

UNIVERSIDADE FEDERAL DO PAMPA

Rafaela Viero Robe

**Aplicação do Método da Imagem Intervalar
nas Funções Densidade de Probabilidade
das Variáveis Aleatórias Contínuas**

Alegrete
2017

Rafaela Viero Robe

**Aplicação do Método da Imagem Intervalar nas
Funções Densidade de Probabilidade das Variáveis
Aleatórias Contínuas**

Trabalho de Conclusão de Curso apresentado
ao Curso de Graduação em Ciência da Com-
putação da Universidade Federal do Pampa
como requisito parcial para a obtenção do tí-
tulo de Bacharel em Ciência da Computação.

Orientador: Profa. Ma. Alice Fonseca Finger

Alegrete
2017

Rafaela Viero Robe

**Aplicação do Método da Imagem Intervalar nas
Funções Densidade de Probabilidade das Variáveis
Aleatórias Contínuas**

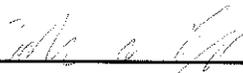
Trabalho de Conclusão de Curso apresentado
ao Curso de Graduação em Ciência da Com-
putação da Universidade Federal do Pampa
como requisito parcial para a obtenção do tí-
tulo de Bacharel em Ciência da Computação.

Trabalho de Conclusão de Curso defendido e aprovado em 29 de novembro de 2017

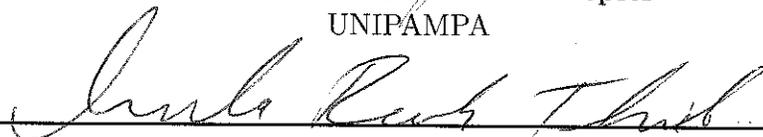
Banca examinadora:



Profa. Ma. Alice Fonseca Finger
Orientador
UNIPAMPA



Prof. Dr. Fabio Natanael Kepler
UNIPAMPA



Prof. Dr. Marcelo Resende Thielo
UNIPAMPA

“isso de querer
ser exatamente aquilo
que a gente é
ainda vai
nos levar além”
(Paulo Leminski)

RESUMO

Quando trabalhamos com cálculos numéricos em ambientes computacionais, operamos sobre números de ponto flutuante. Dessa forma, o resultado é apenas uma aproximação de um valor real e erros gerados por arredondamentos ou truncamentos podem levar a resultados incorretos, não podendo ser afirmada a exatidão das respostas estimadas sem o auxílio de uma análise de erro. Ao utilizar-se intervalos para representação de valores reais, torna-se possível controlar a propagação desses erros, pois resultados intervalares carregam consigo a segurança de sua qualidade. Para transformar a função intervalar a partir de uma função real, geralmente é usado o método da extensão intervalar. Trabalhos utilizando indicadores estatísticos mostram que a complexidade dos algoritmos utilizando o método da imagem intervalar é muito maior do que utilizando o método da extensão intervalar, porém ainda não existe nenhuma análise quanto à utilização do método da imagem intervalar aplicado nas funções com distribuição de probabilidade. Nesse contexto, o presente trabalho possui o objetivo de utilizar o método da imagem intervalar, ao invés da extensão intervalar, para resolver as funções densidade de probabilidade com entradas intervalares, a fim de obter resultados mais exatos em um tempo razoável de processamento. Como último objetivo, foram analisadas as complexidades computacionais para computar tais funções com o método da imagem intervalar. Como resultado, certifica-se que, ao utilizar o método de imagem intervalar, em alguns exemplos os intervalos solução foram retornados com menor diâmetro, porém o esforço computacional aumentou de uma ordem constante para uma ordem exponencial.

Palavras-chave: Análise de complexidade. Extensão Intervalar. Imagem Intervalar.

ABSTRACT

When we work with numerical calculations in computational environments, we operate on floating-point numbers. Thus, the result is only an approximation of a real value and errors generated by rounding or truncations can lead to incorrect results, and the accuracy of the estimated responses can not be affirmed without the aid of an error analysis. By using intervals to represent real values, it becomes possible to control the propagation of these errors, because interval results carry with them the safety of their quality. To transform the interval function from a real function, the interval extension method is usually used. Statistical work shows that the complexity of the algorithms using the interval image method is much greater than using the interval extension method, but there is still no analysis regarding the use of the interval image method applied in the functions with probability distribution. In this context, the present work aims to use the interval image method, instead of the interval extension, to solve the probability density functions with interval inputs, in order to obtain more accurate results in a reasonable processing time. As a last objective, computational complexities were analysed to compute such functions with the interval image method. As a result, it is ensured that when using the interval image method, in some examples the solution intervals were returned with smaller diameter, but the computational effort increased from a constant order to a exponential order.

Key-words: Analysis of complexity. Interval Extension. Interval Image.

LISTA DE FIGURAS

Figura 1 – Representação Geométrica do Conjunto IR	19
Figura 2 – Representação Geométrica da Distância em R	22
Figura 3 – Representação Geométrica do Módulo de um intervalo em R	22
Figura 4 – Representação Geométrica do Diâmetro de um intervalo em R	22
Figura 5 – Representação Geométrica do Ponto Médio de um intervalo em R	23
Figura 6 – Indicadores Estatísticos Intervalares	29
Figura 7 – Ilustração no plano para as operações de adição, subtração, multiplicação e divisão utilizando aritmética multidimensional RDM	31
Figura 8 – Gráfico do Comportamento da Distribuição Uniforme	51
Figura 9 – Gráfico do Comportamento da Distribuição Exponencial	52
Figura 10 – Gráfico do Comportamento da Distribuição de Pareto	54
Figura 11 – Gráfico do Comportamento da Distribuição Weibull	55

LISTA DE TABELAS

Tabela 1 – Operação de Multiplicação	21
Tabela 2 – Operação de Divisão	21
Tabela 3 – Esforço computacional das distribuições	27
Tabela 4 – Indicadores Estatísticos Intervalares e Complexidade dos Problemas dos Indicadores Estatísticos Intervalares	29
Tabela 5 – Problemas da Estatística Descritiva Intervalar, complexidade dos Problemas com extensão intervalar e com imagem intervalar	30
Tabela 6 – Funções Implementadas por Varjão	36
Tabela 7 – Aplicação com a primitiva da função nas formas real e intervalar dos Exemplos 1 e 2, utilizando o método da extensão intervalar.	43
Tabela 8 – Aplicação com a primitiva da função na forma intervalar do Exemplo 1 e 2, utilizando o método da imagem intervalar	44
Tabela 9 – Aplicação com a primitiva da função nas formas real e intervalar dos Exemplos 1 e 2, utilizando o método da extensão intervalar.	45
Tabela 10 – Aplicação com a primitiva da função na forma intervalar dos Exemplos 1 e 2, utilizando o método da imagem intervalar.	45
Tabela 11 – Aplicação com a primitiva da função nas formas real e intervalar dos Exemplos 1 e 2, utilizando o método da extensão intervalar.	46
Tabela 12 – Aplicação com a primitiva da função na forma intervalar dos Exemplos 1 e 2, utilizando o método da imagem intervalar.	46
Tabela 13 – Aplicação com a primitiva da função nas formas real e intervalar dos Exemplos 1 e 2, utilizando o método da extensão intervalar.	47
Tabela 14 – Aplicação com a primitiva da função na forma intervalar dos Exemplos 1 e 2, utilizando o método da imagem intervalar.	47
Tabela 15 – Análise dos Diâmetros da Distribuição Uniforme	49
Tabela 16 – Análise dos Erros da Distribuição Uniforme	50
Tabela 17 – Análise dos Diâmetros da Distribuição Exponencial	51
Tabela 18 – Análise dos Erros da Distribuição Exponencial	52
Tabela 19 – Análise dos Diâmetros da Distribuição de Pareto	53
Tabela 20 – Análise dos Erros da Distribuição de Pareto	53
Tabela 21 – Análise dos Diâmetros da Distribuição Weibull	54
Tabela 22 – Análise dos Erros da Distribuição Weibull	55
Tabela 23 – Intervalo Dividido em 35 Subdivisões.	56
Tabela 24 – Análise Numérica dos Intervalos com 35 subdivisões.	56
Tabela 25 – Esforço Computacional das Distribuições	57

SUMÁRIO

1	INTRODUÇÃO	15
1.1	Objetivos	16
1.2	Organização do Trabalho	17
2	FUNDAMENTAÇÃO TEÓRICA	19
2.1	Operações Aritméticas em IR	19
2.1.1	Otimização de Cálculos para Multiplicação e Divisão Intervalar	21
2.2	Topologia do Conjunto IR	22
2.3	Funções Intervalares	23
2.3.1	Imagem Intervalar de uma Função Real	23
2.3.2	Extensão Intervalar de uma Função Real	24
2.4	Complexidade de Algoritmos	25
2.4.1	Classes de Comportamento Assintótico	26
3	TRABALHOS RELACIONADOS	27
3.1	Extensão intervalar para as variáveis aleatórias com distribuições Uniforme, Normal, Gama, Exponencial e Pareto	27
3.2	Análise da Complexidade Computacional de Problemas de Estatística Descritiva com Entradas Intervalares	28
3.3	Diferenças entre Aritmética de Moore e Aritmética Intervalar RDM	30
3.4	Definição Intervalar da Função Densidade de Probabilidade com Distribuição Beta: Aritmética de Moore e Aritmética Multidimensional RDM	34
3.5	Aspectos Formais da Correção e Otimização da Computação Intervalar	35
3.6	IntPy: Computação Científica Auto Validável em Python	36
4	FUNÇÕES DENSIDADE DE PROBABILIDADE	39
4.1	Distribuição Uniforme	39
4.2	Distribuição Exponencial	40
4.3	Distribuição de Pareto	40
4.4	Distribuição Weibull	41
5	IMAGEM INTERVALAR PARA AS VARIÁVEIS ALEATÓRIAS COM DISTRIBUIÇÃO UNIFORME, EXPONENCIAL, PARETO E WEIBULL	43
5.1	Distribuição Uniforme	43
5.2	Distribuição Exponencial	44

5.3	Distribuição de Pareto	45
5.4	Distribuição Weibull	46
6	ANÁLISE NUMÉRICA	49
6.1	Distribuição Uniforme	49
6.2	Distribuição Exponencial	51
6.3	Distribuição de Pareto	53
6.4	Distribuição Weibull	54
6.5	Análise Numérica com mais Subdivisões	56
7	ANÁLISE DE COMPLEXIDADE	57
7.1	Imagem Intervalar	57
8	CONCLUSÃO	59
8.1	Trabalhos Futuros	60
	REFERÊNCIAS	61
	ANEXOS	63
	ANEXO A – ANEXO A- IMPLEMENTAÇÃO DO MÉTODO DA IMAGEM INTERVALAR	65
	ANEXO B – ANEXO B - ERROS	69

1 INTRODUÇÃO

O primeiro relato da utilização de intervalos como estimativa formal de uma grandeza foi o algoritmo de Arquimedes para estimar o valor de π . Entre 3.000 e 4.000 anos atrás, o valor de π era obtido por tentativa e erro, sem fazer nenhum cálculo para considerar esses possíveis erros. A busca pelo valor exato de π levou ao desenvolvimento de novos conceitos e técnicas, como limites e algoritmos iterativos, além de proporcionar maior precisão nos resultados. Arquimedes, através de uma sucessão de polígonos inscritos e circunscritos com número de lados crescente, conseguiu gerar uma sequência convergente de intervalos para estimar essa constante. Por outro lado, a aritmética intervalar associada à computação científica surgiu com alguns estudos isolados na década de 1950, tendo esses estudos chamado a atenção de vários pesquisadores. Em 1958, Sunaga apresenta seu trabalho, onde são investigadas as regras que definem as operações aritméticas entre intervalos. Porém, somente em 1962, com o primeiro livro publicado por Moore (1966), esses resultados passaram a receber mais atenção.

Os problemas da computação científica se concentram fundamentalmente em três aspectos: na criação do modelo computacional que reflita de forma mais fiel possível a realidade em questão, no controle e análise dos erros que ocorrem no processo computacional e na escolha das técnicas de programação adequadas para desenvolvimento de software científico. Algoritmos convencionais, normalmente utilizados em computação científica, chamados de algoritmos pontuais, computam uma estimativa para uma resposta, e, talvez, um erro estimado. O usuário não pode afirmar a exatidão da resposta estimada sem o auxílio de uma análise de erro, que é extensa, dispendiosa e nem sempre viável.

Existem três fontes de erros em computação numérica: a propagação de erro nos dados e parâmetros iniciais, que é a mais difícil de ser contornada, porque não é possível torná-la arbitrariamente pequena via computação adicional; o erro de arredondamento e o erro de truncamento. As técnicas intervalares consistem em uma alternativa para alcançar limites garantidos para os resultados de computações científicas, através do controle rigoroso e automático do erro do resultado (MESQUITA, 2004). Justificando-se, assim, o emprego de técnicas intervalares para contornar erros gerados pela aritmética de ponto flutuante.

Técnicas intervalares manipulam dados e parâmetros iniciais como intervalos, com o indicativo do erro máximo presente nestes valores antes que os mesmos sejam introduzidos no computador. A propagação do erro nos dados iniciais e a acumulação do erro de arredondamento em qualquer sequência finita de operações aritméticas podem ser ambas rigorosamente controladas simplesmente pela utilização de aritmética de máquina (MESQUITA, 2004). Desta forma algoritmos intervalares, em contraste com os algoritmos pontuais, computam um intervalo como solução, com a garantia de que a resposta pertence a este intervalo.

A matemática intervalar fornece uma alternativa aos erros ocasionados por aproximações utilizando intervalos, os quais foram definidos com o propósito de automatizar a análise do erro computacional. Na aritmética intervalar, um valor real x é aproximado por um intervalo fechado X que possui limites inferior e superior, da forma $X = [\underline{x}, \bar{x}]$, de modo que este intervalo contenha o valor real x (MOORE, 1966). Dessa forma, todos os cálculos reais precisam ser substituídos por cálculos que utilizem aritmética intervalar, além de que o comprimento deste intervalo pode ainda ser utilizado como medida para avaliar a qualidade da aproximação obtida. Para transformar uma função intervalar a partir de uma função real, geralmente são usados dois métodos: o método da extensão intervalar, é um método mais simples de fazer a transformação; ou o método da imagem intervalar, que é mais trabalhoso, porém em muitos casos é o mais exato.

Dentre as diversas soluções de implementação possíveis para um dado problema, é preciso avaliar qual é a mais adequada e, depois disso, julgar quanto à otimalidade da solução desenvolvida. Para isso, existe a análise de algoritmos, a qual tem por objetivo melhorar, quando possível, o desempenho desse algoritmo e qualificá-lo em termos de sua eficiência. Existem diversos critérios para a avaliação de um algoritmo, dentre eles: quantidade de memória ocupada, exatidão da saída, processamento requerido e tempo de processamento (TOSCANI; VELOSO, 2009).

Na literatura as funções de densidade de probabilidade das variáveis aleatórias só foram implementadas utilizando o método da extensão intervalar, baseado na matemática intervalar de Moore (1966) e na aritmética de exatidão máxima de Kulisch e Miranker (2014). O objetivo deste trabalho é utilizar o método da imagem intervalar para definir essas funções intervalares a partir de funções reais e assim poder fazer uma comparação com os resultados já obtidos quando utilizado o método da extensão intervalar. Para tanto, a implementação do método da imagem intervalar será desenvolvido utilizando o pacote de extensão para programação intervalar da linguagem de programação Python, denominado IntPy (BARRETO, 2016), o qual já possui definido o tipo de intervalo e operações sobre esse. Por fim, será analisada a qualidade dos intervalos obtidos a fim de comparar resultados com as funções em que foram utilizados o método da extensão intervalar.

