



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE TRANSPORTES E
GESTÃO TERRITORIAL

Leonardo Sales Duarte

**Estudo comparativo entre modelos clássicos de regressão e de redes neurais artificiais
em avaliação imobiliária na área central de Florianópolis**

Florianópolis

2023

Leonardo Sales Duarte

**Estudo comparativo entre modelos clássicos de regressão e de redes neurais artificiais
em avaliação imobiliária na área central de Florianópolis**

Dissertação submetida ao Programa de Pós-Graduação em Engenharia de Transportes e Gestão Territorial da Universidade Federal de Santa Catarina como requisito para a obtenção do título de Mestre em Engenharia de Transportes e Gestão Territorial.

Orientadora: Prof^ª. Vivian da Silva Celestino Reginato, Dr^ª
Coorientador: Prof. Norberto Hochheim, Dr.

Florianópolis

2023

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Duarte, Leonardo Sales

Estudo comparativo entre modelos clássicos de regressão e de redes neurais artificiais em avaliação imobiliária na área central de Florianópolis / Leonardo Sales Duarte ; orientadora, Vivian da Silva Celestino Reginato, coorientador, Norberto Hochheim, 2023.

181 p.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós-Graduação em Engenharia de Transportes e Gestão Territorial, Florianópolis, 2023.

Inclui referências.

1. Engenharia de Transportes e Gestão Territorial. 2. Rede Neural Artificial. 3. Regressão Linear Clássica. 4. Engenharia de Avaliações. I. Reginato, Vivian da Silva Celestino. II. Hochheim, Norberto. III. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia de Transportes e Gestão Territorial. IV. Título.

Leonardo Sales Duarte

**Estudo comparativo entre modelos clássicos de regressão e de redes neurais artificiais
em avaliação imobiliária na área central de Florianópolis**

O presente trabalho em nível de mestrado foi avaliado e aprovado no dia 07/07/2023 por
banca examinadora composta pelos seguintes membros:

Prof. Alexandre Hering Coelho, Dr.
Universidade Federal de Santa Catarina (UFSC)

Prof. Marco Aurélio Stumpf González, Dr.
Universidade do Vale do Rio dos Sinos (UNISINOS)

Prof. Rubens Alves Dantas, Dr.
Universidade Federal de Pernambuco (UFPE)

Certificamos que esta é a versão original e final do trabalho que foi julgado adequado para
obtenção do título de Mestre em Engenharia de Transportes e Gestão Territorial.

Prof^ª. Ana Maria Benciveni Franzoni, Dr^a
Coordenação do Programa de Pós-Graduação

Prof^ª. Vivian da Silva Celestino Reginato, Dr^a
Orientadora

Florianópolis, 2023

AGRADECIMENTOS

Em primeiro lugar, expresso minha profunda gratidão a Deus pela vida e pelo Seu constante cuidado. Aos meus amados pais, Ercílio e Ozileide, agradeço por todo o carinho, investimento, abnegação e dedicação em cada etapa da minha jornada de aprendizado. Ao meu irmão Lauro, por seu incentivo na escolha do curso ainda no começo de minha caminhada acadêmica. Ao meu amor e melhor amiga, Thainá, agradeço o apoio e compreensão demonstrados ao longo do mestrado.

Agradeço também à Prof.^a Vivian e ao Prof. Norberto por terem aceitado ajudar na condução deste trabalho, compartilhando seus valiosos conhecimentos, orientações e proporcionando apoio em publicações científicas. Ao Professor Alexandre, expresso minha gratidão por suas orientações e instruções na disciplina de programação. Aos demais membros da banca, Prof. González e Prof. Dantas, por terem aceitado o convite e por suas contribuições.

Também sou grato à CAPES pela bolsa concedida durante parte do mestrado.

"Deus não introduziu o homem na escola do mundo para a apreciação da sua multiforme sabedoria? De qualquer maneira, nisto se reconhece o propósito tanto do mundo criado como dos homens nele colocados."

Jan Amos Comenius

RESUMO

Rede Neural Artificial (RNA) é um dos métodos citados pela NBR 14653-2 (ABNT, 2011) para realizar avaliação de imóveis. Esse método pode ser usado para inferir as relações entre as variáveis independentes e a variável dependente do imóvel a partir de uma amostra de treinamento. A tendência é que as RNA ganhem gradativamente mais espaço em avaliações imobiliárias devido sua capacidade de lidar com problemas não lineares, ao contrário da Regressão Linear Clássica (RLC). Apesar disso, a RNA ainda tem sido considerada um método “caixa-preta” e a própria norma NBR 14653-2 (ABNT, 2011) apresenta o método em caráter informativo, sem maiores detalhes, o que dificulta a sua disseminação e utilização entre os avaliadores. Diante do cenário exposto, é possível compreender a pertinência de um estudo que contribua para o avanço da compreensão de RNA e sua aplicação em avaliações imobiliárias. Assim, o objetivo deste trabalho é realizar uma análise comparativa entre o modelo RLC e modelos de RNA em avaliações imobiliárias na área central do município de Florianópolis. Especificamente, os objetivos são: analisar o desempenho do melhor modelo de RLC obtido com auxílio da abordagem de força-bruta e verificar seus pressupostos; simular modelos de RNA com e sem transformações e identificar aqueles com melhor capacidade preditiva ao variar diferentes parâmetros; comparar modelos de RNA com RLC. Para isso foi realizada uma coleta de dados (apartamentos) que resultou em uma amostra total de 231 elementos com 8 variáveis independentes. A RLC de referência possui R^2 de 96,39% e foram verificados a linearidade, normalidade, homocedasticidade, ausência de autocorrelação, de multicolinearidade, de pontos influenciadores e de *outliers*. Ao todo, foram realizadas 5.364 simulações com RNA, sendo 1.800 em cenários com aplicação de transformações e 3.564 sem transformações, nas quais foram testados diferentes parâmetros. Diante dos resultados foi possível perceber que não existem valores fixos para os parâmetros de uma RNA que devam ser sempre empregados, pois cada problema é único e deve ser tratado segundo as suas peculiaridades. O melhor caminho é realizar o ajuste fino dos parâmetros até encontrar modelos com boa capacidade preditiva. Por fim, ao comparar modelos de RNA com RLC, foi possível observar que todos os modelos de RNA utilizados para predição dos valores da amostra de avaliação apresentaram resultados superiores à RLC, ou seja, melhor capacidade preditiva, por exemplo, a RNA-1 apresentou RMSE, MAE e MAPE, respectivamente, iguais a 329.338,34, 212.364,91 e 14,00%; a RNA-11 forneceu 333.445,70, 214.502,72 e 13,80%; e o modelo de RLC obteve 408.853,13, 240.121,12 e 14,70%. Com relação à análise de elasticidade das variáveis, o modelo de RLC e o modelo de RNA-1 apresentaram um melhor comportamento, já o modelo RNA-11 apresentou um comportamento indesejado na variável “suítes”, que poderia ser descartada. Isso suscita uma reflexão: apesar de métricas superiores, redes neurais podem, na verdade, apresentar comportamento indesejado em alguma variável. Isso mostra que análise de elasticidade deve ser sempre considerada na validação de modelos de RNA.

Palavras-chave: Rede Neural Artificial; Regressão Linear Clássica; Engenharia de avaliações

ABSTRACT

Artificial Neural Network (ANN) is one of the methods cited by NBR 14653-2 (ABNT, 2011) to evaluate properties. This method can be used to infer the relationships between the independent variables and the property's dependent variable from a training sample. The tendency is for ANN to gradually gain more space in real estate valuations due to its ability to deal with non-linear problems, unlike Classical Linear Regression (RLC). Despite this, ANN has still been considered a “black box” method and the NBR 14653-2 (ABNT, 2011) presents the method in an informative capacity, without further details, which makes its dissemination and use among professionals difficult. Given the above scenario, it is possible to understand the relevance of a study that contributes to advancing the understanding of ANN and its application in real estate valuations. Therefore, the objective of this work is to carry out a comparative analysis between the RLC model and ANN models in real estate valuations in the central area of the city of Florianópolis. Specifically, the objectives are: to analyze the performance of the best RLC model obtained with the help of the brute-force approach and verify its assumptions; simulate ANN models with and without transformations and identify those with better predictive capacity when varying different parameters; compare ANN models with RLC. To this end, data collection was carried out (apartments) which resulted in a total sample of 231 elements with 8 independent variables. The reference RLC has an R^2 of 96.39% and linearity, normality, homoscedasticity, absence of autocorrelation, multicollinearity, influencing points and outliers were verified. In total, 5,364 simulations were carried out with ANN, 1,800 in scenarios with the application of transformations and 3,564 without transformations, in which different parameters were tested. Given the results, it was possible to realize that there are no fixed values for the parameters of an ANN that must always be used, as each problem is unique and must be treated according to its peculiarities. The best way is to fine-tune the parameters until you find models with good predictive capacity. Finally, when comparing ANN models with RLC, it was possible to observe that all ANN models used to predict the values of the evaluation sample presented results superior to RLC, that is, better predictive capacity, for example, ANN-1 presented RMSE, MAE and MAPE, respectively, equal to 329,338.34, 212,364.91 and 14.00%; RNA-11 provided 333,445.70, 214,502.72 and 13.80%; and the RLC model obtained 408,853.13, 240,121.12 and 14.70%. Regarding the elasticity analysis of the variables, the RLC model and the RNA-1 model showed better behavior, while the RNA-11 model showed undesirable behavior in the “suites” variable, which could be discarded. This raises a reflection: despite superior metrics, neural networks can, in fact, present undesired behavior in some variable. This shows that elasticity analysis should always be considered when validating ANN models.

Keywords: Artificial Neural Network; Classical Linear Regression; Appraisal Engineering.

LISTA DE FIGURAS

Figura 1 – Fluxograma básico de trabalho	6
Figura 2 – Área de estudo: Centro de Florianópolis - SC	8
Figura 3 – Distribuição espacial dos dados coletados no Centro de Florianópolis/SC.....	14
Figura 4 – Distribuição espacial dos valores coletados.....	15
Figura 5 – Distribuição espacial das amostras de treinamento e validação.....	16
Figura 6 – Curva Normal ($\mu=0, \sigma=1$)	20
Figura 7 – Distribuição de resíduos com e sem variância constante	21
Figura 8 – Verificação gráfica da autocorrelação.....	22
Figura 9 – Verificação gráfica da autocorrelação espacial.....	22
Figura 10 – Verificação gráfica de um ponto influenciante	25
Figura 11 – Gráfico $e_i \times \hat{Y}_i$	25
Figura 12 – Fluxograma da força bruta	28
Figura 13 – Fluxograma do <i>stepwise (forward-selection)</i>	31
Figura 14 – Fluxograma básico para verificação dos pressupostos em RLC.....	35
Figura 15 – Fluxograma do “ <i>Geoda work book</i> ” para casos de Regressão Espacial	38
Figura 16 – Resíduos padronizados x valores ajustados.....	41
Figura 17 – Distâncias de Cook.....	41
Figura 18 – <i>Heatmap</i> referente à matriz de correlações	43
Figura 19 – Histograma \times Curva Normal	45
Figura 20 – Histograma cumulativo	45
Figura 21 – Q-Q Plot	46
Figura 22 – Resíduos padronizados x Valores ajustados.....	47
Figura 23 – Verificação gráfica da Multicolinearidade.....	48
Figura 24 – VIF e IC calculados no <i>Gretl</i> após o uso do “ <i>centering</i> ”	50
Figura 25 – Verificação gráfica da autocorrelação espacial.....	51
Figura 26 – Índice de Moran - RLC	51
Figura 27 – Gráficos de linearidade	53
Figura 28 – Gráficos de Predição	54
Figura 29 – Representação de uma Rede Neural Artificial	61
Figura 30 – Neurônio biológico	61
Figura 31 – Neurônio artificial	62
Figura 32 – Tipos de Arquiteturas de Redes Neurais.....	64
Figura 33 – RNA <i>feedforward</i> com única camada de neurônios (saída)	65
Figura 34 – RNA <i>feedforward</i> com camadas múltiplas (oculta + saída)	65
Figura 35 – Cenários possíveis de uma RNA em treinamento.....	71
Figura 36 – Interrupção do treinamento por meio do <i>Early Stopping</i>	72
Figura 37 – Modelo original de uma RNA e seus modelos podados	73
Figura 38 – <i>Trade-off</i> entre porcentagem de poda e perda de acurácia.....	74
Figura 39 – Intervalos de normalização conforme a função usada.	77
Figura 40 – Combinações possíveis para uma quantidade fixa de neurônios	80
Figura 41 – $R^2 \times$ categorias (1HLES).....	82
Figura 42 – $R^2 \times$ N° de neurônios (1HLES)	82

Figura 43 – $R^2 \times$ categorias (2HLES).....	83
Figura 44 – $R^2 \times$ N° de neurônios (2HLES)	83
Figura 45 – $R^2 \times$ categorias (3HLES)	84
Figura 46 – $R^2 \times$ N° de neurônios (3HLES)	84
Figura 47 – $R^2 \times$ categorias (1HLPR)	87
Figura 48 – $R^2 \times$ N° de neurônios (1HLPR)	87
Figura 49 – $R^2 \times$ categorias (2HLPR)	88
Figura 50 – $R^2 \times$ N° de neurônios (2HLPR)	88
Figura 51 – $R^2 \times$ categorias (3HLPR)	89
Figura 52 – $R^2 \times$ N° de neurônios (3HLPR).....	89
Figura 53 – $R^2 \times$ categorias (1HL)	92
Figura 54 – $R^2 \times$ N° de neurônios (1HL).....	92
Figura 55 – $R^2 \times$ categorias (2HL)	93
Figura 56 – $R^2 \times$ N° de neurônios (2HL).....	93
Figura 57 – $R^2 \times$ categorias (3HL)	94
Figura 58 – $R^2 \times$ N° de neurônios (3HL).....	94
Figura 59 – Curva de treinamento \times validação (1HL – $N_n = 50$)	96
Figura 60 – Curva de treinamento \times validação (2HL – $N_n = 50$)	96
Figura 61 – Curva de treinamento \times validação (3HL – $N_n = 50$)	96
Figura 62 – Gráfico de predição da amostra de validação e treinamento (1HL – $N_n = 50$)	97
Figura 63 – Gráfico de predição da amostra de validação e treinamento (2HL – $N_n = 50$)	97
Figura 64 – Gráfico de predição da amostra de validação e treinamento (3HL – $N_n = 50$)	97
Figura 65 – RMSE médios nas RNA com <i>Early Stopping</i>	98
Figura 66 – RMSE \times N° de neurônios (1HLES)	100
Figura 67 – RMSE \times N° de neurônios (2HLES)	100
Figura 68 – RMSE \times N° de neurônios (3HLES)	100
Figura 69 – RMSE médios nas RNA com <i>Pruning</i>	101
Figura 70 – RMSE \times N° de neurônios (1HLPR)	103
Figura 71 – RMSE \times N° de neurônios (2HLPR)	103
Figura 72 – RMSE \times N° de neurônios (3HLPR).....	103
Figura 73 – Curvas de convergência das RNA escolhidas.....	106
Figura 74 – Gráfico de predição da amostra de validação – RNA	107
Figura 75 – Gráfico de predição da amostra de validação – RLC.....	108
Figura 76 – Distribuição espacial dos resíduos e Moran (RLC)	109
Figura 77 – Distribuição espacial dos resíduos e Moran (RNA1).....	109
Figura 78 – Distribuição espacial dos resíduos e Moran (RNA2).....	109
Figura 79 – Distribuição espacial dos resíduos e Moran (RNA3).....	110
Figura 80 – Distribuição espacial dos resíduos e Moran (RNA4).....	110
Figura 81 – Distribuição espacial dos resíduos e Moran (RNA5).....	110
Figura 82 – Distribuição espacial dos resíduos e Moran (RNA6).....	111
Figura 83 – Distribuição espacial dos resíduos e Moran (RNA7).....	111
Figura 84 – Distribuição espacial dos resíduos e Moran (RNA8).....	111
Figura 85 – Distribuição espacial dos resíduos e Moran (RNA9).....	112
Figura 86 – Distribuição espacial dos resíduos e Moran (RNA10).....	112

Figura 87 – Distribuição espacial dos resíduos e Moran (RNA11).....	112
Figura 88 – Distribuição espacial dos resíduos e Moran (RNA12).....	113
Figura 89 – Variação do valor em função da Área.....	120
Figura 90 – Variação do valor em função do N° de WC.....	120
Figura 91 – Variação do valor em função do N° de suítes.....	121
Figura 92 – Variação do valor em função do N° Vagas.....	121
Figura 93 – Variação do valor em função do Padrão.....	122
Figura 94 – Variação do valor em função da Idade.....	122
Figura 95 – Variação do valor em função da Distância à Beira Mar.....	123
Figura 96 – Variação do valor em função do Norte.....	123

LISTA DE QUADROS

Quadro 1 – Itens descritos na NBR 14653-1 relativos à pesquisa de dados	11
Quadro 2 – Ressalvas sobre o <i>stepwise</i>	29
Quadro 3 - Comparação básica entre força bruta e <i>stepwise</i>	32
Quadro 4 - Condições para verificação da autocorrelação pelo teste de Durbin-Watson	37
Quadro 5 - Equações utilizadas para definir a quantidade de neurônios em RNA	67
Quadro 6 - Cenários simulados	79
Quadro 7 – Definir as transformações possíveis para cada variável.	171
Quadro 8 – Cálculo dos modelos RLC.	171
Quadro 9 – Cálculo de métricas de RLC	172
Quadro 10 – Resíduos padronizados e distâncias de Cook	173
Quadro 11 – Teste de Kolmogorov–Smirnov.....	173
Quadro 12 – Testes de White e Breusch-Pagan	174
Quadro 13 – Matriz de correlação, índice condicional e VIF.	174
Quadro 14 – Testes de autocorrelação.....	174
Quadro 15 – Testes de linearidade.	175
Quadro 16 – Dividir uma amostra em treinamento e validação.	175
Quadro 17 – Cálculo das RNA com transformações.....	176
Quadro 18 – Cálculo das RNA sem transformações com Early Stopping.	177
Quadro 19 – Cálculo das RNA sem transformações com <i>Pruning</i>	179
Quadro 20 – Arquivo de configurações.....	180

LISTA DE TABELAS

Tabela 1 – Combinações possíveis de transformações.....	27
Tabela 2 – Coeficientes da regressão.....	40
Tabela 3 – Poder de explicação do modelo	41
Tabela 4 – Correlações entre variável dependente e independentes	42
Tabela 5 – Matriz de correlações	42
Tabela 6 – ANOVA do modelo de Regressão	44
Tabela 7 – Significância global do modelo.....	44
Tabela 8 – Significância dos parâmetros.....	44
Tabela 9 – Porcentagens observadas versus teóricas	46
Tabela 10 – Teste de normalidade	46
Tabela 11 – Testes de homocedasticidade	47
Tabela 12 – VIF para cada variável	49
Tabela 13 – Teste de autocorrelação.....	50
Tabela 14 – Testes para autocorrelação espacial.....	52
Tabela 15 – Testes de Linearidade.....	52
Tabela 16 – Valores estimados <i>versus</i> valores observados no mercado (amostra de validação).....	54
Tabela 17 – Intervalos de confiança e intervalos de predição.....	56
Tabela 18 – Valores máximos e mínimos conforme o campo de arbítrio	57
Tabela 19 – Análise exploratória de dados por categoria em HLES.....	85
Tabela 20 – Testes de normalidade por categorias em HLES	85
Tabela 21 – Testes para comparação entre as categorias em HLES	85
Tabela 22 – Análise exploratória de dados por categoria em HLPR	90
Tabela 23 – Testes de normalidade por categorias em HLPR.....	90
Tabela 24 – Testes para comparação entre as categorias em HLPR.....	90
Tabela 25 – Análise exploratória de dados por categoria em HL.....	95
Tabela 26 – Teste de normalidade por categorias em HL.....	95
Tabela 27 – Testes para comparação entre as categorias em HL	95
Tabela 28 – RMSE médios nas RNA com <i>Early Stopping</i>	98
Tabela 29 – Teste de normalidade por categorias em HLES.....	99
Tabela 30 – Testes para comparação entre as camadas em HLES.....	99
Tabela 31 – RMSE médios nas RNA com <i>Pruning</i>	101
Tabela 32 – Teste de normalidade por categorias em HLPR.....	102
Tabela 33 – Testes para comparação entre as camadas em HLPR	102
Tabela 34 – Melhores modelos de RNA (com aplicação de transformação).....	104
Tabela 35 – Melhores modelos de RNA (sem aplicação de transformação)	105
Tabela 36 – Índices de Moran (RLC x RNA)	108
Tabela 37 – Valores obtidos por RNA (HLES), RLC e Mercado	116
Tabela 38 – Diferenças monetárias do mercado em relação aos modelos RNA e RLC.....	117
Tabela 39 – Diferenças percentuais do mercado em relação aos modelos RNA e RLC	118
Tabela 40 – Amostra coletada no Centro de Florianópolis	137
Tabela 41 – Nome da edificação e coordenadas UTM dos dados pesquisados.....	142
Tabela 42 – Métricas para verificação de <i>outliers</i> e pontos influenciantes.	148

