



UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE CIÊNCIAS
DEPARTAMENTO DE COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO
MESTRADO ACADÊMICO EM CIÊNCIA DA COMPUTAÇÃO

NYKOLAS MAYKO MAIA BARBOSA

MODELOS DE REGRESSÃO LINEAR PARA DADOS INTERVALARES USANDO
TRANSFORMAÇÕES LOGARÍTMICAS

FORTALEZA

2019

NYKOLAS MAYKO MAIA BARBOSA

MODELOS DE REGRESSÃO LINEAR PARA DADOS INTERVALARES USANDO
TRANSFORMAÇÕES LOGARÍTMICAS

Dissertação apresentada ao Curso de Mestrado Acadêmico em Ciência da Computação do Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Ciência da Computação. Área de Concentração: Teoria da Computação.

Orientador: Prof. Dr. João Paulo Pordeus Gomes.

Coorientador: Prof. Dr. César Lincoln Cavalcante Mattos.

FORTALEZA

2019

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

B21m Barbosa, Nykolas Mayko Maia.

Modelos de regressão linear para dados intervalares usando transformações logarítmicas / Nykolas Mayko Maia Barbosa. – 2019.
48 f. : il. color.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Ciências, Programa de Pós-Graduação em Ciência da Computação, Fortaleza, 2019.

Orientação: Prof. Dr. João Paulo Pordeus Gomes.

Coorientação: Prof. Dr. César Lincoln Cavalcante Mattos.

1. Regressão linear. 2. Dados intervalares. 3. Aprendizado sequencial. I. Título.

CDD 005

NYKOLAS MAYKO MAIA BARBOSA

MODELOS DE REGRESSÃO LINEAR PARA DADOS INTERVALARES USANDO
TRANSFORMAÇÕES LOGARÍTMICAS

Dissertação apresentada ao Curso de Mestrado Acadêmico em Ciência da Computação do Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Ciência da Computação. Área de Concentração: Teoria da Computação.

Aprovada em: 31/10/2019

BANCA EXAMINADORA

Prof. Dr. João Paulo Pordeus Gomes (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. César Lincoln Cavalcante
Mattos (Coorientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. João Fernando Lima Alcântara
Universidade Federal do Ceará (UFC)

Prof. Dr. Leonardo Ramos Rodrigues
Instituto de Aeronáutica e Espaço (IAE)

A Deus que guiou-me e iluminou-me durante a
jornada deste trabalho.

AGRADECIMENTOS

Agradeço a Deus por iluminar minhas ideias, abrindo-me caminhos durante toda a minha caminhada até a conclusão deste trabalho.

Meu muito obrigado aos meu pais, Jesus e Ângela, minha irmã Stephanny, minha namorada Larissa Rocha e a todos que fizeram parte dessa jornada com todo o apoio, carinho e o suporte necessário durante as dificuldades encontradas no caminho.

Destaco, também, o meu agradecimento para o meu Professor Orientador João Paulo Pordeus Gomes e ao Coorientador César Lincoln Cavalcante Mattos, que conduziram-me durante todo o trabalho de forma brilhante, repassando todos seus grandiosos conhecimentos a fim de capacitar-me como um melhor profissional na área de pesquisa abordado nesta dissertação.

Por fim, agradeço a todos os meus colegas e amigos que estiveram juntos, além dos demais professores do Departamento de Ciências da Computação da Universidade Federal do Ceará, que foram de extrema importância para o meu desenvolvimento com os conhecimentos passados.

“Sem uma teoria, os fatos são silenciosos.”

(HAYEK, 1993, p. 6.)

RESUMO

Resolver problemas de aprendizagem de máquina com conjuntos de dados intervalares é uma tarefa desafiadora que pode surgir em muitas aplicações do mundo real, por exemplo, na predição da pressão arterial (sistólica e diastólica). Diante disso, muitos pesquisadores propuseram métodos de regressão linear para lidar com dados intervalares nos últimos anos. Embora muitos esforços tenham sido aplicados nesses problemas, todos os métodos disponíveis dependem da modelagem do problema como uma tarefa de otimização restrita, as quais podem levar a resultados sub-ótimos. Além disso, nenhum trabalho anterior forneceu uma maneira de treinar um modelo de forma incremental, o qual é fundamental para problemas de *big data*. Nesse trabalho, atacou-se ambos esses problemas pela proposição de dois diferentes métodos de regressão linear baseados em transformações logarítmicas. Os métodos propostos serão referidos como Mínimos Quadrados Ordinários *Log* Transformados (*Log-transformed Ordinary Least Squares for Interval Data*) (LOID) e Mínimos Quadrados Estocásticos *Log* Transformados (*Log-transformed Least Mean Squares for Interval Data*) (LLID) para dados intervalares, ambos foram comparados com o método estado da arte em conjuntos de dados sintéticos e reais. Os resultados obtidos indicam a viabilidade de ambas abordagens propostas por este trabalho. Além disso, destaca-se que, diante do que foi pesquisado, o *LLID* é o primeiro método de regressão linear sequencial para dados intervalares.

Palavras-chave: regressão linear; dados intervalares; aprendizado sequencial.

ABSTRACT

Solving linear regression problems on interval-valued data is a challenging task that may arise in many applications, for example, blood pressure prediction (systolic and diastolic). Because of that, many researchers have designed methods for such task in recent years. Although much effort has been devoted to this problem, all available methods rely on modeling the problem as a constrained optimization task, which may lead to sub-optimal results. Moreover, no previous work provide a way to train a model in a incremental way, which is fundamental for big data problems. In this paper, we address both problems by proposing two different linear regression methods based on log-transformations. The proposed methods, referred as Log-transformed OLS for interval data (LOID) and Log-transformed LMS for interval data (LLID), are compared to state-of-the-art methods on both synthetic and real-world datasets. The obtained results indicate the feasibility of our approaches. Furthermore, to the best of our knowledge, LLID is the first sequential linear regression method for interval valued.

Keywords: linear regression; interval data; sequential learning.

LISTA DE FIGURAS

Figura 1 – Ilustração da convergência com diferentes valores de α	23
Figura 2 – Ilustração do Vetor de Intervalo $X = (X_1, X_2)$	24
Figura 3 – Raio, Centro e Limites de um Intervalo	25
Figura 4 – Gráficos das Funções Logarítmica e Exponencial	33
Figura 5 – Gráficos do Conjunto de Dados C_4^1	40
Figura 6 – Gráficos do Conjunto de Dados A_1^1	41

LISTA DE TABELAS

Tabela 1 – Configuração dos Conjuntos de Dados - Tipo <i>C</i>	40
Tabela 2 – Configuração dos Conjuntos de Dados - Tipo <i>D</i>	40
Tabela 3 – Comparação entre LOID , LLID and CCRM nos Conjuntos de Dados Sintéticos 1-Dimensional Usando o <i>RMSE</i> como Métrica para o Cálculo do Erro.	42
Tabela 4 – Comparação entre LOID , LLID and CCRM nos Conjuntos de Dados Sintéticos 3-Dimensional Usando o <i>RMSE</i> como Métrica para o Cálculo do Erro.	43
Tabela 5 – Descrição dos Conjuntos de Dados Reais	44
Tabela 6 – Comparação entre LOID , LLID and CCRM nos Conjuntos de Dados Reais usando o <i>RMSE</i> como Métrica para o Cálculo do Erro.	45

LISTA DE ABREVIATURAS E SIGLAS

CCRM	Método do Centro e Raio Restritos (<i>Constrained Center and Range Method</i>)
CRM	Método do Centro e Raio (<i>Center and Range Method</i>)
LLID	Mínimos Quadrados Estocásticos <i>Log</i> Transformados (<i>Log-transformed Least Mean Squares for Interval Data</i>)
LMS	Mínimos Quadrados Estocásticos (<i>Least Mean Squares</i>)
LOID	Mínimos Quadrados Ordinários <i>Log</i> Transformados (<i>Log-transformed Ordinary Least Squares for Interval Data</i>)
OLS	Mínimos Quadrados Ordinários (<i>Ordinary Least Squares</i>)

LISTA DE SÍMBOLOS

x	Número Escalar
X	Variável Intervalar
x^C	Centro de um intervalo de entrada
x^R	Raio de um intervalo de entrada
x^L	Límite inferior de um intervalo de entrada
x^U	Límite superior de um intervalo de entrada
\mathbf{x}^C	Vetor de centros de intervalos de entrada
\mathbf{x}^R	Vetor de raios de intervalos de entrada
\mathbf{x}^L	Vetor de limites inferiores de intervalos de entrada
\mathbf{x}^U	Vetor de limites superiores de intervalos de entrada

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Justificativa e Objetivos	15
1.2	Objetivos	16
2	FUNDAMENTOS TEÓRICOS	17
2.1	Regressão Linear	17
<i>2.1.1</i>	<i>Mínimos Quadrados Ordinários (OLS)</i>	17
<i>2.1.2</i>	<i>Mínimos Quadrados Estocásticos (LMS)</i>	19
<i>2.1.2.1</i>	<i>Otimização Irrestrita</i>	21
<i>2.1.2.2</i>	<i>Método Gradiente Descendente</i>	22
2.2	Conceitos Básicos de Intervalos	23
2.3	Regressão Linear para Dados Intervalares	25
<i>2.3.1</i>	<i>Método do Centro (CM)</i>	25
<i>2.3.2</i>	<i>Método do Centro e Raio (CRM)</i>	27
<i>2.3.3</i>	<i>Método do Centro e Raio Restritos (CCRM)</i>	28
3	MÉTODOS PROPOSTOS	31
<i>3.0.1</i>	<i>Mínimos Quadrados Ordinários Log Transformados para Dados Intervalares (LOID)</i>	31
<i>3.0.2</i>	<i>Mínimos Quadrados Estocásticos Log Transformados para Dados Intervalares (LLID)</i>	34
4	EXPERIMENTOS E RESULTADOS	38
<i>4.0.1</i>	<i>Conjuntos de Dados Sintéticos</i>	39
<i>4.0.1.1</i>	<i>Descrição dos Conjuntos de Dados</i>	39
<i>4.0.1.1.1</i>	<i>Tipo - C</i>	39
<i>4.0.1.1.2</i>	<i>Tipo - D</i>	39
<i>4.0.1.1.3</i>	<i>Tipo - A</i>	40
<i>4.0.1.2</i>	<i>Resultados</i>	42
<i>4.0.2</i>	<i>Conjuntos de Dados Reais</i>	43
<i>4.0.2.1</i>	<i>Descrição dos Conjuntos</i>	43
<i>4.0.2.2</i>	<i>Resultados</i>	44
5	CONCLUSÕES E TRABALHOS FUTUROS	46

REFERÊNCIAS 47

1 INTRODUÇÃO

Modelos de regressão linear são largamente utilizados para prever valores numéricos que são relacionados a um conjunto de medidas independentes (NETO; CARVALHO, 2017). Para ajustar o modelo ao dado, é necessário estimar um vetor de pesos que definem o relacionamento linear entre as variáveis independentes e as saídas, onde o vetor de pesos é, normalmente, obtido pela minimização da função de perda pré definida. Entre todas funções de perda possíveis, a soma dos erros quadráticos é a mais popular e define a abordagem de regressão chamada de mínimos quadrados. Embora o método de mínimos quadrados tenha sido usado satisfatoriamente em muitas aplicações, todas as formulações básicas são desenhadas para trabalhar com conjuntos de dados onde cada coluna representa uma variável. Nesse sentido, tais modelos não são adequados para trabalhar com dados intervalares.

Dados intervalares consistem na representação onde cada componente de uma dada amostrada é modelada como um intervalo, em oposição a único valor. Esse dados podem surgir, segundo (GIORDANI, 2011), por exemplo:

- devido a cálculos imprecisos gerados por dispositivos;
- dados que são intrinsecamente intervalares como pressão arterial ou sanguínea;
- usualmente, em pesquisas que utilizam questionários, tem-se a necessidade de se utilizar faixas de intervalos para descrever perfis de renda, altura, idade e/ou peso das pessoas que respondem ao questionário;
- dados flutuantes em casos de registros gravados durante um intervalo de tempo específico.

Tendo em vista a variedade de problemas e situações que envolvem dados com essas características, há anos, muitos autores tem pesquisado e apresentado modelos de regressão linear para dados intervalares. Um dos primeiros trabalhos foi apresentado por (BILLARD; DIDAY, 2000), onde eles propuseram um modelo que considerava apenas os pontos médios (ou centros) dos intervalos para estimar os coeficientes da regressão, por exemplo, uma variável representada pelo intervalo [15, 25], tem seu ponto médio igual a 20. Nesse sentido, (NETO; CARVALHO, 2008) apresentaram um método chamado de Método do Centro e Raio (*Center and Range Method*) (CRM), que melhora o trabalho anterior por considerar dois problemas de regressão linear independentes, onde o primeiro modela o ponto médio (centro) e o segundo modela o raio (*ranges*). Em (NETO; CARVALHO, 2010), os mesmos autores apresentaram uma nova versão de seu método, agora, chamado de Método do Centro e Raio Restritos (*Constrained Center and Range Method*) (CCRM), onde eles usam uma regressão linear restrita para assegurar

a coerência dos resultados, isto é, os valores preditos para os limites superiores, dos intervalos, devem ser maiores que os preditos para os limites inferiores. Além disso, o desenho do modelo linear esparso foi objeto de estudo do trabalho desenvolvido por (GIORDANI, 2011). O autor propôs uma variante do algoritmo *LASSO* para dados intervalares.

Analisando os trabalhos supracitados, pode-se notar que a restrição de coerência em ambos (NETO; CARVALHO, 2010) e (GIORDANI, 2011) impõe restrições desnecessárias ao modelo de regressão final. Em (NETO; CARVALHO, 2010), os autores asseguram que o raio predito sempre será positivo devido à restrição dos pesos da regressão serem positivos. Essa regra pode levar a resultados sub-ótimos pois não permitem que o processo de otimização encontre uma relação entre entradas e saídas dadas, sejam por um coeficiente de valor negativo, mesmo que se isso ocorra na geração dos dados. Utilizando uma configuração diferente, o autor (GIORDANI, 2011) impõe uma restrição para fazer que a predição do raio seja positiva apenas no conjunto de treinamento, assim, não assegurando que a predição do conjunto de dados de teste irá ser sempre positiva. Outra desvantagem desses trabalhos apresentados anteriormente é que nenhum deles adaptou os modelos de regressão linear para fornecer uma maneira de encontrar os coeficientes iterativamente.