1.1 Objetivos

O objetivo principal deste trabalho é utilizar o método da imagem intervalar para definir funções de densidade de probabilidade das variáveis aleatórias contínuas com entradas intervalares a partir de funções reais e, assim, poder comparar com os resultados já obtidos em outros trabalhos que utilizaram o método da extensão intervalar. A fim de atingir o objetivo principal, os objetivos específicos a serem executados são listados e detalhados logo abaixo:

- Transformar as funções de densidade de probabilidade das variáveis aleatórias contínuas reais em intervalares a partir do método da imagem intervalar;
- Estudar as funções de densidade de probabilidade das variáveis aleatórias contínuas Uniforme, Exponencial, Pareto e Weibull nas suas formas reais;
- Aplicar nas funções estudadas o método da imagem intervalar, esse desenvolvido na linguagem de programação Python;
- Coletar os resultados da aplicação do método da imagem intervalar nas funções de densidade;
- Após aplicado o método da imagem intervalar e coletado os resultados, será feito um estudo e a análise numérica desses;
- Por fim, realizar a análise de complexidade desses resultados, verificando em que classe de complexidade esses estarão, comparando com os resultados já presentes na literatura utilizando o método de extensão intervalar.

1.2 Organização do Trabalho

O presente trabalho organiza-se da seguinte forma:

O [Capítulo 2](#) apresenta uma fundamentação teórica sobre matemática intervalar, citando os principais conceitos, as operações aritméticas básicas, a otimização de cálculos para multiplicação e divisão intervalar, a topologia do conjunto \mathbb{R} , o método da imagem intervalar e o método da extensão intervalar. Também são apresentadas as descrições sobre complexidade de algoritmos, com sua definição e a sua divisão em classes de problemas. Também é abordado nesse capítulo a complexidade intervalar.

No [Capítulo 3](#), alguns trabalhos já desenvolvidos, relacionados ao tema matemática intervalar e o uso do método de extensão intervalar, são apresentados de modo que esses são contribuintes em conceitos e auxiliam no presente trabalho. No [Capítulo 4](#) descrevem-se as funções densidade de probabilidade das variáveis aleatórias contínuas com distribuições nas duas formas, com entradas reais e intervalares, apresentando suas respectivas fórmulas.

No [Capítulo 5](#) são apresentados os intervalos encapsuladores gerados a partir do método da imagem intervalar para as funções de densidade de probabilidade das variáveis aleatórias contínuas com distribuições Uniforme, Exponencial, de Pareto e Weibull.

No [Capítulo 6](#), desenvolve-se a análise numérica onde são apresentados os resultados obtidos a partir da computação da função densidade de probabilidade com distribuição Uniforme, Exponencial, de Pareto e Weibull, na forma intervalar. A análise de complexidade a partir das métricas de erros é apresentada no [Capítulo 7](#) com o objetivo de apurar a qualidade dos resultados intervalares obtidos.

No [Capítulo 8](#) apresentam-se as conclusões obtidas no presente trabalho. Reservou-se este espaço para também apresentar as perspectivas de trabalhos futuros, os quais se deseja desenvolver a partir do conhecimento adquirido no presente trabalho e dar continuidade em trabalhos relacionados.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão apresentadas as principais definições que se fazem necessárias no estudo da Matemática Intervalar, como: um breve histórico sobre essa, mostrando como surgiu; a definição do conjunto de números reais intervalares IR ; principais operações aritméticas. Por fim, apresenta-se uma breve revisão sobre as funções intervalares, que se farão importantes para o desenvolvimento desse trabalho.

2.1 Operações Aritméticas em IR

Define-se IR como sendo o conjunto de todos os intervalos reais.

$$IR = \{[\underline{x}_1; \overline{x}_2] | \underline{x}_1; \overline{x}_2 \in R, \underline{x}_1 \leq \overline{x}_2\}$$

Abaixo apresenta-se uma representação geométrica do conjunto IR .

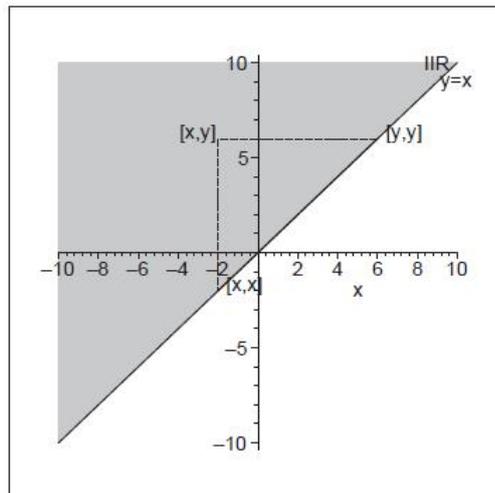


Figura 1 – Representação Geométrica do Conjunto IR

Podemos observar na [Figura 1](#) de [Moore \(1966\)](#), os pontos na diagonal onde $y=x$ entre (x, x) e (y, y) que representam os números reais contidos no intervalo $[x; y]$ e este intervalo é representado pelo vértice (x, y) do triângulo. Também podemos observar que todo número real $x \in R$ pode ser visto como um intervalo de IR . Para tanto, basta identificar os pontos $x \in R$ com os intervalos pontuais $X = [x; x] \in IR$.

Sejam $A, B \in IR$ dois intervalos de reais. As operações de soma, subtração, multiplicação e divisão em IR são definidas por $A * B = \{a * b | a \in A, b \in B\}$, onde $*$ $\in \{+, -, \cdot, /\}$ é quaisquer uma das quatro operações aritméticas. Assume-se também no caso da divisão que $0 \notin B$.

Soma Intervalar: Sejam $A, B \in IR$ dois intervalos de reais, com $A = [\underline{a}_1; \overline{a}_2]$ e $B = [\underline{b}_1; \overline{b}_2]$.

$$\text{Então, } A + B = [(\underline{a}_1 + \underline{b}_1); (\overline{a}_2 + \overline{b}_2)]$$

Exemplo: Sejam $A = [2; 3]$ e $B = [4; 5]$. Temos $A + B = [2; 3] + [4; 5] = [2+4; 3+5] = [6; 8]$

Subtração Intervalar: Sejam $A, B \in IR$ dois intervalos de reais, com $A = [\underline{a}_1; \overline{a}_2]$ e $B = [\underline{b}_1; \overline{b}_2]$.

$$\text{Então, } A - B = [(\underline{a}_1 - \overline{b}_2); (\overline{a}_2 - \underline{b}_1)]$$

Exemplo: Sejam $A = [1; 5]$ e $B = [3; 7]$. Temos $A - B = [1; 5] - [3; 7] = [1 - 7; 5 - 3] = [-6; 2]$

Negativo de um Intervalo: Seja $A \in IR$ um intervalo de reais, com $A = [\underline{a}_1; \overline{a}_2]$.

$$\text{Então, } -A = [-\overline{a}_2; -\underline{a}_1]$$

Exemplo: Seja $A = [8; 9]$. Temos $-A = -[8; 9] = [-9; -8]$

Multiplicação Intervalar: Sejam $A, B \in IR$ dois intervalos de reais, com $A = [\underline{a}_1; \overline{a}_2]$ e $B = [\underline{b}_1; \overline{b}_2]$.

$$\text{Então, } A.B = [\min\{\underline{a}_1.\underline{b}_1, \underline{a}_1.\overline{b}_2, \overline{a}_2.\underline{b}_1, \overline{a}_2.\overline{b}_2\}; \max\{\underline{a}_1.\underline{b}_1, \underline{a}_1.\overline{b}_2, \overline{a}_2.\underline{b}_1, \overline{a}_2.\overline{b}_2\}]$$

Prova: Para que a operação de multiplicação seja fechada em IR , precisamos que $a.b \in A.B$, sempre que $a \in A$ e $b \in B$, ou seja, que $A.B = [\min\{a.b/a \in A, b \in B\}; \max\{a.b/a \in A, b \in B\}]$.

Exemplo: Sejam $A = [-2; 3]$ e $B = [-4; 5]$.

Temos $A.B = [-2; 3].[-4; 5] = [\min\{(-2).(-4), (-2).5, 3.(-4), 3.5\}; \max\{(-2).4, (-2).5, 3.(-4), 3.5\}] = [\min\{8, -10, -12, 15\}; \max\{8, -10, -12, 15\}] = [-12; 15]$.

Divisão Intervalar: Sejam $A, B \in IR$ dois intervalos de reais, com $A = [\underline{a}_1; \overline{a}_2]$ e $B = [\underline{b}_1; \overline{b}_2]$ e $0 \notin B$.

$$\text{Então, } \frac{A}{B} = A.B^{-1} = \left[\min \left\{ \frac{a_1}{b_2}, \frac{a_1}{b_1}, \frac{a_2}{b_2}, \frac{a_2}{b_1} \right\}; \max \left\{ \frac{a_1}{b_2}, \frac{a_1}{b_1}, \frac{a_2}{b_2}, \frac{a_2}{b_1} \right\} \right] \text{ com } 0 \notin [b_1; b_2].$$

Exemplo: Sejam $A = [2; 3]$ e $B = [4; 5]$.

$$\begin{aligned} \text{Temos } \frac{[2; 3]}{[4; 5]} &= \left[\min \left\{ \frac{2}{5}, \frac{2}{4}, \frac{3}{5}, \frac{3}{4} \right\}; \max \left\{ \frac{2}{5}, \frac{2}{4}, \frac{3}{5}, \frac{3}{4} \right\} \right] \\ &= \left[\frac{2}{5}; \frac{3}{4} \right] \end{aligned}$$

2.1.1 Otimização de Cálculos para Multiplicação e Divisão Intervalar

Para fins de implementação em computadores, podemos otimizar os cálculos feitos no caso da multiplicação e da divisão, desenvolvidos por OLIVEIRA (1997), considerando-se os sinais dos extremos dos intervalos, que nos levam a analisar os seguintes nove casos: Sejam $A, B \in IR$ dois intervalos de reais, com $A = [\underline{a}_1; \overline{a}_2]$ e $B = [\underline{b}_1; \overline{b}_2]$. A Tabela 1 exemplifica algumas operações de multiplicação conforme os valores dos sinais dos extremos dos intervalos.

Tabela 1 – Operação de Multiplicação

1.	$\underline{a}_1 \geq 0$ e $\underline{b}_1 \geq 0$	$\rightarrow A.B = [\underline{a}_1.\underline{b}_1; \overline{a}_2.\overline{b}_2]$
2.	$\underline{a}_1 \geq 0$ e $\underline{b}_1 < 0 \leq \overline{b}_2$	$\rightarrow A.B = [\overline{a}_2.\underline{b}_1; \overline{a}_2.\overline{b}_2]$
3.	$\underline{a}_1 \geq 0$ e $\overline{b}_2 < 0$	$\rightarrow A.B = [\overline{a}_2.\underline{b}_1; \underline{a}_1.\overline{b}_2]$
4.	$\underline{a}_1 < 0 \leq \overline{a}_2$ e $\underline{b}_1 \geq 0$	$\rightarrow A.B = [\underline{a}_1.\overline{b}_2; \overline{a}_2.\overline{b}_2]$
5.	$\underline{a}_1 < 0 \leq \overline{a}_2$ e $\underline{b}_1 \leq 0 \leq \overline{b}_2$	$\rightarrow A.B = [\min\{\underline{a}_1.\overline{b}_2, \overline{a}_2.\underline{b}_1\}; \max\{\underline{a}_1.\underline{b}_1, \overline{a}_2.\overline{b}_2\}]$
6.	$\underline{a}_1 < 0 \leq \overline{a}_2$ e $\overline{b}_2 < 0$	$\rightarrow A.B = [\overline{a}_2.\underline{b}_1; \underline{a}_1.\underline{b}_1]$
7.	$\overline{a}_2 < 0$ e $\underline{b}_1 \geq 0$	$\rightarrow A.B = [\underline{a}_1.\overline{b}_2; \overline{a}_2.\underline{b}_1]$
8.	$\overline{a}_2 < 0$ e $\underline{b}_1 < 0 \leq \overline{b}_2$	$\rightarrow A.B = [\underline{a}_1.\overline{b}_2; \underline{a}_1.\underline{b}_1]$
9.	$\overline{a}_2 < 0$ e $\overline{b}_2 < 0$	$\rightarrow A.B = [\overline{a}_2.\overline{b}_2; \underline{a}_1.\underline{b}_1]$

A Tabela 2 exemplifica algumas operações de divisão intervalar conforme os valores dos sinais dos extremos dos intervalos.

Tabela 2 – Operação de Divisão

1.	$\underline{a}_1 > 0$ e $\underline{b}_1 > 0$	$\rightarrow A/B = \left[\frac{\underline{a}_1}{\underline{b}_1}; \frac{\overline{a}_2}{\overline{b}_2} \right]$
2.	$\underline{a}_1 > 0$ e $0 \in [\underline{b}_1; \overline{b}_2]$	$\rightarrow A/B = \text{não definido}$
3.	$\underline{a}_1 > 0$ e $\overline{b}_2 < 0$	$\rightarrow A/B = \left[\frac{\overline{a}_2}{\overline{b}_2}; \frac{\underline{a}_1}{\underline{b}_1} \right]$
4.	$\underline{a}_1 < 0 < \overline{a}_2$ e $\underline{b}_1 > 0$	$\rightarrow A/B = \left[\frac{\underline{a}_1}{\underline{b}_1}; \frac{\overline{a}_2}{\overline{b}_2} \right]$
5.	$\underline{a}_1 < 0 < \overline{a}_2$ e $0 \in [\underline{b}_1; \overline{b}_2]$	$\rightarrow A/B = \text{não definido}$
6.	$\underline{a}_1 < 0 < \overline{a}_2$ e $\overline{b}_2 < 0$	$\rightarrow A/B = \left[\frac{\overline{a}_2}{\overline{b}_2}; \frac{\underline{a}_1}{\underline{b}_1} \right]$
7.	$\overline{a}_2 < 0$ e $\underline{b}_1 > 0$	$\rightarrow A/B = \left[\frac{\underline{a}_1}{\underline{b}_1}; \frac{\overline{a}_2}{\overline{b}_2} \right]$
8.	$\overline{a}_2 < 0$ e $0 \in [\underline{b}_1; \overline{b}_2]$	$\rightarrow A/B = \text{não definido}$
9.	$\overline{a}_2 < 0$ e $\overline{b}_2 < 0$	$\rightarrow A/B = \left[\frac{\overline{a}_2}{\overline{b}_2}; \frac{\underline{a}_1}{\underline{b}_1} \right]$

Esses nove casos, tanto para multiplicação quanto para divisão, servem para simplificar e otimizar os cálculos com intervalos.

2.2 Topologia do Conjunto \mathbb{R}

Os conceitos topológicos aqui vistos serão de suma importância para os resultados futuros desse trabalho. Destaca-se o conceito do diâmetro de um intervalo, pois esse, segundo [CLAUDIO D. M.; MARINS \(1989\)](#) é um indicativo da influência dos erros dos dados de entrada e dos erros de arredondamento e truncamento do resultado final obtido.

Distância entre Dois Intervalos: Sejam $A = [a_1; \bar{a}_2]$ e $B = [b_1; \bar{b}_2]$ dois intervalos de \mathbb{R} conforme [Figura 2](#). Definimos a distância de A e B como sendo o número real não negativo $d(A, B) = \max\{|a_1 - b_1|, |\bar{a}_2 - \bar{b}_2|\}$. Note que: $A = B \Leftrightarrow dist(A, B) = 0$.



Figura 2 – Representação Geométrica da Distância em \mathbb{R} .

Geometricamente, a distância entre dois intervalos é o comprimento do maior segmento que separa os respectivos extremos dos intervalos.

Módulo de um Intervalo: Sejam $A = [a_1; \bar{a}_2]$ um intervalo de \mathbb{R} conforme [Figura 3](#). Definimos o módulo do intervalo A como $|A| = |[a_1; \bar{a}_2]| = dist(A, 0) = \max\{|a_1|, |\bar{a}_2|\} \geq 0$.

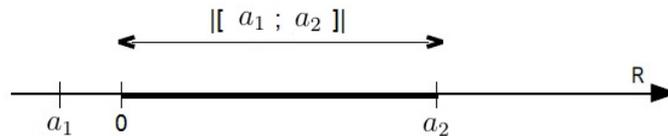


Figura 3 – Representação Geométrica do Módulo de um intervalo em \mathbb{R} .

Geometricamente, o módulo de um intervalo é o comprimento do maior segmento que une cada um dos extremos do intervalo à origem.