Tabela 43 – Coeficientes de determinação variando-se a quantidade de N_n para cada cenário	151
Tabela 44 – RMSE (R\$) variando-se a quantidade de N_n para cada cenário	156
Tabela 45 – Métricas das RNA com transformação e usando <i>Early Stopping</i>	161
Tabela 46 – Melhores das RNA com transformação e usando <i>Pruning</i>	163
Tabela 47 – Melhores das RNA sem transformação e usando <i>Early Stopping</i>	166
Tabela 48 – Melhores das RNA sem transformação e usando <i>Pruning</i>	168

LISTA DE ABREVIATURAS E SIGLAS

DW – Durbin-Watson

e – Erro/resíduo

HL – *Hidden Layer* (camada oculta)

ID – Identificador;

k – Número de variáveis independentes

MAE – Mean Absolute Error

MAPE - Mean Absolute Percentual Error

ML – *Machine Learning*

MLP – *Multilayer Layer Perceptron* (*perceptron* de múltiplas camadas)

MSE – *Mean Squared Error*

n – Quantidade de dados da amostra

NBR – Norma Brasileira

N_n – Quantidade de neurônios na camada oculta

R^2 – coeficiente de determinação

RLC – Regressão Linear Clássica

RMSE – *Root Mean Squared Error*

RNA – Rede Neural Artificial

SUMÁRIO

1. INTRODUÇÃO	1
1.1. JUSTIFICATIVA	3
1.2. OBJETIVOS	4
1.3. HIPÓTESE	4
1.4. ESTRUTURA DA PESQUISA.....	5
2. APRESENTAÇÃO DE MATERIAIS, ÁREA DE ESTUDO E MÉTODO	6
2.1. MATERIAIS	7
2.2. CARACTERIZAÇÃO DA ÁREA DE ESTUDO	7
2.3. APRESENTAÇÃO DO MÉTODO DE TRABALHO	8
2.3.1 Pesquisa de dados.....	8
2.3.2 Produção de modelo por Regressão Linear Clássica (RLC).....	8
2.3.3 Produção de modelo por Rede Neural Artificial (RNA)	9
2.3.4 Análise comparativa entre os modelos produzidos na pesquisa	9
3. MERCADO IMOBILIÁRIO E PESQUISA DE DADOS	10
3.1. MERCADO IMOBILIÁRIO	10
3.2. QUALIDADE DA PESQUISA DE DADOS	10
3.3. CARACTERIZAÇÃO DA PESQUISA DE DADOS	12
3.4. AMOSTRAS DE TREINAMENTO E DE VALIDAÇÃO	15
4. PRODUÇÃO DE MODELOS POR REGRESSÃO LINEAR CLÁSSICA (RLC)	17
4.1. DEFINIÇÃO E PRESSUPOSTOS DA RLC	17
4.1.1 Linearidade.....	19
4.1.2 Normalidade	19
4.1.3 Homocedasticidade	20
4.1.4 Autocorrelação	21
4.1.5 Multicolinearidade.....	23
4.1.6 Pontos influenciantes e <i>outliers</i>	24
4.2. BUSCA PELO MELHOR MODELO DE RLC: FORÇA-BRUTA E STEPWISE	26
4.2.1 Força bruta.....	26
4.2.2 Stepwise.....	28
4.3. DESCRIÇÃO DO MÉTODO.....	31
4.3.1 Definição de parâmetros da RLC	31
4.3.2 Definição das transformações a serem testadas	32
4.3.3 Abordagem utilizada para buscar os melhores modelos RLC.....	32
4.3.4 Verificação dos pressupostos	34
4.4. RESULTADOS DO MODELO ESCOLHIDO DE RLC.....	39
4.4.1 Combinação utilizada	39
4.4.2 Coeficientes da regressão	39
4.4.3 Função de regressão e de estimação	40
4.4.4 Poder de explicação do modelo.....	40
4.4.5 Resíduos padronizados e Distâncias de Cook.....	41

4.4.6	Correlações	42
4.4.7	Erro padrão (<i>Se</i>) e coeficiente de variação (<i>CV</i>)	43
4.4.8	ANOVA e testes de significância	43
4.4.9	Verificação da Normalidade	45
4.4.10	Verificação da Homocedasticidade	47
4.4.11	Verificação da não Multicolinearidade	47
4.4.12	Verificação da não Autocorrelação	50
4.4.13	Verificação da Linearidade	52
4.4.14	Gráfico de Predição	54
4.4.15	Valores para os imóveis avaliados	54
4.5.	CONCLUSÕES PARCIAIS	59
5.	PRODUÇÃO DE MODELO POR REDES NEURAIS ARTIFICIAIS (RNA)	60
5.1.	DEFINIÇÃO	60
5.2.	NEURÔNIO BIOLÓGICO	61
5.3.	NEURÔNIO ARTIFICIAL	61
5.4.	APLICAÇÕES	63
5.5.	ARQUITETURAS DE RNA	63
5.6.	PARÂMETROS COMUMENTE UTILIZADOS NA DEFINIÇÃO DAS RNA	66
5.6.1	Quantidade de neurônios	66
5.6.2	Quantidade de camadas intermediárias	69
5.6.3	Taxa de aprendizagem e número de iterações	69
5.6.4	Funções de ativação	70
5.6.5	Otimizadores e inicializadores de pesos	70
5.7.	OVERFITTING E UNDERFITTING	71
5.8.	EARLY STOPPING	72
5.9.	PRUNING	72
5.10.	DESCRIÇÃO DO MÉTODO	74
5.10.1	Definição dos parâmetros de cálculo	74
5.10.2	Transformações em RNA	76
5.10.3	Escalonar/normalizar amostras	76
5.10.4	Treinar a RNA	77
5.10.5	Verificações dos modelos	78
5.11.	SIMULAÇÕES DE RNA COM A UTILIZAÇÃO DE TRANSFORMAÇÕES	79
5.11.1	Influência do número de neurônios (N_n) na capacidade preditiva de RNA	79
5.11.1.1	Cenários simulados	79
5.11.1.2	Cenários com <i>Early Stopping</i> : 1HLES, 2HLES e 3HLES	80
5.11.1.3	Cenários com <i>Pruning</i> : 1HLPR, 2HLPR e 3HLPR	86
5.11.1.4	Cenários sem <i>Early Stopping</i> ou <i>Pruning</i> : 1HL, 2HL e 3HL	91
5.11.2	Influência do número de camadas ocultas na capacidade preditiva de RNA	98
5.11.2.1	Comparações entre 1HLES, 2HLES e 3HLES	98

5.11.2.2	Comparações entre 1HLPR, 2HLPR e 3HLPR.....	101
5.11.3	Modelos selecionados.....	104
5.12.	SIMULAÇÕES DE RNA SEM A UTILIZAÇÃO DE TRANSFORMAÇÕES.....	104
5.12.1	Cenários simulados.....	104
5.12.2	Modelos selecionados.....	105
5.13.	VERIFICAÇÕES DE QUALIDADE PARA MODELOS DE RNA.....	105
5.13.1	Gráficos de convergência.....	105
5.13.2	Gráficos de Predição.....	107
5.13.3	Verificação da dependência espacial nos modelos RNA.....	108
5.14.	CONCLUSÕES PARCIAIS.....	113
6.	COMPARAÇÃO ENTRE MODELOS DE RLC E RNA: RESULTADOS E ANÁLISES.....	115
6.1.	MODELOS SEPARADOS PARA REALIZAR A AVALIAÇÃO.....	115
6.2.	VALORES OBTIDOS E COMPARAÇÕES.....	115
6.3.	VARIAÇÃO DOS VALORES EM FUNÇÃO DAS VARIÁVEIS EXPLICATIVAS.....	120
6.4.	CONCLUSÕES PARCIAIS.....	123
7.	CONCLUSÕES FINAIS.....	125
	REFERÊNCIAS BIBLIOGRÁFICAS.....	129
	APÊNDICE A – AMOSTRA COLETADA NO CENTRO DE FLORIANÓPOLIS.....	137
	APÊNDICE B – LOCALIZAÇÕES DOS DADOS DA AMOSTRA COLETADA.....	142
	APÊNDICE C – RESÍDUOS PADRONIZADOS E DISTÂNCIAS DE COOK.....	148
	APÊNDICE D – R² DAS RNA COM TRANSFORMAÇÃO.....	151
	APÊNDICE E – RMSE DAS RNA COM TRANSFORMAÇÃO.....	156
	APÊNDICE F – MÉTRICAS DAS MELHORES RNA COM TRANSFORMAÇÃO.....	161
	APÊNDICE G – MÉTRICAS DAS MELHORES RNA SEM TRANSFORMAÇÃO.....	166
	APÊNDICE H – CÓDIGOS EM PYTHON.....	171

1. INTRODUÇÃO

Engenharia de Avaliações é o campo de conhecimento que busca inferir o valor de um bem com o menor erro possível. Na busca desse valor, segundo a NBR 14653-2 (ABNT, 2011), deve ser dada preferência ao método comparativo direto de dados de mercado, no qual o avaliador utiliza um conjunto de dados semelhantes ao imóvel avaliado e recorre a algum método de cálculo para o tratamento dos dados.

Os métodos para tratamento científico prescritos pela NBR 14653-2 (ABNT, 2011) podem ser consultados em seus anexos (A, C, D e E). São eles: regressão linear, regressão espacial, envoltória de dados e redes neurais artificiais.

Segundo Sen e Srivastava (1990, p. 1), a regressão linear é uma “análise de dados que objetiva descobrir como uma ou mais variáveis (chamadas de variáveis independentes, variáveis preditoras ou regressores) afetam outras variáveis (chamadas de variáveis dependentes ou variáveis resposta)”. E, conforme consta em Hoffmann (2016, p. 3), a regressão é “a técnica estatística mais usada em estudos que exigem a análise de relações entre duas ou mais variáveis”.

Quanto à aplicação da regressão em modelos de avaliação de imóveis, a NBR 14653-2 (ABNT, 2011) afirma que esse é o método mais utilizado. Nos casos de existir apenas uma variável independente, o método é chamado de regressão linear simples; e quando há mais de uma variável independente, é chamado de regressão linear múltipla (RAO *et al.*, 2008). Em avaliações imobiliárias, o mais comum é serem utilizados modelos de regressão linear múltipla, pois há vários fatores que interferem na formação do valor (DANTAS, 2012). E no caso de haver dependência espacial que não pode ser tratada pela regressão clássica, deve ser usada a regressão espacial (ABNT, 2011).

A análise envoltória de dados é uma técnica consagrada para avaliar a produtividade e eficiência de imóveis, que se baseia em modelagem econométrica para estimar uma função de produção formada pelas unidades analisadas mais eficientes, onde a partir dos dados coletados é definido um espaço viável de produção, delimitado pela envoltória representativa dos melhores resultados, orientada segundo a minimização dos insumos ou maximização dos produtos. A eficiência de cada uma das unidades observadas é determinada pela distância normalizada à envoltória, através da solução de problemas de programação linear. A análise envoltória de dados sob dupla ótica explica as variações observadas em uma ou mais variáveis de interesse (dependentes), utilizando outras variáveis independentes (ABNT, 2011).

As Redes Neurais Artificiais (RNA) são “modelos matemáticos assemelhados às estruturas neurais biológicas e que podem, entre outras finalidades, ser utilizadas para o aprendizado e posterior

generalização¹” (ABNT, 2011, p. 47). O termo “aprendizado” refere-se à etapa na qual o modelo de RNA é treinado com dados da amostra coletada. Nessa etapa, o objetivo é que a RNA aprenda as relações entre as variáveis independentes (área, idade, padrão construtivo etc.) e a sua variável dependente (valor). Caso, ao final dessa etapa, a RNA seja capaz de fazer boas previsões para imóveis semelhantes aos da amostra usada no treino, ter-se-á alcançado a capacidade de generalização.

As RNA têm fundamentos em diversas áreas de conhecimento e o prognóstico é que esse método se consolide cada vez mais (HAYKIN, 2001). Essa previsão tem sido confirmada com a crescente automatização de tarefas por computadores viabilizada pela expansão da inteligência artificial. E ainda que, segundo a NBR 14653-2 (ABNT, 2011), o método mais utilizado seja a Regressão Linear Clássica (RLC), a tendência é que as RNA ganhem gradativamente mais espaço em avaliações imobiliárias devido sua capacidade natural de lidar com variáveis de natureza não linear, ao contrário da RLC (GONZÁLEZ; FORMOSO, 2000; BAPTISTELLA; CUNICO; STEINER, 2007); além disso, pode também lidar com dados não tabulares, por exemplo, imagens.

Apesar das vantagens conferidas pelo método, existem ainda lacunas na literatura técnica sobre o uso de redes neurais. Uma das questões debatidas é a determinação da quantidade ideal de neurônios a serem utilizados no modelo. Os modelos de RNA são formados por um conjunto de estruturas menores dispostas em camadas, os neurônios, que têm despertado pesquisas de muitos cientistas sobre a quantidade ideal de neurônios na construção de redes neurais (YOTOV; HADZHIKOLEV; HADZHIKOLEVA, 2020). No entanto, ainda não há consenso na literatura técnica sobre um número ótimo de neurônios que garanta os melhores resultados.

A determinação do número de neurônios é quase sempre feita de modo intuitivo ou usualmente por tentativa e erro, onde o objetivo é encontrar a quantidade de neurônios mais adequada (WANAS *et al.*, 1998; SHILIN *et al.*, 2010; SHEELA e DEEPA, 2013a). A própria NBR 14653-2 (ABNT, 2011) não determina um intervalo ou quantidade específica de neurônios, nem mesmo orienta o uso de qualquer equação para definir esse número. Aliás, a referida norma cita RNA em caráter apenas informativo, sem especificações mais detalhadas, o que pode ser entendido como um problema.

Intuitivamente, para definir o número de neurônios, é comum acreditar que, quanto maior a quantidade, melhores serão os resultados. No entanto, segundo Silva, Spatti e Flauzino (2010), esse aumento indiscriminado de neurônios produz um fenômeno denominado *overfitting*², no qual a RNA

1 Generalização é “a capacidade da rede responder adequadamente a padrões que não fizeram parte do conjunto de treinamento” (LIMA; PINHEIRO; SANTOS, 2014, p. 55). Ou seja, é a capacidade da RNA fazer boas previsões fora do conjunto utilizado para treiná-la.

2 Fenômeno caracterizado pelo excesso de treinamento em modelos de RNA. “Isto tem como resultado uma piora na capacidade de generalização da rede” (BRAGA; CARVALHO; LUDERMIR, 2000).

apenas decora o conjunto de dados. Na prática, um modelo com *overfitting* produz bons resultados apenas na amostra para a qual foi treinada, pois decorou os valores, mas é incapaz de fazer boas previsões para imóveis fora da amostra de treinamento. Por outro lado, ainda conforme os autores, o uso de pequenas quantidades de neurônios também é problemático, pois gera modelos de RNA com *underfitting*³ (grandes erros).

Outra lacuna é a definição da quantidade de camadas ocultas (ou intermediárias). A quantidade ideal de camadas ocultas varia de acordo com o problema: alguns problemas podem ser solucionados com apenas uma camada oculta, enquanto outros exigem múltiplas camadas (UZAIR; JAMIL, 2020). Na prática, o mais comum é encontrar RNA com uma ou duas camadas ocultas, conforme consta em Kaastra e Boyd (1996), no entanto, a NBR 14653-2 (ABNT, 2011) não estabelece uma quantidade específica de camadas ocultas, deixando a pesquisa e determinação desse parâmetro a cargo do avaliador, que deve usar algum algoritmo para isso.

Além do número de neurônios e camadas intermediárias, existem ainda outros parâmetros utilizados na definição da arquitetura de RNA como função de ativação, taxa de aprendizagem, otimizadores e inicializadores dos pesos, os quais não devem ser escolhidos de modo trivial.

Por fim, vale ressaltar que o método de RNA tem sido visto como uma espécie de “caixa-preta” em alguma medida (LUDWIG JR.; MONTGOMERY, 2007; MCCLUSKEY *et al.*, 2013; CHANASIT *et al.*, 2021; CHIAPPINI *et al.*, 2023). Vale ressaltar que no ramo da engenharia de avaliações essa percepção não é diferente; e a própria NBR 14653-2 (ABNT, 2011) apresenta o método em caráter informativo, sem maiores detalhes, o que dificulta a sua disseminação entre os avaliadores.

1.1. JUSTIFICATIVA

Em engenharia de avaliações, a maior parte da literatura utiliza Regressão Linear Clássica (RLC) como método para realizar avaliações. A RNA, por outro lado, tem sido pouco explorada, apesar dos conhecidos benefícios que podem ser obtidos e que já foram mencionados na introdução deste capítulo. A própria norma vigente não aborda com detalhes quais diretrizes relacionadas às RNA devem ser seguidas, o que deixa os avaliadores sem uma metodologia clara para utilizar e validar seus modelos.

Por exemplo, a norma não entra em detalhes com relação as definições de: arquitetura de RNA; quantidade de neurônios/camadas intermediárias; critérios para evitar *overfitting*; funções de ativação; normalização dos dados; amostras de treinamento/validação e; verificação dos modelos.

3 Fenômeno caracterizado pela insuficiência de treinamento em modelos de RNA.

Fica a critério do avaliador entender as entrelinhas do texto, pesquisar e testar o que parece ser mais adequado para cada caso avaliado.

Desta forma, a falta de definição normativa pode contribuir para uma escolha equivocada dos parâmetros utilizados nos modelos de RNA, impactando os resultados das avaliações e, até mesmo, inviabilizando-os em decorrência dos erros obtidos, tanto em situações de *overfitting*, quanto em *underfitting*. Esses erros, em avaliações imobiliárias, podem representar variações consideráveis nos valores negociados, prejudicando alguma das partes (comprador ou vendedor) interessadas.

Diante do cenário exposto, é possível compreender a pertinência de um estudo que contribua para o incremento no estado da arte de RNA aplicadas em avaliações imobiliárias.

1.2.OBJETIVOS

O objetivo geral deste trabalho é realizar uma análise comparativa entre o modelo clássico de regressão e modelos de redes neurais artificiais em avaliações imobiliárias na área central do município de Florianópolis. Especificamente, os objetivos são:

- Analisar o desempenho do melhor modelo de RLC obtido com auxílio da abordagem de força-bruta e tendo seus pressupostos verificados;
- Produzir RNA com e sem transformações e identificar modelos que apresentem boa capacidade preditiva ao variar diferentes parâmetros;
- Comparar as avaliações realizadas usando RLC com aquelas efetuadas usando RNA.