Nesse trabalho, objetiva-se enfrentar os problemas mencionados pela proposição de duas abordagens de regressão linear para dados intervalares baseados em transformações logarítmicas sobre os dados disponíveis. A grosso modo, utiliza-se a configuração de centros e os raios das variáveis intervalares e aplica-se uma transformação nos raios usando-se uma função logarítmica. Após isso, computam-se os coeficientes usando os algoritmos Mínimos Quadrados Ordinários (*Ordinary Least Squares*) (OLS) e Mínimos Quadrados Estocásticos (*Least Mean Squares*) (LMS). Os métodos propostos incluem um passo de pré-processamento para a normalização dos raios, através dessa normalização antes de se aplicar a transformação, assegura-se que o dado transformado é uma região *quasi*-linear da função logarítmica. Esse processo é usado para minimizar os efeitos da transformação não-linear sobre o processo de estimação dos pesos. A efetividade de ambos os métodos são verificados pela comparação com o CCRM em vários conjuntos de dados reais e sintéticos.

1.1 Justificativa e Objetivos

A adaptação ou criação de mecanismos que visam manipular dados intervalares tem sido bastante incentivada nos últimos anos, nesse contexto, sabe-se que manipular intervalos a fim

de se trabalhar com extensões de abordagens clássicas em aprendizado de máquina possibilitam resguardar sua integridade, evitando-se a perda de informações além de se conseguir resultados mais coerentes em relação aos problemas. Portanto, este trabalho propõe duas extensões de dois métodos clássicos da literatura, OLS e LMS, realizando-se, assim, uma importante contribuição para área que trabalha com dados intervalares.

1.2 Objetivos

Objetivo Geral: estender os métodos OLS e LMS para resolver problemas com dados intervalares. Além desse, são objetivos específicos:

- formular uma variante do LMS considerando o centro e raio dos intervalos como entrada, porém o processo de treinamento e estimação das saídas são feitas de forma independentes;
- formular uma variante do OLS considerando o centro e raio dos intervalos como entrada, resolvendo o problema de forma análoga a dita acima;
- aplicar a transformação logarítmica e verificar sua eficácia;
- realizar a normalização dos dados para reduzir os efeitos da não-linearidade nos dados

Este trabalho está organizado da seguinte forma: o Capítulo 2 detalha os conceitos e versões originais do OLS e LMS, além de descrever um pouco sobre intervalos e regressão linear para dados intervalares; Capítulo 3 desenvolve os dois modelos propostos: OLS e LMS, respectivamente, para dados intervalares; o Capítulo 4 descreve a metodologia utilizada nos experimentos de validação dos métodos de regressão, além dos resultados obtidos em experimentos numéricos (em conjuntos de dados reais e sintéticos) e os discute e o Capítulo 5 conclui o trabalho.

2 FUNDAMENTOS TEÓRICOS

Neste capítulo, se descrever-se-ão, nas sub-seções seguintes, os conceitos de intervalos, suas propriedades, bem como demonstrando exemplos de como estes dados se comportam no plano cartesiano. Além disso, se definirão os métodos clássicos de regressão linear, OLS e LMS, respectivamente. Por fim, serão descritos as abordagens de regressão para dados intervalares, CM, CRM e CCRM.

2.1 Regressão Linear

Análise de regressão é uma técnica que pode ser usada para prever os valores de uma variável quantitativa dependente (ou saída) como uma função de valores de uma variável quantitativa independente (ou entrada) (KLEINBAUM *et al.*, 1988). O objetivo de uma tarefa de regressão é encontrar um modelo que se ajuste aos dados. Assim, é necessário estimar um vetor de parâmetros $\hat{\mathbf{w}}$ usando o vetor de dados de saída \mathbf{y} e a matriz de entrada \mathbf{X} . A etapa de estimação dos parâmetros (coeficientes), $\hat{\mathbf{w}}$, pode ser visto como um problema de otimização. Nesse trabalho, consideram-se os algoritmos mínimos quadrados e gradiente descendente para resolver tais otimizações. Nesse sentido, faz-se necessário descrevê-los para um melhor entendimento ao longo da leitura desse trabalho, portanto, inicia-se descrevendo-se o método dos Mínimos Quadrados Ordinários (*OLS*).

2.1.1 Mínimos Quadrados Ordinários (*OLS*)

O método dos Mínimos Quadrados Ordinários (*OLS*) é um método padrão usado para obter um único valor para um parâmetro físico a partir de várias medições desse parâmetro, ou parâmetros relacionados a ele por um modelo matemático desconhecido. O primeiro uso do *OLS* é, geralmente, atribuído à Karl Friedrich Gauss em 1795, Gauss inventou o *OLS* para permití-lo estimar o movimento orbital dos planetas a partir de medições de um telescópio (WELLS *et al.*, 1985).

Muitas áreas e campos de pesquisas tem como ferramenta de estimação o método dos mínimos quadrados, entre eles, o aprendizado de máquina, econometria, química, dentre outras áreas que buscam fazer predição de variáveis a partir de um conjunto observações obtidas por algum equipamento, por exemplo.

Nesse sentido, apresenta-se a definição básica do método dos mínimos quadrados de

acordo com (POLLOCK; POLLOCK, 1979).

Uma equação de regressão da forma:

$$\begin{aligned} y_i &= x_{i1}w_1 + x_{i2}w_2 + \cdots + x_{iP}w_P + \varepsilon_i \\ &= \mathbf{x}_i^T \mathbf{w} + \varepsilon_i \end{aligned} \quad (2.1)$$

explica que o valor da variável dependente y_i em termos do conjunto de P variáveis observáveis em $\mathbf{x}_i = [x_{i1}, x_{i2}, \dots, x_{iP}]^T$ e uma variável aleatória não observada ε_i . O vetor $\mathbf{w} = [w_1, w_2, \dots, w_P]^T$ contém os parâmetros de uma combinação linear das variáveis em \mathbf{x}_i . Um conjunto de N sucessivas realizações do relacionamento da regressão, indexados por $i = 1, 2, \dots, N$, pode ser compilado dentro do sistema

$$\mathbf{y} = \mathbf{X}\mathbf{w} + \boldsymbol{\varepsilon}, \quad (2.2)$$

onde $\mathbf{y} = [y_1, y_2, \dots, y_N]^T$ e $\boldsymbol{\varepsilon} = [\varepsilon_1, \varepsilon_2, \dots, \varepsilon_N]^T$ são os vetores de ordem N e $\mathbf{X} = [x_{iP}]$ é uma matriz de ordem $N \times P$. Deve-se assumir que \mathbf{X} é uma matriz não estocástica com posto $P(\mathbf{X}) = i$ o qual requer $N \geq i$.

De acordo com hipóteses clássicas, os elementos do vetor de ruídos $\boldsymbol{\varepsilon}$ são independentes e identicamente distribuídos com média 0 e variância σ^2 . Assim,

$$\mathcal{E}[\boldsymbol{\varepsilon}] = \mathbf{0} \quad e \quad \mathcal{D}[\boldsymbol{\varepsilon}] = \mathcal{E}[\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^T] = \sigma^2 \mathbf{I}_N. \quad (2.3)$$

A matriz $\mathcal{D}[\boldsymbol{\varepsilon}]$, a qual é descrita como matriz de covariância ou matriz de dispersão de $\boldsymbol{\varepsilon}$, contém a variância comum $\sigma^2 = \mathcal{E}[\{\varepsilon_i - \mathcal{E}[\varepsilon_i]\}^2]$ em cada uma de suas diagonais, suas demais posições são 0, onde cada um deles corresponde a covariância $\mathcal{E}[\{\varepsilon_i - \mathcal{E}[\varepsilon_i]\}\{\varepsilon_j - \mathcal{E}[\varepsilon_j]\}^T]$ de dois elementos distintos de $\boldsymbol{\varepsilon}$.

O valor de \mathbf{w} pode ser estimado de acordo com o princípio da regressão dos mínimos quadrados ordinários através da minimização da função quadrática

$$S = \boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon} = (\mathbf{y} - \mathbf{X}\mathbf{w})^T (\mathbf{y} - \mathbf{X}\mathbf{w}). \quad (2.4)$$

O problema pode ser visto como encontrar um valor para $\hat{\mathbf{y}} = \mathbf{X}\mathbf{w}$ com a menor distância em relação ao vetor \mathbf{y} . Essa interpretação vem do reconhecimento que a função $S = \boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon} = (\mathbf{y} - \mathbf{X}\mathbf{w})^T (\mathbf{y} - \mathbf{X}\mathbf{w})$ representa o quadrado da distância Euclidiana (YOUNG; HOUSEHOLDER, 1938) entre dois vetores. A minimização do valor de \mathbf{w} é encontrado pela diferenciação da função $S(\mathbf{w})$ em relação ao \mathbf{w} e igualando o resultado a 0, tem-se a condição

$$\frac{\partial S}{\partial \mathbf{w}} = 2\mathbf{w}^T \mathbf{X}^T \mathbf{X} - 2\mathbf{y}^T \mathbf{X} = 0 \quad (2.5)$$

Rearranjando a equação acima, obtém-se

$$\mathbf{X}^T \mathbf{X} \mathbf{w} = \mathbf{X}^T \mathbf{y}, \quad (2.6)$$

onde a solução da estimação dos mínimos quadrados ordinários são os parâmetros da regressão:

$$\hat{\mathbf{w}} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}. \quad (2.7)$$

A estimação da componente sistemática das equações da regressão é

$$\begin{aligned} \mathbf{X} \hat{\mathbf{w}} &= \mathbf{X} (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} \\ &= P \mathbf{y}. \end{aligned} \quad (2.8)$$

Aqui, $P = \mathbf{X} (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T$ é chamado de projetor ortogonal ou perpendicular sobre o *manifold* de \mathbf{X} , é uma matriz simétrica com propriedades $P = P^T = P^2$.

2.1.2 Mínimos Quadrados Estocásticos (LMS)

Nesta seção, descreve-se o mais popular algoritmo de aprendizagem iterativo conhecido por *LMS*, o qual foi desenvolvido por (WIDROW *et al.*, 1960). O *LMS* foi o primeiro método linear de filtragem adaptativa para solucionar problemas de predições e equalização de canais de comunicação (HAYKIN *et al.*, 2009).

A principal vantagem do *LMS* é que ele se estabeleceu não apenas como um cavalo de batalha para aplicações de filtragem adaptativa, mas também como referência para outros algoritmos. As razões por trás dessa vantagem são:

- Em termos de complexidade computacional, o *LMS* tem complexidade linear em relação aos parâmetros ajustáveis, o qual faz o algoritmo ser computacionalmente eficiente, além de ser efetivo;
- O algoritmo é simples de codificar e fácil de construir;
- Sobretudo, ele é robusto a distúrbios externos.

Seja, uma matriz de entrada, \mathbf{X}_{NP} , com $\mathbf{x}_i = [x_{i1}, x_{i2}, \dots, x_{ij}]^T$, onde $i = (1, 2, \dots, N)$ e $j = (1, 2, \dots, P)$, e o vetor de saída $\mathbf{y}_N = [y_1, y_2, \dots, y_N]^T$. O problema que deseja-se atacar é como desenhar um modelo de múltiplas entradas e uma única saída de função desconhecida é construído. O modelo opera sob a influência de um algoritmo que controla a necessidade de ajustes dos parâmetros, \mathbf{w} , da função a ser modelada seguindo os seguintes pontos:

- O método inicia a partir de uma configuração arbitrária dos parâmetros \mathbf{w} ;
- Ajustes dos parâmetros (pesos) em resposta as variações do comportamento do sistema são feitas de forma contínua (isto é, são incorporadas dentro da construção do algoritmo);
- Os cálculos dos ajustes dos pesos são realizados, completamente, dentro do intervalo de uma iteração;

O fluxo de trabalho do método consiste em 2 processos contínuos:

1. Processo de filtragem, o qual envolve o cálculo de 2 variáveis:

- uma saída, denotada por \hat{y}_i , que é produzida em resposta aos P elementos do vetor de entrada \mathbf{x}_i ;
- um erro, denotado por ε_i , que é obtido pela comparação da saída \hat{y}_i com a saída y_i correspondente produzida pela função desconhecida. Em outras palavras, y_i é a resposta desejada ou saída alvo.

2. Processo adaptativo, o qual envolve o ajuste automático dos pesos de acordo com o error ε_i

Então, a combinação desses dois processos, trabalhando juntos, constituem um *loop* atuando em torno da função.

Sabe-se que a função é linear, logo, a saída y_i é exatamente:

$$y_i = \sum_{k=1}^P w_{ik} x_{ik} \quad (2.9)$$

onde $w_{i1}, w_{i2}, \dots, w_{iP}$ são os P parâmetros da função, calculada na iteração i . Em forma de matriz, pode-se expressar y_i como um produto interno do vetores \mathbf{x}_i e \mathbf{w}_i como

$$y_i = \mathbf{x}_i^T \mathbf{w}_i \quad (2.10)$$

onde

$$\mathbf{w}_i = [w_{i1}, w_{i2}, \dots, w_{iP}]^T. \quad (2.11)$$

Portanto, pode-se calcular, também, o erro através da equação abaixo

$$\varepsilon_i = y_i - \hat{y}_i. \quad (2.12)$$

A maneira na qual o erro ε_i é usado para controlar os ajustes dos pesos da função é determinado pela função de custo usada pra derivar o algoritmo de filtragem adaptativa de interesse, assim, tem-se um problema de otimização, nesse contexto, portanto, faz-se necessário explicar, rapidamente, o processo de otimização irrestrita.

2.1.2.1 Otimização Irrestrita

Considere uma função de custo $J(\mathbf{w})$ que é uma função continuamente diferenciável com algum vetor de pesos \mathbf{w} desconhecido. A função $J(\mathbf{w})$ mapeia os elementos de \mathbf{w} dentro dos números reais. É o cálculo de como escolher o vetor de parâmetros \mathbf{w} de um algoritmo de filtragem adaptativa para que se comporte da melhor maneira possível. Deseja-se encontrar uma solução \mathbf{w}^* que satisfaça a condição

$$J(\mathbf{w}^*) \leq J(\mathbf{w}) \quad (2.13)$$

isto é, precisa-se resolver um problema de otimização irrestrita, declarada como se segue:

$$\underset{\mathbf{w}}{\text{minimize}} J(\mathbf{w}).$$

A condição necessária para otimalidade é

$$\nabla J(\mathbf{w}^*) = \mathbf{0} \quad (2.14)$$

onde ∇ é o operador gradiente,

$$\nabla = \left[\frac{\partial}{\partial w_1}, \frac{\partial}{\partial w_2}, \dots, \frac{\partial}{\partial w_P} \right]^T \quad (2.15)$$

e $\nabla J(\mathbf{w})$ é o vetor gradiente da função de custo,

$$\nabla J(\mathbf{w}) = \left[\frac{\partial J}{\partial w_1}, \frac{\partial J}{\partial w_2}, \dots, \frac{\partial J}{\partial w_P} \right]^T \quad (2.16)$$

Uma classe de algoritmos de otimização irrestrita são particularmente bem adequadas para o desenho de filtros adaptativos baseados na ideia de uma descida iterativa. Espera-se que o algoritmo irá, eventualmente, convergir para a solução ótima \mathbf{w}^* . Fala-se que "espera-se" pois há

uma possibilidade remota do método divergir, ou seja, tornar-se instável, se algumas precauções não forem tomadas.