Diâmetro de um Intervalo: Seja $A = [a_1; \bar{a}_2]$ um intervalo de \mathbb{R} conforme [Figura 4](#). Definimos o diâmetro do intervalo A como sendo o número real não negativo $diam(A) = diam([a_1; \bar{a}_2]) = \bar{a}_2 - a_1 \geq 0$.

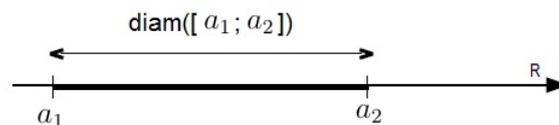


Figura 4 – Representação Geométrica do Diâmetro de um intervalo em \mathbb{R} .

Geometricamente, o diâmetro de um intervalo é o comprimento do segmento que une os extremos do intervalo.

Ponto Médio de um Intervalo: Seja $A = [a_1; \bar{a}_2]$ um intervalo de IR conforme [Figura 5](#). Definimos o ponto médio do intervalo A como sendo o número real $med(A) = med([a_1; \bar{a}_2]) = \frac{a_1 + a_2}{2}$.

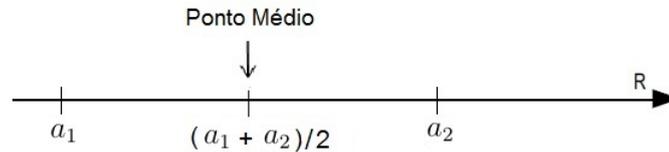


Figura 5 – Representação Geométrica do Ponto Médio de um intervalo em R .

2.3 Funções Intervalares

A aritmética intervalar nos forneceu uma importante ferramenta que é a limitação da faixa de valores de uma função ([KEARFOTT, 1997](#)). Uma função real pode ser transformada em uma função intervalar seguindo algumas definições que veremos a seguir, como a imagem intervalar e a avaliação intervalar de funções reais. Essas funções serão usadas para o desenvolvimento deste trabalho.

Exemplo: A função $f = R \rightarrow R$ dada por $f(x) = ax + b$ pode ser transformada na função $f1(A, B, x) = A.x + B$ na variável x com parâmetros de intervalos A e B. Para $A = [1;2]$, $B=[4;8]$ e $x = [3;7]$, temos:

$$f1(A, B, x) = [1; 2].[3; 7] + [4; 8] = [3; 14] + [4; 8] = [7; 22]$$

As raízes desta função podem ser encontradas, fazendo $f1(A, B, x) = 0$, ou seja, $[1; 2].x + [4; 8] = 0$. Então $[1; 2].x = [-8, -4]$ e portanto $x = \frac{[-8; -4]}{[1; 2]} = [-8; -2]$. Logo, os zeros de $f1(A, B, x)$ são possíveis no intervalo $[-8;-2]$.

2.3.1 Imagem Intervalar de uma Função Real

Seja f uma função real de variável real e X um intervalo tal que $X \subseteq Dom(f)$ e f é contínua em X . Define-se como imagem intervalar da função f em X , ou simplesmente imagem de f em X , o intervalo definido por:

$$I(f, X) = [\min\{f(x)|x \in X\}, \max\{f(x)|x \in X\}].$$

A imagem intervalar de uma função f no intervalo X é definida como o intervalo limitado pelo mínimo da imagem de $f(x)$ e pelo máximo da imagem de $f(x)$, sendo x um elemento do intervalo X ([OLIVEIRA, 1997](#)).

Exemplo: Seja $f(x) = x^2 - x$ e $X = [0; 2] \subseteq R = Dom(f)$.

$$\begin{aligned} \text{Temos } I(f, X) &= I(x^2 - x, [0; 2]) = [\min\{x^2 - x \mid x \in [0; 2]\}, \max\{x^2 - x \mid x \in [0; 2]\}] \\ &= [0; 2]^2 - [0; 2] = [0; 2] \cdot [0; 2] - [0; 2] \\ &= [\min\{0.0, 0.2, 0.2, 2.2\}, \max\{0.0, 0.2, 0.2, 2.2\}] - [0; 2] \\ &= [\min\{0, 0, 0, 4\}, \max\{0, 0, 0, 4\}] - [0; 2] \\ &= [0; 4] - [0; 2] \\ &= [-2; 4] \end{aligned}$$

Nota-se que essa é uma maneira natural de se definir funções intervalares a partir de funções reais, ou seja, $Y = f(X) = I(f, X)$, onde f é uma função real e X é um intervalo contido no domínio da função f . Nota-se também que se $X = [x, x]$ é um intervalo pontual, então $Y = f(X)$ também será um intervalo pontual dado por $Y = [f(x), f(x)]$. Se $X = [a, b]$ é um intervalo com $diam(X) > 0$, então $I(f, X)$ é o intervalo de menor diâmetro que contém todos os valores reais de $f(X)$, quando $x \in X$.

Exemplo:

- Seja $f(x) = x^2 - x$ e $X = [0; 2] \subseteq R = Dom(f)$. Assim, $I(f, X) = (x^2 - x, [0; 2]) = [\min x^2 - x \mid x \in [0; 2], \max x^2 - x \mid x \in [0; 2]] = \left[\frac{-1}{4}; 2 \right]$.
- Seja $g(x) = x(x - 1)$ e $X = [0; 2] \subseteq R = Dom(g)$. Assim, $I(g, X) = (x(x - 1), [0; 2]) = [\min x(x - 1) \mid x \in [0; 2], \max x(x - 1) \mid x \in [0; 2]] = \left[\frac{-1}{4}; 2 \right]$.

Observe que $I(f, X) = I(g, X)$, pois elas representam a mesma função real, porém escritas com expressões diferentes, ficando provado que a imagem intervalar independe das diferentes expressões (OLIVEIRA, 1997).

2.3.2 Extensão Intervalar de uma Função Real

Segundo Ferson et al. (2002) o método da extensão intervalar foi o primeiro a computar o intervalo solução de uma função. Baseado nos conceitos de Moore, a extensão intervalar foi proposta como forma de generalizar funções reais em funções intervalares.

Seja f uma função real de variável real $x \in X$ um intervalo. Define-se extensão intervalar de f em X como sendo a função intervalar $F(X)$, definida de maneira que cada ocorrência da variável real x é substituída pela variável intervalar X e cada operação $(+, -, \cdot, /)$ é substituída pela respectiva operação intervalar de tal modo que, quando $X = [x; x]$ for um intervalo pontual, então $F(X) = f(x)$ (MOORE, 1976).

Exemplo: Seja $f(x) = x^2 - x = x.x - x$ e Então $F(X) = X.X - X$.

$$\begin{aligned} \text{Para } X = [0; 2], \text{ temos } F([0; 2]) &= [0; 2].[0; 2] - [0; 2] \\ &= [\min\{0.0, 0.2, 0.2, 2.2\}, \max\{0.0, 0.2, 0.2, 2.2\}] - [0; 2] \\ &= [\min\{0, 0, 0, 4\}, \max\{0, 0, 0, 4\}] - [0; 2] \\ &= [0; 4] - [0; 2] \\ &= [-2; 4] \end{aligned}$$

Ao contrário da imagem intervalar, a extensão intervalar depende da forma com que a expressão está descrita. Por exemplo, se tomarmos uma função g , tal que $g(x) = x(x - 1)$, que é a mesma função $f(x) = x^2 - x$, só que expressa de forma diferente, tem-se:

$$\begin{aligned} \text{Para } X = [0; 2], \text{ temos } G(X) &= X.(X - [1; 1]) \\ G = ([0; 2]) &= [0; 2].([0; 2] - [1; 1]) \\ &= [0; 2].([-1; 1]) \\ &= [\min\{0. - 1, 0.1, 2. - 1, 2.1\}, \max\{0. - 1, 0.1, 2. - 1, 2.1\}] \\ &= [\min\{0, 0, -2, 2\}, \max\{0, 0, -2, 2\}] \\ &= [-2; 2] \end{aligned}$$

Assim, a extensão da função f descrita no exemplo acima é diferente da extensão da função g . Portanto, $F(X) \neq G(X)$.

2.4 Complexidade de Algoritmos

Um algoritmo é uma descrição passo a passo de como um problema é solucionado. A descrição deve ser finita e os passos devem ser bem definidos, sem ambiguidades, além de executáveis computacionalmente (LORETO, 2006). A complexidade de um algoritmo consiste na quantidade de trabalho necessária para a sua execução, expressa em função das operações fundamentais, as quais variam de acordo com o algoritmo, e em função dos dados (TOSCANI; VELOSO, 2009).

Existem vários aspectos a considerar na determinação da complexidade de um algoritmo, geralmente esses são medidos levando em consideração medidas de desempenho como tempo e espaço requeridos por um algoritmo, relacionados à velocidade e à quantidade de memória, respectivamente (TOSCANI; VELOSO, 2009).

As análises podem ser feitas para o pior caso, melhor caso e o caso médio. Geralmente, o critério de avaliação mais utilizado, entre as medidas de complexidade, é a complexidade no pior caso (TOSCANI; VELOSO, 2009). Um algoritmo, para ser razoável ou não, vai depender de quantos passos computacionais ele necessita para chegar a solução de um problema. Em geral, interessa-se pelo pior caso, o maior número de operações usadas para qualquer entrada de tamanho n , isto é, estuda-se o comportamento assintótico das funções de custo, ou seja, $f(n)$, onde representa o limite do comportamento do custo quando n cresce.

2.4.1 Classes de Comportamento Assintótico

Se f é uma função de complexidade para um algoritmo F , então $O(f)$ é considerada a complexidade assintótica do algoritmo F . As principais classes de problemas possuem as funções de complexidade descritas a seguir.

- $f(n) = O(1)$. Algoritmos de complexidade $O(1)$ são ditos de complexidade constante. O uso do algoritmo independe do tamanho de n , ou seja, as instruções do algoritmo são executadas um número fixo de vezes.
- $f(n) = O(n)$. Um algoritmo de complexidade $O(n)$ é dito de complexidade linear. Em geral, um pequeno trabalho é realizado sobre cada elemento de entrada. Cada vez que n dobra de tamanho, o tempo de execução também dobra.
- $f(n) = O(n^2)$. Um algoritmo de complexidade $O(n^2)$ é dito de complexidade quadrática. Algoritmos dessa ordem de complexidade ocorrem quando os itens de dados são processados aos pares, muitas vezes em um laço dentro de outro.
- $f(n) = O(2^n)$. Um algoritmo com complexidade $O(2^n)$ é dito de complexidade exponencial. Algoritmos dessa ordem de complexidade geralmente não são úteis do ponto de vista prático. Eles ocorrem na solução de problemas quando se usa força bruta para resolvê-los. Quando n dobra, o tempo de execução fica elevado ao quadrado.

3 TRABALHOS RELACIONADOS

Neste capítulo serão apresentados os principais trabalhos relacionados, que possuem relevância com o que está sendo proposto no presente trabalho. Os conceitos aqui apresentados, mostram o que já existe na área, assim como alguns resultados que serão apresentados se farão necessários para o desenvolvimento dos objetivos finais.

3.1 Extensão intervalar para as variáveis aleatórias com distribuições Uniforme, Normal, Gama, Exponencial e Pareto

Finger (2014) em sua dissertação, utilizou o método de extensão intervalar, baseado na matemática intervalar de Moore (1966) e na aritmética de exatidão máxima de Kulisch e Miranker (2014) para definir de forma intervalar as funções densidade de probabilidade das variáveis aleatórias com distribuições Gama e Pareto. As distribuições para Uniforme, Exponencial e Normal, foram desenvolvidas anteriormente por SANTOS (2010). Após definidas, realizou a análise da qualidade dos intervalos encapsuladores para as funções densidade de probabilidade das variáveis aleatórias com distribuições Uniforme, Exponencial, Normal, Gama e Pareto.

A análise foi feita a partir de resultados obtidos com as implementações utilizando o pacote de programação intervalar IntPy. Dentre as diversas soluções ou implementações para determinado problema, foi preciso escolher a mais eficiente. Para isso, foi utilizada a análise de algoritmos, a qual tem como objetivo melhorar, se possível, seu desempenho e escolher, entre os algoritmos disponíveis, o melhor. O resultado da análise, pode ser observado na Tabela 3, esses foram obtidos a partir dos algoritmos implementados para cada distribuição utilizando a primitiva da função ou Método de Simpson. Assim, a autora confirmou através dessa, que ao utilizar intervalos para representar essas funções, foi possível controlar a propagação de erros e, no mínimo, manter o esforço computacional da forma real.

Tabela 3 – Esforço computacional das distribuições

Distribuição	Complexidade Real		Complexidade Intervalar	
	Primitiva da Função	Simpson	Primitiva da Função	Simpson
Uniforme	$O(1)$	-	$O(1)$	-
Exponencial	$O(1)$	$O(2^n)$	$O(1)$	$O(2^n)$
Normal	-	$O(2^n)$	-	$O(2^n)$
Gama	-	$O(2^n)$	-	$O(2^n)$
Pareto	$O(1)$	$O(2^n)$	$O(1)$	$O(2^n)$

3.2 Análise da Complexidade Computacional de Problemas de Estatística Descritiva com Entradas Intervalares

O trabalho desenvolvido por Loreto (2006) faz uma abordagem sobre a complexidade dos problemas da computação intervalar. No decorrer das pesquisas sobre problemas que envolvem computação intervalar, ela verificou que a maioria dos problemas pesquisados pertenciam à classe de problemas NP-Difícil. E diante desses resultados passou a pesquisar sobre a computação intervalar e trabalhos relacionados. O ponto de partida foi o PBCI (problema básico da computação intervalar) (KREINOVICH et al., 2013), ou seja, o problema de computar o intervalo imagem de uma função contínua e computável f .

Uma aplicação clássica deste problema é, por exemplo, o das medições de quantidade de óleo em um poço (KREINOVICH et al., 2013). Geralmente se conhecem os valores aproximados \tilde{x}_1 de uma quantidade física x_i , a exatidão Δ_i de cada medição e o algoritmo $f(x_1, \dots, x_n)$ que transforma os valores x_i no valor da quantidade desejada y . Como resultado, sabe-se que x_i pertence ao intervalo $x_i = [\tilde{x}_1 - \Delta_i, \tilde{x}_1 + \Delta_i]$ e se deseja conhecer o conjunto de possíveis valores de y . Para uma função contínua $f(x_1, \dots, x_n)$, este conjunto é um intervalo denotado por $y = [\underline{y}, \bar{y}]$. Considerando esse exemplo, surgiu a questão: “É possível ter um algoritmo que sempre calcula a imagem exata, isto é, os extremos \underline{y} e \bar{y} do intervalo y em tempo razoável?”. (KREINOVICH et al., 2013), procurou responder esta questão analisando a complexidade do PBCI, verificando que o mesmo pertence a classe de problemas NP-Difícil, ou seja, é pelo menos tão difícil de resolver quanto qualquer problema NP. Em outras palavras, é um problema que não tem algoritmo de tempo de processamento polinomial conhecido.

É importante observar que a NP-dificuldade do PBCI está relacionada com o processamento dos dados de entrada do problema. No PBCI, os dados de entrada de uma função contínua f são valores intervalares, e a forma de calcular o intervalo imagem do PBCI é através da imagem intervalar (OLIVEIRA, 1997). Isto é, no pior caso quando tem-se m número arbitrário de valores e deve-se considerar todos os valores compreendidos entre \underline{x} e \bar{x} do intervalo X para encontrar estes extremos. Outro fato importante, e que completa a caracterização do problema como NP-Difícil, é que o problema em questão é um problema de decisão, pois deseja-se saber se existe o intervalo $y = [\underline{y}, \bar{y}]$ que contenha a solução aproximada do problema. Por fim, Loreto conclui que o problema PBCI é computável; que não foi provado que o problema PBCI é NP; e que a questão em relação a complexidade deste problema está em aberto.

Num segundo momento a autora traz como tema a complexidade dos problemas de cálculo de indicadores estatísticos com entradas intervalares. Para isso, foram definidos em nível intervalar, devido a ausência de tais definições na literatura, alguns indicadores estatísticos através do método de extensão intervalar, tais como: as separatrizes coeficiente de variação intervalar, mediana intervalar, quartil, decil e percentil intervalares.