1.3.HIPÓTESE

A hipótese deste trabalho é que modelos produzidos a partir de RNA podem apresentar capacidade preditiva superior aos modelos produzidos a partir de RLC. Essa suposição se baseia na expectativa de que, quando devidamente treinados, modelos de RNA podem capturar relações complexas entre as variáveis de interesse.

1.4. ESTRUTURA DA PESQUISA

O trabalho é composto por sete capítulos, onde o primeiro aborda a contextualização, a justificativa, os objetivos, a hipótese e a estrutura da pesquisa.

Os materiais utilizados, a caracterização da área de estudo e o método proposto são apresentados no capítulo dois.

No terceiro capítulo é apresentada a pesquisa de dados realizada no centro de Florianópolis.

No quarto capítulo é apresentada a produção, validação e avaliação usando modelo de RLC.

A seguir, no quinto capítulo, são realizadas simulações em RNA variando diversos parâmetros a fim de encontrar modelos com boa capacidade preditiva.

No sexto capítulo, foram conduzidas avaliações utilizando RNA selecionadas, cujos resultados foram posteriormente comparados com o modelo de RLC.

No sétimo e último capítulo são apresentadas as conclusões finais acerca deste trabalho.

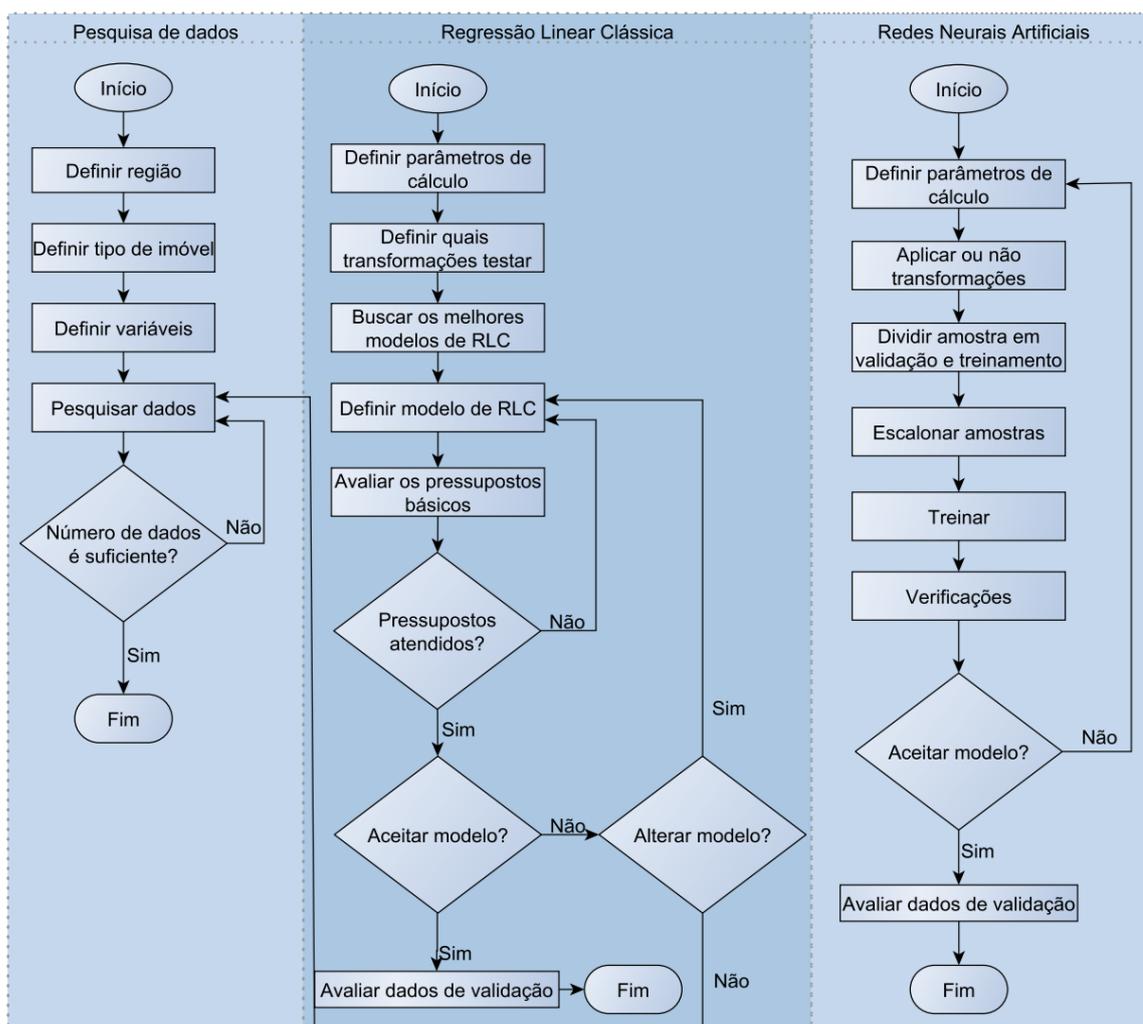
2. APRESENTAÇÃO DE MATERIAIS, ÁREA DE ESTUDO E MÉTODO

Este capítulo tem por objetivo apresentar os materiais, a caracterização da área de estudo e os procedimentos metodológicos utilizados neste trabalho de forma a alcançar os objetivos propostos. Para tanto foi realizada pesquisa qualitativa exploratória em artigos científicos, teses, dissertações, normas, entre outros, para verificar o estado da arte sobre os temas relacionados à avaliação imobiliária e os métodos utilizados para produzir modelos preditivos.

A pesquisa também utiliza como método a pesquisa quantitativa através de estudo de caso aplicado em uma área de estudo, visando realizar a coleta de dados amostrais, gerar modelos e validar os resultados. Também se vale de pesquisa qualitativa analítica crítica para embasar a aceitação ou não da hipótese formulada.

O fluxograma básico, que resume as etapas de trabalho, pode ser observado na figura 1.

Figura 1 – Fluxograma básico de trabalho



2.1.MATERIAIS

Na pesquisa foram utilizados os seguintes materiais:

- *Hardware*: Notebook, Intel Core i5, 8GB RAM;
- Sistema operacional: Windows 10;
- Linguagem de programação utilizada para automação de procedimentos concernentes à RLC e RNA: Python 3.8;
- IDE⁴ (*Integrated Development Environment*) - ferramenta auxiliar na atividade de programação e aumento de produtividade: *Pycharm Community Edition 2021.2.2*;
- Gerenciador⁵ de ambientes e pacotes: *Miniconda 4.3.10*;
- *Software E-Valuation e scripts*: utilizados nessa pesquisa para o desenvolvimento de modelos preditivos de RLC e RNA;
- Editor de textos planos: *Notepad++ 8.1.5*;
- Sistema⁶ de informação geográfico utilizado: *QGIS 3.10.10*

2.2.CARACTERIZAÇÃO DA ÁREA DE ESTUDO

A área utilizada para o estudo de caso está situada no bairro Centro de Florianópolis e pode ser observada na figura 2. É uma área com boa disponibilidade de imóveis, conta com infraestrutura, oportunidades de trabalho e vários serviços disponíveis (comércios, escolas, hospitais, parques, órgãos governamentais *etc.*). Além disso, é próxima à orla da Beira Mar Norte, faixa bastante valorizada na região. Tudo isso contribui para que essa área seja altamente atrativa para investimentos imobiliários, tanto para residências quanto para empreendimentos comerciais. Esses fatores somados à alta demanda por imóveis têm feito dessa localidade uma das regiões com metro quadrado mais valorizado de Florianópolis (10.531,00⁷ R\$/m²).

4 <<https://www.codecademy.com/articles/what-is-an-ide>> Acesso em 6 de outubro de 2021.

5 <<https://docs.conda.io/projects/conda/en/latest/>> Acesso em 6 de outubro de 2021.

6 <<https://qgis.org/en/site/>> Acesso em 6 de outubro de 2021.

7 <<https://downloads.fipe.org.br/indices/fipezap/indice-fipezap-202304-residencial-venda.pdf>> Acesso em 5 de junho de 2023.

Figura 2 – Área de estudo: Centro de Florianópolis - SC



Fonte: Google Earth, 2023.

2.3. APRESENTAÇÃO DO MÉTODO DE TRABALHO

São apresentadas nos itens a seguir os procedimentos metodológicos utilizados neste trabalho.

2.3.1 Pesquisa de dados

Nesta etapa são definidos o tamanho da amostra, os tipos de imóveis avaliados e as variáveis independentes, de forma a permitir a avaliação pelos modelos preditivos. Também é apresentada a amostra separada para os testes finais dos modelos propostos.

2.3.2 Produção de modelo por Regressão Linear Clássica (RLC)

Nesta etapa é produzido o modelo de avaliação imobiliária usando RLC. Primeiramente, são definidos os parâmetros de cálculo e as transformações testadas. Depois disso, utilizando o *software E-Valuation*, é realizada a busca pelos melhores modelos de RLC. É definido, então, o melhor modelo tendo como critério o maior R^2 obtido e as verificações dos pressupostos básicos da RLC. Por fim, o modelo é utilizado para prever valores a partir da amostra de validação.

2.3.3 Produção de modelo por Rede Neural Artificial (RNA)

Nesta etapa são produzidos modelos usando RNA. Ao total, são realizadas 5.364 simulações variando diversos parâmetros para definir quais modelos apresentam melhor capacidade preditiva. Foi considerada a aplicação de transformações nas variáveis em algumas RNA e em outras não. A amostra foi dividida entre treinamento e validação, as quais foram devidamente escalonadas para permitir os treinos e validações. Por fim, são selecionados alguns modelos para verificação de qualidade.

2.3.4 Análise comparativa entre os modelos produzidos na pesquisa

Por fim, são realizadas comparações entre RNA e RLC por meio de métricas, diferenças monetárias e percentuais. Nesse contexto, é verificada se a hipótese de que modelos de RNA podem apresentar capacidade preditiva superior à RLC foi atendida. Além disso, é efetuada a variação dos valores em função das variáveis explicativas (análise de elasticidade). Assim, é possível indicar quais são as similaridades e diferenças no comportamento dos modelos produzidos por RNA em relação ao modelo produzido por RLC.

3. MERCADO IMOBILIÁRIO E PESQUISA DE DADOS

Neste capítulo são apresentados conceitos relativos ao mercado imobiliário e aqueles utilizados para definir tamanho de amostra, quantidade de variáveis e como foi realizada a pesquisa de dados para subsidiar a produção dos modelos preditivos.

3.1. MERCADO IMOBILIÁRIO

Segundo o IAAO (2018, p. 46), mercado pode ser definido como “a área temática de interesse comum na qual compradores e vendedores interagem”. No caso de mercado imobiliário, o interesse comum gira em torno dos imóveis (casas, apartamentos, terrenos etc.). Esse mercado, segundo Dantas (2012), é formado pela tríade: bens imóveis, parte que oferece (vendedores) e parte que adquire (compradores) e dele são coletados os dados necessários para compor a amostra utilizada em avaliações imobiliárias. Essas características devem ser entendidas como variáveis, consideradas importantes na formação do valor do imóvel, ou seja, o valor monetário é função das variáveis coletadas.

3.2. QUALIDADE DA PESQUISA DE DADOS

Se o valor final é função das variáveis, é evidente que a qualidade da coleta de dados tem forte impacto no resultado da avaliação. “Os bons avaliadores sabem que os resultados de suas avaliações terão a mesma qualidade dos dados coletados em sua pesquisa. Em outras palavras, não é seguro construir uma tese sobre falsas premissas” (OLIVEIRA; GRANDISKI, 2014, p. 32).

Segundo Berrini (1949), é na fase da coleta de dados que está a maior dificuldade do processo avaliatório, e isso ocorre devido à falta de qualidade das informações disponíveis. Por isso, é desejável que o avaliador seja bastante criterioso na coleta de dados, pois a pesquisa de dados é a etapa que fundamenta o processo avaliatório. Caso contrário, não serão alcançados resultados satisfatórios, por melhor que seja o método de cálculo utilizado. Em última instância, a capacidade preditiva do método de cálculo reflete a qualidade da pesquisa de campo.

A etapa de coleta de dados tem caráter investigativo, onde o avaliador precisa entender quais variáveis são de fato importantes naquela região. “Na estrutura da pesquisa são eleitas as variáveis que, em princípio, são relevantes para explicar a tendência de formação de valor e estabelecidas as

supostas relações entre si e com a variável dependente⁸” (ABNT, 2011, p. 13). “Contudo, no decorrer dos trabalhos, outras variáveis podem se revelar como importantes”. (DANTAS, 2012, p. 16). Assim, o avaliador deve iniciar sua coleta de dados assumindo alguns pressupostos sobre quais variáveis independentes são importantes para explicar a variável dependente, no entanto, isso pode mudar durante a coleta, seja acrescentando ou retirando variáveis independentes⁹.

Para orientar o pesquisador na coleta de dados, a NBR 14653-1 (ABNT, 2011) apresenta alguns itens a serem verificados, conforme resumo apresentado no quadro 1.

Quadro 1 – Itens descritos na NBR 14653-1 relativos à pesquisa de dados

Itens da NBR 14653-1	Comentário
Buscar dados de mercado com atributos mais semelhantes possíveis aos do bem avaliando.	Delimitação que favorece a composição de uma amostra homogênea, situação desejável em modelos preditivos.
Identificar e diversificar as fontes de informação, sendo que as informações devem ser cruzadas, tanto quanto possível, com o objetivo de aumentar a confiabilidade dos dados de mercado.	Garante a confirmação da veracidade de informações que possam suscitar desconfiança no pesquisador.
Identificar e descrever a características relevantes dos dados de mercado coletados.	Menciona a necessidade da caracterização das variáveis importantes na formação do valor. Isso deve preceder o ato de coletar dados, pois o avaliador não pode iniciar sua pesquisa sem definir claramente quais variáveis devem ser coletadas
Buscar dados de mercado de preferência contemporâneos com a data de referência da avaliação.	Indica a necessidade da delimitação temporal do estudo, pois o mercado imobiliário é dinâmico e sofre alterações ao longo do tempo.

Por fim, é evidente que a quantidade de variáveis não significa, necessariamente, melhoria na qualidade no modelo. O excesso de variáveis explicativas pode ser até mesmo prejudicial por causa da multicolinearidade. Segundo Oliveira e Grandiski (2014, p. 63) “quando são usadas muitas variáveis independentes, aumenta a possibilidade de elas serem correlacionadas, propiciando excesso de ajustamento do modelo aos dados pesquisados, com pequeno aumento de predição do modelo”. Assim, é importante que o avaliador escolha de modo adequado variáveis que possam explicar seu modelo, evitando situações com variáveis em excesso, possivelmente, correlacionadas.

8 Segundo a NBR 14653-2 (ABNT, 2011, p. 8) é a “variável cujo comportamento se pretende explicar pelas variáveis independentes”.

9 “As variáveis independentes referem-se às características físicas (por exemplo, área, frente), de localização (como bairro, logradouro, distância ao polo de influência, entre outros) e econômicas (como oferta ou transação, época e condição do negócio – à vista ou a prazo)” (ABNT, 2011, p. 13).

3.3. CARACTERIZAÇÃO DA PESQUISA DE DADOS

Neste trabalho foram escolhidas oito variáveis independentes (7 quantitativas e 1 qualitativa), definidas, considerando o tipo de imóvel (apartamento) e a localização (Centro de Florianópolis), região que tem como polo valorizador a orla da Beira Mar Norte. As variáveis definidas foram:

- a) **Área privativa:** variável quantitativa mensurada em metros quadrados (m^2) que indica o tamanho do imóvel. A relação esperada entre a área e o valor do imóvel é diretamente proporcional, pois quanto maior a área de um imóvel, maior costuma ser o seu valor;
- b) **Nº de banheiros (WC):** variável quantitativa, onde a relação esperada entre o número de banheiros e o valor do imóvel é diretamente proporcional, pois quanto mais banheiros há no apartamento, maior tende a ser o valor;
- c) **Nº de suítes:** variável quantitativa, onde a relação esperada entre o número de suítes e o valor do imóvel é diretamente proporcional, pois quanto mais suítes, maior costuma ser o valor;
- d) **Nº de vagas:** variável quantitativa, onde a relação esperada entre o número de vagas de garagem e o valor é diretamente proporcional, pois quanto mais vagas disponíveis, maior tende a ser o valor do imóvel;
- e) **Padrão construtivo:** variável qualitativa, categorizada em alto, médio e simples (1, 2 e 3). A relação esperada entre o padrão construtivo e o valor do imóvel é inversamente proporcional, pois a escala da categoria é: 1º – alto, 2º – médio e 3º – simples; ou seja, quanto menor o número, maior o padrão (maior o valor);
- f) **Idade:** variável quantitativa mensurada em anos. A relação esperada entre a idade do imóvel e o valor do imóvel é inversamente proporcional, pois quanto maior é a idade, mais depreciado é o valor do imóvel;
- g) **Distância à Beira-mar:** variável quantitativa mensurada em metros (m) que indica a menor distância euclidiana do imóvel à orla. A relação esperada entre a distância à orla e o valor do imóvel é inversamente proporcional, pois quanto menor é a distância até a orla, mais valorizados costumam ser os imóveis;

h) **Norte**¹⁰: variável quantitativa mensurada em metros (m) que representa a localização do imóvel com relação ao eixo Norte-Sul. A relação esperada entre essa variável e o valor do imóvel é diretamente proporcional, pois quanto mais ao norte, mais valorizado costumam ser os imóveis na região estudada.

A coordenada Leste não foi considerada, pois apresentou pequena correlação ($r = 0,01$) com os valores dos imóveis e, também, problemas em sua significância ($>30\%$), que impede o enquadramento mínimo exigido pela NBR 14653-2 (ABNT, 2011).

A partir da definição das variáveis, foi definido o tamanho da amostra, segundo os preceitos da NBR 14653-2 (2011) que diz que a quantidade mínima de dados necessários, deve ser estabelecida de acordo com o grau de fundamentação¹¹ e em função da quantidade (k) de variáveis independentes, conforme apresentado a seguir:

- 6 ($k+1$) para grau de fundamentação III;
- 4 ($k+1$) para grau de fundamentação II;
- 3 ($k+1$) para grau de fundamentação I.

Considerando as 8 variáveis explicativas apresentadas, é necessária uma amostra com, no mínimo, 6($8+1$) elementos para o enquadramento no grau de fundamentação III. Como a amostra saneada possui 156 elementos, o modelo obedece à prescrição normativa, pois $156 > 54$.

Para evitar a micronumerosidade¹², a NBR 14653-2 (ABNT, 2011, p. 34) orienta obedecer às seguintes quantidades de dados (n):

- $n \geq 3 (k + 1)$;
- para $n \leq 30$, $n_i \geq 3$;
- para $30 < n \leq 100$, $n_i \geq 10\% n$;
- para $n > 100$, $n_i \geq 10$.

Onde n_i “é o número de dados de mesma característica, no caso de utilização de variáveis dicotômicas e variáveis qualitativas expressas por códigos alocados ou códigos ajustados” (ABNT, 2011, p. 34). No caso dessa pesquisa, como o $n > 100$, é requerido que cada variável qualitativa tenha

¹⁰ Foi feita uma translação da origem das coordenadas para diminuir a escala da variável. A coordenada 6.944.745,26 foi considerada como origem.

¹¹ Grau de fundamentação é uma classificação estabelecida pela NBR 14653-2 (2011) para os laudos de avaliação. Existem três graus (III, II e I) e o enquadramento em cada um deve seguir as prescrições do item 9.2.1 da NBR 14653-2 (2011) para modelos de regressão.

