Manager (NVM). O NVM permite alternar facilmente entre diferentes versões do Node.js.

1. **Instalação do NVM:** Se você ainda não tem o NVM instalado, clique para baixar o nvm-setup.exe da versão mais recente no link:

```
https://github.com/coreybutler/nvm-windows/releases
```

Após a instalação, reinicie o terminal

2. **Listar versões disponíveis:** Para ver uma lista de todas as versões do Node.js disponíveis para instalação:

```
nvm ls-remote
```

3. **Instalar a versão necessária:** Para instalar uma versão específica do Node.js, use:

```
nvm install vX.X.X
```

Onde `vX.X.X` representa a versão do Node.js desejada.

4. **Utilizar uma versão específica:** Para definir uma versão específica do Node.js para o seu projeto, execute:

```
nvm use vX.X.X
```

5. **Definir a versão padrão:** Se quiser definir uma versão padrão do Node.js para o seu ambiente, use:

```
nvm alias default vX.X.X
```

6. **Instalar a versão correta do Angular:** Uma vez que esteja utilizando a versão correspondente do node, basta usar:

```
npm install @angular/cli -g
```

Dessa forma você estará utilizando a versão do Angular correspondente à do projeto.

4.5 EXECUTANDO O PROJETO ANGULAR

Para navegar até o diretório do projeto e iniciar o servidor de desenvolvimento, execute os seguintes comandos:

```
cd nome-do-projeto  
ng serve
```

o diretório deve ser aquele diretamente pai da pasta `.angular` (você deve conseguir visualizá-la olhando para os filhos).

Por padrão, o servidor de desenvolvimento será iniciado em `http://localhost:4200`. Você pode abrir este URL em seu navegador para ver seu projeto Angular em funcionamento.

4.6 ESTRUTURA DO PROJETO ANGULAR

A estrutura do projeto estará da seguinte forma:

```

nome-do-projeto/
├── e2e/
├── node_modules/
├── src/
│   ├── app/
│   │   ├── components/
│   │   └── main/
│   ├── assets/
│   ├── environments/
│   ├── index.html
│   ├── main.ts
│   └── styles.css
├── angular.json
├── package.json
└── tsconfig.json

```

- `src/`: Contém o código-fonte do aplicativo.
- `app/`: Diretório principal para os componentes e módulos do Angular.
- `components/`: Contém os componentes auxiliares do site, como a navbar e o gráfico.
- `main/`: Contém as páginas onde estarão exibidas cada um dos tópicos abordados.
- `assets/`: Diretório para arquivos estáticos, como imagens, fontes e vídeos.
- `environments/`: Diretório para arquivos de configuração de ambiente.
- `index.html`: O arquivo HTML principal.
- `main.ts`: O arquivo principal de entrada do Angular.
- `styles.css`: O arquivo principal de estilos.

4.7 CRIANDO COMPONENTES

Componentes são os blocos de construção principais de um aplicativo Angular. Para criar um novo componente, use o seguinte comando:

```
ng generate component nome-do-componente
```

Isso criará um novo diretório com arquivos para o componente no diretório `src/app/`.

4.8 RESUMO

Agora que uma noção básica do projeto já foi dada, basta estudar o básico de como o framework funciona e como o projeto está estruturado para continuar a implementação do site.

5 EXPERIÊNCIA DOS ALUNOS

Este capítulo aborda a percepção e o feedback dos alunos em relação ao site interativo desenvolvido para facilitar o estudo de cálculo numérico. Para avaliar a eficácia do projeto e identificar potenciais melhorias, foi elaborado um formulário direcionado aos estudantes da universidade, que representam o público-alvo principal da plataforma.

O questionário contemplou diversos aspectos, como a usabilidade do site, a clareza do conteúdo teórico, a qualidade e a funcionalidade dos simuladores interativos, entre outros. O objetivo era entender como os recursos disponíveis contribuíram para a experiência de aprendizado dos usuários, assim como obter sugestões para futuros aperfeiçoamentos.

A seguir, serão apresentadas as respostas dos alunos, analisadas quantitativa e qualitativamente, fornecendo uma visão crítica sobre a eficácia da plataforma e seu potencial como ferramenta educacional. Esses dados são essenciais para validar o impacto do projeto e para explorar possíveis evoluções da plataforma.