Ela reuniu ainda outros indicadores estatísticos já definidos na forma intervalar, como a média intervalar, moda intervalar, variância intervalar, desvio padrão intervalar, covariância intervalar e coeficiente de correlação intervalar. A Figura 6 contém uma relação de todas as expressões dos indicadores intervalares contidos na tese de Loreto.

Média Intervalar:	$ME_v = [\underline{me}, \overline{me}] = \frac{1}{n} [\sum_{i=1}^n x_i, \sum_{i=1}^n \bar{x}_i]$
Mediana Intervalar:	$MD_v = [\underline{md}, \overline{md}] = \begin{cases} \frac{(x_{(\frac{n}{2})} + x_{(\frac{n}{2}+1)})}{2}, & \text{se } n \text{ for par} \\ x_{(\frac{n+1}{2})}, & \text{se } n \text{ for ímpar.} \end{cases}$
Moda Intervalar:	$MO_v = [\underline{mo}, \overline{mo}] = [mo_{1 \leq i \leq n} \{x_i\}, mo_{1 \leq i \leq n} \{\bar{x}_i\}]$
Amplitude total Intervalar:	$AT_v = [\underline{at}, \overline{at}] = \begin{cases} [\underline{ma} - \underline{mi}, \overline{ma} - \underline{mi}], & \text{se } \underline{ma} \geq \underline{mi} \\ [0, \overline{ma} - \underline{mi}], & \text{se } \underline{ma} < \underline{mi}. \end{cases}$
Variância Intervalar:	$VA_v = [\underline{va}, \overline{va}] = \frac{1}{n-1} \sum_{i=1}^n (x_i - ME)^2$
Desvio padrão Intervalar:	$DP_v = [\underline{dp}, \overline{dp}] = \sqrt{VA} = +\sqrt{[\underline{va}, \overline{va}]} = [+ \sqrt{\underline{va}}, + \sqrt{\overline{va}}]$
Coefficiente de variação Intervalar:	$CV_v = [\underline{cv}, \overline{cv}] = \frac{DP}{ME} = \frac{[\underline{dp}, \overline{dp}]}{[\underline{me}, \overline{me}]}$
Covariância Intervalar:	$CO_v = [\underline{co}, \overline{co}] = \frac{1}{n-1} \sum_{i=1}^n (x_i - ME_X)(y_i - ME_Y)$
Coefficiente de correlação Intervalar:	$CC_v = [\underline{cc}, \overline{cc}] = \frac{CO}{DP_X DP_Y} = \left(\frac{[\underline{co}, \overline{co}]}{[\underline{dp}_X, \overline{dp}_X][\underline{dp}_Y, \overline{dp}_Y]} \right)$
Separatrizes Intervalares:	$Q_v = [q, \bar{q}], D_v = [d, \bar{d}], P_v = [p, \bar{p}], pos = (n-1)\alpha + 1$

Figura 6 – Indicadores Estatísticos Intervalares

Após definir esses indicadores estatísticos com entradas intervalares, Loreto investigou a complexidade computacional dos problemas de computar os valores dessas medidas com entradas intervalares. Por fim, realizou uma comparação entre os resultados obtidos com os encontrados na bibliografia. A primeira tabela, Tabela 4, refere-se aos problemas que não possuem operações aritméticas intervalares, a Tabela 5 apresenta os problemas que possuem operações aritméticas intervalares. Na Tabela 4 apresenta-se os resultados de Loreto de complexidade dos problemas de computar os valores da mediana, moda e separatrizes com entradas intervalares.

Tabela 4 – Indicadores Estatísticos Intervalares e Complexidade dos Problemas dos Indicadores Estatísticos Intervalares

Indicadores Estatísticos Intervalares	Complexidade dos Problemas dos Indicadores Estatísticos Intervalares
Mediana Intervalar	P
Moda Intervalar	P
Separatrizes Intervalares	P

Na Tabela 5 apresenta-se a classificação quanto a classe de complexidade dos problemas, e, também, descreve-se todos os resultados de complexidade destes problemas encontrados na bibliografia, realizando uma comparação com os resultados de Loreto. Nas tabelas a seguir utiliza-se o símbolo “-” para indicar que não foram encontrados resultados de complexidade para o referido problema.

Tabela 5 – Problemas da Estatística Descritiva Intervalar, complexidade dos Problemas com extensão intervalar e com imagem intervalar

Problemas de Estatística Descritiva Intervalar	Complexidade dos Problemas com Extensão Intervalar	Complexidade dos Problemas com Imagem Intervalar
Média Intervalar	P	-
Amplitude Intervalar	P	-
Variância Intervalar	P	$\underline{va} \rightarrow P$ $\overline{va} \rightarrow NP - Difícil$
Desvio Padrão	P	-
Coefficiente de Variação Intervalar	P	-
Covariância Intervalar	P	$\underline{co} \rightarrow NP - Difícil$ $\overline{co} \rightarrow NP - Difícil$
Coefficiente de Correlação Intervalar	P	$\underline{cc} \rightarrow NP - Difícil$ $\overline{cc} \rightarrow NP - Difícil$

A fim de mostrar a qualidade de aproximação nos intervalos solução para os indicadores estatísticos intervalares, foi realizado um estudo de caso onde comparou-se os resultados reais com os intervalares para cada um dos indicadores, avaliando se os intervalos resultantes englobam a resposta real exata. Para esse fim, ela utilizou exemplos numéricos através da aplicação de valores de índice de massa corporal de alunos do ensino fundamental. A efetiva certificação foi feita por meio do cálculo da medida de erro, onde, em todos os exemplos considerados, verificou-se que a qualidade no intervalo solução foi mantida.

3.3 Diferenças entre Aritmética de Moore e Aritmética Intervalar RDM

O trabalho desenvolvido por Landowski (2015), faz um comparativo entre a aritmética de Moore e a aritmética intervalar multidimensional RDM, apresentando as operações aritméticas básicas de soma, subtração, multiplicação e divisão definidas nas duas aritméticas, utilizando exemplos numéricos como meio comparativo entre as duas metodologias apresentadas. Além das operações básicas, o trabalho apresenta propriedades como comutatividade, associatividade, elementos neutros da multiplicação e divisão, inversão de elementos, sub distributivo e cancelamento.

O autor também apresenta ilustrações que permitem perceber como os valores dos intervalos dipostos no plano ficam, salientando que os valores entre os extremos, inferior e superior, do intervalo são considerados na aritmética multidimensional. A Figura 7 mostra como são construídas essas ilustrações, nela são apresentadas disposições no plano tridimensional, na qual são demonstradas cada uma das operações sob a abordagem multidimensional RDM. Dois intervalos foram utilizados nos cálculos: $A = [1, 2]$ e $B = [3, 4]$.

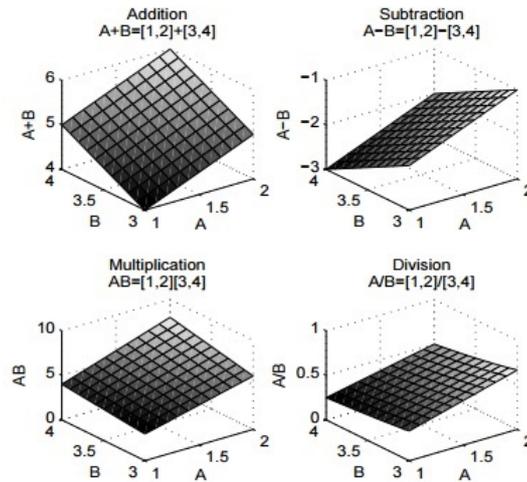


Figura 7 – Ilustração no plano para as operações de adição, subtração, multiplicação e divisão utilizando aritmética multidimensional RDM

Em busca de aplicações da aritmética intervalar é encontrada uma grande quantidade de trabalhos que aplicam aritmética de Moore (1966). Porém, estudos mais recentes provam que essa abordagem possui deficiência e não é a mais correta. Um problema clássico da aritmética de Moore (1966), quando se é usado o método da extensão intervalar é que conforme o modo que uma mesma função é descrita, os resultados retornam valores diferentes, o que justifica o uso da imagem intervalar para a transformação de uma função real para intervalar.

Um importante problema que não é amplamente discutido na literatura é o chamado “problema do lado direito de equações intervalares”. Suponha uma equação básica não intervalar $f(x) = 0$. Sua extensão intervalar pode ser obtida substituindo as variáveis por intervalos e as operações aritméticas pelas operações intervalares correspondentes. Como resultado teremos uma equação intervalar $[f]([x]) = 0$. Observa-se que esta equação apresenta em sua parte esquerda um valor intervalar, enquanto a parte direita não é um intervalo degenerado zero. Obviamente, se $[f](x) = [\underline{f}, \bar{f}]$, então a equação $[f]([x]) = 0$ é verdadeira somente quando $\underline{f} = \bar{f} = 0$. Em geral, a equação $[f]([x]) = 0$ somente pode ser verificada para o intervalo $[x]$ invertido, ou seja, quando $\bar{x} < \underline{x}$.

Intervalos invertidos são analisados em outras aritméticas intervalares, mas é difícil e muitas vezes até impossível encontrar uma situação real onde $\bar{x} < \underline{x}$ é significativo. Sabe-se que se uma expressão pode ser apresentada em diferentes, mas algebricamente equivalentes, formas, seus resultados podem ser diferentes quando é feita a extensão intervalar. O mesmo é verdadeiro para equações.

Considera-se a extensão intervalar da equação linear mais simples

$$ax = b \quad (3.1)$$

e suas formas algébricas equivalentes

$$x = \frac{b}{a}, \quad (3.2)$$

$$ax - b = 0 \quad (3.3)$$

para a, b intervalos ($0 \notin a$).

Sejam $[a] = [\underline{a}, \bar{a}]$ e $[b] = [\underline{b}, \bar{b}]$ intervalos. Por simplicidade, primeiro considera-se o caso onde $[a] > 0$, $[b] > 0$, ou seja, $\underline{a}, \bar{a} > 0$ e $\underline{b}, \bar{b} > 0$. A extensão intervalar para a equação 3.1 é $[\underline{a}, \bar{a}][\underline{x}, \bar{x}] = [\underline{b}, \bar{b}]$. Assim, obtém-se $[\underline{ax}, \bar{ax}] = [\underline{b}, \bar{b}]$. A igualdade do lado direito com o esquerdo só é possível se $\underline{ax} = \underline{b}$ e $\bar{ax} = \bar{b}$ e finalmente tem-se

$$\underline{x} = \frac{\underline{b}}{\underline{a}}, \bar{x} = \frac{\bar{b}}{\bar{a}}. \quad (3.4)$$

A extensão intervalar para a equação 3.2 resulta nas expressões

$$\underline{x} = \frac{\underline{b}}{\underline{a}}, \bar{x} = \frac{\bar{b}}{\underline{a}}. \quad (3.5)$$

$$\underline{x} = \frac{\underline{b}}{\bar{a}}, \bar{x} = \frac{\bar{b}}{\bar{a}}. \quad (3.6)$$

Agora considere alguns exemplos:

Exemplo 1: Sejam $a = [3, 4]$ e $[b] = [1, 2]$. Pela equação 3.4 $\underline{x} = 0.333$, $\bar{x} = 0.5$ e pela equação 3.6 $\underline{x} = 0.25$, $\bar{x} = 0.666$.

Exemplo 2: Sejam $a = [1, 2]$ e $[b] = [3, 4]$. Pela equação 3.4 $\underline{x} = 3$, $\bar{x} = 2$ e pela equação 3.6 $\underline{x} = 1.5$, $\bar{x} = 4$.

Exemplo 3: Sejam $a = [0.1, 0.3]$ e $[b] = [1, 1]$ (ou seja, b é um número real). Pela equação 3.4 $\underline{x} = 10$, $\bar{x} = 3.333$ e pela equação 3.6 $\underline{x} = 3.333$, $\bar{x} = 10$.

Pode-se perceber que a extensão intervalar da equação 3.1 pode resultar em intervalos impróprios (invertidos), ou seja, $\underline{x} > \bar{x}$ como pode ser visto nos exemplos 1 e 2, enquanto que a equação 3.2 retorna intervalos corretos, ou seja, $\underline{x} < \bar{x}$.

Para nossos propósitos, basta declarar que a extensão de intervalo da equação 3.2 fornece intervalos resultantes corretos em todos os casos, enquanto a extensão de intervalo da equação 3.1 pode resultar em intervalos invertidos praticamente sem sentido.

Nota-se que no Exemplo 3, a extensão intervalar formal da equação 3.1 leva a uma equação intervalar contraditória, uma vez que no lado direito tem-se um intervalo degenerado b (valor real), enquanto que o lado esquerdo é um intervalo.

Colocar os intervalos degenerados no lado direito da equação 3.1 deveria ser equivalente ao requisito de reduzir uma incerteza do lado esquerdo até zero. Isso é possível somente no caso de intervalo inverso $1/2x$ que, por sua vez, pode ser interpretado como um pedido para introduzir entropia negativa no sistema.

A extensão intervalar padrão para a equação 3.3 é $[\underline{ax}, \bar{ax}] - [\underline{b}, \bar{b}] = 0$. Assim, obtém-se $[\underline{ax} - \bar{b}, \bar{ax} - \underline{b}] = 0$ e

$$\underline{x} = \frac{\bar{b}}{\underline{a}}, \bar{x} = \frac{\underline{b}}{\bar{a}}.$$

É fácil notar que em alguns casos o resultado será um intervalo invertido.

Em resumo, pode-se dizer que somente a equação 3.2 pode ser considerada como base razoável para a extensão intervalar. Por outro lado, a partir dessa base obtém-se a equação 3.6, a qual muitas vezes resulta em uma diferença grande do intervalo de entrada comparado ao intervalo de saída, como pode ser visto no Exemplo 3.

Sejam A, B, C intervalos. Abaixo são apresentadas as propriedades mais importantes da aritmética RDM:

- 1) $A + B = B + A, AB = BA$: comutatividade da adição e multiplicação.
- 2) $A + (B + C) = (A + B) + C, A(BC) = (AB)C$: associatividade da adição e multiplicação.
- 3) Para cada $A \in R$ existe $A^{-1} \in R$, tal que $A + (A^{-1}) = (A^{-1}) + A = 0$. A^{-1} é o inverso aditivo de A .
- 4) $A(B + C) = (AB) + (AC)$: distributividade a esquerda, $(B + C)A = (BA) + (CA)$: distributividade a direita.
- 5) Para cada $A \in R, 0 \notin A$, existe $A^{-1} = 1/A \in R$, tal que $AA^{-1} = A(1/A) = 1$. A^{-1} é o inverso multiplicativo de A .
- 6) $A + C = B + C \rightarrow A = B$: cancelamento da adição.
- 7) $CA = CB \rightarrow A = B$: cancelamento da multiplicação.

A aritmética de Moore não possui as regras 3, 4, 5 e 7. Consequentemente, transformações de fórmulas não podem ser feitas. Por exemplo, na equação $A + X = C$ colocar o A no lado direito, $X = C - A$, não é permitido, pois a regra 3 não existe. Como algumas transformações não são permitidas, problemas algébricos e matemáticos mais complexos não podem ser resolvidos.

O autor conclui então, que os resultados da aritmética de Moore são unidimensionais, ou seja, não mostram uma solução completa, já a aritmética RDM dá uma solução multidimensional. Em alguns casos, nas soluções na aritmética de Moore, são utilizadas o método da extensão intervalar, vale salientar que esse método depende da forma da equação, então Landowski sugere que a aritmética de Moore não pode resolver corretamente problemas mais complicados, fazendo uma comparação com a aritmética RDM, que para diferentes formas da equação dão os mesmos resultados.