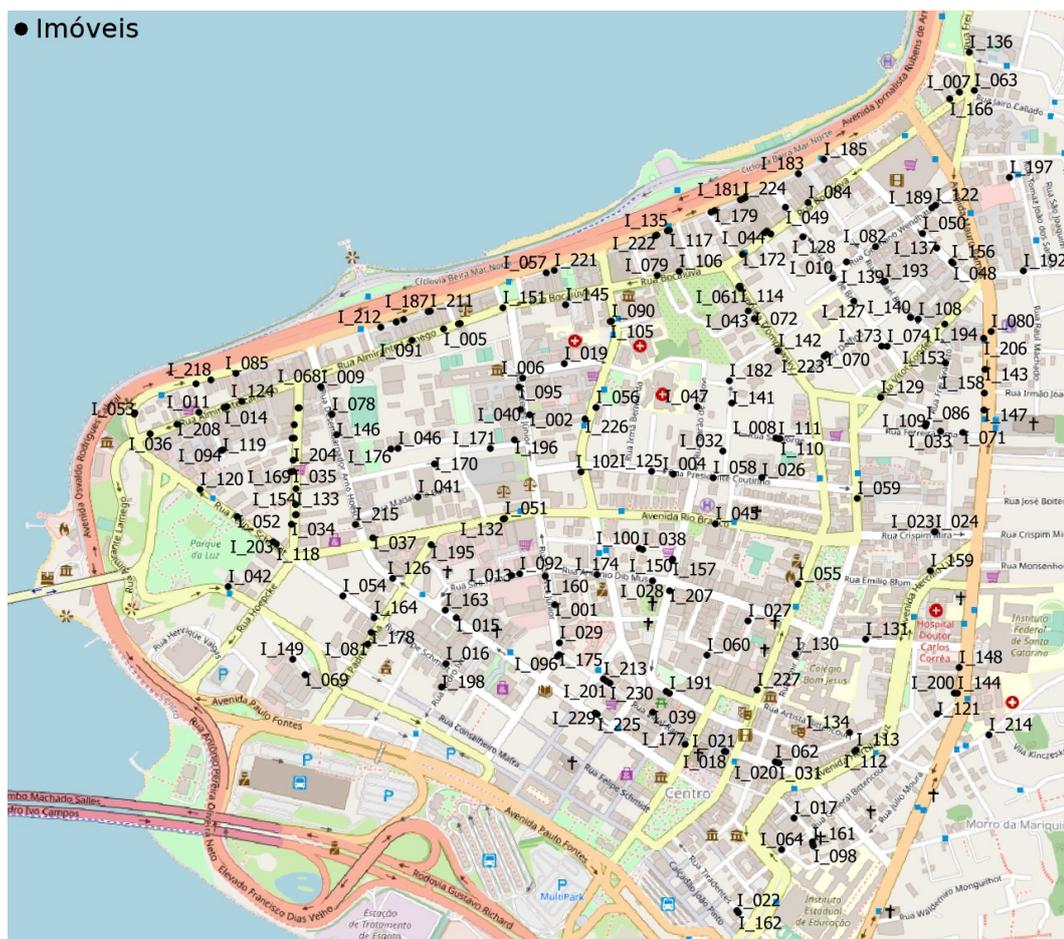
¹² Termo usado para se referir ao problema de “pequeno tamanho da amostra” (GOLDBERGER; GOLDBERGER, 1991).

no mínimo 10 dados com a mesma característica. Como a variável qualitativa “Padrão” possui 25 dados para a categoria “simples”, 51 dados para a categoria “médio” e 80 dados para a categoria “alto”, o requisito da NBR é atendido.

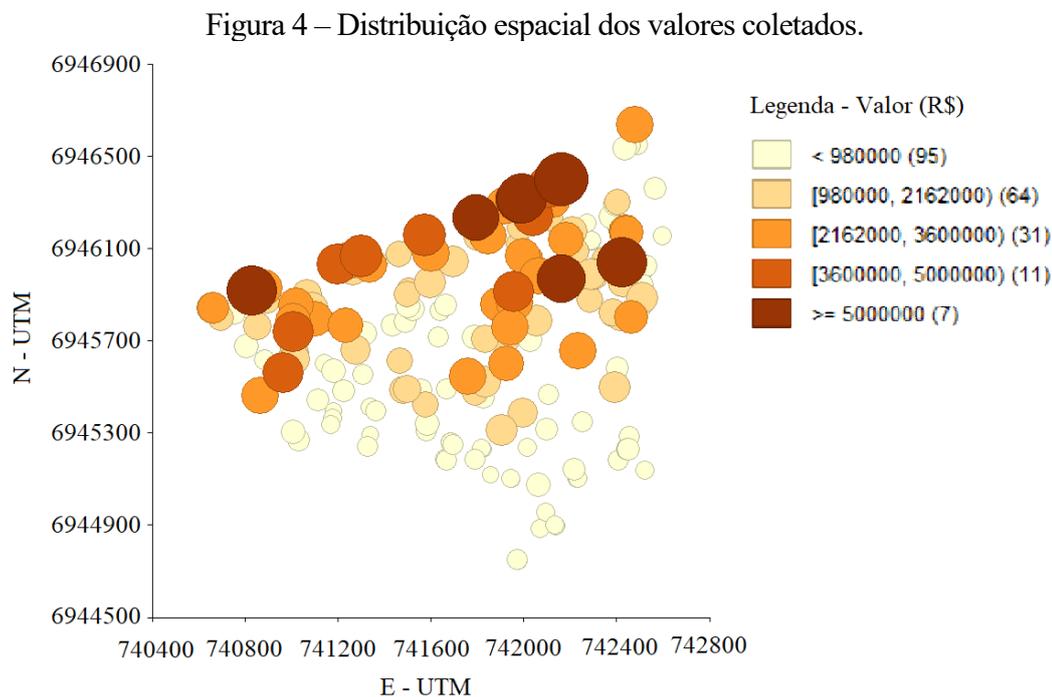
Assim, foram coletados 231 dados (apartamentos) na região de estudo entre os meses de agosto e setembro de 2021 nos seguintes *sites* de anúncios de imóveis: amrimoveis.com.br; lfenegociosimobiliarios.com.br; piramides.com.br; r8imoveis.com.br; chavesnamao.com.br; estivaleteimoveis.com.br; imobiliariacordova.com.br; imovelweb.com.br; investprime.com.br; joaquimalfredoimoveis.com.br; lopes.com.br; vivareal.com.br e zapimoveis.com.br (ver apêndice A).

Alguns dos imóveis coletados foram retirados da amostra bruta, os quais estão indicados no apêndice A por meio de asteriscos (*). A figura 3 apresenta a distribuição espacial dos imóveis utilizados no Centro de Florianópolis. As coordenadas UTM e os nomes dos condomínios/edifícios dos elementos pesquisados podem ser consultados no apêndice B.

Figura 3 – Distribuição espacial dos dados coletados no Centro de Florianópolis/SC.



A figura 4 apresenta a distribuição espacial dos valores dos imóveis, onde é possível notar que a região mais ao Norte é a que concentra os imóveis mais valorizados.



3.4. AMOSTRAS DE TREINAMENTO E DE VALIDAÇÃO

Para produção dos modelos (RLC e RNA) foi usado um conjunto de dados denominado amostra¹³ de treinamento e para comparar os modelos foi utilizada a amostra de validação.

A amostra total deve ser subdividida em: treinamento e validação. Para isso, em primeiro lugar, é criado um elemento (*lista_dados_extremos*) que armazena os índices dos dados extremos (máximos ou mínimos). O motivo para ser criado tal elemento é especificar quais são os dados que obrigatoriamente devem constar na amostra de treinamento. E a razão pela qual a amostra de treinamento deve conter os valores extremos é para garantir que todos os elementos da amostra de validação estejam dentro do escopo treinado, desse modo, é garantido o princípio da não extrapolação de variáveis recomendado pela NBR 14653-2 (ABNT, 2011).

De posse dos itens extremos, esses são inseridos na lista que representa os elementos a serem treinados (*indices_random_train*). Posteriormente, são acrescentados nessa lista outros elementos de

¹³ Os dados das amostras de treinamento e validação podem ser obtidos na pasta “data” do projeto no [GitHub](#).

maneira randômica, por meio da função *randint* da biblioteca *random*¹⁴. Esse acréscimo acontecerá até que o número de elementos da amostra de treinamento atinja o número ideal conforme a porcentagem de separação definida pelo avaliador. Por fim, é criada a *indices_random_test*, que contém os elementos que não foram incluídos na lista *indices_random_train*. No apêndice H é apresentado o código para essa divisão. As respectivas localizações dos dados das amostras de treinamento e validação constam na figura 5.

Figura 5 – Distribuição espacial das amostras de treinamento e validação.



¹⁴<https://docs.python.org/3/library/random.html>. Acesso em 22 de outubro de 2021.

4. PRODUÇÃO DE MODELOS POR REGRESSÃO LINEAR CLÁSSICA (RLC)

Neste capítulo, inicialmente, são apresentados no item 4.1 definições e pressupostos básicos para validar os modelos de RLC e no item 4.2 as abordagens que constam na literatura para a seleção de modelos RLC. A partir daí, é possível avançar na descrição do método no item 4.3. No item 4.4 são apresentados os resultados do modelo escolhido e as avaliações realizadas. Por fim, no item 4.5, são tecidas as conclusões parciais do capítulo.

4.1.DEFINIÇÃO E PRESSUPOSTOS DA RLC

Segundo consta em Kmenta (1988) ou Gujarati e Porter (2011), a função de regressão para uma amostra de n elementos pode ser representada conforme a função linear da equação 1 e, sua forma matricial, conforme as equações 2 e 3.

$$Y_i = \hat{b}_0 + \hat{b}_1 X_{1i} + \hat{b}_2 X_{2i} + \dots + \hat{b}_k X_{ki} + \hat{u}_i \quad (1)$$

$$Y = X\hat{b} + \hat{u} \quad (2)$$

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \dots \\ Y_n \end{bmatrix} = \begin{bmatrix} 1 & X_{11} & X_{21} & \dots & X_{k1} \\ 1 & X_{12} & X_{22} & \dots & X_{k2} \\ \dots & \dots & \dots & \dots & \dots \\ 1 & X_{1n} & X_{2n} & \dots & X_{kn} \end{bmatrix} \cdot \begin{bmatrix} \hat{b}_0 \\ \hat{b}_1 \\ \dots \\ \hat{b}_k \end{bmatrix} + \begin{bmatrix} \hat{u}_1 \\ \hat{u}_2 \\ \dots \\ \hat{u}_n \end{bmatrix} \quad (3)$$

Onde Y representa a variável dependente; X_1, X_2, \dots, X_k representam as k variáveis explicativas; \hat{b} representa os estimadores dos coeficientes de regressão; \hat{u} representa os resíduos (GUJARATI; PORTER, 2011).

O objetivo em avaliações imobiliárias é encontrar a função de estimação (valor monetário) a partir da função de regressão calculada. Para isso, pode ser realizada a estimação dos coeficientes de regressão pelo método dos mínimos quadrados (MMQ).

“O princípio da estimação dos mínimos quadrados envolve a minimização da soma dos desvios ao quadrado, dos valores observados a partir da média. Isto é, temos de encontrar o valor médio que faça a soma exigida tão pequena quanto possível” (KMENTA, 1988, p. 249). “De acordo com o método de mínimos quadrados, o estimador b deve ser aquele que minimiza o valor da soma de quadrados dos desvios”. (HOFFMANN, 2016, p. 6). Em suma, o objetivo do MMQ é encontrar os estimadores (coeficientes) que minimizem os erros entre valores preditos e valores observados.

Em modelos de regressão linear simples, para cálculo de \hat{b}_0 e \hat{b}_1 , Kmenta (1988) apresenta as equações 4 e 5:

$$\hat{b}_0 = Y - b_1 \cdot X \quad (4)$$

$$\hat{b}_1 = \frac{\sum((X_i - \bar{X}) \cdot (Y_i - \bar{Y}))}{\sum(X_i - \bar{X})^2} \quad (5)$$

No caso de regressão linear múltipla, para o cálculo de \hat{b} , Kmenta (1988) apresenta:

$$\hat{b} = (X'X)^{-1}(X'Y) \quad (6)$$

Onde, $Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \dots \\ Y_n \end{bmatrix}$, $X = \begin{bmatrix} 1 & X_{11} & X_{21} & \dots & X_{k1} \\ 1 & X_{12} & X_{22} & \dots & X_{k2} \\ \dots & \dots & \dots & \dots & \dots \\ 1 & X_{1n} & X_{2n} & \dots & X_{kn} \end{bmatrix}$

Uma vez determinados os coeficientes da RLC, é necessário verificar se o modelo atende aos pressupostos ou hipóteses básicas da RLC, a fim de garantir o bom funcionamento do modelo. Gujarati e Porter (2011, p. 325) apresentam as hipóteses para modelos de RLC:

- Hipótese 1. O modelo de regressão é linear nos parâmetros;
- Hipótese 2. Os valores dos regressores, os X , são fixos, os valores de X são independentes do termo de erro. Aqui, isso significa que exigimos covariância zero entre u_i e cada variável X ;
- Hipótese 3. Para os X dados, o valor médio do erro u_i é zero;
- Hipótese 4. Para os X dados, a variância de u_i é constante ou homocedástica;
- Hipótese 5. Para os X dados, não há autocorrelação, ou correlação serial, entre os termos do erro;
- Hipótese 6. O número de observações n deve ser maior que o número de parâmetros a serem estimados;
- Hipótese 7. Deve haver variação suficiente nos valores das variáveis X ;
- Hipótese 8. Não há colinearidade exata entre as variáveis X ;
- Hipótese 9. O modelo está especificado corretamente, logo não há viés de especificação;
- Hipótese 10. O termo estocástico (de erro) u_i é distribuído normalmente.

Visando garantir tais pressupostos, o anexo A da NBR 14653-2 (ABNT, 2011), em seu item A.2.1, orienta que sejam verificados os seguintes itens: linearidade, normalidade, homocedasticidade, não autocorrelação, não multicolinearidade, ausência de pontos influenciantes e *outliers*. Considerando a relevância de verificar tais itens, na sequência são fornecidas definições e maneiras para identificar cada um.

4.1.1 Linearidade

Há dois tipos de linearidades: a linearidade com relação às variáveis X e com relação aos parâmetros \hat{b} ; no primeiro caso, tem-se uma relação linear quando X não tem expoente diferente de 1 e não se combina com outra variável; no segundo caso, é linear quando o \hat{b} não é elevado a expoente que não seja 1 e não se combina com nenhum outro parâmetro; essa é a linearidade a que se refere o termo “regressão linear” (GUJARATI; PORTER, 2011).

Assim, é fundamental garantir a linearidade na RLC, pois o método pressupõe esse tipo de comportamento. Uma saída para alcançar tal comportamento é a utilização de transformações nas variáveis. As transformações citadas pela norma são artifícios matemáticos para alcançar relações lineares. Ou seja, ainda que a relação original não seja linear, transformações podem ser aplicadas às variáveis visando tornar o modelo especificado como linear (KRÄMER; SONNBERGER, 1986).

Sobre isso, a NBR 14653-2 (ABNT, 2011, p. 35) afirma:

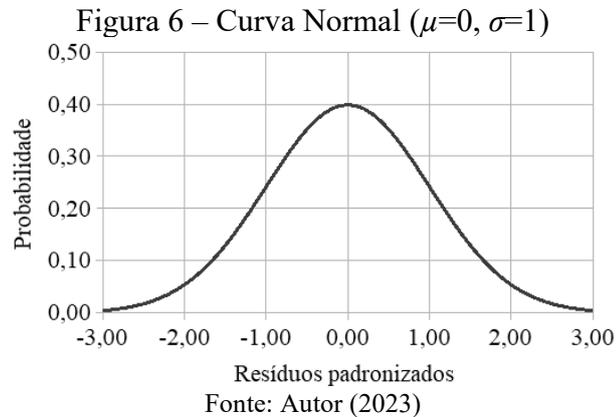
“As transformações utilizadas para linearizar o modelo devem, tanto quanto possível, refletir o comportamento do mercado, com preferência pelas transformações mais simples de variáveis, que resultem em modelo satisfatório [...]. Após as transformações realizadas, se houver, examina-se a linearidade do modelo, pela construção de gráficos dos valores observados para a variável dependente versus cada variável independente, com as respectivas transformações”.

4.1.2 Normalidade

Essa hipótese espera que os resíduos (diferenças entre valores observados e preditos), estejam dispostos de modo semelhante ao comportamento da curva gaussiana/normal. A curva normal é “uma clássica distribuição de curva em forma de sino, determinada por dois parâmetros: sua média μ (*mi*) e o desvio padrão σ (*sigma*)” (GRUS, 2016, p. 75).

Capellano (2014, p. 133) explica que “a média e o desvio padrão definem a forma da curva normal. A média determina sua posição em relação ao eixo das ordenadas [...] e o desvio padrão determina sua altura e largura”. Assim, usando os resíduos padronizados, o formato da curva normal com $\mu=0$ e $\sigma=1$ deve ser conforme mostrado na figura 6. A ideia, portanto, é que os resíduos

padronizados da RLC estejam dispostos de modo semelhante ao observado nessa figura, de modo que possa ser garantido o comportamento normal.



Para a verificação da normalidade, a NBR 14653-2 (ABNT, 2011) orienta algumas opções:

- a) pelo exame de histograma dos resíduos padronizados, com o objetivo de verificar se sua forma guarda semelhança com a da curva normal;
- b) pela análise do gráfico de resíduos padronizados versus valores ajustados, que deve apresentar pontos dispostos aleatoriamente, com a grande maioria situada no intervalo $[-2, +2]$;
- c) pela comparação da frequência relativa dos resíduos amostrais padronizados nos intervalos de $[-1; +1]$, $[-1,64, +1,64]$ e $[-1,96, +1,96]$, com as probabilidades da distribuição normal padrão nos mesmos intervalos, ou seja, 68%, 90% e 95%;
- d) pelo exame do gráfico dos resíduos ordenados padronizados versus quantis da distribuição normal padronizada, que deve se aproximar da bissetriz do primeiro quadrante¹⁵;
- e) pelos testes de aderência não paramétricos, como, por exemplo, o qui-quadrado, o de Kolmogorov-Smirnov ajustado por Stephens e o de Jarque-Bera.

4.1.3 Homocedasticidade

A homocedasticidade é definida por Gujarati e Porter (2011, p. 370) da seguinte maneira: “igual ou homogêneo (*homo*) espalhamento (*cedasticidade*), isto é, variância igual”.

Isso significa que o modelo precisa apresentar resíduos aleatórios e uniformes, isto é, sem viés ao longo dos valores analisados, de modo a evitar o comportamento heterocedástico.

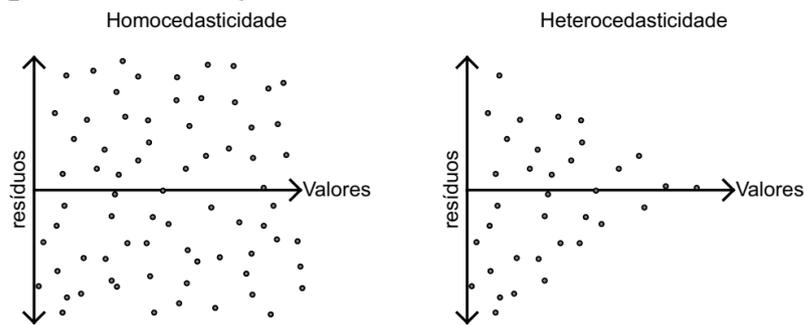
“Heterocedasticidade é a falta de variância residual constante através da amplitude dos valores previstos. Em outras palavras, os erros são maiores em algumas porções da amplitude do que em outras” (BRUCE; BRUCE, 2019, p. 166).

¹⁵ Essa representação gráfica é comumente referenciada como “*Q-Q plot*”.