Assim, para este trabalho, utilizou-se o método do gradiente descendente proposto por (BERTSEKAS; TSITSIKLIS, 2000) para se resolver o problema de otimização da função de custo do algoritmo *LMS*, na seção seguinte, faz-se uma breve descrição de como funciona o método do gradiente descendente.

2.1.2.2 Método Gradiente Descendente

No método do gradiente descendente, os sucessivos ajustes aplicados ao vetor de pesos \mathbf{w} são na direção à minimização do gradiente, isto é, na oposição oposta ao vetor gradiente $\nabla \mathbf{J}(\mathbf{w})$.

O algoritmo do gradiente descendente é formalmente escrito por

$$\hat{\mathbf{w}}_{i+1} = \hat{\mathbf{w}}_i - \alpha \nabla \mathbf{J}(\hat{\mathbf{w}}_i) \quad (2.17)$$

onde α é uma constante positiva chamado de passo de aprendizagem e $\nabla \mathbf{J}(\hat{\mathbf{w}}_i)$ é o vetor gradiente calculado na iteração i . Indo a partir da iteração i à $i + 1$, o algoritmo aplica a correção

$$\begin{aligned} \Delta \hat{\mathbf{w}}_i &= \hat{\mathbf{w}}_{i+1} - \hat{\mathbf{w}}_i \\ &= -\alpha \nabla \mathbf{J}(\hat{\mathbf{w}}_i) \end{aligned} \quad (2.18)$$

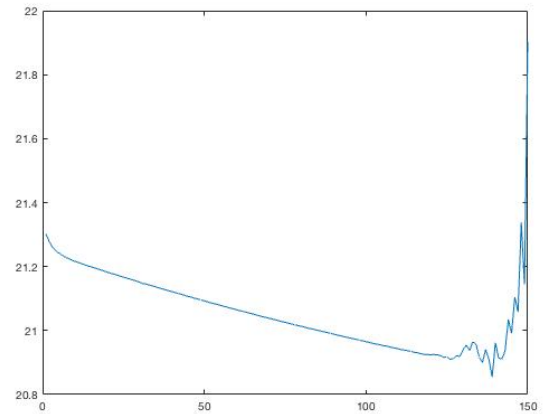
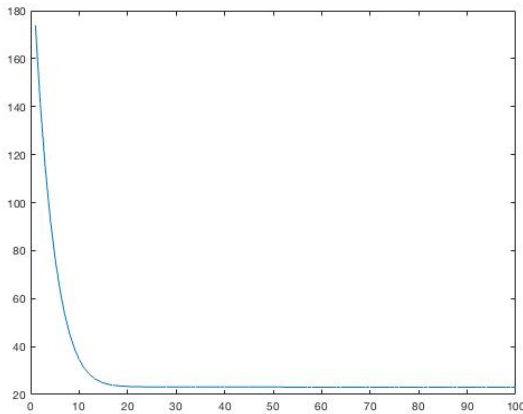
Portanto, o método o gradiente descendente converge para a solução $\hat{\mathbf{w}}^*$ lentamente. Além disso, o passo de aprendizagem α tem uma profunda influência no comportamento de convergência:

- Quando α é pequeno, a resposta do algoritmo é superdimensionada, pois a trajetória traçada por $\hat{\mathbf{w}}_i$ segue um caminho suave no plano, como pode ser verificado pela Figura 1a;
- Quando é grande, a resposta do algoritmo é subestimada, pois a trajetória de $\hat{\mathbf{w}}_i$ segue um caminho em zigue-zague (oscilatório);
- Quando α excede um certo valor crítico, o algoritmo se torna instável, ou seja, diverge, conforme ilustrado na Figura 1b.

Figura 1 – Ilustração da convergência com diferentes valores de α

(a) Convergência para um valor de α pequeno.

(b) Convergência para um valor de α grande.



2.2 Conceitos Básicos de Intervalos

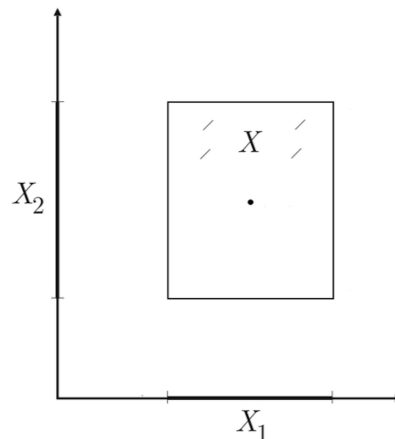
Como descrito em (BOCK; DIDAY, 2012), um dado intervalar é uma classe de dado simbólico que pode ser usado com o objetivo de resumir um grande conjunto de dados. Além disso, há muitas situações, no mundo moderno, onde os valores de uma variável específica podem ser representados como um intervalo, por exemplo, cálculo da pressão sanguínea ou informações de renda média familiar, o qual, normalmente, é considerado em termos de intervalos do salário mínimo, isto é, seja $X = [1, 4]$, onde X é um intervalo de 1 à 4 salários mínimos de um indivíduo ou família. Assim, como enfatizado por (NETO; CARVALHO, 2017), a disponibilidade desse tipo de dado e o recente crescimento do aprendizado de máquina como uma importante forma de análise de dados, tem trazido a necessidade de estender as técnicas de modelagem para dados intervalares. A importância desse tipo de dado tem criado um outro campo de pesquisa o qual estuda e formula todas as operações aritméticas sobre variáveis intervalares, como pode ser visto em (KEARFOTT, 1996).

É relevante mencionar que, seguindo o que foi escrito em (MOORE *et al.*, 2009), todas as variáveis intervalares usadas nesse trabalho serão um intervalo fechado. Denota-se, então, uma variável intervalar $X = [x^L, x^U]$ com um conjunto de números reais dados por:

$$X = [x^L, x^U] = \{x \in \mathcal{R} : x^L \leq x \leq x^U\}.$$

Conforme mostrado na lista de símbolos, adota-se, neste trabalho, letras maiúsculas para representar uma variável intervalar, seus limites (*endpoints* ou *bounds*) x^L e x^U são representados por letras minúsculas, pois são números escalares.

Figura 2 – Ilustração do Vetor de Intervalo $X = (X_1, X_2)$



Portanto, pode-se, agora, definir alguma das propriedades e operações aplicadas às variáveis intervalares conforme descrito abaixo e definido por (MOORE *et al.*, 2009):

- dois intervalos X e Y são ditos serem iguais se eles tem os mesmos conjuntos. Na prática, isso acontece se eles possuem limites iguais:

$$X = Y \text{ se } x^L = y^L \text{ e } x^U = y^U$$

- fala-se que X é degenerado se $x^L = x^U$. Como um intervalo contém um único número real x , por convenção, concorda-se em identificá-lo como um intervalo degenerado $[x, x]$ com o número real x . Nesse sentido, pode-se escrever como:

$$0 \longrightarrow [0, 0]$$

- O raio (*range*) de um intervalo X é definido e denotado por

$$x^R = \frac{(x^U - x^L)}{2}$$

- O ponto médio ou centro (*center*) de um intervalo X é computado seguindo a equação

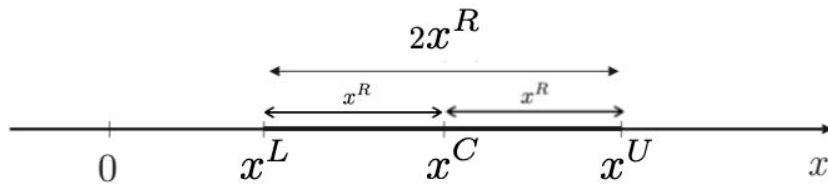
$$x^C = \frac{(x^L + x^U)}{2}$$

Na Figura 3 pode-se ver a ilustração do centro, raio e os limites inferior e superior de um intervalo no plano.

Portanto, exemplificadas as definições acima, pode-se, também, representar um intervalo usando-se apenas seus valores de centro e raio

$$X = [x^C - x^R, x^C + x^R]$$

Figura 3 – Raio, Centro e Limites de um Intervalo



2.3 Regressão Linear para Dados Intervalares

Ao contrário dos métodos clássicos para tratamento de dados que foram desenvolvidos ao longo de mais de um século, os métodos para tratamento de dados intervalares estão crescendo a cada dia. Portanto, nessa sub-seção apresentar-se-ão alguns dos algoritmos de regressão linear para dados intervalares baseados nos mínimos quadrados (*OLS*). Os três principais métodos para esse tipo de dado são: método do centro (*CM*), método do centro e do raio (*CRM*) e o método do centro e do raio restrito (*CCRM*) (FAGUNDES, 2013).

2.3.1 Método do Centro (*CM*)

Um dos primeiros métodos de regressão linear para dados do tipo intervalo, foi apresentado por (BILLARD; DIDAY, 2000), chama-se Método do Centro (*CM*), o qual tem como objetivo ajustar um modelo de regressão linear utilizando apenas os centros das variáveis intervalares, ou seja, construir um modelo onde todas as entradas serão o centros calculados a partir dos dados do tipo intervalo, em seguida, aplica-se esse modelo aos limites inferiores e superiores dos intervalos das variáveis de entrada para prever, respectivamente, o limite inferior e superior da variável de saída. Portanto, a função de custo desse método a qual busca-se minimizar será a soma dos quadrados dos erros dos centros dos intervalos.

Seja $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ um conjunto de exemplos, onde cada $\mathbf{x}_i = [X_{i1}, \dots, X_{iP}]^T$, $i = (1, 2, \dots, N)$, $X_{ij} = [x_{ij}^L, x_{ij}^U] \in \Omega = \{[x_{ij}^L, x_{ij}^U] \mid x_{ij}^L \leq x_{ij}^U; x_{ij}^L, x_{ij}^U \in \mathcal{R}^1\} (j = 1, \dots, P)$ e $Y_i = [y_i^L, y_i^U] \in \Omega$.

Considerando que o conjunto de variáveis \mathbf{x}_i como variáveis predictoras independentes relacionadas linearmente com uma variável resposta dependente Y_i seguindo o modelo:

$$\begin{aligned} y_i^L &= w_0 + w_1 x_{i1}^L + w_2 x_{i2}^L + \dots + w_P x_{iP}^L + \varepsilon_i^L \\ y_i^U &= w_0 + w_1 x_{i1}^U + w_2 x_{i2}^U + \dots + w_P x_{iP}^U + \varepsilon_i^U \end{aligned} \quad (2.19)$$

Seguindo as Equações 2.19, pode-se encontrar os parâmetros minimizando-se a soma dos quadrados dos erros dos limites inferiores e superiores dado por:

$$\sum_{i=1}^N (\varepsilon_i^L + \varepsilon_i^U)^2 = \sum_{i=1}^N (y_i^L - w_0 - w_1 x_{i1}^L - \dots - w_P x_{iP}^L + y_i^U - w_0 - w_1 x_{i1}^U - \dots - w_P x_{iP}^U)^2, \quad (2.20)$$

observando-se, então, a Equação 2.20, verifica-se que a estimação dos limites inferiores e superiores utilizam os mesmos parâmetros $\mathbf{w} = (w_0, w_1, \dots, w_P)$, os quais serão estimados utilizando os valores dos centros como dito anteriormente, portanto, pode-se reescrever a expressão dos parâmetros estimados em notação matricial, conforme mostrado abaixo:

$$\hat{\mathbf{w}} = (\mathbf{A})^{-1} \mathbf{b} \quad (2.21)$$

onde \mathbf{A} é uma matriz de dimensão $(P+1) \times (P+1)$ e \mathbf{b} é um vetor com dimensão $(P+1) \times 1$, calculados como:

$$\mathbf{A} = \begin{pmatrix} N & \sum_i x_{i1}^C & \dots & x_{iP}^C \\ \sum_i x_{i1}^C & \sum_i (x_{i1}^C)^2 & \dots & \sum_i x_{iP}^C x_{i1}^C \\ \vdots & \vdots & \vdots & \vdots \\ \sum_i x_{iP}^C & \sum_i x_{i1}^C x_{iP}^C & \dots & \sum_i (x_{iP}^C)^2 \end{pmatrix} \quad \text{e } \mathbf{b} = \left(\sum_i y_i^C, \sum_i y_i^C x_i \right) \quad (2.22)$$

Portanto, a Equação 2.21 pode ser representada seguindo o que foi explicado na seção 2.1.1:

$$\hat{\mathbf{w}} = [(\mathbf{X}^C)^T \mathbf{X}^C]^{-1} (\mathbf{X}^C)^T \mathbf{y}^C \quad (2.23)$$

onde, $\mathbf{X}^C = ((\mathbf{x}_1^C)^T, (\mathbf{x}_2^C)^T, \dots, (\mathbf{x}_N^C)^T)$, $\mathbf{x}_i^C = (1, x_{i1}^C, \dots, x_{iP}^C)^T$ e $\mathbf{y}^C = (y_1^C, y_2^C, \dots, y_N^C)^T$. Além disso, tem-se que $x_{ij}^C = \frac{x_{ij}^L + x_{ij}^U}{2}$ e $y_i^C = \frac{y_j^L + y_j^U}{2}$.

Logo, ao se aplicar o modelo para prever \mathbf{y} tem-se:

$$\begin{aligned} \hat{\mathbf{y}}^L &= (\mathbf{X}^L)^T \hat{\mathbf{w}} \\ \hat{\mathbf{y}}^U &= (\mathbf{X}^U)^T \hat{\mathbf{w}} \end{aligned} \quad (2.24)$$

Como pode se ver, esse método não assegura que $Y^L \leq Y^U$. Se isso não é garantido, uma importante regra dos dados intervalares será quebrada e o resultado não será considerado um intervalo válido.

2.3.2 Método do Centro e Raio (CRM)

A ideia básica do método do centro e raio proposto por (NETO; CARVALHO, 2008) foi considerar uma nova abordagem para ajustar um modelo de regressão linear aos dados intervalares usando a informação contida nos centros e raios dos intervalos com o objetivo de melhorar a performance de predição do modelo em comparação com o método do centro (CM).