3.4 Definição Intervalar da Função Densidade de Probabilidade com Distribuição Beta: Aritmética de Moore e Aritmética Multidimensional RDM

O trabalho desenvolvido por Maraschin (2016), teve como objetivo principal criar a definição intervalar para a função densidade de probabilidade da variável aleatória contínua com distribuição Beta. Para isso, o autor aplicou o método de extensão intervalar, de modo que todas as entradas que antes comporiam entradas reais, passaram a receber entradas intervalares para os parâmetros da função densidade. O autor fez uso da linguagem de programação Python, operando com o pacote IntPy e utilizou duas abordagens da matemática intervalar para as implementações. A aritmética intervalar desenvolvida por Moore (1976) e a aritmética intervalar multidimensional RDM desenvolvida por Landowski (2015). Por fim, ele realizou a análise de complexidade dos algoritmos implementados a fim de analisar o esforço computacional despendido para calcular a função densidade de probabilidade da variável aleatória com distribuição Beta, nas formas real e intervalar.

A distribuição Beta é um modelo probabilístico para uma variável aleatória contínua X , cujos valores possíveis são limitados superior e inferiormente. Na forma da distribuição Beta padronizada, a variável X é definida no intervalo $[0, 1]$. Nesse caso, para $0 \leq x \leq 1, \alpha > 0, \beta > 0$, a função densidade de probabilidade Beta é expressa pela equação abaixo:

$$f_X(x) = \frac{1}{B(\alpha, \beta)} \int_0^a x^{\alpha-1} (1-x)^{\beta-1} dx.$$

Seguindo o método da extensão intervalar [subseção 2.3.2](#), ele substituiu todas as entradas de x da equação acima, onde obteve a seguinte equação:

$$f_{Bint}(\mathbf{x}, \alpha, \beta) = \frac{1}{B(\alpha, \beta)} \int_0^a x^{\alpha-1} (1-\mathbf{x})^{\beta-1} dx.$$

O autor após a análise de complexidade para cada um dos algoritmos implementados para computar a função densidade com distribuição Beta, tanto sob as diferentes abordagens da aritmética intervalar utilizadas, quanto para a sua forma real, chegou a conclusão que a complexidade se manteve igual, usando a aritmética intervalar de Moore,

a aritmética intervalar RDM e a do algoritmo com entradas reais. Assim sendo, ele pode justificar o uso de técnicas intervalares para computar cálculos numéricos, onde pode-se obter maior exatidão nos resultados com controle automático de erros além de manter o esforço computacional em um nível aceitável, se mantendo igual, neste caso.

3.5 Aspectos Formais da Correção e Otimização da Computação Intervalar

Visto que a análise de intervalo, proposta por Moore (1966), diz respeito à descoberta de funções de intervalos para produzir limites na precisão dos resultados numéricos que são garantidos para serem nítidos e corretos. O último critério, correção, é o principal, uma vez que estabelece que o resultado de uma computação de intervalo deve sempre conter o valor da função real relacionada. Santiago, Bedregal e Acioly (2006) mostraram algumas relações em termos de aspectos topológicos de intervalos (topologias de Scott e Moore). A ideia do artigo foi mostrar algumas propriedades de algoritmos de intervalos corretos, dando uma relação com algum outro tipo de funções, como extensões intervalares, apresentadas na subseção 2.3.2 .

A partir da definição da aritmética de intervalo, o conjunto de funções de intervalo é dividido em duas classes: funções de intervalo racional e funções de intervalo irracional. As funções de intervalo racional são aquelas calculadas usando as quatro operações aritméticas, essas vistas na seção 2.1 e as funções irracionais são calculadas usando aproximações racionais. Portanto, tudo o que pode ser feito com métodos padrão de ponto flutuante também pode ser feito com sua extensão de intervalo. A correção repousa sobre o teorema fundamental da aritmética de intervalo, assim os autores mostram que o teorema pode ser informalmente lido como: Se F é uma extensão de intervalo de inclusão monotônica de uma função real f , então $F(X)$ contém o intervalo $f(X)$. Então, esse teorema mostra que a classe de funções de intervalo de inclusão monotônica tem propriedade de correção e como todas as operações de aritmética são inclusão monotônica, então qualquer função de intervalo racional também está correta.

As extensões de intervalo foram propostas por Moore como uma forma de generalizar funções reais em termos de intervalos. Nesse artigo, os autores usaram a terminologia "representações de intervalo" (KEARFOTT, 1997), essa no entanto tem um significado diferente de Moore. Eles adotaram a terminologia de Moore e exploraram a noção de representações sozinhas principalmente em termos do ponto de vista topológico. Após a definição da extensão intervalar os autores fizeram algumas proposições:

- Existem extensões intervalares que não são representações de intervalo. Prova: A função $F([a, b]) = [m([a, b]), m([a, b])]$ estende a função de identidade $Id(x) = x$, mas não garante a correção.
- Existem representações de intervalos que não são extensões intervalares. Prova: A função $F([a, b]) = [a - 1, b + 1]$ satisfaz claramente o requisito.

- A interseção entre as famílias de extensões intervalares com a família de representações de intervalo não está vazia. Prova: A aritmética de Moore e a função de identidade, por exemplo, satisfazem isso.
- Existem extensões intervalares de funções contínuas que não são nem Moore-contínuas nem Scott-contínuas.

Portanto, nesse trabalhos os autores chegaram a conclusão de que os conceitos de extensão sozinhos são fracos, uma vez que não garantem a correção e não preservam a continuidade das funções reais nas funções de continuidade das funções de intervalo Moore ou Scott.

3.6 IntPy: Computação Científica Auto Validável em Python

Varjao (2011), teve como objetivo do seu trabalho desenvolver funções matemáticas específicas para a linguagem de programação Python, assim como o cálculo de probabilidades intervalares para variáveis aleatórias, além de analisar o desempenho das rotinas implementadas em Barreto (2016) com as mesmas implementadas no IntLab, pacote de desenvolvimento intervalar, que já havia sido implementado. O suporte para a realização de operações matemáticas sobre intervalos é o uso de arredondamentos direcionados (KULISCH; MIRANKER, 2014) nos extremos dos intervalos, ou seja, o limite inferior \underline{x} é arredondado para baixo (∇) e o limite superior \bar{x} é arredondado para cima (Δ). Por exemplo, o intervalo $[2.71828, 3.14159]$ se arredondado utilizando a técnica de arredondamento direcionado, para quatro casas decimais, o intervalo resultante torna-se $[2.7183, 3.1416]$. A lista completa de funções implementadas por Varjão é apresentada na Tabela 6, a qual contém um quadro apresentando os atributos referentes às operações e sua descrição de retorno de resultado.

Tabela 6 – Funções Implementadas por Varjão

Atributo	Retorno
pow	Potência do intervalo
sqrt	Raiz Quadrada do intervalo
log	Logaritmo do intervalo
exp	Exponencial do intervalo
sin	Seno do intervalo
cos	Cosseno do intervalo
tan	Tangente do intervalo
asin	Arco Seno do intervalo
acos	Arco Cosseno do intervalo
atan	Arco Tangente do intervalo
sinh	Seno Hiperbólico do intervalo
cosh	Cosseno Hiperbólico do intervalo
tanh	Tangente Hiperbólica do intervalo

Tais funções estão contidas na classe `stdfunc` (*standard functions*), a qual herda o atributo *IReal* e, como característica principal, possui a extensão das operações listadas acima sobre o atributo *IReal*. Ele conclui que o IntPy obteve resultados satisfatórios quando comparado aos resultados encontrados no Intlab, em tempo de processamento e em precisão de máquina, não encontrando dificuldades para realizar as implementações das funções intervalares em Python.

4 FUNÇÕES DENSIDADE DE PROBABILIDADE

Nesta seção serão apresentadas as funções densidade de probabilidade das variáveis aleatórias contínuas utilizadas no presente trabalho, nas suas formas real e intervalar.

4.1 Distribuição Uniforme

A distribuição uniforme é a mais simples distribuição contínua, entretanto é muito utilizada dentro da teoria da probabilidade.

- Forma Real: uma variável aleatória contínua X , cujos valores possíveis x encontram-se restritos à condição $a \leq x \leq b$, é distribuída uniformemente se a probabilidade de que ela esteja compreendida em qualquer intervalo $[m, n]$, contido em $[a, b]$, for diretamente proporcional ao comprimento $(m - n)$. Se a constante de proporcionalidade for denotada por ρ , então,

$$P(m \leq X \leq n) = \rho(m - n) \text{ se } a \leq m \leq n \leq b.$$

Uma vez que $P(a \leq X \leq b) = 1$, é fácil verificar que $\rho = 1/(b - a)$. Portanto, para qualquer $a \leq x \leq b$, a função de distribuição acumulada da distribuição Uniforme é dada por

$$F_x(x) = \frac{x - a}{b - a} \quad (4.1)$$

Se $x < a$, $F_x(x) = 0$ e, se $x > b$, $F_x(x) = 1$. A função densidade da distribuição Uniforme decorre da diferenciação da [Equação 4.1](#) e tem a seguinte expressão:

$$f_x(x) = \frac{1}{b-a} \text{ se } a \leq x \leq b.$$

Assim, é possível calcular a probabilidade do valor calculado estar compreendido entre a e b .

- Forma Intervalar: a distribuição Uniforme possui densidade com primitiva na forma analítica, podendo ser resolvida pela aplicação do Teorema Fundamental do Cálculo Integral. Analisando-se os possíveis casos de soluções para esta distribuição, [SANTOS \(2010\)](#) definiu a função *UNIF*, a qual é uma proposta de como calcular um intervalo encapsulador para probabilidades da Uniforme no intervalo $[a, b]$. Enfatiza-se que *UNIF* é uma extensão intervalar monotônica para a inclusão Moore e fornece o intervalo de menor comprimento para encapsular probabilidades para esta distribuição.

$$UNIF(c, d) = \left[\frac{w([c, d] \cap [a, b])}{b - a}, \overline{\frac{w([c, d] \cap [a, b])}{b - a}} \right], [c, d] \cap [a, b] \neq \emptyset.$$

onde $[c, d]$ e $[a, b]$ são intervalos $\subset R$.

4.2 Distribuição Exponencial

A distribuição exponencial é frequentemente utilizada para determinar o tempo entre ocorrências de eventos sucessivos. Possui inúmeras aplicações em diversas áreas do conhecimento humano e às variáveis hidrológicas.

- Forma Real: define-se a função densidade de probabilidade da distribuição Exponencial de parâmetro α por:

$$f(x) = \begin{cases} \alpha e^{-\alpha x}, & 0 \leq x < \infty, \\ 0, & x < 0. \end{cases}$$

com $\alpha > 0$. A probabilidade de que um número $x \in (a, b)$ é:

$$P(a \leq x \leq b) = \alpha \int_a^b e^{-\alpha x} dx = e^{-\alpha a} - e^{-\alpha b}.$$

Sendo a aplicação da integral a resolução de um número x estar compreendido no intervalo entre os valores a e b .

- Forma Intervalar: Segundo [SANTOS \(2010\)](#), como a probabilidade da variável aleatória X , assumir valores negativos é 0 pode-se restringir, na aplicação do método de Simpson intervalar, o domínio de f e de $f^{(4)}$ ao conjunto R_+ . A extensão intervalar definida por Santos para esta função densidade de probabilidade é calculada através da seguinte função:

$$F_E(X) = \alpha [e^{-\alpha \underline{x}}, e^{-\alpha \bar{x}}], \underline{x} > 0 \text{ e } G_E(X) = \alpha^5 [e^{-\alpha \underline{x}}, e^{-\alpha \bar{x}}]$$

4.3 Distribuição de Pareto

A distribuição de Pareto recebeu esse nome devido ao economista italiano Vilfredo Pareto que utilizou essa distribuição, no início do século XX, para descrever a distribuição da riqueza na Itália, onde uma minoria possuía a maior parte da riqueza e a maioria da população possuía uma pequena parte. Mais precisamente, ele reparou que 20% da população possuía 80% da riqueza. Em seguida, constatou que esta assimetria se verificava nas mesmas proporções em outros países. Foi baseado neste fato que, mais tarde, o gestor de negócios, Joseph M. Juran, estabeleceu o Princípio de Pareto ou a regra dos 80-20 que diz que, em muitos fenômenos, 80% dos efeitos vem de 20% das causas. Por exemplo, num negócio de vendas, 80% das vendas provêm, de 20% dos clientes.

- Forma Real: Define-se a função de densidade de probabilidade da distribuição de Pareto, por:

$$f(x) = \begin{cases} \frac{\alpha\beta^\alpha}{x^{\alpha+1}}, & \text{se } x \geq \beta \\ 0, & \text{se } x < \beta \end{cases}$$

com $\alpha > 1$ e $\beta > 0$.

$$\int_{-\infty}^{+\infty} f(x)dx = \int_{\beta}^{+\infty} \frac{\alpha\beta^\alpha}{x^{\alpha+1}}dx.$$

Pode-se, então, calcular a probabilidade de um valor distribuído conforme Pareto, através da integral apresentada acima.

- Forma Intervalar: seja $X \sim P[a, b]$, $a < b$, uma variável aleatória com distribuição de Pareto com parâmetros α e c . Para computar o intervalo encapsulador da probabilidade intervalar para esta variável aleatória utiliza-se a seguinte fórmula desenvolvida por [Finger \(2014\)](#):

$$F_P(X) = \alpha \frac{c^\alpha}{X^{\alpha+1}} = \alpha \left(\left[\frac{c^\alpha}{\bar{x}^{\alpha+1}}, \frac{c^\alpha}{\underline{x}^{\alpha+1}} \right] \right) \text{ para } 0 \notin X = [\underline{x}, \bar{x}],$$

onde α e $c \in R$.

Define-se a função densidade aplicando a extensão intervalar, ou seja, para cada operação de $F_P(X)$, se é conhecido o intervalo X , computa-se a imagem através da aritmética intervalar definida por [Moore \(1976\)](#).

4.4 Distribuição Weibull

A distribuição de Weibull é uma distribuição de probabilidade para variáveis aleatórias contínuas que é aplicada em diversos casos devido a sua versatilidade e relativa simplicidade.

- Forma Real: a distribuição de Weibull possui dois parâmetros de entrada, os quais definem sua forma e escala.

$$f(x) = \int_a^b \alpha\beta x^{\alpha-1} e^{-(\beta x)^\alpha} = e^{-(\beta b)^\alpha} - e^{-(\beta a)^\alpha}, x > 0,$$

onde α representa o parâmetro de forma e β o parâmetro de escala da função.

- Forma Intervalar: a definição intervalar da função densidade de probabilidade de uma variável aleatória contínua com distribuição Weibull, usando o método de extensão intervalar é expressa na equação a seguir:

$$Weibull(X, \alpha, \beta) = \int_a^b \alpha \beta X^{\alpha-1} e^{-(\beta X)^\alpha} = e^{-(\beta b)^\alpha} - e^{-(\beta a)^\alpha},$$

onde X representa o intervalo.

5 IMAGEM INTERVALAR PARA AS VARIÁVEIS ALEATÓRIAS COM DISTRIBUIÇÃO UNIFORME, EXPONENCIAL, PARETO E WEIBULL

Para uma função real ser transformada em uma função intervalar existem basicamente dois métodos principais: o método da imagem intervalar, visto na [subseção 2.3.1](#) e o método da extensão intervalar, presente na [subseção 2.3.2](#). O objetivo desse presente capítulo é verificar se utilizando o método da imagem intervalar nas funções densidade de probabilidade das variáveis aleatórias com distribuição retornariam resultados melhores e com uma complexidade aceitável para os algoritmos.

Para comparar os resultados encontrados e a exatidão dos mesmos foram executados os exemplos das funções no método de imagem intervalar como resolução, para todos os exemplos testados foram utilizados intervalos divididos em 5 e 10 e 35 subdivisões. Após a execução foi realizada a análise numérica dos resultados obtidos, comparando os resultados encontrados com os que já haviam na literatura. Na [Capítulo 6](#) será mostrada a análise da qualidade desses intervalos solução e a análise de complexidade dos algoritmos intervalares, a fim de comparar com os resultados já obtidos por [Finger \(2014\)](#).

5.1 Distribuição Uniforme

Apresentam-se na [Tabela 7](#) os valores obtidos em cada uma das soluções utilizadas para a forma real e para a forma intervalar, utilizando o método da extensão intervalar, bem como o tempo em segundos que cada solução levou para computar o cálculo.