Para verificar essa hipótese, a NBR 14653-2 (ABNT, 2011) cita a análise gráfica dos resíduos e os testes de Park¹⁶ e White, deixando o avaliador livre para usar também outras verificações.

No caso da análise gráfica, quando os pontos são distribuídos de modo aleatório, há indício de variância constante, mas se estiverem distribuídos obedecendo alguma tendência, há suspeita de variância não constante (DANTAS, 2011). A figura 7, por exemplo, mostra duas situações: uma distribuição homocedástica e outra heterocedástica, respectivamente.

Figura 7 – Distribuição de resíduos com e sem variância constante



Fonte: Autor (2023)

4.1.4 Autocorrelação

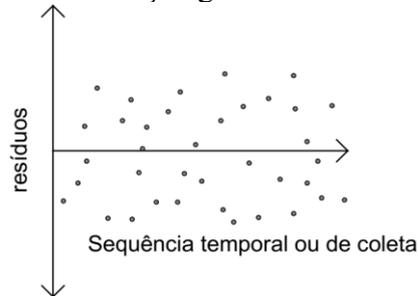
Em modelos de regressão linear, é importante não haver relação entre os termos de erro, ou seja, um erro de um determinado dado não deve ser influenciado ou influenciar outro erro qualquer (KMENTA, 1988; GUJARATI; PORTER, 2011). Isso implica que o erro em um determinado momento não afeta o erro em outro (KMENTA, 1988).

Desta forma, Kendall e Buckland (1971, p. 8) *apud* Gujarati e Porter (2011) definem autocorrelação da seguinte maneira: “correlação entre integrantes de séries de observações ordenadas no tempo [como as séries temporais] ou no espaço [como nos dados de cortes transversal]”. A autocorrelação de séries temporais costuma ser menos comum em avaliações imobiliárias, pois as coletas de dados são realizadas em um período bem definido.

¹⁶ Goldfeld e Quandt (1972) *apud* Gujarati e Porter (2011) chamam atenção para a possibilidade de problemas no teste de Park, onde os resíduos da regressão auxiliar usada nesse teste podem violar pressupostos do Método dos Mínimos Quadrados e até mesmo apresentarem heterocedasticidade. Gujarati e Porter (2011) encorajam o uso desse teste para fins estritamente exploratórios.

Gazola (2002) apresenta uma imagem que mostra a forma de verificar a autocorrelação para o caso de dados ordenados no tempo; nessa situação, são dispostos os resíduos em sequência temporal ou de coleta, os quais devem estar distribuídos aleatoriamente para não haver autocorrelação (figura 8).

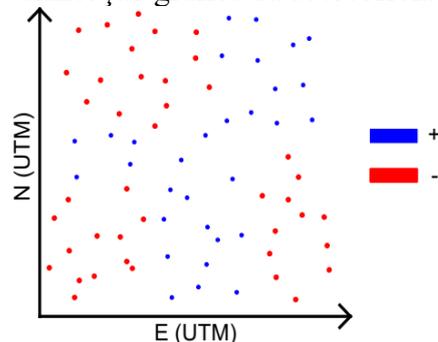
Figura 8 – Verificação gráfica da autocorrelação



Fonte: Adaptado de Gazola (2002)

Já a autocorrelação espacial tem a ver com a capacidade que um dado tem de influenciar imóveis próximos e pode ser verificada no mapa de resíduos (e_i), que precisa apresentar e_i positivos e negativos espalhados randomicamente (DANTAS, 2011). A figura 9 indica a presença da autocorrelação espacial, pois há um inequívoco agrupamento de resíduos de mesmo sinal.

Figura 9 – Verificação gráfica da autocorrelação espacial



Fonte: Autor (2023)

A antiga versão da NBR 14653-2 (ABNT, 2004) trazia como uma das formas de fazer a verificação de autocorrelação o teste de Durbin-Watson. Na versão atual, a NBR 14653-2 (ABNT, 2011) retirou essa verificação e manteve apenas a indicação de fazer um gráfico de resíduos e valores ajustados, que deve ter pontos dispostos aleatoriamente.

4.1.5 Multicolinearidade

Multicolinearidade acontece quando variáveis independentes contribuem de modo muito similar para explicar as variações na variável dependente. Nesse caso, o modelo possui variáveis independentes em excesso, pois a mesma explicação está sendo dada por várias.

Em modelos de regressão, o que o avaliador espera obter são fortes correlações entre a variável Y_i e as X_k , por outro lado, cada variável X_k deve ter um comportamento independente das outras X_k (OLIVEIRA, GRANDISKI, 2014). Ainda, conforme os autores, se houver correlação forte entre as variáveis X_k , ocorre a colinearidade (duas variáveis) ou multicolinearidade (mais de duas) e esse problema está ligado ao comportamento dos preços, que dependem de variáveis X_k que se comportam de modo coeso.

Um exemplo disso é um modelo de regressão que depende de variáveis como: área, suítes, garagens, idade, padrão construtivo e localização. É esperado que imóveis com áreas maiores também tenham mais suítes e garagens (e vice-versa), ou seja, há uma correlação natural entre essas variáveis. A questão a ser respondida não é se há ou não multicolinearidade, mas se o grau da correlação identificada é aceitável (KMENTA, 1988). Ou seja, em última análise, é certo que haverá correlações entre variáveis independentes, a questão é mensurar se tais correlações são aceitáveis ou não.

“Para a verificação da multicolinearidade deve-se, em primeiro lugar, analisar a matriz das correlações, que espelha as dependências lineares de primeira ordem entre as variáveis independentes, com atenção especial para resultados superiores a 0,80” (ABNT, 2011, p. 36). No entanto, essa análise deve ser cautelosa, pois, segundo Gujarati e Porter (2011), não é necessário a ocorrência de altas correlações para haver multicolinearidade.

Uma das formas citadas por Gujarati e Porter (2011) para a verificação do grau de multicolinearidade é número condicional (n_c) e o índice condicional (IC), onde $100 < n_c < 1.000$ ou $10 < IC < 30$ indicam multicolinearidade moderada a forte e $n_c > 1000$ e $IC > 30$ indicam multicolinearidade grave.

Outra métrica citada pelos autores é o fator de inflação variada (FIV ou VIF). À medida que o VIF aumenta, mais desfavorável é o comportamento, sendo que uma forte multicolinearidade pode ser detectada para valores de $VIF > 10$ (KLEINBAUM; KUPPER; MULLER, 1988 apud GUJARATI; PORTER, 2011). Outros autores são mais conservadores e defendem que casos graves de multicolinearidade acontecerão se o VIF ultrapassar limites como 5 ou 10 (MONTGOMERY; PECK, 1982 apud VAN GAANS; VRIEND, 1990).

No caso de alto grau de multicolinearidade, uma saída é identificar qual variável, dentre as fortemente correlacionadas, é a mais significativa para o modelo (BRASIL, 2018). Depois disso, é

possível excluir as variáveis menos significativas que estejam contribuindo para um alto grau de multicolinearidade. Outra solução é expandir a amostra e verificar se o novo modelo tem o seu grau de multicolinearidade reduzido (GUJARATI; PORTER, 2011). Essa solução, no entanto, pode ser de difícil aplicação em cenários com escassez de dados.

Apesar dos esforços para contornar a multicolinearidade, é possível que ela faça parte do modelo, ou seja, seja intrínseca ao mercado estudado. Por isso a própria NBR 14653-2 (ABNT, 2011) permite a utilização de modelos com multicolinearidade (não perfeita), desde que os imóveis avaliados também sigam o mesmo padrão observado no modelo.

Por fim, é importante verificar se a multicolinearidade detectada é real ou está inflada. Para isso pode ser aplicada a técnica de escalonamento *centering*¹⁷. Nesse contexto é importante saber que existem dois tipos de multicolinearidade: a não-essencial e a essencial, sendo que a primeira é removida por meio de escalonamento, mas a segunda não (MARQUARDT; 1980 apud AFSHARTOUS; PRESTON, 2011). Ou seja, essa técnica reduz apenas a parte não essencial, mantendo a parte essencial, como demonstrado por Dalal e Zickar (2012) e deve ser considerada sempre que houver dúvidas com relação aos resultados da multicolinearidade. “O *centering* é, portanto, uma etapa importante ao testar os efeitos de interação na regressão múltipla para obter uma interpretação significativa dos resultados” (ROBINSON; SCHUMACKER, 2009).

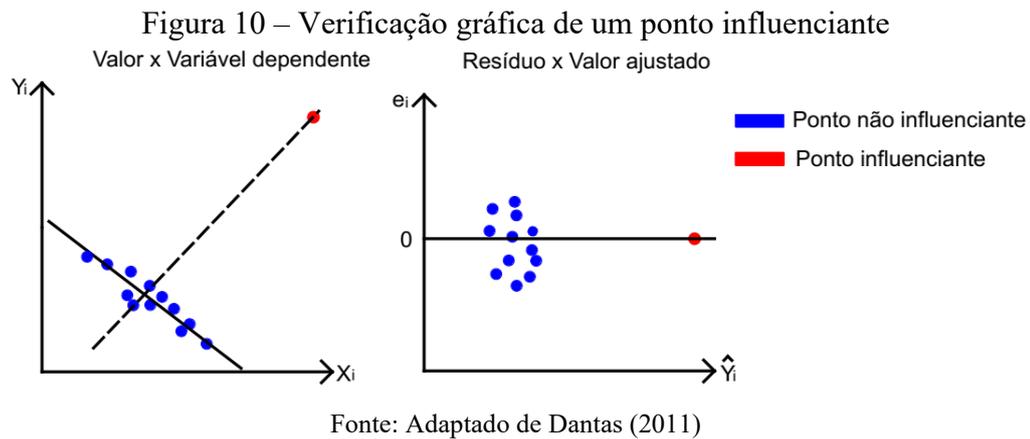
4.1.6 Pontos influenciantes e *outliers*.

“A existência de pontos influenciantes ou pontos atípicos pode ser verificada pelo gráfico dos resíduos versus cada variável independente, como também em relação aos valores ajustados, ou usando técnicas estatísticas mais avançadas, como a estatística de Cook ou a distância de Mahalanobis para detectar pontos influenciantes”. (ABNT NBR 14653-2, 2011, p. 37).

Esses pontos são caracterizados por possuírem pequenos resíduos, mas, ao mesmo tempo, distanciando-se demasiadamente do conjunto de dados, tendo assim a capacidade de alterar o comportamento da tendência esperada para os valores do mercado. (DANTAS, 2011, p. 114).

Dantas (2011) cita ainda três formas para identificar um ponto influenciante: a) através da distância de *Cook*; b) Y_i versus X_j ; c) e_i versus X_j . A figura 10 mostra o comportamento de um ponto influenciante caracterizado por apresentar pequenos resíduos e estar fora do alinhamento formado pelos demais pontos.

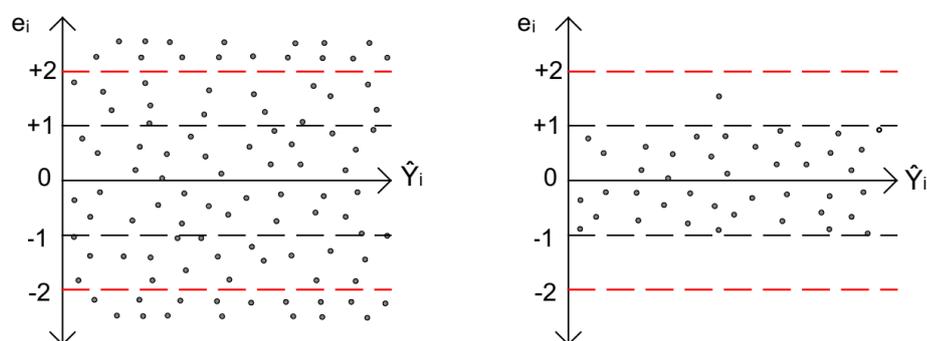
¹⁷ A técnica consiste em subtrair todos os elementos de uma variável pela média da variável.



Com relação aos outliers, são apresentadas duas definições: a) “Entende-se por *outlier* um dado que contém grande resíduo em relação aos demais que compõem a amostra” (DANTAS, 2011, p. 113); b) “os elementos discrepantes ou “*outliers*” são observações cujo comportamento é muito diferenciado das demais” (Oliveira; Grandiski, 2014, p. 87).

É possível notar nas duas definições o cuidado em ressaltar a comparação entre o discrepante e os demais dados da amostra. Oliveira e Grandiski (2014), por exemplo, afirmam que um dado considerado *outlier* tem um erro alto, no entanto, esse erro é alto quando comparado com os demais dados. Um critério comumente utilizado é considerar discrepante elementos fora do intervalo $[-2; +2]$, que é um critério fundamentado na distribuição normal dos dados (Oliveira; Grandiski, 2014). Porém, Dantas (2011) destaca que há situações em que os dados extrapolam o intervalo $[-2; +2]$ e não parece haver *outliers*; e há situações que mesmo sem pontos fora do intervalo $[-2; +2]$ ocorre *outlier*. A figura 11 esclarece tal comportamento, o gráfico da esquerda, mesmo com pontos fora do intervalo $[-2; +2]$, tem um conjunto homogêneo (sem discrepantes), ao contrário do gráfico à direita.

Figura 11 – Gráfico $e_i \times \hat{Y}_i$



Fonte: Adaptado de Dantas (2011)

4.2.BUSCA PELO MELHOR MODELO DE RLC: FORÇA-BRUTA E STEPWISE

Esse item é importante para fundamentar a escolha de qual abordagem será adotada para realizar a busca pelos melhores modelos de RLC, que é uma etapa que demanda tempo quando não se dispõe de *softwares* para auxiliar a tarefa. A razão dessa demora é que, dependendo do número de variáveis independentes (k), pode haver milhares ou milhões de combinações diferentes para serem analisadas. O número total de combinações possíveis (n_c) é dado pela equação 7:

$$n_c = k_1^{TR1} \cdot k_2^{TR2} \quad (7)$$

Onde k_1 é quantidade de variáveis independentes que não tem nenhum valor igual a “0”; k_2 é quantidade de variáveis independentes que tem, pelo menos, um valor igual a “0”; $TR1$ é a quantidade de transformações que as variáveis sem valores “0” podem assumir; $TR2$ é a quantidade de transformações que as variáveis com pelo menos um valor “0” podem assumir.

Há duas técnicas muito conhecidas para buscar os melhores modelos de regressão: força bruta (*brute-force*) e o *stepwise*.

4.2.1 Força bruta

Embora, comumente, referenciado como um método, força bruta é mais precisamente uma abordagem computacional aplicada na elaboração de algoritmos e seu uso na computação consiste em verificar cada solução/informação disponível dentro de um campo de possibilidades, o qual contém certamente a solução/informação desejada (HLAVÁČ; JEFFERY; WIEDERMANN, 2000). A força bruta é, portanto, uma forma de busca exaustiva, onde cada solução (uma por uma) é verificada.

O termo “força bruta” é mais utilizado nas ciências da computação. Por isso, fora desse contexto, é possível que sejam utilizadas outras designações para essa abordagem. Por exemplo, na obra de Sen e Srivastava (1990), o termo utilizado para designar força bruta é “*search over all possible subsets*”, e Zhang (2016) utiliza o termo “*best subset regression*” para designar o procedimento de força bruta. Apesar de nomenclaturas diferentes, todos seguem a mesma lógica: verificar todas as soluções (uma a uma).

Conforme Sen e Srivastava (1990), a força bruta é a abordagem mais útil na seleção de variáveis (se não houver muitas para serem analisadas). Os autores fazem essa ressalva tendo em vista o tempo de processamento, no entanto, reconhecem que o desenvolvimento tecnológico tende a

possibilitar a manipulação de muitas variáveis. O tempo de processamento também é uma ressalva feita por Hlaváč, Jeffery e Wiedermann (2000), pois existem três fatores que impactam esse tempo:

- a) Complexidade do problema;
- b) Volume de dados;
- c) Processador utilizado.

Dos fatores citados, não é possível alterar a complexidade do problema, nem o volume de dados. Por outro lado, o surgimento de processadores com melhor performance tem ajudado a aprimorar o desempenho do processo de força bruta e, assim, o tempo de cálculo tem se tornado cada vez menos um fator limitante na utilização dessa abordagem.

Apesar da ressalva com relação ao tempo de processamento, Hlaváč, Jeffery e Wiedermann (2000) afirmam que a força bruta sempre garante sucesso. E isso acontece porque todas as possibilidades são verificadas, ou seja, não há como o melhor modelo ficar de fora da análise do avaliador.

Na força bruta, cada solução (modelo) representa uma combinação dentro de um conjunto com todas as combinações possíveis. Existem vários modelos de regressão por causa das diferentes transformações que podem ser aplicadas em cada variável. Para exemplificar, a tabela 1 apresenta a quantidade de combinações possíveis para diferentes valores de k (variáveis independentes) considerando sete transformações. Pela quantidade de combinações apresentadas nessa tabela, fica evidente que é impraticável verificar cada solução sem automação computacional.

Tabela 1 – Combinações possíveis de transformações

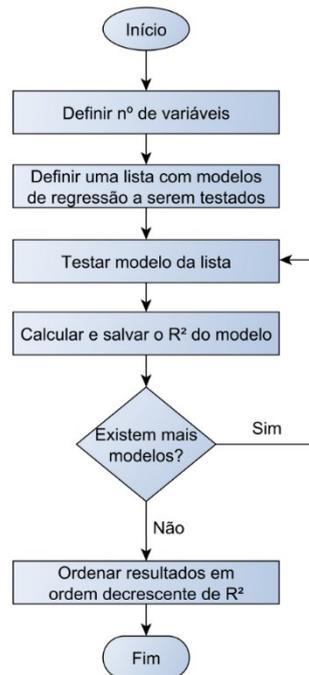
k	Quantidade de combinações
2	49
4	2.401
6	117.649
8	5.764.801
10	282.475.249

Na utilização da força bruta, é necessário ainda escolher alguma métrica para avaliar a capacidade preditiva de cada modelo e, assim, julgar se um modelo é melhor que outro. No caso da RLC, é muito comum a utilização do R^2 , o coeficiente de determinação do modelo.

Por fim, o procedimento da força bruta não tem como objetivo substituir o ser humano na tomada de decisão sobre o modelo final. Essa abordagem é comumente utilizada para listar e

apresentar ao profissional as melhores opções encontradas. A partir daí, a decisão final é tomada pelo avaliador. Além disso, a remoção de variáveis do modelo é decisão tomada pelo avaliador e nunca delegada à máquina. Isso é importante em cenários onde alguma variável se comporta de modo incoerente e, ainda assim, o modelo apresenta um valor considerável de R^2 . A figura 12 apresenta um fluxograma de como o processo de força bruta pode ser aplicado para calcular e apresentar os melhores modelos de regressão.

Figura 12 – Fluxograma da força bruta



4.2.2 Stepwise

O termo *stepwise*¹⁸ sugere algo marcado ou processado em etapas (de modo gradual). No contexto de regressão linear, esse termo é utilizado para designar um procedimento pelo qual é definido o modelo de regressão a ser utilizado. Diferentemente da força bruta, onde a decisão final sobre a escolha das variáveis é responsabilidade do ser humano, o *stepwise* propõe fazer a seleção automaticamente.

Basicamente, o *stepwise* é uma abordagem que trabalha com inserção/remoção gradual e controlada de variáveis independentes para a construção do modelo final de regressão (PENNSTATE,

18 <<https://www.merriam-webster.com/dictionary/stepwise>>. Acesso em 2 de abril de 2022.

2023). Essa seleção de variáveis é usada quando o pesquisador se depara com uma série de preditores potenciais, mas não tem (ou não usa) o conhecimento necessário que lhe permita especificar, de antemão, as variáveis importantes a serem incluídas no modelo (HARRELL, 2001).