Seja $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ um conjunto de exemplos, onde cada $\mathbf{x}_i = (X_{i1}, \dots, X_{iP})^T$, $i = (1, 2, \dots, N)$, $X_{ij} = [x_{ij}^L, x_{ij}^U] \in \Omega = \{[x_{ij}^L, x_{ij}^U] \mid x_{ij}^L \leq x_{ij}^U; x_{ij}^L, x_{ij}^U \in \mathcal{R}^1\} (j = 1, \dots, P)$ e $Y_i = [y_i^L, y_i^U] \in \Omega$. Seja, também, \mathbf{y}^C e \mathbf{X}^C o conjunto de centros das variáveis quantitativas ou de saída e o conjunto de centros das variáveis de entrada, respectivamente, \mathbf{y}^R e \mathbf{X}^R os respectivos conjuntos de raios correspondentes a cada variável y_i e \mathbf{x}_i , onde o cálculo utilizado para se obter os centros e raios dos intervalos seguiram as equações descritas na seção 2.2.

Assim, consideram-se \mathbf{y}^C e \mathbf{y}^R os conjuntos de variáveis dependentes, \mathbf{X}^C e \mathbf{X}^R os conjuntos de variáveis preditoras independentes que são relacionadas de acordo com a seguinte regressão linear:

$$\begin{aligned} y_i^C &= w_0^C + w_1^C x_{i1}^C + \dots + w_P^C x_{iP}^C + \varepsilon_i^C \\ y_i^R &= w_0^R + w_1^R x_{i1}^R + \dots + w_P^R x_{iP}^R + \varepsilon_i^R. \end{aligned} \quad (2.25)$$

Então, a soma dos quadrados dos erros dos centros e raios do Método do Centro e Raio (CRM) é dado por

$$\begin{aligned} S_{crm} &= \sum_{i=1}^N (\varepsilon_i^C)^2 + \sum_{i=1}^N (\varepsilon_i^R)^2 \\ &= \sum_{i=1}^N (y_i^C - w_0^C - w_1^C x_{i1}^C - \dots - w_P^C x_{iP}^C)^2 + \sum_{i=1}^N (y_i^R - w_0^R - w_1^R x_{i1}^R - \dots - w_P^R x_{iP}^R)^2, \end{aligned} \quad (2.26)$$

Então, a minimização da equação acima é equivalente a se ajustar dois modelos de regressão independentes sobre os centros e os raios dos intervalos, respectivamente.

Portanto, a regressão linear para os centros pode ser escrita como:

$$\mathbf{y}^C = \mathbf{X}^C \mathbf{w}^C + \boldsymbol{\varepsilon}^C, \quad (2.27)$$

onde $\mathbf{y}^C = (y_1^C, y_2^C, \dots, y_N^C)^T$, $\mathbf{X}^C = ((\mathbf{x}_1^C)^T, (\mathbf{x}_2^C)^T, \dots, (\mathbf{x}_N^C)^T)^T$,
 $(\mathbf{x}_i^C)^T = (1, x_{i1}^C, \dots, x_{iP}^C)$, $\mathbf{w}^C = (w_0^C, w_1^C, \dots, w_P^C)^T$, $\boldsymbol{\varepsilon}^C = (\varepsilon_1^C, \varepsilon_2^C, \dots, \varepsilon_N^C)^T$.

De forma similar, a equação da regressão linear sobre os raios é dada como se segue:

$$\mathbf{y}^R = \mathbf{X}^R \mathbf{w}^R + \boldsymbol{\varepsilon}^R, \quad (2.28)$$

$$\text{onde } \mathbf{y}^R = (y_1^R, y_2^R, \dots, y_N^R)^T, \mathbf{X}^R = ((\mathbf{x}_1^R)^T, (\mathbf{x}_2^R)^T, \dots, (\mathbf{x}_N^R)^T)^T, \\ (\mathbf{x}_i^R)^T = (1, x_{i1}^R, \dots, x_{iP}^R), \mathbf{w}^R = (w_0^R, w_1^R, \dots, w_P^R)^T, \boldsymbol{\varepsilon}^R = (\varepsilon_1^R, \varepsilon_2^R, \dots, \varepsilon_N^R)^T.$$

Portanto, usando o método dos mínimos quadrados para estimar $\hat{\mathbf{w}}^C$ e $\hat{\mathbf{w}}^R$ nas equações mostradas acima, podemos escrever a estimação como:

$$\hat{\mathbf{w}}^C = ((\mathbf{X}^C)^T \mathbf{X}^C)^{-1} (\mathbf{X}^C)^T \mathbf{y}^C \\ \hat{\mathbf{w}}^R = ((\mathbf{X}^R)^T \mathbf{X}^R)^{-1} (\mathbf{X}^R)^T \mathbf{y}^R \quad (2.29)$$

Assim, dado um novo exemplo $\mathbf{x} = (X_1, X_2, \dots, X_P)^T$, onde $\mathbf{x}^C = (x_1^C, x_2^C, \dots, x_P^C)^T$ e $\mathbf{x}^R = (x_1^R, x_2^R, \dots, x_P^R)^T$ o valor de $\hat{y} = [\hat{y}^L, \hat{y}^U]$ predito a partir dos valores preditos para \hat{y}^C e \hat{y}^R é calculado como mostrado abaixo:

$$\hat{y}^L = \hat{y}^C - \hat{y}^R \\ \hat{y}^U = \hat{y}^C + \hat{y}^R. \quad (2.30)$$

Pode-se observar que, diferentemente, do método do centro, o *CRM* utiliza duas regressões lineares independentes sobre os centros e os raios das variáveis intervalares a fim de prever os valores dos limites inferiores e superiores, seguindo as propriedades já explicadas na primeira seção deste capítulo. Contudo, mesmo diante dessa melhora no modelo, não há, porém, nenhuma restrição na regressão sobre os raios, ou seja, pode-se prever um valor para um dado $\hat{y}^R < 0$, o que invalida a existência de um intervalo válido, pois isso viola a regra que para todo intervalo, os limites inferiores devem ser menores ou iguais que os limites superiores.

2.3.3 Método do Centro e Raio Restritos (CCRM)

O método do centro e do raio mostrado na seção anterior, demonstrou a importância de incluir as informações fornecidas pelo centro e o raio de um intervalo em um modelo de regressão linear, porém, o modelo que foi proposto não garante, matematicamente, que $\hat{y}_i^L \leq \hat{y}_i^U$. Portanto, nesse contexto, os mesmos autores propuseram em (NETO; CARVALHO, 2010), um novo modelo de regressão linear, agora, restringindo os valores dos parâmetros $\hat{\mathbf{w}}^R$ estimados pelo algoritmo, desse modo, conseguiram assegurar que o $\hat{y}_i = [\hat{y}_i^L, \hat{y}_i^U]$, $\hat{y}_i^L \leq \hat{y}_i^U$.

Assim, considere, novamente, \mathbf{y}^C e \mathbf{y}^R sejam variáveis dependentes e X_j^C e X_j^R ($j = 1, 2, \dots, P$) as variáveis preditoras independentes, mas, agora, relacionadas de acordo com a regressão linear:

$$\begin{aligned} y_i^C &= w_0^C + w_1^C x_{i1}^C + \dots + w_P^C x_{iP}^C + \varepsilon_i^C \\ y_i^R &= w_0^R + w_1^R x_{i1}^R + \dots + w_P^R x_{iP}^R + \varepsilon_i^R, \end{aligned} \quad (2.31)$$

com restrições $w_j^R \geq 0$, $j = 0, \dots, P$.

Observe que o modelo de regressão ajustado aos raios dos intervalos irá ser calculado considerando restrições de desigualdades cujo objetivo é assegurar que $y_i^R \geq 0$, o qual significa que \hat{y}_i^L será sempre menor ou igual ao \hat{y}_i^U . Vale ressaltar, sempre, que os parâmetros \mathbf{w}^C não possuem nenhuma restrição.

Assim, no método do centro e raio restritos (*CCRM*), a soma dos quadrados dos erros dos centros e dos raios é dado por:

$$\begin{aligned} \mathbf{S}_{ccrm} &= \sum_{i=1}^N (\varepsilon_i^C)^2 + \sum_{i=1}^N (\varepsilon_i^R)^2 \\ &= \sum_{i=1}^N (y_i^C - w_0^C - w_1^C x_{i1}^C - \dots - w_P^C x_{iP}^C)^2 + \sum_{i=1}^N (y_i^R - w_0^R - w_1^R x_{i1}^R - \dots - w_P^R x_{iP}^R)^2, \end{aligned} \quad (2.32)$$

restrito por $w_j^R \geq 0$, $j = 0, \dots, P$,

Na seção 2.1.1, apresentou-se o método dos mínimos quadrados ordinários sem restrições de \mathbf{w}^C e \mathbf{w}^R . Contudo, observe que o modelo proposto na Equação 2.32 será ajustado de acordo com as restrições de desigualdade $w_j^R \geq 0$, $j = 0, \dots, P$. Pode-se reescrever o *CCRM* utilizando-se notação matricial como

$$\begin{aligned} \mathbf{y}^C &= \mathbf{X}^C \mathbf{w}^C + \varepsilon_C \\ \mathbf{y}^R &= \mathbf{X}^R \mathbf{w}^R + \varepsilon_R, \end{aligned} \quad (2.33)$$

com restrições $w_j^R \geq 0$, $j = 0, \dots, P$,

onde, \mathbf{X}^C e \mathbf{X}^R possuem posto completo $P + 1 \leq N$, $\mathbf{y}^C = (y_1^C, \dots, y_N^C)^T$, $\mathbf{X}^C = ((\mathbf{x}_1^C)^T, \dots, (\mathbf{x}_N^C)^T)^T$, $(\mathbf{x}_i^C)^T = (1, X_{i1}^C, \dots, X_{iP}^C)$, $\mathbf{w}^C = (w_0^C, \dots, w_P^C)^T$, $\varepsilon^C = (\varepsilon_1^C, \dots, \varepsilon_N^C)^T$, $\mathbf{y}^R = (y_1^R, \dots, y_N^R)^T$, $\mathbf{X}^R = ((\mathbf{x}_1^R)^T, \dots, (\mathbf{x}_N^R)^T)^T$, $(\mathbf{x}_i^R)^T = (1, X_{i1}^R, \dots, X_{iP}^R)$, $\mathbf{w}^R = (w_0^R, \dots, w_P^R)^T$ com restrições $w_j^R \geq 0$, $j = 0, \dots, P$ e $\varepsilon^R = (\varepsilon_1^R, \dots, \varepsilon_N^R)^T$.

Sabendo-se que o processo de estimação dos parâmetros \mathbf{w}^C não possuem restrições, pode-se, então, aplicar o método dos mínimos quadrados conforme a Equação 2.7 para estimar os valores de $\hat{\mathbf{w}}^C$:

$$\hat{\mathbf{w}}^C = ((\mathbf{X}^C)^T \mathbf{X}^C)^{-1} (\mathbf{X}^C)^T \mathbf{y}^C. \quad (2.34)$$

Contudo, a segunda equação na Expressão 2.33 deve ser ajustada de acordo com as restrições de desigualdade $\mathbf{w}_R \geq \mathbf{0}$, nesse sentido, o método dos mínimos quadrados ordinários, em sua formulação original, como dito anteriormente, não funciona com restrições, então, (NETO; CARVALHO, 2010) utilizaram uma solução desenvolvida para esse problema através de técnicas iterativas de programação quadrática apresentadas por (JUDGE; TAKAYAMA, 1966) cujo algoritmo da solução foi apresentado em (LAWSON; HANSON, 1995), assegurando-se a positividade do vetor de parâmetros \mathbf{w}^R estimados pelo mínimos quadrados, se \mathbf{X}^R tem posto completo $P + 1 \leq N$. A ideia básica do algoritmo proposto por (LAWSON; HANSON, 1995) é identificar os valores que são incompatíveis com as restrições e alterá-los para valores não negativos através de um processo de re-ponderação.

Assim, uma vez definidas as equações das regressões lineares propostas pelo (CCRM), pode-se descrever a etapa de predição dos valores de saída do modelo. Dado um novo exemplo $\mathbf{x} = (X_1, X_2, \dots, X_P)$, onde $\mathbf{x}^C = (x_1^C, x_2^C, \dots, x_P^C)$ e $\mathbf{x}^R = (x_1^R, x_2^R, \dots, x_P^R)$ o valor de $\hat{y} = [\hat{y}^L, \hat{y}^U]$ predito a partir dos valores preditos para \hat{y}^C e \hat{y}^R é calculado como mostrado abaixo:

$$\begin{aligned} \hat{y}^L &= \hat{y}^C - \hat{y}^R \\ \hat{y}^U &= \hat{y}^C + \hat{y}^R, \end{aligned} \quad (2.35)$$

onde, $\hat{\mathbf{y}}^C = (\mathbf{x}^C)^T \mathbf{w}^C$, $\hat{\mathbf{y}}^R = (\mathbf{x}^R)^T \mathbf{w}^R$, $(\mathbf{x}^C)^T = (1, x_1^C, \dots, x_P^C)$,
 $(\mathbf{x}^R)^T = (1, x_1^R, \dots, x_P^R)$, $(\hat{\mathbf{w}}^C)^T = (\hat{w}_0, \hat{w}_1^C, \dots, \hat{w}_P^C)^T$ e $(\hat{\mathbf{w}}^R)^T = (\hat{w}_0, \hat{w}_1^R, \dots, \hat{w}_P^R)^T$
 com $\hat{w}_j^R \geq 0$, $j = 0, \dots, P$.

Novamente, nota-se que, apesar das melhorias propostas pelos autores do CCRM, pode-se visualizar uma falha fundamental na concepção do modelo de regressão restrito ajustado sobre os valores de \mathbf{X}^R , pois, uma vez que os pesos ajustados são forçados a serem maiores ou iguais a 0, o modelo não conseguirá prever os valores das variáveis de saída de forma adequada, quando houver uma relação linear onde algum dos valores dos pesos que multiplicam \mathbf{X}^R forem negativos.

3 MÉTODOS PROPOSTOS

Uma vez que têm-se descritos os fundamentos teóricos, os quais foram utilizados para o desenvolvimento deste trabalho, observou-se que cada uma das soluções propostas para regressão linear intervalar, desde o método do centro proposto por (BILLARD; DIDAY, 2000), passando pelo método do centro e raio descrito em (NETO; CARVALHO, 2008) e chegando ao método do centro e raio restrito desenvolvido em (NETO; CARVALHO, 2010), possuem problemas que ainda precisam ser melhorados a fim de se obter modelos de regressão linear mais adequados aos dados do tipo intervalo.

Nesse contexto, os problemas que motivaram o desenvolvimento desses dois próximos métodos foram a necessidade da geração de uma regressão onde os pesos sejam otimizados iterativamente, pois em caso de conjunto de dados muito grandes todas as soluções propostas utilizando os mínimos quadrados ordinários, necessitam de um alto poder de processamento, dada a característica do algoritmo de fazer a otimização em lote.