- Exemplo 1: Intervalo A = [0.0, 3.0], Intervalo B = [1.0, 2.0]
- Exemplo 2: Intervalo A = [0.0, 3.0], Intervalo B = [0.5, 4.0/7.0]

Tabela 7 – Aplicação com a primitiva da função nas formas real e intervalar dos Exemplos 1 e 2, utilizando o método da extensão intervalar.

Método	Resultados Exemplo 1
Real	0.3333333333333333
Tempo (s)	0.00008
Intervalar	[0.3333333333333333 , 0.3333333333333334]
Tempo (s)	0.00042
Método	Resultados Exemplo 2
Real	0.02380952380952
Tempo (s)	0.00008
Intervalar	[0.02380952380952, 0.02380952380953]
Tempo (s)	0.00039

Analisando a [Tabela 7](#), é possível verificar que o valor real está contido no intervalo solução nos dois exemplos. Na [Tabela 8](#), serão mostrados os resultados dos exemplos acima utilizando o método da imagem intervalar.

Tabela 8 – Aplicação com a primitiva da função na forma intervalar do Exemplo 1 e 2, utilizando o método da imagem intervalar

Dividindo em 5 subdivisões	
Resultado Exemplo 1	[0.33333333333324766 , 0.33333333333343806]
Tempo (s)	0.00021
Resultado Exemplo 2	[0.023809523809509219 , 0.023809523809534459]
Tempo (s)	0.00042
Dividindo em 10 subdivisões	
Resultado Exemplo 1	[0.33333333333322990 , 0.33333333333343806]
Tempo (s)	0.00218
Resultado Exemplo 2	[0.023809523809508147 , 0.023809523809534459]
Tempo (s)	0.00158

Analisando a [Tabela 8](#), é possível verificar que o valor real está contido no intervalo solução em todos os exemplos. Quanto ao tempo que cada solução requer para gerar o resultado, a solução aqui apresentada retorna tempos maiores e somente no exemplo 1, quando o intervalo é dividido em 5 subdivisões, retorna um tempo igual à solução gerada utilizando o método da extensão intervalar.

5.2 Distribuição Exponencial

A densidade de probabilidade intervalar da distribuição Exponencial, definida por [SANTOS \(2010\)](#), foi implementada utilizando a primitiva da função para obter o intervalo solução. Apresentam-se na [Tabela 9](#) os valores obtidos em cada uma das soluções utilizadas para a forma real e para a forma intervalar, utilizando o método da extensão intervalar, bem como o tempo em segundos que cada solução levou para computar o cálculo.

- Exemplo 1: Intervalo $A = [20, 50]$, $\alpha = 0.01$, subintervalos = 5000.
- Exemplo 2: Intervalo $A = [10, 40]$, $\alpha = 0.05$, subintervalos = 5000.

Tabela 9 – Aplicação com a primitiva da função nas formas real e intervalar dos Exemplos 1 e 2, utilizando o método da extensão intervalar.

Método	Resultados Exemplo 1
Real	0.212200093365665335
Tempo (s)	0.00003
Intervalar	[0.212200093365665335, 0.212200093365665336]
Tempo (s)	0.00036
Método	Resultados Exemplo 2
Real	0.471195376476020722
Tempo (s)	1.19616
Intervalar	[0.471195376476020722, 0.471195376476020722]
Tempo (s)	0.00093

Analisando a [Tabela 9](#), é possível verificar que o valor real está contido no intervalo solução nos dois exemplos. Na [Tabela 10](#), serão mostrados os resultados dos exemplos acima utilizando o método da imagem intervalar.

Tabela 10 – Aplicação com a primitiva da função na forma intervalar dos Exemplos 1 e 2, utilizando o método da imagem intervalar.

Dividindo em 5 subdivisões	
Resultado Exemplo 1	[0.212200093365348147, 0.212200093365348813]
Tempo (s)	0.00036
Resultado Exemplo 2	[0.471195376475991856, 0.471195376476023331]
Tempo (s)	0.00035
Dividindo em 10 subdivisões	
Resultado Exemplo 1	[0.212200093365347481, 0.212200093365349396]
Tempo (s)	0.00096
Resultado Exemplo 2	[0.471195376476013672, 0.471195376476020833]
Tempo (s)	0.00108

Analisando a [Tabela 10](#), é possível verificar que o valor real está contido no intervalo solução em todos os exemplos. Quanto ao tempo que cada solução requer para gerar o resultado, a solução aqui apresentada retorna um tempo menor somente no exemplo 2, quando o intervalo é dividido em 5 subdivisões.

5.3 Distribuição de Pareto

A implementação da distribuição de Pareto foi realizada utilizando-se a primitiva da função. Assim, apresenta-se uma maneira para computar o valor com entradas reais e uma para computar o intervalo solução para esta distribuição com exemplos já presentes na literatura revisada e apresentada nos trabalhos relacionados. Apresentam-se na [Tabela 11](#) os valores obtidos em cada uma das soluções utilizadas para a forma real e para a forma intervalar, utilizando o método da extensão intervalar, bem como o tempo em segundos que cada solução levou para computar o cálculo.

- Exemplo 1: Intervalo = $[1.0, 2.0]$, $\alpha = 0.25$, $c = 1.0$, subintervalos = 5000.
- Exemplo 2: Intervalo = $[1.0, 2.0]$, $\alpha = 0.50$, $c = 1.0$, subintervalos = 5000.

Tabela 11 – Aplicação com a primitiva da função nas formas real e intervalar dos Exemplos 1 e 2, utilizando o método da extensão intervalar.

Método	Resultados Exemplo 1
Real	0.15910358474629
Tempo (s)	0.00006
Intervalar	[0.15910358474629 , 0.15910358474630]
Tempo (s)	0.00039
Método	Resultados Exemplo 2
Real	0.29289321881345
Tempo (s)	0.00019
Intervalar	[0.29289321881345 , 0.29289321881346]
Tempo (s)	0.00036

Analisando a [Tabela 11](#) , é possível verificar que o valor real está contido no intervalo solução nos dois exemplos. Na [Tabela 12](#), será mostrado os resultados dos exemplos acima utilizando o método da imagem intervalar.

Tabela 12 – Aplicação com a primitiva da função na forma intervalar dos Exemplos 1 e 2, utilizando o método da imagem intervalar.

Dividindo em 5 subdivisões	
Resultado Exemplo 1	[0.159103584746208421 , 0.159103584746321969]
Tempo (s)	0.00044
Resultado Exemplo 2	[0.292893218813299105 , 0.292893218813515432]
Tempo (s)	0.00043
Dividindo em 10 subdivisões	
Resultado Exemplo 1	[0.159103584746277782 , 0.159103584746338483]
Tempo (s)	0.00127
Resultado Exemplo 2	[0.292893218813437106 , 0.292893218813557565]
Tempo (s)	0.00135

Analisando a [Tabela 12](#) , é possível verificar que o valor real está contido no intervalo solução em todos os exemplos. Quanto ao tempo que cada solução requer para gerar o resultado, a solução aqui apresentada retorna sempre tempos maiores do que as soluções geradas utilizando o método da extensão intervalar.

5.4 Distribuição Weibull

A implementação da distribuição de Weibull foi realizada utilizando-se a primitiva da função. Assim, apresenta-se uma maneira para computar o valor com entradas reais e uma para computar o intervalo solução para esta distribuição com os exemplos abaixo. Na [Tabela 13](#) mostram-se os valores obtidos em cada uma das soluções utilizadas para a forma real e para a forma intervalar, utilizando o método da extensão intervalar, bem como o tempo em segundos que cada solução levou para computar o cálculo.

- Exemplo 1: Intervalo = $[2.0, 7.0]$, $\alpha = 0.30$, $\beta = 0.1$, $x = 8.760$, subintervalos = 500.
- Exemplo 2: Intervalo = $[2.0, 7.0]$, $\alpha = 1.00$, $\beta = 0.7$, $x = 5.450$, subintervalos = 500.

Tabela 13 – Aplicação com a primitiva da função nas formas real e intervalar dos Exemplos 1 e 2, utilizando o método da extensão intervalar.

Método	Resultados Exemplo 1
Real	0.1323719964625507
Tempo (s)	1.78813
Intervalar	[0.1323719964625506 , 0.1323719964625507]
Tempo (s)	0.00024
Método	Resultados Exemplo 2
Real	0.239150380870682189
Tempo (s)	1.19209
Intervalar	[0.239150380870682186 , 0.239150380870682201]
Tempo (s)	0.00014

Analisando a [Tabela 13](#), é possível verificar que o valor real está contido no intervalo solução nos dois exemplos. Na [Tabela 14](#), será mostrado os resultados dos exemplos acima utilizando o método da imagem intervalar.

Tabela 14 – Aplicação com a primitiva da função na forma intervalar dos Exemplos 1 e 2, utilizando o método da imagem intervalar.

Dividindo em 5 subdivisões	
Resultado Exemplo 1	[0.132371996462549213 , 0.132371996462551295]
Tempo (s)	0.00069
Resultado Exemplo 2	[0.239150380870630397 , 0.239150380870687268]
Tempo (s)	0.00054
Dividindo em 10 subdivisões	
Resultado Exemplo 1	[0.132371996462548963 , 0.132371996462552183]
Tempo (s)	0.00205
Resultado Exemplo 2	[0.239150380870630397 , 0.239150380870687268]
Tempo (s)	0.00225

Analisando a [Tabela 14](#), é possível verificar que o valor real está contido no intervalo solução em todos os exemplos. Quanto ao tempo que cada solução requer para gerar o resultado, a solução aqui apresentada retorna sempre tempos maiores do que as soluções geradas utilizando o método da extensão intervalar.

6 ANÁLISE NUMÉRICA

O objetivo do presente capítulo é analisar e comparar os intervalos encapsuladores obtidos para cada distribuição, esses definidos anteriormente pelo método da extensão intervalar e os intervalos obtidos nesse presente trabalho utilizando o método da imagem intervalar. Para verificar a qualidade dos resultados realiza-se a análise numérica sobre os intervalos resultantes, para isso a computação utilizando intervalos fornece a medida do diâmetro do intervalo ($w(x)$), o qual é calculado pela diferença entre o limite superior e o limite inferior ($w(x) = \bar{x} - \underline{x}$), além das seguintes estimativas para o erro: erro Absoluto, referente a diferença entre o valor exato de um número e seu valor aproximado; e erro Relativo, estando relacionado ao erro percentual podendo ser calculado pelo quociente entre o erro absoluto e o valor aproximado.

- Erro Absoluto: $|x - m(x)| < \frac{w(x)}{2}$, onde $m(x) = (\frac{x+\bar{x}}{2})$ é o ponto médio do intervalo x ;
- Erro Relativo: $|\frac{x-m(x)}{x}| \leq \frac{w(x)}{2\min|x|}$, se $0 \notin x$;

Tais medidas de erros foram aplicadas nos valores de resultados intervalares obtidos na computação das funções densidade com distribuição Uniforme, Exponencial, Pareto e Weibull com o objetivo de verificar a qualidade do intervalo solução, além de que o cálculo do diâmetro forneceu uma medida de qualidade de aproximação do intervalo solução em relação ao valor real.

6.1 Distribuição Uniforme

Na [Tabela 15](#) mostram-se os valores dos diâmetros obtidos em cada uma das soluções para a forma intervalar, utilizando o método da extensão e da imagem intervalar.

Tabela 15 – Análise dos Diâmetros da Distribuição Uniforme

Análise dos Diâmetros dos Intervalos			
Distribuição	Método	Exemplo	Diâmetro
Uniforme	Extensão Intervalar	1	1.00×10^{-14}
	Extensão Intervalar	2	1.00×10^{-14}
Uniforme	Imagem Intervalar	1 *Dividindo em 5	1.90×10^{-14}
	Imagem Intervalar	1 *Dividindo em 10	2.08×10^{-14}
	Imagem Intervalar	2 *Dividindo em 5	2.52×10^{-14}
	Imagem Intervalar	2 *Dividindo em 10	2.63×10^{-14}

Analisando os resultados, observa-se que todos os diâmetros são nulos até a 14^a casa decimal, verifica-se que o intervalo encapsulador possui qualidade de aproximação do valor real.

A Tabela 16 apresenta os valores obtidos para a estimativa de erro absoluto e erro relativo em relação aos intervalos resultantes da aplicação dos dados dos exemplos à distribuição Uniforme.

Tabela 16 – Análise dos Erros da Distribuição Uniforme

Análise Numérica dos Intervalos com Imagem Intervalar		
Exemplo	Erro Absoluto	Erro Relativo
1 *Dividindo em 5	$8.88 \times 10^{-16} < 9.57 \times 10^{-15}$	$2.66 \times 10^{-15} \leq 2.87 \times 10^{-14}$
1 *Dividindo em 10	$0.56 \times 10^{-16} < 1.04 \times 10^{-14}$	$1.67 \times 10^{-16} \leq 3.13 \times 10^{-16}$
2 *Dividindo em 5	$1.25 \times 10^{-16} < 1.26 \times 10^{-16}$	$5.28 \times 10^{-13} \leq 5.30 \times 10^{-13}$
2 *Dividindo em 10	$1.20 \times 10^{-14} < 1.31 \times 10^{-14}$	$5.06 \times 10^{-13} \leq 5.52 \times 10^{-13}$

Para o erro absoluto, verifica-se que a diferença entre o valor exato e o ponto médio do intervalo encapsulador, no exemplo 1 dividindo o intervalo em 5 subdivisões apresenta resultados onde a diferença nas casas decimais ocorre na 16^a casa decimal. No exemplo 1, dividindo o intervalo em 5 subdivisões apresenta resultados onde a diferença nas casas decimais ocorre somente na 17^a casa decimal.

Nas duas subdivisões do exemplo 2 a diferença nas casas decimais ocorre na 14^a casa decimal. O resultado da divisão do diâmetro do intervalo pela metade também mantém a mesma característica, certificando que a solução proposta no presente trabalho apresenta qualidade boa quanto a estimativa de erro absoluto. Para o erro relativo, a desigualdade se manteve válida em todos os resultados obtidos, indicando a qualidade do intervalo.

Na Figura 8 temos o gráfico do comportamento da função com distribuição uniforme, onde o mínimo e o máximo da função encontram-se nos extremos. Diante disso, ao usarmos o método da imagem intervalar não teremos uma variação significativa nos resultados esperados, pois a função com distribuição uniforme será sempre constante, não sofrendo variação entre os máximos e mínimos da imagem, o que nesse caso é preferível utilizar o método da extensão intervalar.

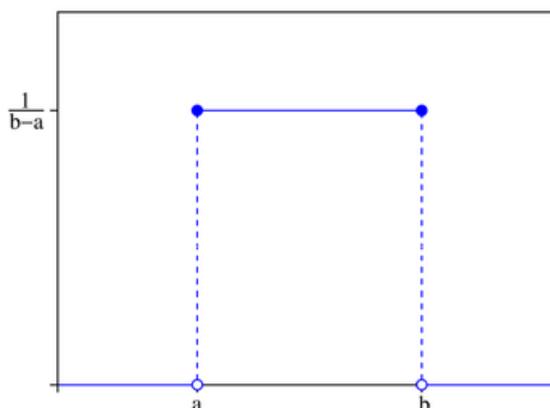


Figura 8 – Gráfico do Comportamento da Distribuição Uniforme

6.2 Distribuição Exponencial

Na Tabela 17 mostram-se os valores dos diâmetros obtidos em cada uma das soluções para a forma intervalar, utilizando o método da extensão e da imagem intervalar.

Tabela 17 – Análise dos Diâmetros da Distribuição Exponencial

Análise dos Diâmetros dos Intervalos			
Distribuição	Método	Exemplo	Diâmetro
Exponencial	Extensão Intervalar	1	1.00×10^{-14}
	Extensão Intervalar	2	0.00×10^{-14}
Exponencial	Imagem Intervalar	1 *Dividindo em 5	6.66×10^{-16}
	Imagem Intervalar	1 *Dividindo em 10	1.91×10^{-15}
	Imagem Intervalar	2 *Dividindo em 5	3.14×10^{-14}
	Imagem Intervalar	2 *Dividindo em 10	7.16×10^{-14}

Analisando os resultados, observa-se que o diâmetro do exemplo 1, utilizando o método da imagem intervalar e dividindo o intervalo em 5 subdivisões é nulo até a 16ª casa decimal, os diâmetros do exemplo 1 e 2, utilizando o método da imagem intervalar e dividindo o intervalo em 10 subdivisões são nulos até a 15ª casa decimal e os diâmetros dos dois exemplos utilizando o método da extensão intervalar e do exemplo 2 utilizando o método da imagem intervalar e dividindo o intervalo em 5 subdivisões são nulos até a 14ª casa decimal.