Essa técnica *stepwise* pode ser dividida em: “*forward-selection*” e “*backward-elimination*”:

O primeiro caso “[...] inicia sem variáveis explicativas e, em seguida, adiciona variáveis, uma a uma, com base em qual variável é a mais estatisticamente significativa, até que não haja mais variáveis estatisticamente significativas”. O segundo caso “[...] inicia com todas as variáveis explicativas possíveis e, em seguida, descarta as variáveis estatisticamente menos significativas, uma a uma” (SMITH, 2018, p. 1-2).

A diferença básica entre *forward-selection* e *backward-elimination*¹⁹ é que o primeiro caso adiciona variáveis e o segundo remove variáveis. Essas inserções/remoções acontecem conforme o critério das significâncias das variáveis. No caso da regressão linear, a forma de ser feita a verificação desse critério é por meio da estatística *t*: quanto maior, mais significativa é a variável para o modelo de regressão (BRUCE; BRUCE, 2019).

Diferente da força bruta, o *stepwise* não faz uma busca exaustiva em todas as soluções possíveis e, por isso, converge mais rapidamente, conferindo mais agilidade ao pesquisador. Além disso, o *stepwise* é uma técnica bem difundida como menciona Harrell (2011), no entanto, esse autor também faz algumas ressalvas sobre o *stepwise* (ver quadro 2).

Quadro 2 – Ressalvas sobre o *stepwise*

Ressalvas de Harrell (2011)	Comentário
Produce valores de R ² tendenciosos;	O valor do R ² pode ser tendencioso/inflado quando variáveis estatisticamente significantes são selecionadas, mas que não deveriam estar juntas no modelo por serem altamente colineares ou por alguma delas ter comportamento incoerente com a realidade.
Produce “p-valores” pequenos (estatísticas t grandes), mas sem significado adequado;	É possível que seja selecionada uma variável estatisticamente significativa, mas inadequada ao modelo, por exemplo, uma variável com <i>p-valor</i> significativo, mas que apresenta gráfico de linearidade incoerente com a realidade. Nesse caso, a seleção automática do <i>stepwise</i> produziria um modelo inadequado.
Não resolve problemas causados por multicolinearidade;	Outra preocupação é a ocorrência da multicolinearidade. Como o critério de seleção do <i>stepwise</i> é a significância da variável, é possível que variáveis colineares sejam selecionadas para o modelo final, já que a significância da variável não é o critério indicado para detectar colinearidade entre variáveis.
Não permite que o avaliador pense sobre o problema;	Além disso, como o <i>stepwise</i> automatiza todo o processo, é retirada do profissional a oportunidade de realizar uma análise mais aprofundada sobre as particularidades da avaliação, das variáveis explicativas do problema e das transformações utilizadas.

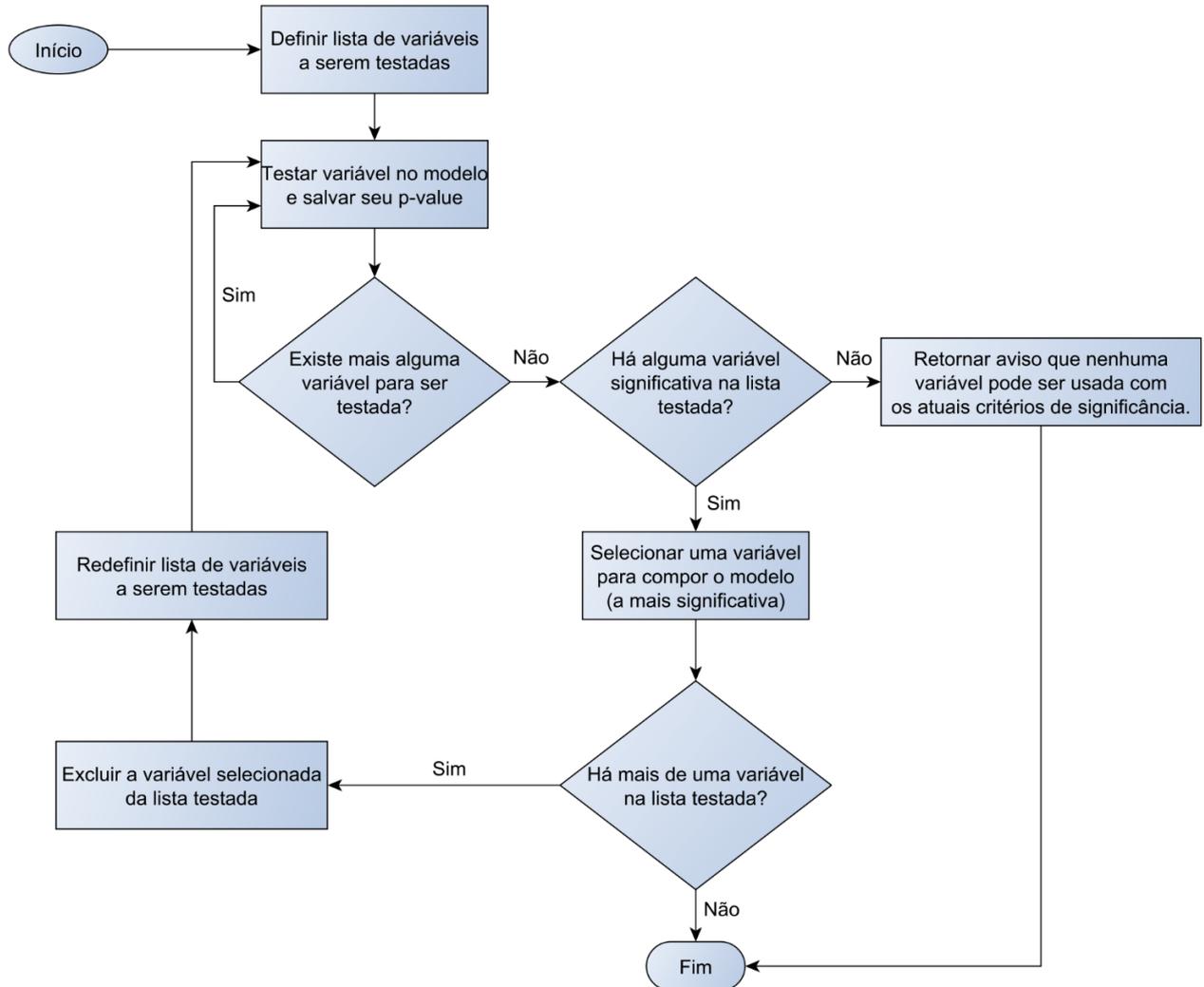
¹⁹ O tipo *backward-elimination* é o menos indicado caso o avaliador desconfie que existam variáveis colineares na sua base de dados. E isso porque esse tipo já inicia com todas as variáveis selecionadas, ou seja, o modelo já inicia com multicolinearidade.

No caso particular de avaliações imobiliárias, é frequente a utilização de transformações nas variáveis. No entanto, no *stepwise*, testar várias transformações para uma mesma variável eleva a chance da ocorrência de dois problemas:

- a) Diferentes transformações da mesma variável podem ser selecionadas para o modelo, gerando assim problemas de multicolinearidade. Para evitar isso, é necessário que o avaliador previamente analise a melhor transformação para cada variável isoladamente. Esse tipo de tratamento acaba anulando o caráter prático do *stepwise*, além disso, é desaconselhável analisar uma variável isoladamente quando se trata de regressão linear múltipla;
- b) As variáveis colineares tendem a concorrer entre si para entrar no modelo conforme consta em Harrell (2011) e nem sempre a melhor transformação é selecionada. Isso acontece porque a melhor transformação não apresenta necessariamente a maior estatística t , isto é, não é apenas essa métrica que define a melhor transformação. Há outros fatores a serem analisados, por exemplo: sua correlação com as demais variáveis (independentes e dependente), sua linearidade, sua coerência com a realidade etc.

Para ilustrar o funcionamento do *stepwise*, é apresentado o fluxograma da figura 13.

Figura 13 – Fluxograma do *stepwise (forward-selection)*.



4.3.DESCRICÃO DO MÉTODO

Neste item são descritos os procedimentos metodológicos adotados neste trabalho com relação à produção do modelo de RLC.

4.3.1 Definição de parâmetros da RLC

O primeiro parâmetro a ser definido é a significância para o cálculo do intervalo de confiança. O item A.10 da NBR 14653-2 (2011) orienta o valor de 20% para a significância, pois determina intervalo de confiança de 80%. O segundo parâmetro é o valor mínimo do R^2 dos modelos de RLC. Esse não é um parâmetro normativo, mas sim do *software E-Evaluation*. Tal definição afeta

a etapa descrita em 4.3.4.2²⁰ pois o R^2 é a métrica usada para a seleção dos melhores modelos de RLC, sendo assim, a alteração desse parâmetro pode excluir ou incluir diferentes modelos. Neste trabalho, é considerado um R^2 mínimo de 80%. O terceiro parâmetro é um número definido para limitar a quantidade de modelos armazenados durante a etapa descrita em 4.3.4.2. Os melhores modelos estão entre aqueles com maior R^2 , desse modo, não há motivo para gravar todos os modelos possíveis. Neste trabalho são gravados até 100 modelos.

4.3.2 Definição das transformações a serem testadas

Essa é a etapa em que são definidas quais transformações são testadas na busca pelo melhor modelo de RLC. As transformações comumente testadas são: x^2 ; \sqrt{x} ; $\ln(x)$; $1/x$; $1/x^2$; $1/\sqrt{x}$. Lembrando que a variável pode também ser mantida em sua escala original e que a NBR 14653-2 (ABNT, 2011) destaca que é preferível a utilização de transformações mais simples.

4.3.3 Abordagem utilizada para buscar os melhores modelos RLC

Tendo em vista as informações apresentadas no item 4.2 sobre as abordagens força-bruta e *stepwise*, o quadro 3 apresenta uma comparação entre alguns aspectos das duas abordagens.

Quadro 3 - Comparação básica entre força bruta e *stepwise*

Força bruta	<i>Stepwise</i>
Todos os modelos são verificados.	Não verifica todos os modelos.
Maior tempo de processamento.	Menor tempo de processamento.
Menor praticidade.	Maior praticidade.
Parte do processo é guiado pela máquina.	Todo o processo é guiado pela máquina.
Requer maior esforço analítico do avaliador na escolha das variáveis a serem utilizadas.	Não requer esforço analítico do avaliador na definição das variáveis utilizadas.

Neste trabalho é utilizado a força bruta para a seleção de variáveis. Essa decisão foi tomada pois é essencial encontrar o melhor modelo de regressão, o qual será tomado como referência para o estudo de Redes Neurais. O uso da força bruta é mais conservador e garante a verificação de todas os modelos possíveis, ainda que, para isso, seja necessário um maior esforço computacional.

²⁰ Etapa onde ocorre a busca pelos melhores modelos de RLC.

O *stepwise* não foi escolhido por causa dos problemas já citados no item 4.2. Acerca dessa abordagem, Judd, McClelland e Ryan (2017) consideram que a regressão *stepwise* e suas variantes apresentam a promessa sedutora de encontrar o melhor modelo sem, praticamente, nenhum esforço do pesquisador ou analista de dados. No entanto, o fascínio é enganoso, pois é possível mostrar que, para preditores redundantes, o procedimento *stepwise* nem sempre encontrará o melhor modelo. Quando os autores usam o termo “preditores redundantes” estão fazendo referência às variáveis colineares. Esse problema de multicolinearidade, em *stepwise*, dificulta a seleção de um bom modelo de regressão. Ainda ressaltam que a sua maior vantagem que era o menor tempo de processamento só se justificava quando os computadores ainda não tinham a eficiência de hoje em dia. Assim utilizar um método automatizado é preferível utilizar algum que permita verificar todas as combinações possíveis.

Neste trabalho, portanto, a tarefa de buscar os melhores modelos de RLC é automatizada por meio de um *software* desenvolvido no âmbito deste trabalho, que faz a busca por força bruta. De modo geral, essa busca pode ser dividida em duas partes:

- a) Definição das transformações aplicáveis à cada variável;
- b) Seleção dos melhores modelos de RLC.

4.3.3.1 Definição das transformações aplicáveis à cada variável

Essa etapa é importante para definir quais variáveis podem assumir as transformações definidas no item 4.3.2. Por exemplo, uma variável que possua algum dado com valor igual a 0 não pode assumir uma transformação logarítmica. Então, o *software* faz a verificação dos dados de cada variável para aplicar apenas as transformações possíveis. O resultado dessa verificação é uma lista (*lista_tr_indice*) contendo números que representam²¹ as transformações a serem testadas em cada variável. A descrição a seguir ajuda a compreender: por exemplo, um modelo com 5 variáveis (1 dependente e 4 independentes) com *lista_tr_indice* = [[1, 4, 5], [1, 4, 5], [1, 4, 5], [1], [1]] significa que a variável dependente e as duas primeiras independentes podem assumir as transformações: 1, 4 e 5; já as duas últimas podem assumir apenas a escala original: 1.

O algoritmo que define a “*lista_tr_indice*” pode ser observado no apêndice H.

²¹ No código, cada número representa uma escala de transformação: 1: Escala original; 2: x^2 ; 3: \sqrt{x} ; 4: $\ln(x)$; 5: $1/x$; 6: $1/x^2$; 7: $1/\sqrt{x}$.

4.3.3.2 Seleção dos melhores modelos de RLC

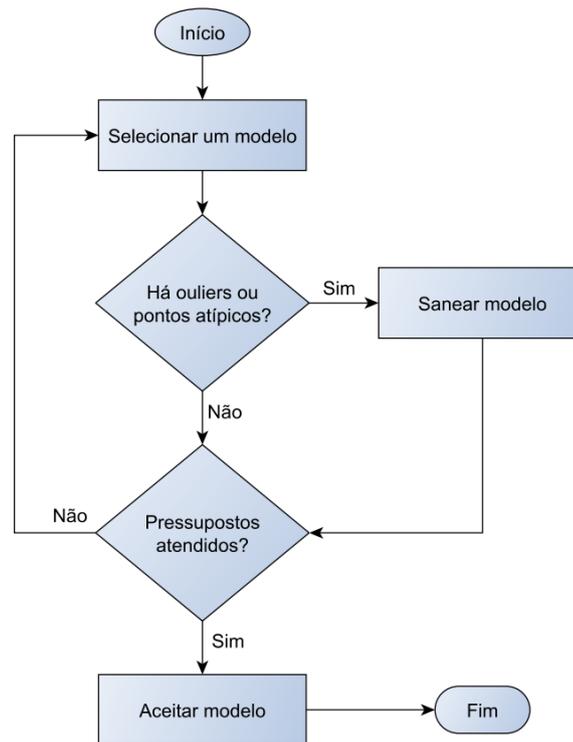
Nessa etapa, os modelos de RLC são testados um a um. Cada modelo é calculado e seu R^2 é tomado como métrica de avaliação. As combinações possíveis são obtidas usando a função *product*, um iterador²² combinatório nativo do Python. Assim, para cada modelo de RLC, é calculado o valor do R^2 , que é utilizado para filtrar quais modelos são aceitos e adicionados à lista final dos melhores resultados. O código para realizar essa seleção assim como o código para obtenção do R^2 e RMSE contam no apêndice H.

4.3.4 Verificação dos pressupostos

Esse item trata sobre a verificação dos pressupostos básicos do modelo definido. O *software E-evaluation*, desenvolvido nesta pesquisa, busca e armazena os melhores modelos de regressão, filtrando e ordenando-os com base no R^2 . No entanto, em última instância, a definição do modelo final é responsabilidade do pesquisador, de modo que não basta apenas escolher o modelo com o maior R^2 (o que pode até funcionar em alguns casos), é necessário considerar, também, os pressupostos citados no item 4.1. Desse modo, nessa etapa, o fluxograma da figura 14 pode ser usado como instrução para a definição do modelo.

22 <<https://docs.python.org/3/library/itertools.html>> Acesso em 21 de outubro de 2021.

Figura 14 – Fluxograma básico para verificação dos pressupostos em RLC.



4.3.4.1 Verificação da ausência de outliers e pontos atípicos

Essa deve ser a primeira verificação realizada, pois para garantir os pressupostos da RLC, é necessário haver um modelo saneado, ou seja, sem a interferência de *outliers* e/ou pontos atípicos.

Neste trabalho, a presença de *outliers* é verificada por meio do gráfico de resíduos padronizados *versus* valores ajustados. Nesse gráfico, são observados quais pontos destoam dos demais, conforme explicado no item 4.1.6. A presença de pontos influenciantes é verificada pelos valores da distância de Cook, que não devem exceder 1,00. Basta, então, observar o gráfico das distâncias de Cook e checar se algum elemento ultrapassa esse limite.

Os valores dos resíduos padronizados e das distâncias de Cook são calculados como consta no código que está no apêndice H. O *software* retorna esses valores e destaca resíduos absolutos maiores que 2,00 e pontos influenciantes maiores que 1,00.

4.3.4.2 Verificação da Normalidade

Neste trabalho, inicialmente, é realizada a verificação visual. Para isso, é observado se o histograma dos resíduos padronizados guarda semelhança com a curva normal e se existe aderência entre os pontos e a bissetriz no gráfico Q-Q. Essas verificações visuais são rápidas e intuitivas, no

entanto, a subjetividade do avaliador pode interferir no processo de verificação, por isso, verificações numéricas também são realizadas para ratificar ou refutar as verificações visuais.

São realizadas as seguintes verificações numéricas: a comparação entre as frequências do histograma e as de referência de uma curva normal e; a utilização do teste de Kolmogorov-Smirnov para verificação da normalidade.

O teste de Kolmogorov-Smirnov é empregado para verificar se há diferenças entre duas distribuições, sendo que a hipótese nula é não haver diferença entre as funções de distribuição das duas amostras comparadas, ou seja, há normalidade (Kolmogorov–Smirnov Test, 2008).

Neste trabalho, o teste de Kolmogorov–Smirnov é realizado por meio da função “*stats.kstest*” da biblioteca *Scipy*²³ como é apresentado no código que consta no apêndice H.

4.3.4.3 Verificação da Homocedasticidade

Neste trabalho, a verificação é realizada por meio da observação gráfica dos resíduos padronizados *versus* os valores ajustados e por meio de testes²⁴ estatísticos. São utilizados os testes de White e Breusch-Pagan²⁵ para verificação da homocedasticidade. Foi incluído Breusch-Pagan como opção porque a significância estatística no teste de White pode estar vinculada a erros de especificação do modelo em vez de um comportamento de fato heterocedástico (GUJARATI; PORTER, 2011). Ou seja, é possível que um modelo diagnosticado por White como heterocedástico não tenha, de fato, esse comportamento, mas esteja apenas com problemas de especificação. Dessa maneira, é interessante a presença de outro teste para garantir boas interpretações.

Neste trabalho, os testes de White e Breusch-Pagan são realizados por meio da biblioteca *StatsModels*²⁶ como apresentado no apêndice H.

4.3.4.4 Verificação da não Multicolinearidade

Neste trabalho é realizada a verificação gráfica (resíduos padronizados *versus* variáveis independentes), apesar de não requerida pela NBR 14653-2. Além disso, são observadas as correlações, o índice condicional e o VIF. A matriz de correlação, o índice condicional e o VIF podem

23 <<https://scipy.org/>>. Acesso em 01 de janeiro de 2022.

24 Devido às limitações mencionadas no item 4.1.3 sobre o teste de Park, ele não é utilizado neste trabalho.

25 É utilizada a versão corrigida do teste. Essa versão elimina o erro no nível de significância caso os resíduos do modelo não estejam distribuídos de modo normal (KOENKER, 1981).

26 <<https://www.statsmodels.org/>>. Acesso em 08 de janeiro de 2022.

ser calculados por meio das bibliotecas *Pandas*²⁷, *Numpy*²⁸ e *StatsModels* como apresentado no apêndice H.