5.1 RESULTADOS

Foi pedido para que os alunos avaliassem, quantitativamente ou categoricamente, diversos aspectos do site, e os resultados recolhidos foram os seguintes:

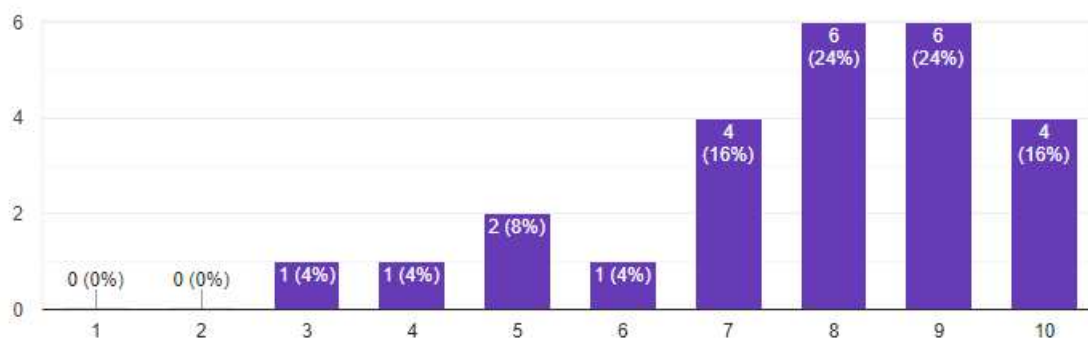


Figura 17 – Avaliação quantitativa de 1 a 10 da interface do site.

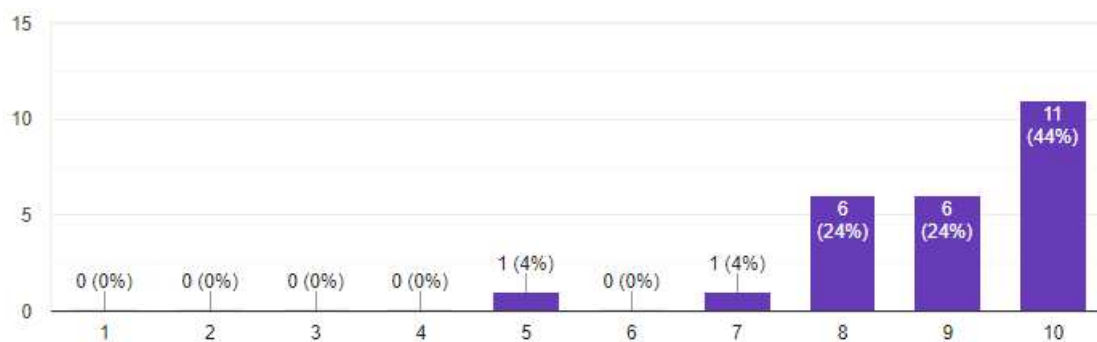


Figura 18 – Avaliação quantitativa de 1 a 10 da usabilidade do site.