A Tabela 18 apresenta os valores obtidos para a estimativa de erro absoluto e erro relativo em relação aos intervalos resultantes da aplicação dos dados dos exemplos à distribuição Exponencial.

Tabela 18 – Análise dos Erros da Distribuição Exponencial

Análise Numérica dos Intervalos com Imagem Intervalar		
Exemplo	Erro Absoluto	Erro Relativo
1 *Dividindo em 5	$0.00 \times 10^{-16} < 3.33 \times 10^{-16}$	$0.00 \times 10^{-16} \leq 1.57 \times 10^{-15}$
1 *Dividindo em 10	$0.28 \times 10^{-16} < 9.71 \times 10^{-16}$	$1.31 \times 10^{-16} \leq 4.57 \times 10^{-15}$
2 *Dividindo em 5	$1.31 \times 10^{-14} < 1.57 \times 10^{-14}$	$2.78 \times 10^{-14} \leq 3.34 \times 10^{-14}$
2 *Dividindo em 10	$3.44 \times 10^{-15} < 3.60 \times 10^{-15}$	$7.30 \times 10^{-15} \leq 7.65 \times 10^{-15}$

Para o erro absoluto, verifica-se que a diferença entre o valor exato e o ponto médio do intervalo encapsulador, no exemplo 1 dividindo o intervalo em 5 subdivisões apresenta resultados onde a diferença nas casas decimais é nula até a 18ª casa decimal. No exemplo 1, dividindo o intervalo em 10 subdivisões apresenta resultados onde a diferença nas casas decimais ocorre somente na 17ª casa decimal. No exemplo 2 dividindo o intervalo em 5 subdivisões apresenta resultados onde a diferença nas casas decimais é nula até a 14ª casa decimal e quando dividido em 10 subdivisões a diferença nas casas decimais é nula até a 15ª casa decimal. Para o erro relativo, a desigualdade se manteve válida em todos os resultados obtidos, indicando a qualidade do intervalo.

Na Figura 9 temos o gráfico do comportamento da função com distribuição exponencial, onde o mínimo e o máximo da função encontram-se nos extremos. Diante disso, ao usarmos o método da imagem intervalar não teremos uma variação significativa nos resultados esperados, pois dependendo dos valores de entrada os resultados não sofrerão variações entre os máximos e mínimos da imagem.

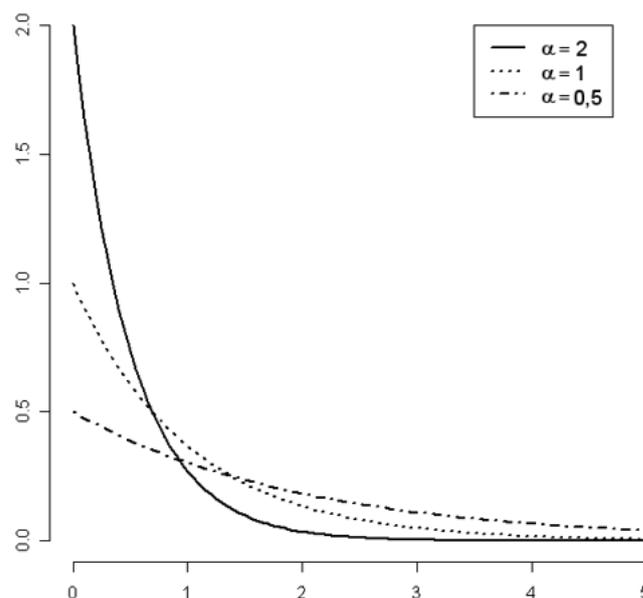


Figura 9 – Gráfico do Comportamento da Distribuição Exponencial

6.3 Distribuição de Pareto

Na [Tabela 19](#) mostram-se os valores dos diâmetros obtidos em cada uma das soluções para a forma intervalar, utilizando o método da extensão e da imagem intervalar.

Tabela 19 – Análise dos Diâmetros da Distribuição de Pareto

Análise dos Diâmetros dos Intervalos			
Distribuição	Método	Exemplo	Diâmetro
Pareto	Extensão Intervalar	1	1.00×10^{-14}
	Extensão Intervalar	2	1.00×10^{-14}
Pareto	Imagem Intervalar	1 *Dividindo em 5	1.13×10^{-13}
	Imagem Intervalar	1 *Dividindo em 10	6.07×10^{-14}
	Imagem Intervalar	2 *Dividindo em 5	2.16×10^{-13}
	Imagem Intervalar	2 *Dividindo em 10	1.20×10^{-13}

Analisando os resultados, observa-se que o diâmetro do exemplo 1 e 2, utilizando o método da extensão intervalar e o diâmetro do exemplo 1, utilizando o método da imagem intervalar, quando o intervalo é dividido em 5 subdivisões é nulo até a 14^a casa decimal, já os demais diâmetros são nulos até a 13^a casa decimal.

A [Tabela 20](#) apresenta os valores obtidos para a estimativa de erro absoluto e erro relativo em relação aos intervalos resultantes da aplicação dos dados dos exemplos à distribuição Pareto.

Tabela 20 – Análise dos Erros da Distribuição de Pareto

Análise Numérica dos Intervalos com Imagem Intervalar		
Exemplo	Erro Absoluto	Erro Relativo
1 *Dividindo em 5	$2.02 \times 10^{-13} < 5.68 \times 10^{-13}$	$1.27 \times 10^{-13} \leq 3.57 \times 10^{-13}$
1 *Dividindo em 10	$2.27 \times 10^{-14} < 3.03 \times 10^{-14}$	$1.42 \times 10^{-13} \leq 1.90 \times 10^{-13}$
2 *Dividindo em 5	$4.50 \times 10^{-14} < 1.08 \times 10^{-13}$	$1.53 \times 10^{-13} \leq 3.69 \times 10^{-13}$
2 *Dividindo em 10	$4.49 \times 10^{-14} < 6.02 \times 10^{-14}$	$1.53 \times 10^{-13} \leq 2.05 \times 10^{-13}$

Para o erro absoluto, verifica-se que a diferença entre o valor exato e o ponto médio do intervalo encapsulador, em todos exemplos apresentam resultados onde a diferença nas casas decimais é nula até a 14^a casa decimal. Para o erro relativo, a desigualdade se manteve válida em todos os resultados obtidos, indicando a qualidade do intervalo.

Na [Figura 10](#) temos o gráfico do comportamento da função com distribuição de pareto, onde o mínimo e o máximo da função encontram-se nos extremos. Diante disso, ao usarmos o método da imagem intervalar não teremos uma variação significativa nos resultados esperados, pois dependendo dos valores de entrada os resultados não sofrerão variações entre os máximos e mínimos da imagem.

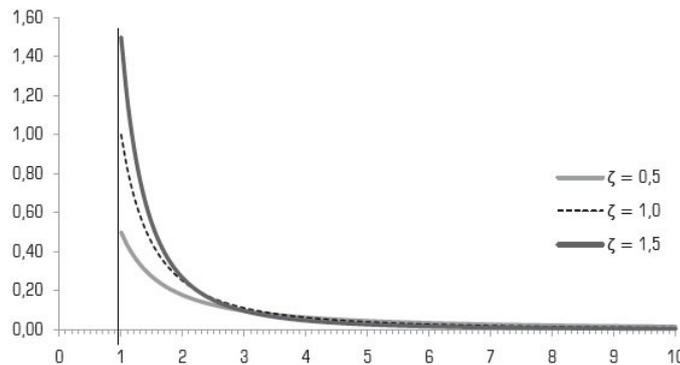


Figura 10 – Gráfico do Comportamento da Distribuição de Pareto

6.4 Distribuição Weibull

Na [Tabela 21](#) mostram-se os valores dos diâmetros obtidos em cada uma das soluções para a forma intervalar, utilizando o método da extensão e da imagem intervalar.

Tabela 21 – Análise dos Diâmetros da Distribuição Weibull

Análise dos Diâmetros dos Intervalos			
Distribuição	Método	Exemplo	Diâmetro
Weibull	Extensão Intervalar	1	1.00×10^{-14}
	Extensão Intervalar	2	1.00×10^{-14}
Weibull	Imagem Intervalar	1 *Dividindo em 5	2.08×10^{-15}
		1 *Dividindo em 10	3.22×10^{-15}
	Imagem Intervalar	2 *Dividindo em 5	5.68×10^{-14}
		2 *Dividindo em 10	5.68×10^{-14}

Analisando os resultados, observa-se que os diâmetros do exemplo 1 e 2, utilizando o método da extensão intervalar são nulos até a 13ª casa decimal. Os diâmetros do exemplo 1, utilizando o método da imagem intervalar são nulos até a 14ª casa decimal, já os diâmetros do exemplo 2 são nulos até a 13ª casa decimal.

A [Tabela 22](#) apresenta os valores obtidos para a estimativa de erro absoluto e erro relativo em relação aos intervalos resultantes da aplicação dos dados dos exemplos à distribuição Weibull.

Tabela 22 – Análise dos Erros da Distribuição Weibull

Análise Numérica dos Intervalos com Imagem Intervalar		
Exemplo	Erro Absoluto	Erro Relativo
1 *Dividindo em 5	$4.72 \times 10^{-15} < 1.05 \times 10^{-14}$	$3.56 \times 10^{-14} \leq 7.96 \times 10^{-14}$
1 *Dividindo em 10	$2.27 \times 10^{-14} < 3.03 \times 10^{-14}$	$5.62 \times 10^{-14} \leq 5.86 \times 10^{-14}$
2 *Dividindo em 5	$4.50 \times 10^{-14} < 1.08 \times 10^{-13}$	$1.53 \times 10^{-13} \leq 3.69 \times 10^{-13}$
2 *Dividindo em 10	$2.33 \times 10^{-14} < 2.84 \times 10^{-14}$	$9.77 \times 10^{-14} \leq 1.18 \times 10^{-13}$

Para o erro absoluto, verifica-se que a diferença entre o valor exato e o ponto médio do intervalo encapsulador, no exemplo 1 dividindo o intervalo em 5 e 10 subdivisões apresenta resultados onde a diferença nas casas decimais é nula até a 15ª casa decimal. No exemplo 2, dividindo o intervalo em 5 e 10 subdivisões apresenta resultados onde a diferença nas casas decimais ocorre somente na 14ª casa decimal. Para o erro relativo, a desigualdade se manteve válida em todos os resultados obtidos, indicando a qualidade do intervalo.

Na [Figura 11](#) temos o gráfico do comportamento da função com distribuição weibull, onde o máximo da função não se encontra extremo. Diante disso, ao usarmos o método da imagem intervalar, poderemos ter uma variação significativa nos resultados esperados, pois dependendo dos valores de entrada os resultados sofrerão variações entre os máximos e mínimos da imagem, o que nesse caso é preferível utilizar o método da imagem intervalar.

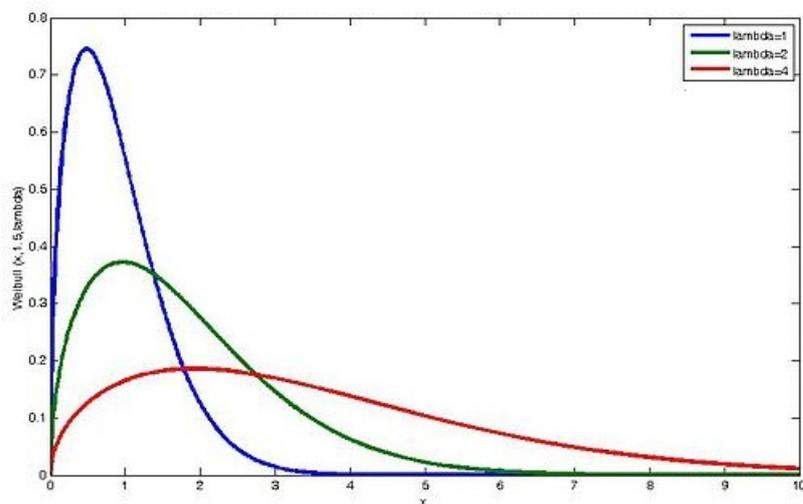


Figura 11 – Gráfico do Comportamento da Distribuição Weibull

6.5 Análise Numérica com mais Subdivisões

Com o intuito de fazer uma análise numérica mais precisa, para cada distribuição com os dois exemplos em questão, o intervalo foi dividido em 35 subdivisões. Na [Tabela 23](#) mostram-se os valores obtidos em cada uma das distribuições para a sua forma intervalar, utilizando o método da imagem intervalar, bem como o diâmetro do intervalo gerado e o tempo em segundos que cada solução levou para computar o cálculo.

Tabela 23 – Intervalo Dividido em 35 Subdivisões.

Intervalo Dividido em 35 Subdivisões			
Distribuição	Resultado	Diâmetro	Tempo (s)
Unif.-Ex.1	[0.33333333333322712 , 0.33333333333344084]	2.13×10^{-14}	0.00943
Unif.-Ex.2	[0.023809523809507148 , 0.023809523809534165]	2.70×10^{-14}	0.00904
Expon.-Ex.1	[0.212200093365347398 , 0.212200093365349396]	1.99×10^{-14}	0.00747
Expon.-Ex.2	[0.471195376476013672 , 0.471195376476020944]	7.27×10^{-14}	0.01568
Pareto-Ex.1	[0.159103584746277810 , 0.159103584746338594]	6.07×10^{-14}	0.01427
Pareto-Ex.2	[0.292893218813437273 , 0.292893218813557732]	1.20×10^{-13}	0.01689
Weibull-Ex.1	[0.132371996462548963 , 0.132371996462552211]	3.24×10^{-15}	0.02559
Weibull-Ex.2	[0.239150380870630785 , 0.239150380870687268]	5.64×10^{-14}	0.01890

Analisando os intervalos gerados, nota-se que a maioria dos diâmetros aumentaram, comparados com os resultados de quando o intervalo foi dividido em 5 e 10 subdivisões, porém de forma pouca significativa, partindo do princípio de que os exemplos usados possuíam intervalos de entrada de comprimento pequeno. A [Tabela 24](#) apresenta os valores obtidos para a estimativa de erro absoluto e erro relativo em relação aos intervalos resultantes da aplicação dos dados dos exemplos à distribuição Uniforme, Exponencial, de Pareto e Weibull.

Tabela 24 – Análise Numérica dos Intervalos com 35 subdivisões.

Análise Numérica dos Intervalos com 35 subdivisões			
Distribuição	Exemplo	Erro Absoluto	Erro Relativo
Uniforme	1	$0.00 \times 10^{-17} < 1.07 \times 10^{-14}$	$0.00 \times 10^{-17} \leq 3.21 \times 10^{-14}$
Uniforme	2	$6.48 \times 10^{-15} < 1.35 \times 10^{-14}$	$2.71 \times 10^{-13} \leq 5.67 \times 10^{-13}$
Exponencial	1	$3.44 \times 10^{-15} < 3.63 \times 10^{-15}$	$7.30 \times 10^{-15} \leq 7.71 \times 10^{-15}$
Exponencial	2	$0.83 \times 10^{-16} < 1.02 \times 10^{-15}$	$0.39 \times 10^{-15} \leq 4.84 \times 10^{-14}$
Pareto	1	$2.27 \times 10^{-13} < 3.04 \times 10^{-13}$	$1.43 \times 10^{-13} \leq 1.91 \times 10^{-13}$
Pareto	2	$4.51 \times 10^{-14} < 6.02 \times 10^{-14}$	$1.54 \times 10^{-13} \leq 2.05 \times 10^{-13}$
Weibull	1	$1.67 \times 10^{-15} < 1.63 \times 10^{-14}$	$1.25 \times 10^{-14} \leq 1.23 \times 10^{-13}$
Weibull	2	$2.31 \times 10^{-14} < 2.82 \times 10^{-14}$	$9.67 \times 10^{-14} \leq 1.18 \times 10^{-13}$

Para o erro absoluto e relativo, a desigualdade se manteve válida em todos os resultados obtidos, indicando a qualidade do intervalo. Cabe salientar ainda que as fórmulas de medidas de erros absoluto, erro relativo e diâmetro dos intervalos também foram codificadas na linguagem Python, possibilitando a computação e obtenção de tais estimativas. O código implementado se encontra no [Apêndice B](#).