4.3.4.5 Verificação da não Autocorrelação

A única forma citada pela NBR 14653-2 (ABNT, 2011) para detectar a autocorrelação é o exame do gráfico de resíduos padronizados *versus* valores ajustados. Como a referida norma não cita nenhum teste para a verificação da autocorrelação, esse trabalho utiliza o teste de Durbin-Watson e o de Breusch-Godfrey.

O teste de Durbin-Watson é o mais conhecido procedimento para a verificação da autocorrelação (KRÄMER; SONNBERGER, 1986). Nesse teste é calculado o parâmetro “ d_w ” de Durbin-Watson, que indica a não ocorrência de autocorrelação se for próximo de “2” (GUJARATI; PORTER, 2011; ASTERIOU; HALL, 2011). O quadro 4 resume a utilização desse teste, onde d_L e d_U são valores tabelados por Durbin-Watson.

Quadro 4 - Condições para verificação da autocorrelação pelo teste de Durbin-Watson

Hipótese nula – H0	Decisão	Condição
Não há autocorrelação positiva	Rejeitar	$0 < d_w < d_L$
Não há autocorrelação positiva	Zona de indecisão	$d_L \leq d_w \leq d_U$
Não há autocorrelação negativa	Rejeitar	$4 - d_L < d_w < 4$
Não há autocorrelação negativa	Zona de indecisão	$4 - d_U \leq d_w \leq 4 - d_L$
Nenhuma autocorrelação (positiva e negativa)	Não rejeitar	$d_U < d_w < 4 - d_U$

Fonte: Adaptado de (GUJARATI; PORTER, 2011)

Apesar de *Durbin-Watson* ser mais conhecido e mais usado, apresenta limitações²⁹, por esse motivo é preferível utilizar *Breusch-Godfrey*, que consegue superar essas limitações (GUJARATI; PORTER, 2011; ASTERIOU; HALL, 2011).

A hipótese nula do teste de *Breusch-Godfrey* é que não ocorre autocorrelação serial (ASTERIOU; HALL, 2011). Dessa maneira, o que se deseja é aceitar a hipótese nula para um determinado nível de significância. Neste trabalho, os testes citados são realizados como apresentado no apêndice H.

27 <<https://pandas.pydata.org/>>. Acesso em 12 de janeiro de 2023.

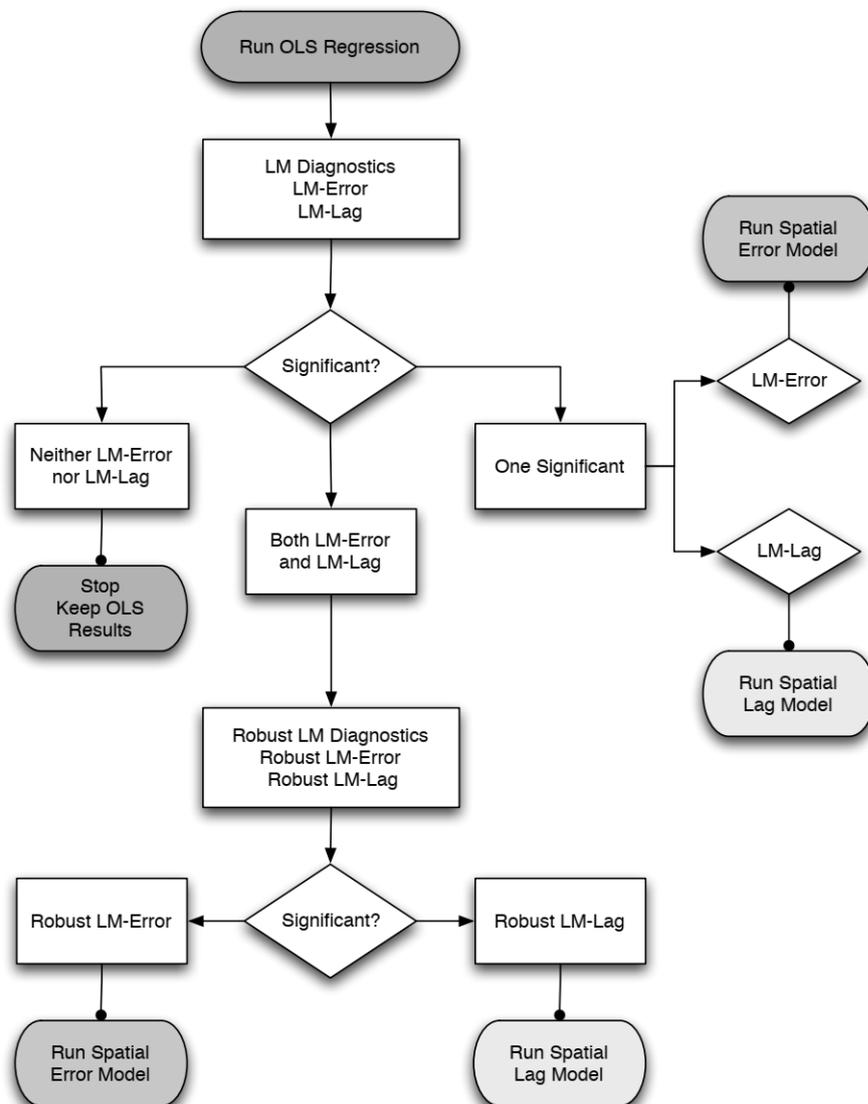
28 <<https://numpy.org/>>. Acesso em 08 de janeiro de 2022.

29 a) pode fornecer resultados inconclusivos; b) não é aplicável quando é utilizada variável dependente defasada; c) não considera ordens mais altas de correlação serial (ASTERIOU; HALL, 2011).

Por fim, também é verificada a autocorrelação espacial por meio da plotagem espacial dos resíduos. E para ratificar essa observação, é realizada também a verificação de dependência espacial usando as estatísticas de Moran, *LM-error* e *LM-lag* calculadas pelo *software* Geoda.

O manual do Geoda fornece um fluxograma para a tomada de decisão (ver figura 15). Primeiramente, é calculado um modelo RLC tradicional e são observados os valores das métricas não robustas “*LM-error*” e “*LM-lag*”; caso nenhum dos dois seja estatisticamente significativo, o modelo tradicional pode ser utilizado, caso contrário, um modelo espacial deve ser utilizado (ANSELIN, 2005).

Figura 15 – Fluxograma do “*Geoda work book*” para casos de Regressão Espacial



Fonte: Anselin (2005)

4.3.4.6 Verificação da Linearidade

Neste trabalho, a verificação da linearidade é realizada pela observação dos gráficos da variável dependente *versus* variáveis independentes e por testes de linearidade. A única forma de verificação citada pela NBR 14653-2 (ABNT, 2011) é a observação dos gráficos, no entanto, visando confirmar os resultados advindos da interpretação visual, esse trabalho faz os testes de linearidade usando as funções “*linear_lm*” e “*linear_rainbow*” disponíveis na biblioteca *StatsModels*.

O teste “*linear_lm*”³⁰ baseia-se no multiplicador de Lagrange e tem como hipótese nula que a especificação linear do modelo está correta (STATSMODELS, 2023b). O teste “*linear_rainbow*” (Teste Rainbown) faz uma comparação entre o ajuste da amostra completa e o ajuste de uma subamostra, onde o esperado é que o primeiro ajuste não seja significativamente inferior ao segundo, caso contrário a hipótese nula de linearidade é rejeitada (BALTAGI, 2011).

Neste trabalho, os testes de linearidade são realizados como apresentado no apêndice H.

4.4.RESULTADOS DO MODELO ESCOLHIDO DE RLC

Este item apresenta o modelo de regressão utilizado (4.4.1 a 4.4.3), as verificações que o validam (4.4.4 a 4.4.14) e a avaliação dos imóveis da amostra de teste (4.4.15).

4.4.1 Combinação utilizada

A busca automática das melhores combinações de transformações das variáveis é realizada pelo método da força bruta. Das opções fornecidas, foi escolhida a seguinte combinação: valor total (VT) em logaritmo natural; área em logaritmo natural; WC em logaritmo natural; suítes na escala original; vagas na escala original; padrão em logaritmo natural; idade na escala original; distância em logaritmo natural; norte na escala original.

4.4.2 Coeficientes da regressão

Os sinais dos coeficientes apresentados na tabela 2 estão coerentes, pois obedecem à lógica esperada e já descrita no item 3.3.

³⁰ É o mesmo teste calculado pelo *software* de econometria *Gretl* <<http://gretl.sourceforge.net/>>. Acesso em 10 de janeiro de 2022.

Tabela 2 – Coeficientes da regressão

Parâmetro	Variável	Valor
b_0	Intercepto	11,741892
b_1	Área	0,553606
b_2	WC	0,268195
b_3	Suítes	0,032717
b_4	Vagas	0,082003
b_5	Padrão	-0,304688
b_6	Idade	-0,014186
b_7	Distância	-0,101334
b_8	Norte	0,000111

4.4.3 Função de regressão e de estimação

A função de regressão fornece a saída na escala logarítmica. Já a função de estimação retorna o valor monetário (escala original).

Função de regressão:

$$LN(VT) = (1.174e+01) + (5.536e-01 * LN(ÁREA)) + (2.682e-01 * LN(WC)) + (3.272e-02 * SUITE) + (8.200e-02 * VAGA) + (-3.047e-01 * LN(PADRAO)) + (-1.419e-02 * IDADE) + (-1.013e-01 * LN(DIST)) + (1.111e-04 * Norte)$$

Função de estimação (R\$):

$$VT = EXP((1.174e+01) + (5.536e-01 * LN(ÁREA)) + (2.682e-01 * LN(WC)) + (3.272e-02 * SUITE) + (8.200e-02 * VAGA) + (-3.047e-01 * LN(PADRAO)) + (-1.419e-02 * IDADE) + (-1.013e-01 * LN(DIST)) + (1.111e-04 * Norte))$$

4.4.4 Poder de explicação do modelo

O coeficiente de determinação³¹ (R^2) é uma métrica comumente utilizada para se avaliar a qualidade de modelos de regressão. Segundo observado na tabela 3, é possível concluir que, 96,39% das variações em relação à média são explicadas pelo modelo proposto; além disso, o coeficiente de correlação (R) igual a 98,18% indica uma relação forte entre a variável dependente e as independentes.

31 Obtido pela razão entre a parte “explicada” e o “total” da ANOVA (tabela 6). É preferível modelos que consigam explicar muito de sua variação total.

Tabela 3 – Poder de explicação do modelo

Coefficientes	Valor
Coefficiente de Determinação (R^2)	0,9639
Coefficiente de Determinação Ajustado (R_a^2)	0,9620
Coefficiente de Correlação (R)	0,9818

4.4.5 Resíduos padronizados e Distâncias de Cook

Os resíduos padronizados plotados na figura 16 indicam não haver *outliers*, visto que os resíduos apresentados pertencem à mesma nuvem de pontos, sem valores que destoam dos demais. As distâncias de Cook da figura 17 indicam não haver pontos influenciantes no modelo, visto que os valores obedecem aos limites aceitáveis (menor que 1,00).

Figura 16 – Resíduos padronizados x valores ajustados

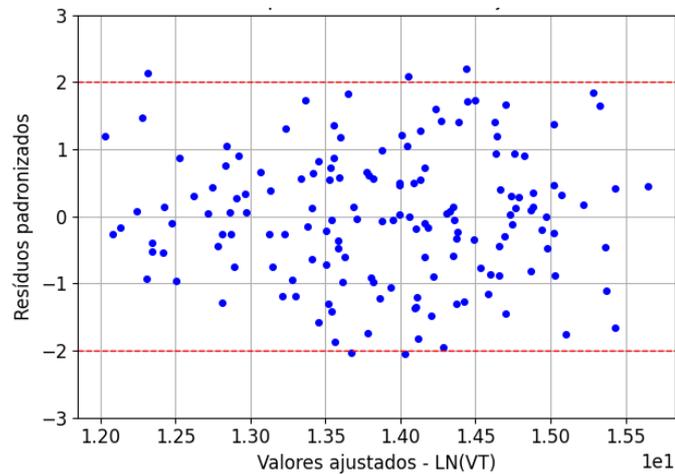
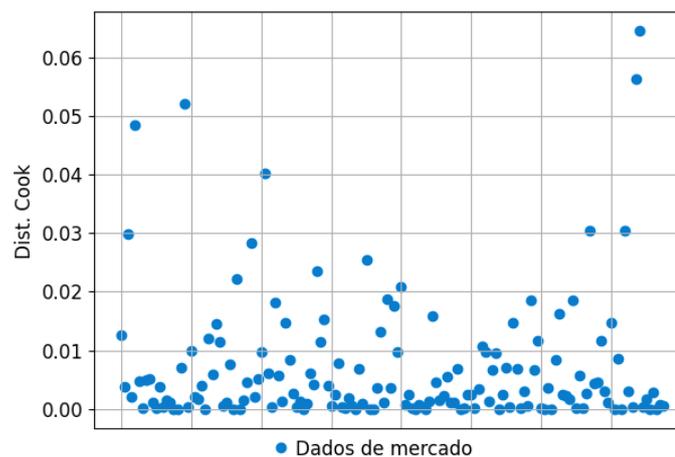


Figura 17 – Distâncias de Cook



Os valores dos resíduos padronizados e distâncias de Cook são apresentados no apêndice C.

4.4.6 Correlações

As correlações podem ser observadas nas tabelas 4 e 5. A tabela 4 apresenta os coeficientes entre VT e as variáveis explicativas, onde é observado que os sinais desses coeficientes são coerentes com a lógica esperada e já descrita no item 3.3.

Tabela 4 – Correlações entre variável dependente e independentes

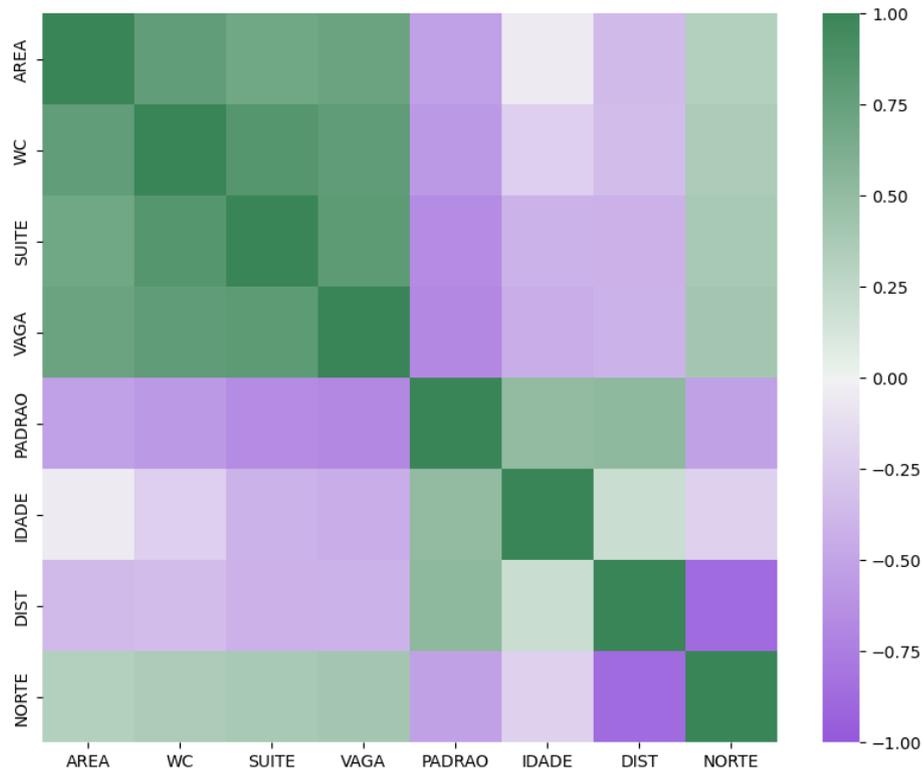
	Área	WC	Suítes	Vagas	Padrão	Idade	Distância	Norte
VT	0,88	0,84	0,83	0,85	-0,79	-0,44	-0,55	0,58

As correlações entre as variáveis independentes são apresentadas na tabela 5 e figura 18. Nesse caso, o desejável é evitar que as correlações sejam altas. A figura 18 ilustra a matriz de correlações da tabela 5, onde é possível notar que as maiores correlações ocorrem entre as variáveis: “Área, WC, Suítes e Vagas” e entre “Distância e Norte”, com destaque para o caso “Área e WC” que ultrapassa 0,80. Conforme consta na NBR 14653-2 (ABNT, 2011), valores maiores que 0,80 devem servir de alerta para a verificação da multicolinearidade.

Assim, para investigar com mais cuidado a presença da multicolinearidade, no item 4.4.11 são realizadas verificações por meio de gráficos e métricas como o VIF e IC.

Tabela 5 – Matriz de correlações

	Área	WC	Suítes	Vagas	Padrão	Idade	Dist	Norte
Área	1.00	0.85	0.72	0.77	-0.61	-0.12	-0.44	0.41
WC	0.85	1.00	0.80	0.76	-0.59	-0.25	-0.28	0.37
Suítes	0.72	0.80	1.00	0.80	-0.66	-0.41	-0.37	0.39
Vagas	0.77	0.76	0.80	1.00	-0.69	-0.45	-0.32	0.41
Padrão	-0.61	-0.59	-0.66	-0.69	1.00	0.50	0.37	-0.48
Idade	-0.12	-0.25	-0.41	-0.45	0.50	1.00	0.04	-0.21
Dist.	-0.44	-0.28	-0.37	-0.32	0.37	0.04	1.00	-0.71
Norte	0.41	0.37	0.39	0.41	-0.48	-0.21	-0.71	1.00

Figura 18 – *Heatmap* referente à matriz de correlações

4.4.7 Erro padrão (*Se*) e coeficiente de variação (*CV*)

Os valores encontrados para o Erro padrão e coeficiente de variação são:

- *Se* (R\$) igual a 337.391,02;
- *CV*: 22,41%.

Porém, a análise isolada do erro padrão pode não ser tão útil caso falte a referência da média dos valores VT analisados. No modelo estudado, a média dos VT é 1.505.358,22 e a razão entre o *Se*/VT (337.391,02/1.505.358,22) é igual a 22,41% (*CV*).

4.4.8 ANOVA e testes de significância

As informações da análise de variância (ANOVA) e os testes de significância são apresentadas nas tabelas 6, 7 e 8, respectivamente.

Tabela 6 – ANOVA do modelo de Regressão

TIPO DE VARIAÇÃO	GL	SSE	MSE
Explicada	8	1,106e+02	1,383e+01
Não explicada	147	4,141e+00	2,817e-02
Total	155	1,148e+02	7,404e-01

A ANOVA orienta o avaliador sobre a capacidade de generalização do modelo: quanto menor for a diferença entre a variação explicado e a total, melhor é o modelo e maior o seu coeficiente de determinação, visto que a razão $(1,106e+02)/(1,148e+02)$ é igual ao R^2 (0,96).

Além disso, com a ANOVA, é possível calcular a estatística F_c conforme a equação 8:

$$F_c = \frac{(MSE_{\text{explicado}})}{(MSE_{\text{nãoexplicado}})} = \frac{(1,383e + 01)}{(2,817e - 02)} = 490,9 \quad (8)$$

No caso da significância global do modelo, para fins de comparação, os valores tabelados de "F" são: 3,214 para Grau III (1%); 2,8269 para Grau II (2%); 2,3092 para Grau I (5%). Conforme a tabela 7, a estatística $F_c > F_{\text{tabelados}}$; assim, é rejeitada a hipótese nula de que nenhuma variável é importante.

Tabela 7 – Significância global do modelo

Estatística (F_c)	P-valor (α)
490,9193	0,0000%

No caso das significâncias dos parâmetros, para efeitos comparativos, os valores tabelados de " $t_{\text{tabelados}}$ " são: 1,6609 para Grau III (10%); 1,2904 para Grau II (20%); 1,0421 para Grau I (30%). Conforme os valores apresentados na tabela 8, nota-se que $|t_c| > |t_{\text{tabelados}}|$ em todos os casos. Nesse caso, a hipótese nula de que a variável não é importante é rejeitada em todos os casos.