Além disso, o método do centro e raio restrito que foi proposto com o objetivo de melhorar algoritmos anteriores resguardando a propriedade de que os limites inferiores serão sempre menores ou iguais aos limites superiores, utilizou-se de uma restrição muito forte, fazendo com que os dados que possuírem uma relação linear com coeficientes negativos para os raios, não consigam ter seus pesos ajustados de maneira a ter boas previsões. Portanto, diante desses 2 problemas encontrados, este trabalho apresenta duas soluções que resolvem tais problemas relatados anteriormente.

3.0.1 Mínimos Quadrados Ordinários Log Transformados para Dados Intervalares (LOID)

Com o objetivo de assegurar que $\mathbf{x}_i^L \leq \mathbf{x}_i^U$ o LOID propõe, assim como o CCRM, o ajuste de duas regressões lineares para os centros e raios, respectivamente, porém, com uma transformação logarítmica nos dados na etapa de treinamento.

Assim, seja \mathbf{X}_{NP} e \mathbf{y}_N a matriz de entrada e o vetor de saída, respectivamente, onde $X_{ij} = [x_{ij}^L, x_{ij}^U]$, $Y_i = [y_i^L, y_i^U]$, $i = (1, 2, \dots, N)$ e $j = (1, 2, \dots, P)$. Tem-se, portanto, \mathbf{X}_{NP}^C e \mathbf{X}_{NP}^R os centros e raios de \mathbf{X} , respectivamente, assim \mathbf{y}_N^C os centros do vetor de variáveis dependentes (ou saída) e \mathbf{y}_N^R os raios dessas variáveis. Em notação matricial, o método LOID propos pode ser escrito como:

$$\begin{aligned}\hat{\mathbf{y}}^C &= \mathbf{X}^C \hat{\mathbf{w}}^C + \boldsymbol{\varepsilon}^C, \\ \hat{\mathbf{y}}^R &= \mathbf{X}^R \hat{\mathbf{w}}^R + \boldsymbol{\varepsilon}^R,\end{aligned}\tag{3.1}$$

onde, $\hat{\mathbf{w}}^C = (\hat{w}_0^C, \hat{w}_1^C, \dots, \hat{w}_p^C)^T$, $\hat{\mathbf{w}}^R = (\hat{w}_0^R, \hat{w}_1^R, \dots, \hat{w}_p^R)^T$, $\boldsymbol{\varepsilon}^C = (\varepsilon_1^C, \dots, \varepsilon_p^C)^T$, and $\boldsymbol{\varepsilon}^R = (\varepsilon_1^R, \dots, \varepsilon_p^R)^T$.

Então, a soma dos quadrados dos erros é dada por

$$\begin{aligned}\varepsilon_{LOID} &= \sum_{i=1}^N (\varepsilon_i^C)^2 + \sum_{i=1}^N (\varepsilon_i^R)^2 \\ &= \sum_{i=1}^N (y_i^C - \hat{w}_0^C - \hat{w}_1^C x_{i1}^C - \dots - \hat{w}_p^C x_{ip}^C)^2 \\ &\quad + \sum_{i=1}^N (y_i^R - \hat{w}_0^R - \hat{w}_1^R x_{i1}^R - \dots - \hat{w}_p^R x_{ip}^R)^2.\end{aligned}\tag{3.2}$$

Os parâmetros $\hat{\mathbf{w}}^C$ na primeira expressão da Equação (3.1) são encontrados via a formulação padrão dos mínimos quadrados ordinários descrito na seção 2.1.1

$$\hat{\mathbf{w}}^C = ((\mathbf{X}^C)^T \mathbf{X}^C)^{-1} (\mathbf{X}^C)^T \mathbf{y}^C,\tag{3.3}$$

onde faz-se necessário enfatizar que os parâmetros $\hat{\mathbf{w}}^C$ não são restritos.

A segunda expressão na Equação (3.1) deve ser ajustada seguindo a restrição $\mathbf{y}^R \geq 0$. Então, asseguram-se que os valores preditos para \mathbf{y}^R são maiores ou iguais a 0, usando uma transformação logarítmica sobre os raios das variáveis dependentes \mathbf{y}^R , a qual será utilizada para ajustar os parâmetros $\hat{\mathbf{w}}^R$.

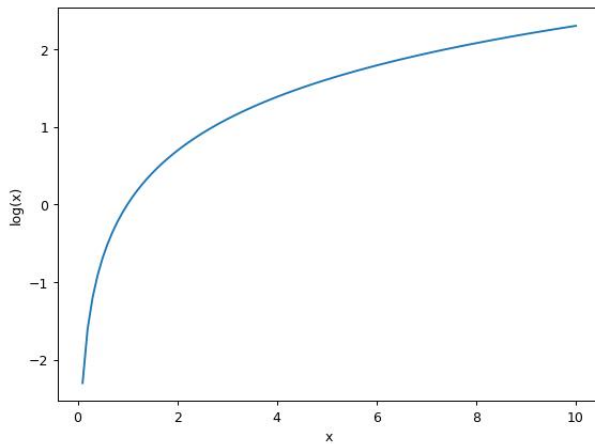
A transformação logarítmica oferece a possibilidade de conseguir valores a partir de qualquer componente de $\hat{\mathbf{w}}^R \leq 0$, essa é uma importante característica caso o conjunto de dados gerado ou obtido tenha esse comportamento, contudo, aplicando-se essa transformação, adiciona-se uma não linearidade aos sobre os dados, embora aja uma maneira de minimizá-la, usando apenas uma simples normalização sobre \mathbf{X} e \mathbf{y} , para isso, dividiu-se todos os elementos por seus respectivos valores máximos, isto é, $\mathbf{X}/\max(\mathbf{X})$ e $\mathbf{y}/\max(\mathbf{y})$.

Conforme ilustrado na Figura 4, pode-se observar a importância da normalização dos dados além de se entender como o processo de transformação assegura que os valores de $\hat{\mathbf{y}}^R$ sempre serão maiores que 0. Na Figura 4a observa-se que a não linearidade quando os

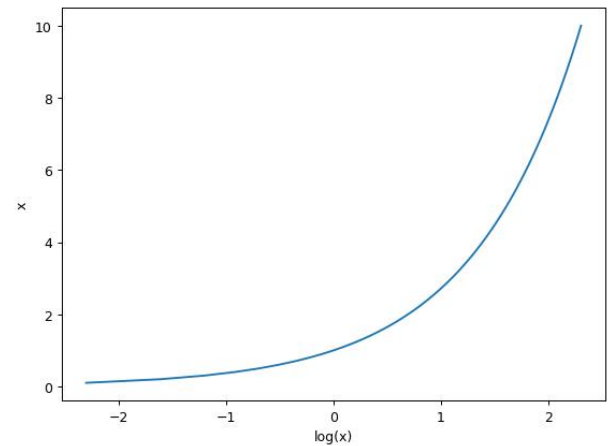
dados então entre 0 e 1 é mínima, portanto, isso faz com que o *LOID* consiga obter êxito na sua proposta, já na Figura 4b, tem-se que é necessário aplicar a função exponencial para se obter, de fato, o $\hat{\mathbf{y}}^R$ pois como pode-se ver os valores serão sempre positivos.

Figura 4 – Gráficos das Funções Logarítmica e Exponencial

(a) Função Logarítmica



(b) Função Exponencial



O algoritmo 1, abaixo, resume de forma geral, a etapa de treinamento da abordagem proposta.

Algoritmo 1: Etapa de Treinamento do **LOID**

Require: \mathbf{X}, \mathbf{y} {conjunto de dados de treinamento}

- 1: $\mathbf{X} \leftarrow \frac{\mathbf{X}}{\max(\mathbf{X})}$ {aplica a normalização em \mathbf{X} }
 - 2: $\mathbf{y} \leftarrow \frac{\mathbf{y}}{\max(\mathbf{y})}$ {aplica a normalização em \mathbf{y} }
 - 3: $N, P \leftarrow \dim(\mathbf{X})$ {obtem a dimensão da matriz de entrada}
 - 4: $\mathbf{y}_{train} \leftarrow \log \mathbf{y}$ {aplica a transformação logarítmica nas amostras de saída}
 - 5: $\mathbf{X}_{train} \leftarrow [\mathbf{1}_N \mathbf{X}]_{P(P+1)}$ {adiciona o *bias* (viés)}
 - 6: $\hat{\mathbf{w}} \leftarrow (\mathbf{X}_{train}^T \mathbf{X}_{train})^{-1} \mathbf{X}_{train}^T \mathbf{y}_{train}$
-

Na primeira linha do Algoritmo 1, faz-se a normalização dos dados, aqui, adota-se uma normalização através da divisão de cada elemento \mathbf{x}_{ij} da matriz \mathbf{X} , onde $i = (1, \dots, N)$ e $j = (1, \dots, P)$ pelo máximo de cada componente j da matriz. O mesmo acontece, na segunda linha, agora para o vetor de *targets* \mathbf{y} , após a normalização dos dados de entrada, calcula-se os valores da quantidade de linhas N e a quantidade de *features* P , então, aplica-se a transformação logarítmica em \mathbf{y} , adiciona-se o viés (*bias*) à matriz de entrada \mathbf{X} , por fim, faz-se a estimação dos parâmetros $\hat{\mathbf{w}}$ através do método dos mínimos quadrados.

Uma vez que têm-se os valores de $\hat{\mathbf{w}}$ pode-se testar o modelo, assim, dado um novo exemplo $\mathbf{x} = (X_1, X_2, \dots, X_P)^T$, onde $\mathbf{x}^C = (x_1^C, x_2^C, \dots, x_P^C)^T$ e $\mathbf{x}^R = (x_1^R, x_2^R, \dots, x_P^R)^T$ o valor de $\hat{y} = [\hat{y}^L, \hat{y}^U]$ pode ser predito de acordo com os passos no Algoritmo 2. É importante salientar que, como visto nas linhas 3 e 4, primeiro, faz-se a predição da saída transformada pela aplicação da função logarítmica no Algoritmo 1, porém, o objetivo é prever a saída a partir de uma entrada sem transformação, assim, deve-se aplicar a função inversa da logarítmica, isto é, a função exponencial, como mostrado acima, seguindo os passos do algoritmo abaixo:

Algoritmo 2: Etapa de Teste do **LOID**

Require: \mathbf{X} {conjunto de dados de teste}

- 1: $N, P \leftarrow \dim(\mathbf{X})$ {obtem a dimensão da matriz de entrada}
 - 2: $\mathbf{X}_{train} \leftarrow [\mathbf{1}^N \mathbf{X}]_{P(P+1)}$ {adiciona o *bias* (viés)}
 - 3: $\hat{\mathbf{y}}_{aux} \leftarrow \mathbf{X}_{test} \hat{\mathbf{w}}^T$ {calcula a saída transformada}
 - 4: $\hat{\mathbf{y}} \leftarrow \exp(\hat{\mathbf{y}}_{aux})$ {calcula a saída}
-

Novamente, o Algoritmo 2, reflete os passos da etapa de teste do *LOID* de forma genérica, como pode-se observar, nas duas primeiras linhas calcula-se apenas as dimensões do conjunto de dados de entrada e adiciona-se o bias, após isso, calcula-se o valor de $\hat{\mathbf{y}}_{aux}$ cujo valor obtido ainda não é o esperado, dado que ele é um valor que ainda encontra-se transformado devido a transformação aplicada aos dados de treinamento, assim, o último passo do algoritmo é realizar a função inversa da transformação, no caso, a função exponencial, para se obter os valores reais de $\hat{\mathbf{y}}$ preditos pelo modelo.

Por fim, faz-se necessário recordar, novamente, que os Algoritmos 1 e 2 são aplicados apenas para se obter o modelo para os raios das variáveis intervalares.

3.0.2 Mínimos Quadrados Estocásticos Log Transformados para Dados Intervalares (LLID)

Nesta subseção, apresenta-se o método proposto denominado de Mínimos Quadrados Estocásticos *Log* Transformados para Dados Intervalares, que diferentemente do *LOID* mostrado na subseção anterior, é uma versão iterativa o qual usará como método de otimização, o Gradiente Descendente que foi descrito na Seção 2.1.2.2 além de ser uma extensão do *LMS* para dados do tipo intervalo, seguindo o algoritmo descrito na subseção 2.1.2.

seja \mathbf{X}_{NP} e \mathbf{y}_N a matriz de entrada e o vetor de saída, respectivamente, onde $X_{ij} = [x_{ij}^L, x_{ij}^U], Y_i = [y_i^L, y_i^U], i = (1, 2, \dots, N)$ e $j = (1, 2, \dots, P)$. Tem-se, portanto, \mathbf{X}_{NP}^C e \mathbf{X}_{NP}^R os centros e raios de \mathbf{X} , respectivamente, assim \mathbf{y}_N^C os centros do vetor de variáveis dependentes (ou

saída) e \mathbf{y}_N^R os raios dessas variáveis.