7 ANÁLISE DE COMPLEXIDADE

O termo complexidade refere-se, em geral, aos requerimentos de recursos necessários para que um algoritmo possa resolver um problema sob o ponto de vista computacional, ou seja, é a quantidade de trabalho dispendido pelo algoritmo (TOSCANI; VELOSO, 2009).

A seguir, apresentam-se as ordens de complexidade encontradas para cada uma das funções de densidade de probabilidade utilizadas no presente trabalho, com entradas intervalares, que utilizam a primitiva da função e computam o intervalo com o método da imagem intervalar.

7.1 Imagem Intervalar

A imagem intervalar de uma função f , contínua no intervalo x , é definida como sendo o intervalo limitado pelo mínimo e pelo máximo da imagem $f(x)$, sendo x um elemento encapsulado pelo intervalo X . Para a transformação de uma função real para a sua forma intervalar, utilizando o método da imagem intervalar, quanto mais subdivisões esse intervalo tiver, teremos um intervalo solução mais correto. Para a implementação das funções de densidade Uniforme, Exponencial, Pareto e Weibull, foram utilizadas as suas respectivas primitivas da função. Como a eficiência desse método depende do número de subdivisões, a complexidade do mesmo para as distribuições é de ordem $O(2^n)$. O algoritmo implementado pode ser encontrado no [Apêndice A](#).

Na [Tabela 25](#) é possível verificar de forma mais clara a ordem de complexidade encontrada para cada distribuição.

Tabela 25 – Esforço Computacional das Distribuições

Esforço Computacional das Distribuições		
Distribuição	Complexidade Intervalar	
	Método: Extensão Intervalar	Método: Imagem Intervalar
Uniforme	$O(1)$	$O(2^n)$
Exponencial	$O(1)$	$O(2^n)$
Normal	-	-
Beta	$O(1)$	-
Pareto	$O(1)$	$O(2^n)$
Gama	-	-
Weibull	$O(1)$	$O(2^n)$

A partir da análise da complexidade computacional dos algoritmos propostos para computar as funções com entradas reais, é possível afirmar que utilizando a primitiva da função como solução, os algoritmos são executados em uma complexidade menor, ou seja, menos trabalho para computar o resultado. Com a utilização da primitiva da função, juntamente do método da extensão intervalar podemos observar que os algoritmos são executados em uma complexidade menor do que comparado com o uso do método da imagem intervalar.

8 CONCLUSÃO

Quando se trabalha com computação numérica, um dos fatores de maior importância é a exatidão da resposta desses cálculos. Com isso, sempre se procura chegar a resultados cada vez mais exatos e com um menor erro possível contido neles, então a matemática intervalar surge com o objetivo principal de realizar um controle automático de erros dos cálculos, retornando respostas com uma maior exatidão. No estudo das variáveis aleatórias sobre o conjunto dos reais, R , um dos problemas está no cálculo da função densidade de probabilidade, visto que seu valor numérico resultante no computador é dado por aproximação devido à representação em aritmética de ponto flutuante, portanto afetado por erros de arredondamento ou truncamento. Desta forma, algoritmos intervalares, em contraste com os algoritmos pontuais, computam um intervalo como solução, com a garantia de que a resposta pertence a este intervalo.

Para transformar uma função intervalar a partir de uma função real, são usados dois métodos, o método da extensão intervalar e o método da imagem intervalar. O primeiro método citado, é definido de maneira que cada ocorrência da variável real x é substituída pela variável intervalar X e cada operação aritmética pela respectiva operação intervalar, porém sabe-se que esse método depende da forma com que a função está expressa, ou seja, se essa função for escrita de uma forma diferente, teremos um resultado diferente, o que torna esse método desvantajoso. Ao contrário da extensão intervalar, a imagem intervalar não depende da forma com que a função está expressa e é definida como o intervalo limitado pelo mínimo e pelo máximo da imagem de $f(x)$, sendo x um elemento do intervalo X .

A partir de estudos na literatura que mostram que o método de imagem intervalar pode resultar em intervalos menores, porém com um esforço computacional maior, o presente trabalho teve como objetivo fazer um comparativo dos métodos de extensão e imagem intervalar aplicados nas funções densidade de probabilidade com distribuições, as quais já foram definidas e resolvidas com o método de extensão.

Feita a análise numérica, a qual busca avaliar a qualidade dos intervalos solução, foi possível perceber que, em alguns casos, o método de imagem foi melhor, como por exemplo na distribuição Exponencial e Weibull, em outros casos não houve diferenças significativas.

Quanto ao esforço computacional, fica claro que o método da imagem intervalar depende do número de subdivisões do intervalo e analisando o algoritmo implementado, verifica-se que ele utiliza a técnica de força bruta ou também chamada de busca exaustiva. É uma técnica trivial, onde são testadas todas as possibilidades existentes de um determinado problema e sabendo que quantos candidatos à solução o problema tem, maior é o custo de execução do seu algoritmo. Por esse motivo, o método da imagem intervalar assume uma ordem de complexidade exponencial.

Com isso, é possível concluir que ainda assim o método de extensão intervalar é preferido quando é preciso resolver uma função densidade de probabilidade a qual se conhece a forma analítica da função. Porém, pelo que foi apresentado nos trabalhos relacionados, existem falhas na aritmética de Moore e, outras aritméticas podem retornar resultados mais confiáveis e corretos do que a extensão intervalar de Moore.

8.1 Trabalhos Futuros

Como ficou claro no presente trabalho, a aritmética definida por Moore apresenta algumas limitações com cálculos de intervalos, como intervalos com diâmetro muito grande, por exemplo, portanto para trabalhos futuros, pretende-se investigar outras aritméticas, como a aritmética multidimensional RDM, a fim de contornar essa e outras limitações e assim retornar resultados mais corretos. Por outro lado, ainda não foi demonstrado na literatura que, em termos computacionais, essa aritmética também será mais eficiente, retornando problemas com uma ordem de complexidade melhor que os resolvidos por Moore. Outro objetivo futuro é de testar exemplos com intervalos maiores, a fim de investigar se o método da imagem se comporta de maneira diferente.

REFERÊNCIAS

- BALBONI, M. D. C. et al. Critérios para análise e escolha de ambientes intervalares. **Revista Jr de Iniciação Científica em Ciências Exatas e Engenharia**, Rio Grande - RS, n. 7, 2014. Nenhuma citação no texto.
- BARRETO, R. M. **IntPy 0.1.3**. 2016. Disponível em: <<https://pypi.python.org/pypi/IntPy/0.1.3>>. Citado 2 vezes nas páginas 16 e 36.
- CLAUDIO D. M.; MARINS, J. M. **Cálculo Numérico Computacional: teoria e prática**. [S.l.]: Atlas, 1989. Citado na página 22.
- FERSON, S. et al. Absolute bounds on the mean of sum, product, max, and min: a probabilistic extension of interval arithmetic. 2002. Citado na página 24.
- FINGER, A. F. **Extensão intervalar para as variáveis aleatórias com distribuições Uniforme, Normal, Gama, Exponencial e Pareto**. Dissertação (Mestrado) — Universidade Federal de Pelotas, 2014. Citado 3 vezes nas páginas 27, 41 e 43.
- GAREY M. E.; JOHNSON, D. S. **Computers and intractability: a guide to the theory of NP-completeness**. [S.l.]: Freeman, 1979. ISBN 9788540701397. Nenhuma citação no texto.
- KEARFOTT, R. B. Interval computations: Introduction, uses and resources. **Department of Mathematics University of Southwestern Louisiana, USA**, Online, 1997. Citado 2 vezes nas páginas 23 e 35.
- KREINOVICH, V. et al. **Computational complexity and feasibility of data processing and interval computations**. [S.l.]: Springer Science & Business Media, 2013. v. 10. Citado na página 28.
- KULISCH, U. W.; MIRANKER, W. L. **Computer arithmetic in theory and practice**. [S.l.]: Academic press, 2014. Citado 3 vezes nas páginas 16, 27 e 36.
- LANDOWSKI, M. Differences between moore and rdm interval arithmetic. **Maritime University of Szczecin, Waly Chrobrego 1-2, 70-500 Szczecin, Poland**, DTIC Document, 2015. Citado 2 vezes nas páginas 30 e 34.
- LORETO, A. B. **Análise da Complexidade Computacional de Problemas de Estatística Descritiva com Entradas Intervalares**. Tese (Doutorado) — UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL, 2006. Citado 2 vezes nas páginas 25 e 28.
- MARASCHIN, D. A. **Definição Intervalar da Função Densidade de Probabilidade com Distribuição Beta: Aritmética de Moore e Aritmética Multidimensional RDM**. [S.l.]: Universidade Federal do Pampa, Alegrete, 2016. Citado na página 34.
- MESQUITA, M. P. de. **Matemática intervalar: princípios e ferramentas c-xsc**. [S.l.]: Online, 2004. Citado na página 15.
- MOORE, R. E. **Interval Analysis**. [S.l.]: Prentice-Hall Englewood Cliffs, 1966. v. 4. Citado 6 vezes nas páginas 15, 16, 19, 27, 31 e 35.

MOORE, R. E. On computing the range of a rational function of n variables over a bounded region. **Computing**, Springer, v. 16, n. 1, p. 1–15, 1976. Citado 3 vezes nas páginas 24, 34 e 41.

OLIVEIRA, P. W. et al. **Fundamentos da Matemática Intervalar**. [S.l.]: Sagra-Luzzatto, 1997. v. 1. Citado 4 vezes nas páginas 21, 23, 24 e 28.

PYTHON 2.7.11 documentation. 2016. Disponível em: <<https://docs.python.org/2/>>. Nenhuma citação no texto.

SANTIAGO, R. H. N.; BEDREGAL, B. R. C.; ACIOLY, B. M. Formal aspects of correctness and optimality of interval computations. 2006. Citado na página 35.

SANTOS, M. G. **Probabilidades Autovalidáveis para as Variáveis Aleatórias Exponencial, Normal e Uniforme**. Tese (Doutorado) — UNIVERSIDADE FEDERAL DE PERNAMBUCO, Recife, 2010. Citado 4 vezes nas páginas 27, 39, 40 e 44.

TOSCANI, L.; VELOSO, P. **Complexidade de Algoritmos: Série Livros Didáticos Informática UFRGS - Vol. 13**. [S.l.]: Bookman, 2009. ISBN 9788540701397. Citado 3 vezes nas páginas 16, 25 e 57.

VARJAO, F. **IntPy: Computação científica auto validável em Python**. Dissertação (Mestrado) — Master's thesis, Universidade Federal de Pernambuco, Recife, 2011. Citado na página 36.

Anexos

ANEXO A – ANEXO A- IMPLEMENTAÇÃO DO MÉTODO DA IMAGEM INTERVALAR

Todas as implementações descritas no decorrer deste anexo foram realizadas utilizando o ambiente de desenvolvimento intervalar composto pela linguagem de programação Python e o pacote de extensão intervalar IntPy, para essa mesma linguagem.

```

1 #from __future__ import division
2 from scipy import integrate
3 import time
4 import math
5 #from intpy import *
6
7 class ImgInt:
8     def __init__(self, x, y, division):
9         if x > y:
10             self.inf = y
11             self.sup = x
12         else:
13             self.inf = x
14             self.sup = y
15         self.increment = float((self.sup - self.inf)/division)
16         #self.sup += self.increment
17
18     def __repr__(self):
19         return (str(self.inf) + " " + str(self.sup) + " " + str(
20             float(self.increment)))
21
22 def executeIntegration(function, a, b, *listVar):
23     auxA = a.inf
24     resultMin = float("inf")
25     resultMax = float("-inf")
26     while auxA <= a.sup:
27         auxB = b.inf
28         while auxB <= b.sup:
29             if auxB > auxA:
30                 result, err = integrate.quad(function, auxA, auxB
31                     , args = listVar)
32                 resultMax = max(resultMax, result)
33                 resultMin = min(resultMin, result)
34             else:
35                 result, err = integrate.quad(function, auxB, auxA
36                     , args = listVar)

```

```

34         resultMax = max(resultMax, result)
35         resultMin = min(resultMin, result)
36         if not b.increment:
37             break
38         auxB += b.increment
39         if not a.increment:
40             break
41         auxA += a.increment
42     return [resultMin, resultMax]
43
44 def FunctionIntervalarIMG(function, listImgVar, listVar = []):
45     for var in listImgVar:
46         if not isinstance(var, ImgInt):
47             return "erro"
48     aux = listImgVar[0].inf
49     resultMin = float("inf")
50     resultMax = float("-inf")
51     if len(listImgVar) <= 1:
52         while aux <= listImgVar[0].sup:
53             listVar.append(aux)
54             result = function(*listVar)
55             resultMax = max(resultMax, result)
56             resultMin = min(resultMin, result)
57             aux+=float(listImgVar[0].increment)
58
59     else:
60         while aux <= listImgVar[0].sup:
61             listVar.append(aux)
62             result = FunctionIntervalarIMG(function, listImgVar
63                 [1:], listVar)
64             resultMax = max(resultMax, result[1])
65             resultMin = min(resultMin, result[0])
66             aux += float(listImgVar[0].increment)
67             del listVar[-1]
68         return [resultMin, resultMax]
69
70 def IntegrateIntervalarIMG(function, ImgVarA, ImgVarB, listImgVar
71     = [], listVar = []):
72     if not listImgVar:
73         return executeIntegration(function, ImgVarA, ImgVarB, *
74             listVar)
75     aux = listImgVar[0].inf

```

```
73     resultMin = float("inf")
74     resultMax = float("-inf")
75     while aux < listImgVar[0].sup:
76         listVar.append(aux)
77         result = IntegrateIntervalarIMG(function, ImgVarA,
78             ImgVarB, listImgVar[1:], listVar)
79         resultMax = max(resultMax, result[1])
80         resultMin = min(resultMin, result[0])
81         aux += float(listImgVar[0].increment)
82         del(listVar[-1])
83     return [resultMin, resultMax]
```


ANEXO B – ANEXO B - ERROS

```

1
2 from intpy import *
3 import sys
4
5 arquivo = open("1.txt", 'r')
6 txt = arquivo.read()
7 entradas = []
8
9 for le in txt.split("\n"):
10     entradas.append(le.split(" "))
11     le = arquivo.readline()
12
13 del entradas[-1]
14 print entradas
15
16 #Calculates Relative Error
17
18 def relativeError(xInt, xReal):
19     try:
20         r = abs((xReal - m(xInt))/xReal)
21         error = (diam(xInt))/(2*xInt.inf)
22     except:
23         r = "NaN"
24         error = "NaN"
25     return '%.18f'%r+" <= "+ '%.18f'%error
26
27
28 #Calculates interval diameter
29 def diam(xInt):
30     return xInt.sup - xInt.inf
31 def m(xInt):
32     return (xInt.inf + xInt.sup)/2.0
33
34 #Calculates absolute error
35
36 def absError(xInt, xReal):
37     try:
38         r = abs((xReal - m(xInt))/xReal)
39         error = (diam(xInt)/2)
40     except:

```

```
41     r = "NaN"
42     error = "NaN"
43     return '%.18f'%r + " <= " + '%.18f'%error
44
45 for x1, x2, real in entradas:
46     x = IReal(x1, x2)
47     print ("intervalo: " + str(x) + " Resultado Real" + real)
48     print ("Erro Absoluto: " + absError(x, float(real)))
49     print ("Erro Relativo: " + relativeError(x, float(real)))
50     print ("=====")
```