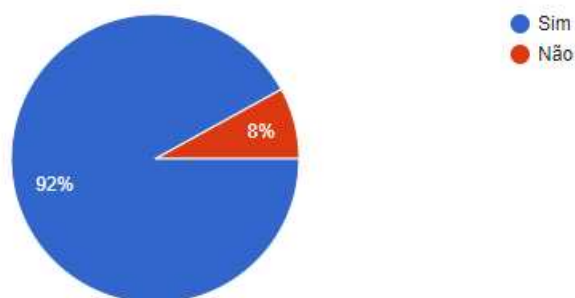


Figura 19 – Se a teoria do site abrange todo o conteúdo visto em sala

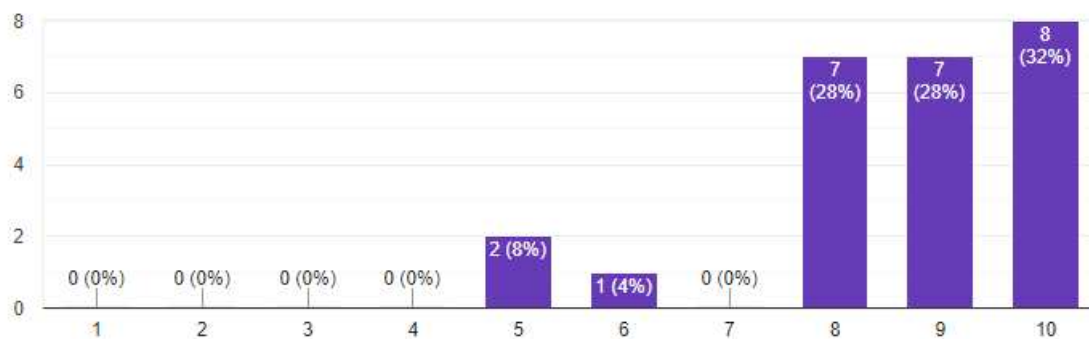


Figura 20 – Avaliação quantitativa da clareza e facilidade de entendimento da teoria

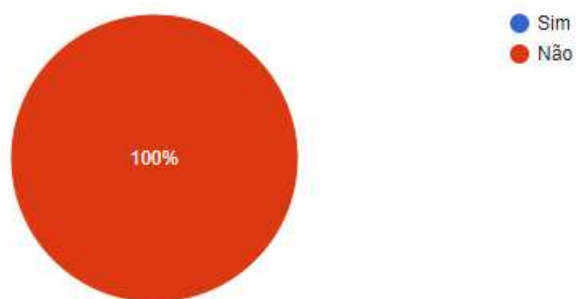


Figura 21 – Se o aluno teve dificuldade em utilizar a aplicação dos métodos

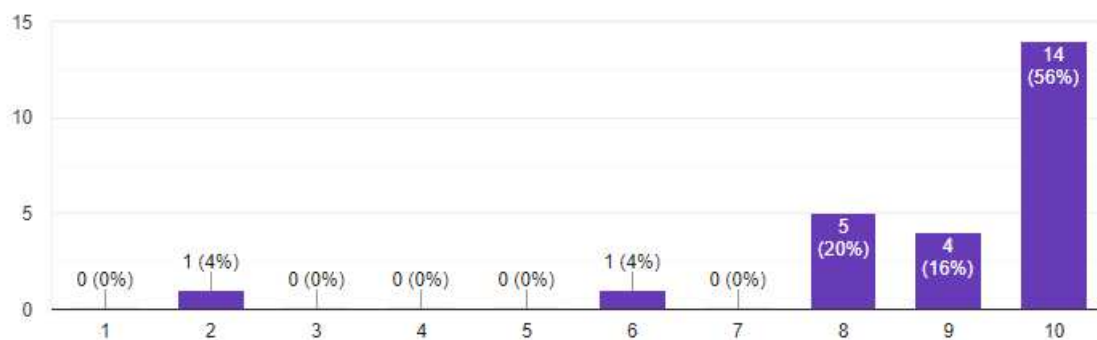


Figura 22 – Avaliação quantitativa do quão útil os alunos acreditam que o site teria sido quando eles cursavam a disciplina

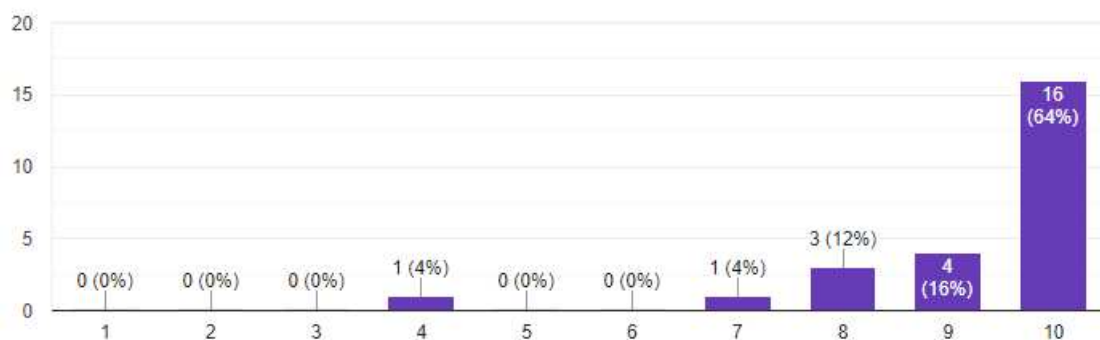


Figura 23 – Avaliação quantitativa do quão útil os alunos acreditam que o site será para futuros alunos da disciplina