Assumindo-se a existência de uma função contínua $f : \mathcal{X} \rightarrow \mathcal{Y}$ entre o espaço de entrada e de saída, o objetivo é estimar os parâmetros de f a partir dos dados disponíveis. Assim, considere a função linear f , isto é:

$$\hat{y}_i = f(\mathbf{x}_i) = \mathbf{x}_i^T \mathbf{w}, \quad (3.4)$$

$$\text{onde, } \mathbf{x}_i = (x_{i1}, \dots, x_{iP})^T \text{ e } \mathbf{w} = (w_1, w_2, \dots, w_P)^T.$$

Então, assim como acontece no método *LOID* apresentado anteriormente, ajustam-se duas diferentes funções, uma baseada em estimar $\hat{\mathbf{w}}^C$ e outra cujo objetivo é encontrar $\hat{\mathbf{w}}^R$. A última, porém, deve considerar a restrição $\hat{\mathbf{y}}^R \geq 0$. Assim, pode-se reescrever a Equação 3.4 duas vezes como se segue:

$$\begin{aligned} \hat{y}_i^C &= f(\mathbf{x}_i^C) = (\mathbf{x}_i^C)^T \hat{\mathbf{w}}^C, \\ \hat{y}_i^R &= f(\mathbf{x}_i^R) = (\mathbf{x}_i^R)^T \hat{\mathbf{w}}^R, \end{aligned} \quad (3.5)$$

$$\text{onde } \mathbf{x}_i^C = (x_{i1}^C, \dots, x_{iP}^C)^T, \mathbf{w}^C = (w_1^C, \dots, w_P^C)^T, \mathbf{x}_i^R = (x_{i1}^R, \dots, x_{iP}^R)^T \text{ e } \mathbf{w}^R = (w_1^R, \dots, w_P^R)^T.$$

Pode-se, então, calcular o erro da saída da i -ésima amostra:

$$\begin{aligned} \varepsilon_i^C &= y_i^C - \hat{y}_i^C, \\ \varepsilon_i^R &= \log(y_i^R) - \hat{y}_i^R. \end{aligned} \quad (3.6)$$

Substituindo-se a Equação (3.5) dentro dessas expressões têm-se

$$\begin{aligned} \varepsilon_i^C &= y_i^C - (\mathbf{x}_i^C)^T \hat{\mathbf{w}}^C, \\ \varepsilon_i^R &= y_i^R - (\mathbf{x}_i^R)^T \hat{\mathbf{w}}^R. \end{aligned}$$

Considere as funções de custo

$$\begin{aligned} J(\hat{\mathbf{w}}^C) &= \frac{1}{2} \sum_{i=1}^N (\varepsilon_i^C)^2 \\ J(\hat{\mathbf{w}}^R) &= \frac{1}{2} \sum_{i=1}^N (\varepsilon_i^R)^2, \end{aligned} \quad (3.7)$$

as quais são funções contínuas e diferenciáveis com vetores de pesos (parâmetros) desconhecidos $\hat{\mathbf{w}}^C$ e $\hat{\mathbf{w}}^R$. A função

$$J(\hat{\mathbf{w}})$$

mapeia os elementos de $\hat{\mathbf{w}}$ dentro do espaço dos números reais. Deseja-se encontrar uma solução ótima $\hat{\mathbf{w}}^*$, i.e., $(\hat{\mathbf{w}}^C)^*$ e $(\hat{\mathbf{w}}^R)^*$, que satisfaçam a condição

$$J(\hat{\mathbf{w}}^*) \leq J(\hat{\mathbf{w}})$$

A solução pode ser obtida através do problema de otimização abaixo:

$$\begin{aligned} &\underset{\mathbf{w}^C}{\text{minimize}} J(\hat{\mathbf{w}}^C), \\ &\underset{\mathbf{w}^R}{\text{minimize}} J(\hat{\mathbf{w}}^R). \end{aligned} \tag{3.8}$$

Seguindo a abordagem de otimização do gradiente descendente, seguindo o mesmo princípio da definição encontrada em 2.1.2.2, diferenciando-se apenas pelo fato de a cada época de treinamento os dados de treinamento são permutados para que se reduza a probabilidade do algoritmo ficar preso em um mínimo local, portanto, é fácil mostrar que, em geral, os parâmetros são atualizados de acordo com o (HAYKIN, 2005; HAYKIN *et al.*, 2009):

$$\hat{\mathbf{w}}_0 = \mathbf{0}$$

$$\varepsilon_i = y_i - \mathbf{x}_i^T \hat{\mathbf{w}}_{i-1},$$

$$\hat{\mathbf{w}}_i = \hat{\mathbf{w}}_{i-1} + \alpha \varepsilon_i \mathbf{x}_i,$$

onde ε é calculado usando as expressões da Equação (3.6) e α é o passo de aprendizagem.

O Algoritmo 3, mostrado a seguir, resume de forma genérica, as operações acima.

Vale ressaltar que os primeiros passos do Algoritmo 3 são os mesmo da etapa de treinamento do *LOID*, portanto, não se faz necessário explicá-los novamente, porém, note que a quarta linha mostra que deve-se inicializar a primeira componente do $\hat{\mathbf{w}}$, o viés (*bias*), com o valor médio de $\mathbf{y}_{train} = \log(\mathbf{y})$, isso faz com que o algoritmo gradiente descendente possa convergir mais rápido uma vez que ele terá um ponto de partida, teoricamente, interessante dentro dos dados.

Algoritmo 3: Etapa de Treinamento do LLID

Require: $\mathbf{X}, \mathbf{y}, \text{epocas}, \alpha$ {conjunto de treinamento, número de épocas e passo de aprendizado}

- 1: $\mathbf{X} \leftarrow \frac{\mathbf{X}}{\max(\mathbf{X})}$ {aplica a normalização em \mathbf{X} }
- 2: $\mathbf{y} \leftarrow \frac{\mathbf{y}}{\max(\mathbf{y})}$ {aplica a normalização em \mathbf{y} }
- 3: $N, P \leftarrow \dim(\mathbf{X})$ {obtem a dimensão da matriz de entrada}
- 4: $\mathbf{X}_{train} \leftarrow [\mathbf{1}_N \mathbf{X}]_{N \times (P+1)}$
- 5: $\mathbf{y}_{train} \leftarrow \log \mathbf{y}$ {aplica a transformação logarítmica nas amostras de saída do conjunto de treinamento}
- 6: $\hat{\mathbf{w}}_{N+1}^T \leftarrow [\bar{\mathbf{y}}_{train} \mathbf{0}_{1 \times N}]$
- 7: **for** $e \in \{1, 2, \dots, \text{epocas}\}$ **do**
- 8: $indices \leftarrow perm(P)$
- 9: **for** $i \in indices$ **do**
- 10: $\hat{\mathbf{y}} \leftarrow (\mathbf{x}_{train})_i \hat{\mathbf{w}}^T$
- 11: $\varepsilon \leftarrow (\mathbf{y}_{train})_i - \hat{\mathbf{y}}$
- 12: $\hat{\mathbf{w}} \leftarrow \hat{\mathbf{w}} + \alpha \varepsilon (\mathbf{x}_{train})_i$
- 13: **end for**
- 14: **end for**

De forma similar ao algoritmo *LOID* detalhado anteriormente, após a etapa de treinamento deve-se executar a etapa de testes seguindo o Algoritmo 4, o qual deve considerar a aplicação da função exponencial no modelo de predições.

Algoritmo 4: Etapa de Testes do LLID

Require: \mathbf{X}, \mathbf{y} {conjunto de teste}

- 1: $N, P \leftarrow \dim(\mathbf{X})$ {obtem a dimensão da matriz/vetor de teste}
- 2: $\mathbf{X}_{train} \leftarrow [\mathbf{1}_N \mathbf{X}]_{N \times (P+1)}$ {adiciona o viés (*bias*)}
- 3: $\hat{\mathbf{y}}_{aux} \leftarrow \mathbf{X}_{test} \hat{\mathbf{w}}^T$ {calcula a saída transformada}
- 4: $\hat{\mathbf{y}} \leftarrow \exp(\hat{\mathbf{y}}_{aux})$ {calcula a saída real, isto é, sem transformação}

Seguindo-se o mesmo princípio e explicação encontrada para a etapa de testes do algoritmo *LOID* mostrado na subseção anterior, nesta mesma etapa do método *LLID* realizam-se as mesmas operações até se chegar aos valores reais preditos para a variável $\hat{\mathbf{y}}$ imediatamente após a aplicação da função exponencial sobre as saídas transformadas preditas.

Por fim, uma vez que os métodos propostos por este trabalho foram completamente detalhados e mostrados seus fundamentos, na próxima seção, apresentar-se-ão os experimentos e resultados obtidos a partir de dados reais e sintéticos, comparando-se suas performance entre si e em relação ao *CCRM* que é o algoritmo do estado da arte para regressões lineares com variáveis do tipo intervalo.

4 EXPERIMENTOS E RESULTADOS

A utilidade dos métodos de regressão linear propostos neste trabalho foram avaliadas através de experimentos com conjuntos de dados sintéticos com diferentes configurações lineares. Inicialmente, avaliou-se a performance dos algoritmos de otimização utilizados por *LOID*, *LLID* e *CCRM* para identificar qual algoritmo apresenta a melhor acurácia para a estimação dos parâmetros.

Após isso, a performance desses métodos foram comparados utilizando-se a métrica que é calculada através da raiz quadrada do erro quadrático médio (*RMSE*) do centro (*RMSE_C*), raio (*RMSE_R*) e a soma do *RMSE* do limites inferiores e superiores (*RMSE_{LU}*) para cada intervalo. Além disso, aplicou-se, também, o método de Monte Carlo (METROPOLIS; ULAM, 1949) com 50 repetições. As funções para o cálculo do *RMSE* foram computadas como:

$$\begin{aligned}
 RMSE_C &= \sqrt{\frac{\sum_{i=1}^N (y_i^C - \hat{y}_i^C)^2}{N}}, \\
 RMSE_R &= \sqrt{\frac{\sum_{i=1}^N (y_i^R - \hat{y}_i^R)^2}{N}}, \\
 RMSE_{LU} &= \sqrt{\frac{\sum_{i=1}^N (y_i^L - \hat{y}_i^L)^2}{N}} + \sqrt{\frac{\sum_{i=1}^N (y_i^U - \hat{y}_i^U)^2}{N}},
 \end{aligned} \tag{4.1}$$

onde N é o número de amostras na etapa de teste.

As métricas acima foram selecionadas com o objetivo de quantificar os erros em cada modelo, separadamente, um para prever \mathbf{y}^R e outro para prever \mathbf{y}^C . Então, através da análise dos valores do *RMSE_R* e *RMSE_C* pode-se calcular a precisão dos métodos avaliadas em cada cenário de teste. Por outro lado, tem-se o objetivo de se computar o erro de intervalo geral alcançado por cada técnica. Portanto, calculou-se o *RMSE_{LU}* que quantifica o erro através da soma dos *RMSE* dos limites do intervalo.

Finalmente, todos os modelos foram também avaliados usando-se conjunto de dados reais. Suas performance foram comparadas usando a técnica de validação cruzada Leave-One-Out (*LOO*) (KOHAVI *et al.*, 1995). O *LOO* foi escolhido devido ao tamanho dos conjunto de dados reais, em média, cada um deles têm 30 amostras.

4.0.1 Conjuntos de Dados Sintéticos

Os conjuntos de dados sintéticos foram gerados usando-se 2 estratégias. A primeira, considera as configurações apresentadas em (NETO; CARVALHO, 2010) seguindo as Tabelas 1 e 2. Além disso, para o desenvolvimento deste trabalho, propõe-se 2 conjuntos sintéticos adicionais. Em todos os casos, separou-se 67% dos dados gerados, para a etapa de treinamento e o restante, 33%, foram utilizados na etapa de teste para o cálculo das métricas de performances já descritas acima.

4.0.1.1 Descrição dos Conjuntos de Dados

4.0.1.1.1 Tipo - C

Conforme apresentado nas Tabelas 1 e 2, os autores em (NETO; CARVALHO, 2010) propuseram dois tipos de configurações diferentes, representados por C e D , para os conjuntos de dados lineares sintéticos. A configuração chamada de C tem os centros e raios independentes, enquanto os conjuntos do tipo D possuem uma dependência entre os pontos médios (centros) e os seus respectivos raios. Ambas configurações foram geradas com dimensão P onde $P = 1$ e $P = 3$.

A construção do conjunto de dados tipo C e seus intervalos correspondentes são obtidos de acordo com os seguintes passos:

1. Gere \mathbf{X}^C , \mathbf{X}^R , e \mathbf{y}^C de acordo com a Tabela 1;
2. Calcule a variável aleatória \mathbf{y}^C :

$$\mathbf{y}^C = (\mathbf{X}^C)^T \mathbf{w} + \boldsymbol{\varepsilon},$$

onde $(\mathbf{X}^C)^T = (1, X_1^C)$ e $w \sim U[c, d]$ em \mathcal{R}^2 , ou $(\mathbf{X}^C)^T = (1, X_1^C, X_2^C, X_3^C)$ and $w \sim U[c, d]$ em \mathcal{R}^4 and $\boldsymbol{\varepsilon} \sim U[e, f]$.

onde U é uma distribuição uniforme. Na Figura 5, mostra-se, como exemplo, um *plot* dos dados intervalares e apenas os centros dos dados, respectivamente, nos quais se percebe a linearidade dos dados.

4.0.1.1.2 Tipo - D

A geração dos conjuntos de dados do tipo D seguem como se descreve abaixo:

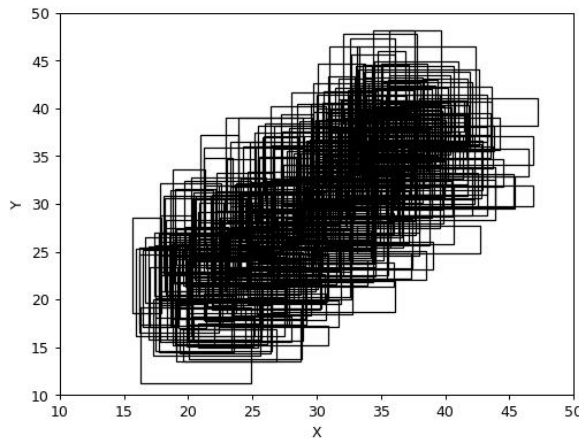
1. Gera \mathbf{X}^C de acordo com a Tabela 2;

Tabela 1 – Configuração dos Conjuntos de Dados - Tipo C

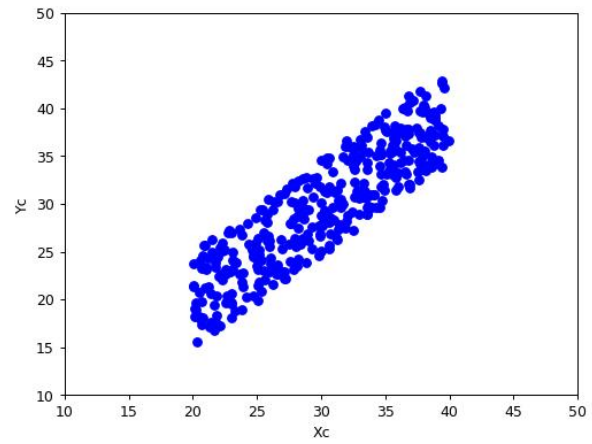
C_1	$X_{ij}^C \sim U[20, 40]$	$X_{ij}^R \sim U[20, 40]$	$Y_{ij}^R \sim U[20, 40]$	$\varepsilon \sim U[-20, 20]$
C_2	$X_{ij}^C \sim U[20, 40]$	$X_{ij}^R \sim U[20, 40]$	$Y_{ij}^R \sim U[20, 40]$	$\varepsilon \sim U[-5, 5]$
C_3	$X_{ij}^C \sim U[20, 40]$	$X_{ij}^R \sim U[1, 5]$	$Y_{ij}^R \sim U[1, 5]$	$\varepsilon \sim U[-20, 20]$
C_4	$X_{ij}^C \sim U[20, 40]$	$X_{ij}^R \sim U[1, 5]$	$Y_{ij}^R \sim U[1, 5]$	$\varepsilon \sim U[-5, 5]$

Figura 5 – Gráficos do Conjunto de Dados C_4^1

(a) Gráfico das variáveis intervalares. No eixo das abscissas têm-se os valores \mathbf{X} e no eixo das ordenadas o valores de \mathbf{y}



(b) Gráfico dos centros das variáveis intervalares



2. Calcule a variável aleatória \mathbf{y}^C : $\mathbf{y}^C = (\mathbf{X}^C)^T w + \varepsilon$, onde $(\mathbf{X}^C)^T = (1, X_1^C)$ e $w \sim U[c, d]$ em \mathcal{R}^2 , ou $(\mathbf{X}^C)^T = (1, X_1^C, X_2^C, X_3^C)$ e $w \sim U[c, d]$ em \mathcal{R}^4 e $\varepsilon \sim U[e, f]$;
3. Calcule a variável aleatória \mathbf{y}^R : $\mathbf{y}^R = \mathbf{y}^C w^* + \varepsilon^*$, onde as variáveis X_j^R ($j = 1, \dots, P$) são relacionadas às variáveis X_j^C de acordo com $X_j^R = X_j^C w^* + \varepsilon^*$, onde $w^* \sim U[g, h]$ e $\varepsilon^* \sim U[i, j]$.

Tabela 2 – Configuração dos Conjuntos de Dados - Tipo D

D_1	$X_{ij}^C \sim U[20, 40]$	$\varepsilon \sim U[-20, 20]$	$\varepsilon^* \sim U[1, 5]$
D_2	$X_{ij}^C \sim U[20, 40]$	$\varepsilon \sim U[-20, 20]$	$\varepsilon^* \sim U[10, 20]$
D_3	$X_{ij}^C \sim U[20, 40]$	$\varepsilon \sim U[-5, 5]$	$\varepsilon^* \sim U[1, 5]$
D_4	$X_{ij}^C \sim U[20, 40]$	$\varepsilon \sim U[-5, 5]$	$\varepsilon^* \sim U[10, 20]$

4.0.1.1.3 Tipo - A

O método *CCRM* restringe os parâmetros preditos $\hat{\mathbf{w}}^R$ serem maiores ou iguais a 0. Neste trabalho, gerou-se 2 conjuntos de dados, 1-dimensional (A^1) e um 3-dimensional (A^3), onde

os verdadeiros parâmetros \mathbf{w}_R são menores que 0. Abaixo, detalhou-se as funções geradoras e os passos necessários para se gerar tais conjuntos. Portanto, inicialmente, descreve-se A^1 seguindo os passos abaixo:

1. Calcule $\mathbf{X}^C = (\mathbf{1}, U[20, 40])$, $\mathbf{X}^R = (\mathbf{1}, U[0, 1])$;
2. Calcule $\mathbf{y}^C = \mathbf{X}^C * (2, 5)^T$ and $\mathbf{y}^R = \mathbf{X}^R(5, -3)^T$;
3. Calcule a variância de $S_y^C = \mathbf{y}^C$ and $S_y^R = \mathbf{y}^R$;
4. Re-calcule $\mathbf{y}^C = \mathbf{y}^C + \sqrt{0.01S_y^C} \boldsymbol{\varepsilon}$ e $\mathbf{y}^R = \mathbf{y}^R + \sqrt{0.01S_y^R} \boldsymbol{\varepsilon}$, where $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$;
5. Para valores $y_i^R < 0$, faça $y_i^R = 0$,

a Fig. 6 mostra o gráfico desse conjunto de dados no plano, mostrando a linearidade dos dados.

Nesse sentido, o conjunto de dados A^3 foi gerado para se haver variância maior entre os valores de \mathbf{X}^R , fazendo com que o *CCRM* obtenha um *RMSE* ainda maior se comparado aos métodos *LOID* e *LLID*. Assim, seguem os passos para se gerar o conjunto A^3 :

1. Calcule $\mathbf{X}^C = (\mathbf{1}, U[20, 40])$, $\mathbf{X}^R = (\mathbf{1}, U[1, 10])$;
2. Calcule $\mathbf{y}^C = \mathbf{X}^C(2, 2, 5, 6)^T$ and $\mathbf{y}^R = \mathbf{X}^R(62, -2, -3, -1)^T$;
3. Calcule a variância de $S_y^C = \mathbf{y}^C$ and $S_y^R = \mathbf{y}^R$;
4. Recompute $\mathbf{y}^C = \mathbf{y}^C + \sqrt{0.01S_y^C} \boldsymbol{\varepsilon}$ and $\mathbf{y}^R = \mathbf{y}^R + \sqrt{0.01S_y^R} \boldsymbol{\varepsilon}$, onde $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$;
5. Para valores $y_i^R < 0$, substitua-os com $y_i^R = 0$.

Figura 6 – Gráficos do Conjunto de Dados A_1^1

(a) Gráfico das variáveis intervalares. No eixo das abscissas

têm-se os valores \mathbf{X} e no eixo das ordenadas o valores de \mathbf{y} (b) Gráfico dos centros das variáveis intervalares

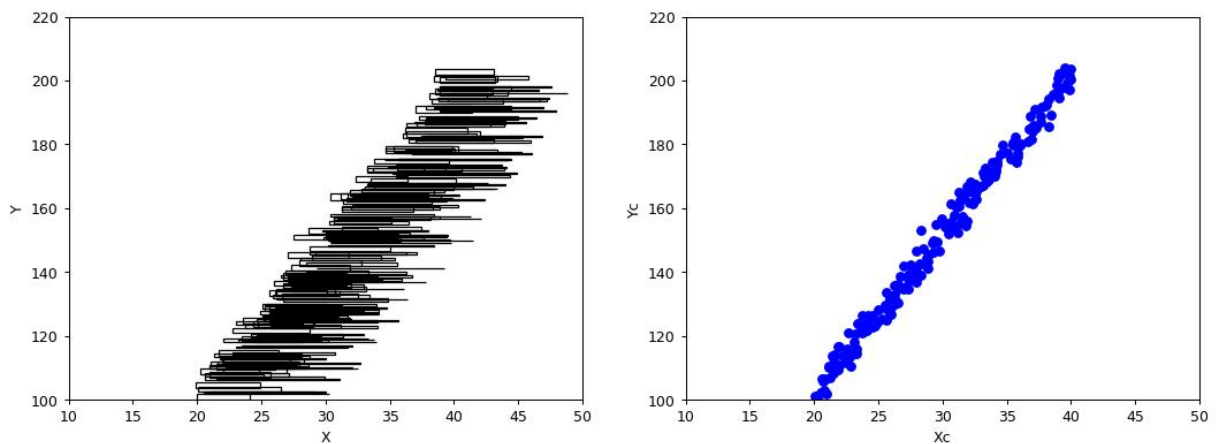


Tabela 3 – Comparação entre **LOID**, **LLID** and **CCRM** nos Conjuntos de Dados Sintéticos 1-Dimensional Usando o *RMSE* como Métrica para o Cálculo do Erro.

		A_1^1	C_1^1	C_2^1	C_3^1	C_4^1	D_1^1	D_2^1	D_3^1	D_4^1
LOID	<i>RMSE_C</i>	$3.11 \pm 1.87e-1$	$1.11e1 \pm 4.15e-1$	$2.88 \pm 9.51e-2$	$1.19e1 \pm 3.77e-1$	$2.86 \pm 7.68e-2$	$1.21e1 \pm 4.01e-1$	$1.15e1 \pm 3.85e-1$	$2.98 \pm 1.00e-1$	$2.85 \pm 9.89e-2$
	<i>RMSE_R</i>	$1.23e-1 \pm 9.03e-3$	$5.78 \pm 2.20e-1$	$5.91 \pm 2.12e-1$	$1.23 \pm 4.21e-2$	$1.15 \pm 4.77e-2$	$5.03e1 \pm 2.12$	$4.77e1 \pm 1.84$	$1.30e1 \pm 6.19e-1$	$1.24e1 \pm 5.76e-1$
	<i>RMSE_{LU}</i>	$6.23 \pm 3.75e-1$	$2.50e1 \pm 7.69e-1$	$1.31e1 \pm 4.01e-1$	$2.39e1 \pm 7.5e-1$	$6.18 \pm 1.42e-1$	$1.00e2 \pm 4.22$	$9.56e1 \pm 3.67$	$2.61e1 \pm 1.24$	$2.50e1 \pm 1.15$
LLID	<i>RMSE_C</i>	$3.11 \pm 1.87e-1$	$1.11e1 \pm 4.16e-1$	$2.88 \pm 9.51e-2$	$1.19e1 \pm 3.77e-1$	$2.86 \pm 7.67e-2$	$1.21e1 \pm 4.01e-1$	$1.15e1 \pm 3.86e-1$	$2.98 \pm 1.00e-1$	$2.85 \pm 9.90e-2$
	<i>RMSE_R</i>	$1.23e-1 \pm 9.03e-3$	$5.78 \pm 2.20e-1$	$5.91 \pm 2.12e-1$	$1.23 \pm 4.22e-2$	$1.15 \pm 4.76e-2$	$5.03e1 \pm 2.12$	$4.77e1 \pm 1.86$	$1.30e1 \pm 6.18e-1$	$1.24e1 \pm 5.75e-1$
	<i>RMSE_{LU}</i>	$6.23 \pm 3.75e-1$	$2.50e1 \pm 7.70e-1$	$1.31e1 \pm 4.01e-1$	$2.39e1 \pm 7.5e-1$	$6.18 \pm 1.42e-1$	$1.00e2 \pm 4.22$	$9.56e1 \pm 3.69$	$2.61e1 \pm 1.24$	$2.50e1 \pm 1.15$
CCRM	<i>RMSE_C</i>	$3.11 \pm 1.87e-1$	$1.11e1 \pm 4.15e-1$	$2.88 \pm 9.51e-2$	$1.19e1 \pm 3.77e-1$	$2.86 \pm 7.68e-2$	$1.21e1 \pm 4.01e-1$	$1.15e1 \pm 3.85e-1$	$2.98 \pm 1.00e-1$	$2.85 \pm 9.89e-2$
	<i>RMSE_R</i>	$8.57e-1 \pm 4.01e-2$	$5.75 \pm 2.06e-1$	$5.88 \pm 2.17e-1$	$1.20 \pm 3.95e-2$	$1.12 \pm 3.89e-2$	$4.94e1 \pm 1.64$	$4.62e1 \pm 1.65$	$1.20e1 \pm 4.47e-1$	$1.17e1 \pm 4.68e-1$
	<i>RMSE_{LU}</i>	$6.46 \pm 3.63e-1$	$2.50e1 \pm 7.67e-1$	$1.31e1 \pm 4.11e-1$	$2.39e1 \pm 7.5e-1$	$6.16 \pm 1.41e-1$	$9.89e1 \pm 3.28$	$9.25e1 \pm 3.31$	$2.41e1 \pm 8.93e-1$	$2.34e1 \pm 9.37e-1$

4.0.1.2 Resultados

Os resultados obtidos nos testes utilizando-se os conjuntos de dados sintéticos são apresentados nas Tabelas 3 e 4, onde os conjuntos foram separados pela suas dimensões, 1D e 3D, respectivamente. Como pode-se notar, na primeira coluna da Tabela 3, mostram-se os resultados do conjunto de dados proposto nesse trabalho, A_1^1 , que possui os pesos da função que multiplicam os X^R gerados menores que 0, percebe-se que, como esperado, o *CCRM* não obteve um bom resultado, apesar de parecer próximo, devido às restrições impostas por esse algoritmo sobre a estimação dos valores de \hat{w}^R .

Além disso, observa-se, também, que nos demais conjuntos de dados as variantes do *LMS* e *OLS* propostas obtiveram bons resultados, com ênfase para o *dataset* C_2^1 onde os métodos *LOID* e *LLID* foram, inclusive, melhores que o *CCRM* obtendo uma variância sensivelmente menor o que faz com que eles obtenham resultados mais precisos. Mesmo nos demais conjuntos de dados sintéticos propostos pelos autores do *CCRM*, as variantes propostas nesse trabalho, conseguiram resultados bastante próximos considerando as três métricas utilizadas para o cálculo do erro durante a etapa de testes.

Seguindo com a discussão e apresentação dos resultados dos conjuntos sintéticos, na Tabela 4, têm-se os resultados conseguidos através dos conjuntos sintéticos de 3 dimensões. Novamente, observa-se que a primeira coluna faz referência ao conjunto proposto neste trabalho, A_1^3 , o qual possui em, sua função geradora, uma relação linear cujos pesos que multiplicam X^R são menores que 0, onde, mais um vez, os métodos propostos, *LOID* e *LLID* obtiveram resultados melhores em relação ao *CCRM*, ficando-se evidenciado que, com mais dimensões, e uma maior magnitude dos valores gerados, o *CCRM* não consegue prever os intervalos dada a

Tabela 4 – Comparação entre **LOID**, **LLID** and **CCRM** nos Conjuntos de Dados Sintéticos 3-Dimensional Usando o *RMSE* como Métrica para o Cálculo do Erro.