6 PLANOS FUTUROS E CONCLUSÃO

6.1 PLANOS FUTUROS

A visão futura para o site é transformá-lo em um sistema centralizado de estudo para as disciplinas do currículo do Bacharelado em Ciências da Computação (BCC) da UFRJ. Este ambiente busca proporcionar aos alunos um espaço confortável e direcionado para o estudo, ajudando-os a evitar reprovações e trancamentos de disciplinas. Com base em experiências pessoais, sabe-se que muitos alunos optam por trancar disciplinas quando se sentem perdidos no conteúdo, uma vez que encontrar material de estudo específico na internet pode ser desafiador para algumas matérias do nosso currículo.

Há também casos de estudantes que, inicialmente, obtiveram baixo desempenho nas primeiras avaliações, mas que, posteriormente, conseguiram acesso a materiais de estudo adequados e optaram por continuar na disciplina, culminando em sua aprovação.

O diferencial deste site é permitir que os próprios alunos, orientados pelos professores, disponibilizem materiais que refletem o conteúdo específico e as demandas das disciplinas, com ênfase nas partes mais importantes. Isso contrasta com a busca tradicional na biblioteca ou na internet por material genérico sobre as disciplinas.

Adicionalmente, a integração de ferramentas interativas no site, algo raramente encontrado na internet, visa aprofundar ainda mais o entendimento dos estudantes. Acredita-se que o site possui potencial para cobrir todo o currículo do BCC e, eventualmente, expandir-se para outros currículos, incluindo o ensino médio.

6.2 CONCLUSÃO

Este TCC apresentou um resumo teórico dos principais tópicos abordados na disciplina de Cálculo Numérico, bem como um relato detalhado do processo de criação do site, desde a concepção da ideia até o produto final. Foram discutidos os insights obtidos pelo desenvolvedor ao longo do processo, incluindo os desafios enfrentados e as soluções implementadas.

O site desenvolvido não só oferece uma plataforma de estudo centralizada e específica para os alunos de Cálculo Numérico da UFRJ, mas também introduz uma nova abordagem para o ensino e a aprendizagem, integrando explicações teóricas, ferramentas interativas e materiais de estudo elaborados com a colaboração de alunos e professores. Este ambiente visa melhorar a compreensão dos alunos, elevar o CR médio e reduzir a taxa de evasão, proporcionando uma alternativa eficiente e direcionada em comparação com a busca por materiais genéricos na internet.

Adicionalmente, o TCC inclui uma introdução ao framework utilizado para o desen-

volvimento do site, oferecendo uma base para implementações futuras e facilitando a expansão do projeto para outras disciplinas e currículos. A aplicação, atualmente em estado Beta, representa um avanço significativo e promissor, marcando o primeiro passo de um projeto com potencial para impactar positivamente a comunidade acadêmica.

REFERÊNCIAS

Angular — angular.dev. <https://angular.dev/>.

Figma: a ferramenta de design de interface colaborativa — figma.com. <https://www.figma.com/pt-br/?context=confirmLocalePref>.

Taxa de convergência – Wikipédia, a enciclopédia livre — pt.wikipedia.org. https://pt.wikipedia.org/wiki/Taxa_de_converg%C3%Aancia.

Teorema do valor intermediário – Wikipédia, a enciclopédia livre — pt.wikipedia.org. https://pt.wikipedia.org/wiki/Teorema_do_valor_intermedi%C3%A1rio.

Burden, R. L., Faires, J. D., and Burden, A. M. *Análise numérica*. Cengage Learning, 2016.

de Oliveira, R. Site do professor raymundo de oliveira. <https://www.raymundodeoliveira.eng.br/calculo.html>.

de Souza, C. C. Calculadora numérica — calculadora-numerica.ic.ufrj.br. <https://calculadora-numerica.ic.ufrj.br/>, 2024.

Mark Otto, J. T., and contributors, B. Bootstrap — getbootstrap.com. <https://getbootstrap.com/>.

Nirschl, G. C. Site de calculo numerico da ifsp. <https://vtp.ifsp.edu.br/nev/#calcnum>.

Ruggiero, M. *Cálculo numérico, aspectos teóricos e computacionais*. Pearson, 2002.