	A_1^3	C_1^3	C_2^3	C_3^3	C_4^3	D_1^3	D_2^3	D_3^3	D_4^3	
LOID	<i>RMSE_C</i>	$5.01 \pm 3.62e-1$	$1.15e1 \pm 4.18e-1$	$2.88 \pm 1.05e-1$	$1.19e1 \pm 4.05e-1$	$2.99 \pm 8.86e-2$	$1.18e1 \pm 3.75e-1$	$1.20e1 \pm 3.95e-1$	$2.90 \pm 9.70e-2$	$2.90 \pm 9.84e-2$
	<i>RMSE_R</i>	$2.51 \pm 5.81e-1$	$6.15 \pm 1.79e-1$	$5.82 \pm 2.08e-1$	$1.17 \pm 4.25e-2$	$1.18 \pm 5.18e-2$	$4.90e1 \pm 1.78$	$4.92e1 \pm 1.91$	$1.24e1 \pm 4.88e-1$	$1.48e1 \pm 7.33e-1$
	<i>RMSE_{LU}</i>	$1.26e1 \pm 8.46e-1$	$2.61e1 \pm 7.32e-1$	$1.29e1 \pm 3.90e-1$	$2.39e1 \pm 8.05e-1$	$6.43 \pm 1.69e-1$	$9.82e1 \pm 3.55$	$9.86e1 \pm 3.81$	$2.48e1 \pm 9.77e-1$	$2.98e1 \pm 1.45$
LLID	<i>RMSE_C</i>	$5.01 \pm 3.62e-1$	$1.15e1 \pm 4.18e-1$	$2.88 \pm 1.05e-1$	$1.19e1 \pm 4.04e-1$	$2.99 \pm 8.86e-2$	$1.18e1 \pm 3.75e-1$	$1.20e1 \pm 3.95e-1$	$2.90, \pm 9.70e-2$	$2.90 \pm 9.89e-2$
	<i>RMSE_R</i>	$2.51 \pm 5.81e-1$	$6.15 \pm 1.81e-1$	$5.82 \pm 2.10e-1$	$1.17 \pm 4.25e-2$	$1.18 \pm 5.18e-2$	$4.90e1 \pm 1.78$	$4.92e1 \pm 1.91$	$1.24e1 \pm 4.88e-1$	$1.48e1 \pm 7.34e-1$
	<i>RMSE_{LU}</i>	$1.26e1, \pm 8.46e-1$	$2.61e1 \pm 7.32e-1$	$1.29e1 \pm 3.92e-1$	$2.39e1 \pm 8.05e-1$	$6.43 \pm 1.69e-1$	$9.82e1 \pm 3.55$	$9.87e1 \pm 3.80$	$2.48e1 \pm 9.77e-1$	$2.98e1 \pm 1.45$
CCRM	<i>RMSE_C</i>	$5.01 \pm 3.62e-1$	$1.15e1 \pm 4.18e-1$	$2.88 \pm 1.05e-1$	$1.19e1 \pm 4.05e-1$	$2.99 \pm 8.86e-2$	$1.18e1 \pm 3.75e-1$	$1.20e1 \pm 3.95e-1$	$2.90, \pm 9.70e-2$	$2.90 \pm 9.84e-2$
	<i>RMSE_R</i>	$9.21 \pm 5.91e-1$	$6.10 \pm 1.64e-1$	$5.76 \pm 2.06e-1$	$1.15 \pm 3.79e-2$	$1.16 \pm 4.34e-2$	$4.79e1 \pm 1.58$	$4.76e1 \pm 1.59$	$1.17e1 \pm 4.30e-1$	$1.32e1 \pm 6.40e-1$
	<i>RMSE_{LU}</i>	$2.09e1 \pm 1.03$	$2.61e1 \pm 7.39e-1$	$1.28e1 \pm 3.94e-1$	$2.39e1 \pm 8.04e-1$	$6.41 \pm 1.64e-1$	$9.60e1 \pm 3.18$	$9.54e1 \pm 3.18$	$2.36e1 \pm 8.60e-1$	$2.66e1 \pm 1.26$

sua fundamentação teórica que restringe os pesos estimados $\hat{\mathbf{w}}^R$.

Nos demais *datasets*, o *CCRM* obteve os melhores resultados, porém, detalhando-os, percebe-se que as variantes propostas ficaram praticamente com os mesmo resultados nos conjuntos C_1^3 , C_2^3 , C_3^3 e C_4^3 onde o *LOID* e *LLID* ficaram com resultados piores apenas na métrica *RMSE_R* o que refletiu, minimamente, nos resultado do *RMSE_{LU}*. Já nos conjunto do tipo *D*, que possuem uma função geradora onde há uma relação entre os centros e os raios, o *CCRM* conseguiu resultados melhores e mais distantes se comparados aos métodos propostos *LOID* e *LLID*.

4.0.2 Conjuntos de Dados Reais

Neste trabalho, foi utilizado 6 conjuntos de dados reais que podem ser encontrados em (FAGUNDES, 2013) e serão, brevemente, descritos na próxima sub-seção.

4.0.2.1 Descrição dos Conjuntos

Uma breve descrição de cada conjunto de dados é apresentada a seguir, além disso, na Tabela 5 pode-se observar a quantidade de amostras e número de *features* de cada um deles:

- **Cardiologia:** O conjunto de dados Cardiologia possui 2 variáveis independentes, Pressão Arterial Sistólica e Pressão Arterial Diastólica, as quais são as entradas do sistema e são usadas para predizer o valor intervalar da taxa de pulsação sanguínea.
- **Carro:** No conjuto Carro considerou-se apenas 2 variáveis de entrada, velocidade máxima e as cilindradas do motor, para prever o intervalo do preço.

- **Carro 7 features:** Nesse conjunto Carro 7 features considerou-se as demais 5 variáveis que não foram utilizadas anteriormente: aceleração, passo, comprimento, largura e altura.
- **Cogumelo:** O conjunto Cogumelo é descrito por 3 variáveis do tipo intervalo, onde as entradas são, tamanho do estipe e espessura do estipe, já a saída é a largura da capa do píleo.
- **Futebol:** O conjunto Futebol possui informações de jogadores profissionais, onde cada jogador é descrito por duas variáveis independentes, altura e idade, e uma variável dependente: peso.
- **NASA:** Um MDP (*Metrics Data Program*) é uma organização que coleta as métricas de 13 projetos de software e gera um repositório para uma dessas informações. Cada projeto de software deve ser associado com 33 métricas de software. O processo de generalização transforma o conjunto de um escalar para uma variável do tipo intervalo através da execução da biblioteca ISDA que foi desenvolvida em (FAGUNDES, 2013). Finalmente, as duas variáveis independentes são: o número de operandos e o número de operadores, os quais são usados para predizer o intervalo do número de linhas.

Tabela 5 – Descrição dos Conjuntos de Dados Reais

	Número de Amostras	Número de Características
Cardiologia	59	2
Carro	33	2
Carro 7 Feature	33	7
Cogumelo	23	2
Futebol	20	2
Nasa	13	2

4.0.2.2 Resultados

Os resultados obtidos através dos testes realizados com os conjuntos de dados reais estão resumidos na Tabela 6. Em geral, exceto pelo conjunto da Nasa, onde o *LOID* obteve um resultado pior em comparação com os demais métodos, para todos os outros, os métodos propostos foram muito próximos ou melhores que o *CCRM*.

Observando na Tabela 6, pode-se observar que no *dataset* Cardiologia, que o *LLOID* foi sensivelmente melhor ao predizer os valores dos centros dos intervalos, isso refletiu no $RMSE_{LU}$ que calcula o erro total da variável intervalar predita, onde o método proposto conseguiu um resultado melhor em relação a todos os outros.

Tabela 6 – Comparação entre **LOID**, **LLID** and **CCRM** nos Conjuntos de Dados Reais usando o *RMSE* como Métrica para o Cálculo do Erro.

		Cardiologia	Carro	Carro 7 Features	Cogumelo	Futebol	Nasa
LOID	<i>RMSE_C</i>	8.96 ± 6.92	2.32e4 ± 1.60e4	2.04e4 ± 2.03e4	1.62 ± 1.27	1.84 ± 1.49	4.92e2 ± 4.52e2
	<i>RMSE_R</i>	5.68 ± 4.90	1.51e4 ± 2.38e4	2.44e4 ± 7.65e4	1.05 ± 7.68e-1	1.10 ± 7.74e-1	5.44e2 ± 7.43e2
	<i>RMSE_{LU}</i>	2.12e1 ± 1.31e1	5.89e4 ± 4.45e4	7.07e4 ± 1.51e4	3.83 ± 2.30	3.95 ± 2.86	1.26e3 ± 1.40e3
LLID	<i>RMSE_C</i>	8.86 ± 6.80	2.20e4 ± 1.80e4	2.20e4 ± 1.81e4	1.65 ± 1.07	1.89 ± 1.59	3.79e2 ± 3.58e2
	<i>RMSE_R</i>	5.70 ± 4.90	1.48e4 ± 2.34e4	1.48e4 ± 2.35e4	1.06 ± 8.39e-1	1.22 ± 9.43e-1	3.56e2 ± 4.79e2
	<i>RMSE_{LU}</i>	2.10e1 ± 1.3e1	5.69e4 ± 4.68e4	5.68e4 ± 4.68e4	3.78 ± 2.06	4.29 ± 3.01	9.14e2 ± 8.73e2
CCRM	<i>RMSE_C</i>	8.96 ± 6.92	2.32e4 ± 1.60e4	2.04e4 ± 2.03e4	1.62 ± 1.27	1.84 ± 1.49	4.92e2 ± 4.53e2
	<i>RMSE_R</i>	5.69 ± 4.68	1.98e4 ± 2.17e4	1.56e4 ± 1.79e4	9.27e-1 ± 6.79e-1	1.07 ± 8.19e-1	3.97e2 ± 3.64e2
	<i>RMSE_{LU}</i>	2.12e1 ± 1.29e1	6.14e4 ± 4.21e4	5.02e4 ± 4.34e4	3.52 ± 2.40	3.93 ± 2.87	9.84e2 ± 9.06e2

O mesmo aconteceu no conjunto de dados Carro que possui é 2-dimensional, neste, porém, o *CCRM* acabou tendo um resultado pior que os métodos propostos *LOID* e *LLID*, o que não aconteceu no conjunto Carro 7 features que possui mais características, nessa etapa de testes, o *CCRM* obteve os melhores resultados.

Nos conjuntos Cogumelo e Futebol, novamente, o *CCRM* obteve bons resultados em todas as métricas de *RMSE*, porém, no último conjunto de dados, da Nasa, o *LLID* foi o melhor entre todos, obtendo um *RMSE_{LU}* bem melhor em relação ao *CCRM*.

5 CONCLUSÕES E TRABALHOS FUTUROS

Considerando-se a crescente necessidade de se implementar algoritmos de Aprendizado de Máquina para tarefas que envolvem dados intervalares, este trabalho, propõe dois modelos de regressão linear para dados do tipo intervalo. Os métodos *LOID* e *LLID*, compreendem-se em uma etapa de normalização dos dados de entrada e saída seguidos por uma transformação logarítmica dos dados de saída. Depois disso, os métodos *OLS* ou *LMS* podem ser usados para estimar os coeficientes do modelo de regressão linear.

A proposta tem duas vantagens sobre o suas contrapartes mais populares: i) ambos, *LOID* e *LLID*, não necessitam de qualquer restrição na etapa de otimização; ii) *LLID* é o primeiro método de regressão linear iterativo para dados intervalares. Esse último fato faz do *LLID* adequado para aplicações que envolvem *big data*.

Com base em um extenso conjunto de experimentos, pode-se verificar que o *LOID* e o *LLID* superaram o CCRM em algumas situações, sobretudo, quando há uma linearidade negativa (coeficientes da função geradora negativos) além de terem obtido resultados bastante próximos nas demais situações. Portanto, acredita-se que ambas as propostas podem ser vistas como uma alternativa válida para enfrentar problemas de regressão linear com dados do tipo intervalo.

Portanto, propõe-se como trabalhos futuros, o estudo e o desenvolvimento de técnicas para se conseguir realizar tarefas de regressão linear utilizando os dados intervalares por completo, isto é, evitando-se a necessidade de modelar duas regressões que modelam os centros e os raios, respectivamente. Entretanto, existe um grande desafio para se elaborar um modelo de regressão com essas características, devido a necessidade das funções de custos de serem diferenciáveis em relação aos pesos, que serão intervalos, nesse caso, além da necessidade de se inverter matrizes com valores do tipo intervalos, assim, este trabalho deixa como estudos posteriores a elaboração dessas técnicas.

REFERÊNCIAS

- BERTSEKAS, D. P.; TSITSIKLIS, J. N. Gradient convergence in gradient methods with errors. **SIAM Journal on Optimization**, SIAM, v. 10, n. 3, p. 627–642, 2000.
- BILLARD, L.; DIDAY, E. Regression analysis for interval-valued data. In: **Data Analysis, Classification, and Related Methods**. [S. l.]: Springer, 2000. p. 369–374.
- BOCK, H.-H.; DIDAY, E. **Analysis of symbolic data: exploratory methods for extracting statistical information from complex data**. [S. l.]: Springer Science & Business Media, 2012.
- FAGUNDES, R. A. A. **Métodos de regressão robusta e kernel para dados intervalares**. Tese (Doutorado) – Universidade Federal de Pernambuco, 2013.
- GIORDANI, P. Regression analysis for interval-valued data based on the lasso technique. **Technical Report n. 7, Department of Statistical Sciences, Sapienza University of Rome**, 2011.
- HAYKIN, S. S. **Adaptive filter theory**. [S. l.]: Pearson Education India, 2005.
- HAYKIN, S. S. *et al.* **Neural networks and learning machines/Simon Haykin**. [S. l.]: New York: Prentice Hall,, 2009.
- JUDGE, G. G.; TAKAYAMA, T. Inequality restrictions in regression analysis. **Journal of the American Statistical Association**, Taylor & Francis, v. 61, n. 313, p. 166–181, 1966.
- KEARFOTT, R. B. Interval computations: Introduction, uses, and resources. **Euromath Bulletin**, Citeseer, v. 2, n. 1, p. 95–112, 1996.
- KLEINBAUM, D. G.; KUPPER, L. L.; MULLER, K. E.; NIZAM, A. **Applied regression analysis and other multivariable methods**. [S. l.]: Duxbury Press Belmont, CA, 1988. v. 601.
- KOHAVI, R. *et al.* A study of cross-validation and bootstrap for accuracy estimation and model selection. In: MONTREAL, CANADA. **Ijcai**. [S. l.], 1995. v. 14, n. 2, p. 1137–1145.
- LAWSON, C. L.; HANSON, R. J. **Solving least squares problems**. [S. l.]: Siam, 1995. v. 15.
- METROPOLIS, N.; ULAM, S. The monte carlo method. **Journal of the American statistical association**, Taylor & Francis, v. 44, n. 247, p. 335–341, 1949.
- MOORE, R. E.; KEARFOTT, R. B.; CLOUD, M. J. **Introduction to interval analysis**. [S. l.]: Siam, 2009. v. 110.
- NETO, E. d. A. L.; CARVALHO, F. D. A. D. Centre and range method for fitting a linear regression model to symbolic interval data. **Computational Statistics & Data Analysis**, v. 52, n. 3, p. 1500–1515, 2008. ISSN 0167-9473. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0167947307001934>.
- NETO, E. d. A. L.; CARVALHO, F. D. A. D. Constrained linear regression models for symbolic interval-valued variables. **Computational Statistics & Data Analysis**, Elsevier, v. 54, n. 2, p. 333–347, 2010.
- NETO, E. d. A. L.; CARVALHO, F. D. A. D. Nonlinear regression applied to interval-valued data. **Pattern Analysis and Applications**, v. 20, n. 3, p. 809–824, Aug 2017. ISSN 1433-755X. Disponível em: <https://doi.org/10.1007/s10044-016-0538-y>.

POLLOCK, D. S. G.; POLLOCK, D. **The algebra of econometrics**. [*S. l.*], 1979.

WELLS, D. E.; VANÍČEK, P.; PAGIATAKIS, S. D. Least squares spectral analysis revisited. Department of Surveying Engineering, University of New Brunswick Fredericton, 1985.

WIDROW, B.; HOFF, M. E. *et al.* **Adaptive switching circuits**. [*S. l.*], 1960. v. 4, n. 1, 96–104 p.

YOUNG, G.; HOUSEHOLDER, A. S. Discussion of a set of points in terms of their mutual distances. **Psychometrika**, Springer, v. 3, n. 1, p. 19–22, 1938.