Pela tabela 8 também é possível inferir o grau de significância das variáveis no modelo de regressão: a mais significativa é a área (maior t_c em módulo) e a menos é o número de suítes (menor t_c em módulo).

Tabela 8 – Significância dos parâmetros

	Área	WC	Suítes	Vagas	Padrão	Idade	Dist.	Norte
Estatística (t_c)	11,1077	4,8113	1,3592	2,7805	-6,1227	-8,9372	-6,7931	2,1220
P-valor (α)	0,0000%	0,0004%	17,6149%	0,6138%	0,0000%	0,0000%	0,0000%	3,5517%

4.4.9 Verificação da Normalidade

Essa verificação é requerida no item A.2.1.2 da NBR 14653-2 (ABNT, 2011). Graficamente, essa verificação pode ser realizada pelo histograma, histograma cumulativo e pelo gráfico Q-Q.

No modelo analisado, é possível notar comportamentos que indicam a normalidade: o histograma dos resíduos padronizados guarda semelhança com a curva da distribuição normal (figura 19), a distribuição cumulativa do histograma assemelha-se com a distribuição normal acumulada (figura 20) e os pontos observados no gráfico Q-Q estão ajustados à bissetriz (figura 21).

Figura 19 – Histograma × Curva Normal

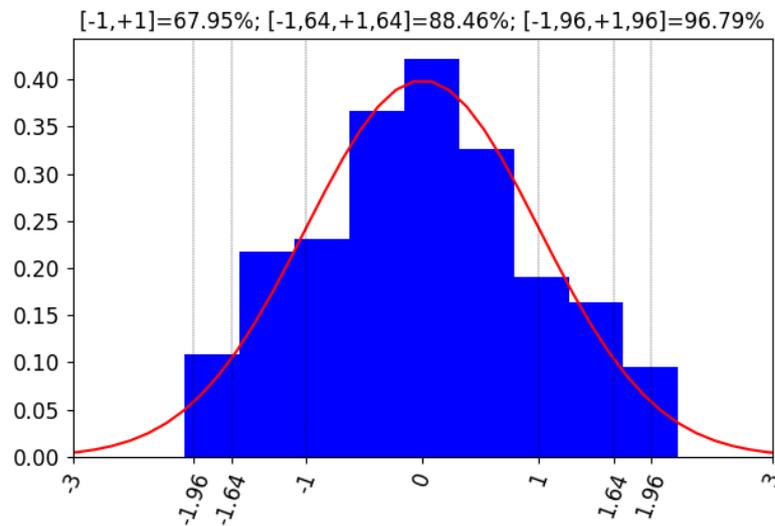


Figura 20 – Histograma cumulativo

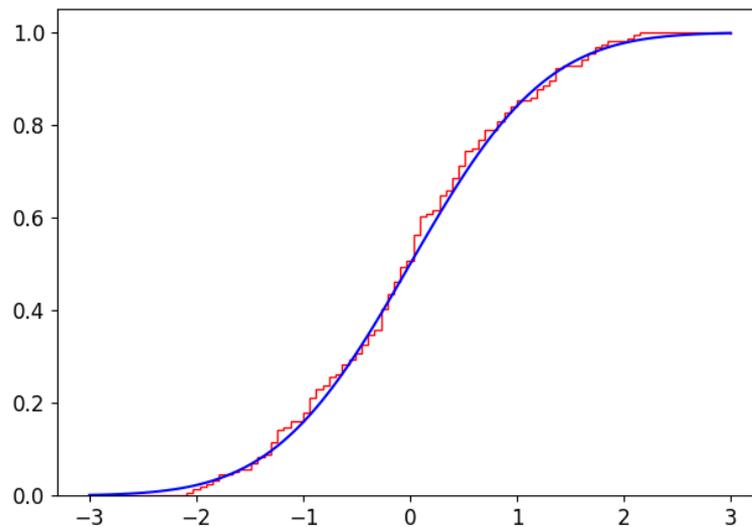
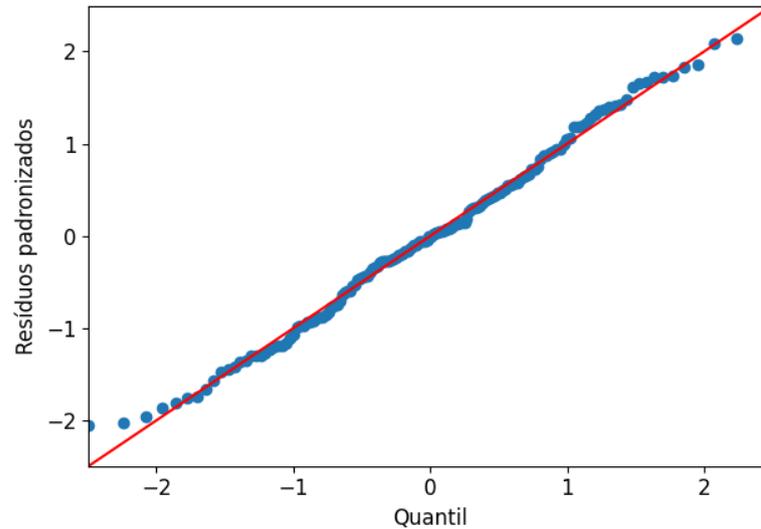


Figura 21 – Q-Q Plot



Outra verificação realizada é a comparação das porcentagens dos resíduos nos intervalos $[-1,00; +1,00]$, $[-1,64; +1,64]$ e $[-1,96; +1,96]$ com as faixas recomendadas pela NBR 14653-2 (ABNT, 2011). As porcentagens observadas no modelo e as teóricas de uma curva normal podem ser observadas na tabela 9, onde é possível notar pequenas diferenças, indicando comportamento normal.

Tabela 9 – Porcentagens observadas versus teóricas

Intervalo	Porcentagem		Diferença absoluta
	Observada	Teórica	
$[-1,00; +1,00]$	67,95%	68,00%	0,05%
$[-1,64; +1,64]$	88,46%	90,00%	1,54%
$[-1,96; +1,96]$	96,79%	95,00%	1,79%

Por fim, o teste de normalidade de Kolmogorov–Smirnov pode ser observado na tabela 10. A hipótese nula é que não há diferença entre as funções de distribuição das duas amostras comparadas (KOLMOGOROV–SMIRNOV TEST, 2008). É comparada a amostra dos resíduos e uma com distribuição normal. O resultado é que, para um nível de significância de 5% ($0,9245 > 0,05$), não é possível rejeitar a hipótese nula, portanto é aceita a condição de normalidade dos resíduos padronizados.

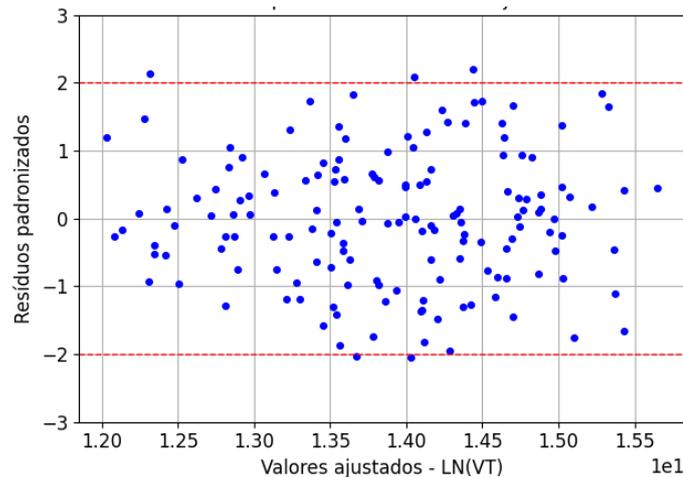
Tabela 10 – Teste de normalidade

Estatística/Significância	Teste
	Kolmogorov–Smirnov
Estatística	0,0429
P-valor (α)	0,9245

4.4.10 Verificação da Homocedasticidade

Essa verificação é requerida no item A.2.1.3 da NBR 14653-2 (ABNT, 2011). Graficamente, a averiguação pode ser realizada plotando os resíduos padronizados *versus* valores ajustados (figura 22). Nessa figura, é possível notar que os pontos do gráfico se distribuem de modo aleatório (sem padrão definido), comportamento que indica homocedasticidade no modelo.

Figura 22 – Resíduos padronizados x Valores ajustados



Além da verificação gráfica, é recomendada a realização de testes para a confirmação da homocedasticidade. Desse modo, são apresentados os testes³² de White e de Breusch-Pagan (tabela 11), onde é possível notar que, a um nível de significância de 5%, a condição homocedástica do modelo é comprovada, já que a hipótese nula não pode ser negada ($0,7843 > 0,05$ e $0,6538 > 0,05$).

Tabela 11 – Testes de homocedasticidade

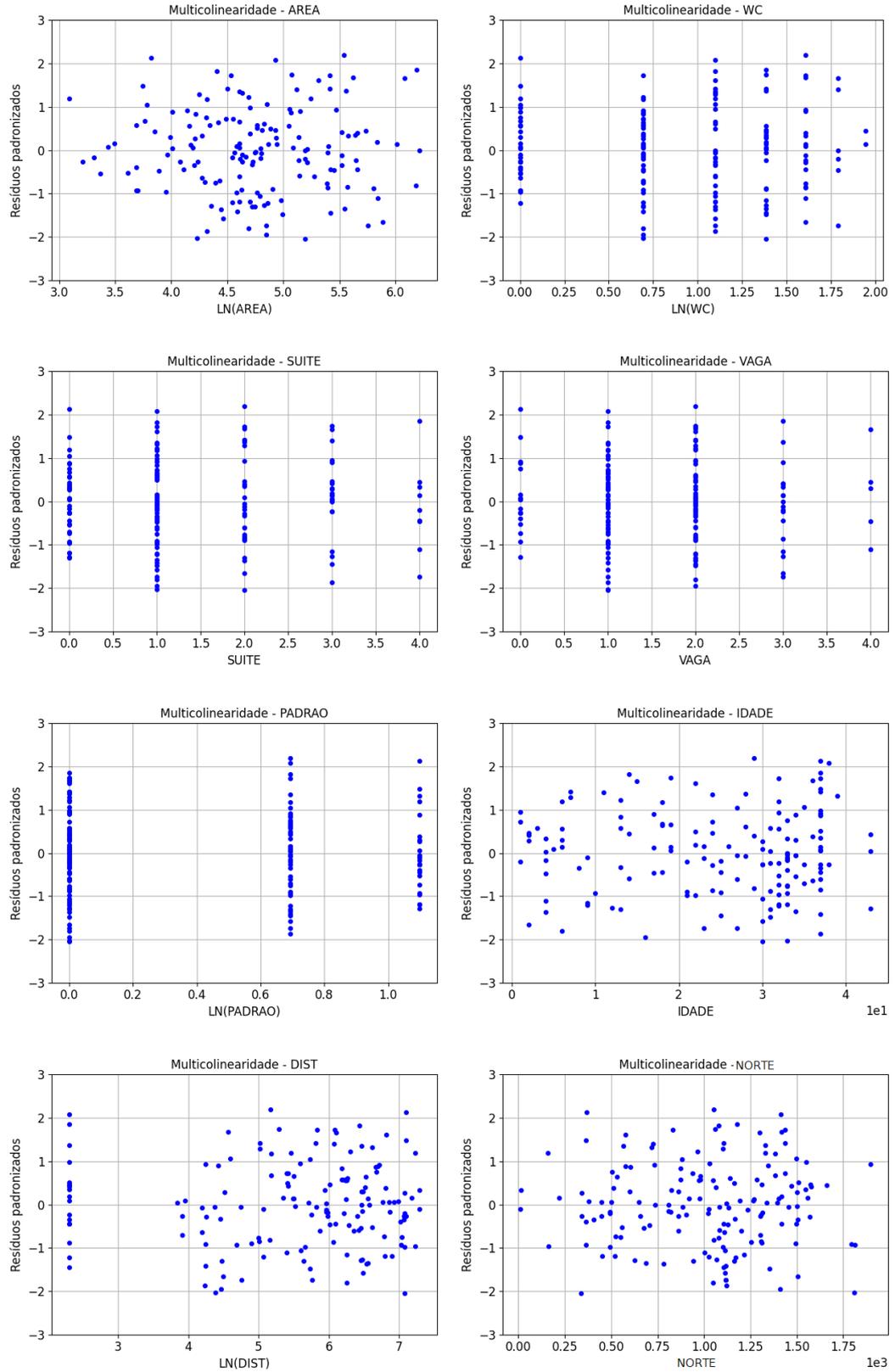
Estatística/Significância	Teste	
	White	Breusch-Pagan
Estatística	36,426	5,9418
P-valor (α)	0,7843	0,6538

4.4.11 Verificação da não Multicolinearidade

Graficamente, a verificação de multicolinearidade é realizada com auxílio da figura 23.

³² Para esses dois testes, a hipótese nula é que os erros são homocedásticos (STATSMODELS, 2023a).

Figura 23 – Verificação gráfica da Multicolinearidade



Nesse tipo de análise, a multicolinearidade é percebida quando os gráficos de variáveis diferentes guardam semelhanças entre si (ABUNAHMAN, 2008). Ou seja, se os resíduos de diferentes variáveis apresentam comportamento semelhante, isso pode indicar que essas variáveis contribuem de modo similar para explicar as mudanças na variável dependente.

Conforme a matriz de correlações apresentada no item 4.4.6, a correlação que inspira uma análise mais cuidadosa é da Área *versus* WC. No entanto, pela figura 23 não é possível concluir se há ou não problemas de multicolinearidade entre essas variáveis. Isso ocorre devido à natureza discreta da variável WC, que varia de unidade em unidade, diferentemente da variável contínua Área.

É possível perceber que a verificação gráfica é intuitiva, porém abre espaço para subjetividade, pois a avaliação de equivalência entre os gráficos é realizada apenas visualmente e não há como mensurar se a equivalência visual caracteriza um grau elevado de multicolinearidade. Por isso, devem ser utilizadas métricas mais objetivas como VIF e IC para avaliar o grau da multicolinearidade.

O VIF de cada variável podem ser observados na tabela 12. Todos os valores são menores que 5 ou 10, caracterizando um grau de multicolinearidade dentro dos limites recomendados na literatura técnica. Os resultados dos VIF ratificam o que foi observado na matriz de correlações, isto é, as maiores correlações de fato ocorrem entre as variáveis Área e WC (comportamento já esperado do mercado).

Tabela 12 – VIF para cada variável

Parâmetro	Área	WC	Suítes	Vagas	Padrão	Idade	Distância	Norte
VIF	5,87	5,45	4,22	4,39	2,58	1,87	2,53	2,44

Já o valor encontrado para o Índice Condicional (IC) foi 68,7. Existe aqui uma incoerência entre os resultados dos VIF e do IC: os resultados dos VIF indicam grau de multicolinearidade dentro dos limites aceitáveis, já o IC não. Nesse caso, é utilizado o “*centering*” (técnica mencionada no item 4.1.5) e verificado se o IC sofre alteração.

A centralização é feita nas variáveis Área, WC, Suítes e Vagas, pois são as que apresentam os maiores valores de VIF. Os resultados obtidos no *software* Gretl são apresentados na figura 24. É possível observar que o IC é alterado para 25,79. Nesse caso, a multicolinearidade removida foi apenas a não essencial (a parcela essencial permanece) e o IC encontrado indica que a multicolinearidade existente não é grave. Portanto, os resultados do VIF e IC estão coerentes e dentro dos limites aceitáveis.

Figura 24 – VIF e IC calculados no *Gretl* após o uso do “centering”

Fatores de Inflacionamento da Variância (VIF)
 Valor mínimo possível = 1,0
 Valores > 10,0 podem indicar um problema de colinearidade

AREA	5.865
WC	5.446
SUITE	4.224
VAGA	4.387
PADRAO	2.583
IDADE	1.870
DIST	2.535
Norte	2.443

VIF(j) = 1/(1 - R(j)^2), onde R(j) é o coeficiente de correlação múltipla entre a variável j e a outra variável independente

Diagnósticos de colinearidade de Belsley-Kuh-Welsch:

lambda	cond	const	AREA	WC	SUITE	VAGA	PADRAO	IDADE	DIST	Norte
4.447	1.000	0.000	0.002	0.002	0.003	0.003	0.009	0.004	0.001	0.001
3.413	1.141	0.000	0.009	0.010	0.012	0.012	0.000	0.001	0.000	0.002
0.367	3.482	0.001	0.143	0.076	0.060	0.082	0.060	0.024	0.002	0.010
0.253	4.190	0.000	0.100	0.077	0.334	0.038	0.182	0.004	0.001	0.021
0.201	4.703	0.000	0.001	0.017	0.083	0.593	0.256	0.004	0.002	0.031
0.146	5.519	0.001	0.099	0.485	0.370	0.000	0.078	0.052	0.027	0.001
0.103	6.557	0.001	0.032	0.087	0.043	0.007	0.302	0.087	0.060	0.217
0.063	8.401	0.006	0.566	0.146	0.011	0.264	0.100	0.758	0.039	0.002
0.007	25.793	0.991	0.048	0.100	0.084	0.002	0.014	0.066	0.868	0.715

4.4.12 Verificação da não Autocorrelação

Conforme consta na NBR 14653-2 (2011), a forma gráfica de investigar a autocorrelação é verificando se o gráfico de resíduos *versus* valores ajustados é formado por pontos dispostos de modo aleatório, condição essa que caracteriza a não autocorrelação. Tal condição já foi verificada na figura 22.

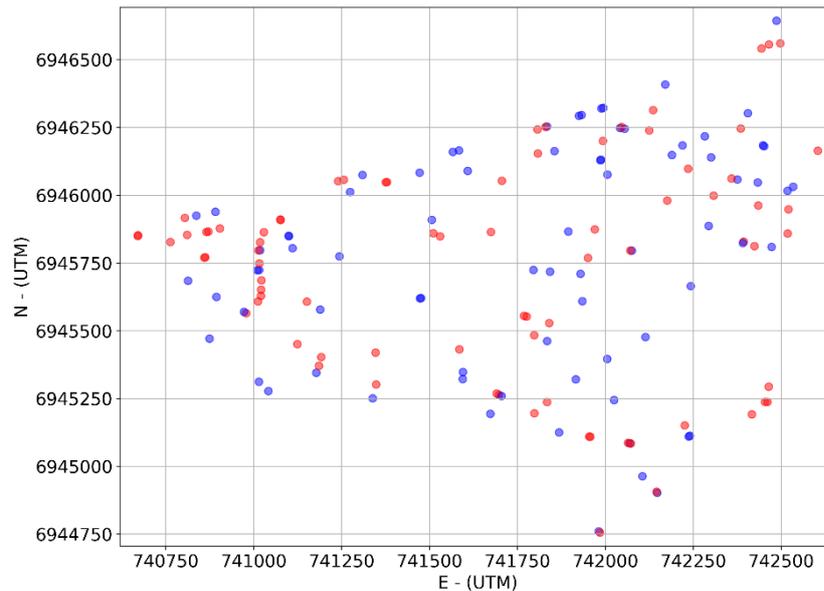
Além da verificação gráfica, são realizados testes de *Durbin-Watson* e *Breusch-Godfrey*. O teste de *Durbin-Watson* ($dw = 1,9905$), com dw próximo a 2,00 indica ausência de autocorrelação e o teste de *Breusch-Godfrey* (tabela 13) que apresenta resultados que indicam, a um nível de significância de 5%, que não é possível rejeitar a hipótese nula de não autocorrelação.

Tabela 13 – Teste de autocorrelação

Estatística/Significância	Breusch-Godfrey
Estatística	10,9387
P-valor (α)	0,3623

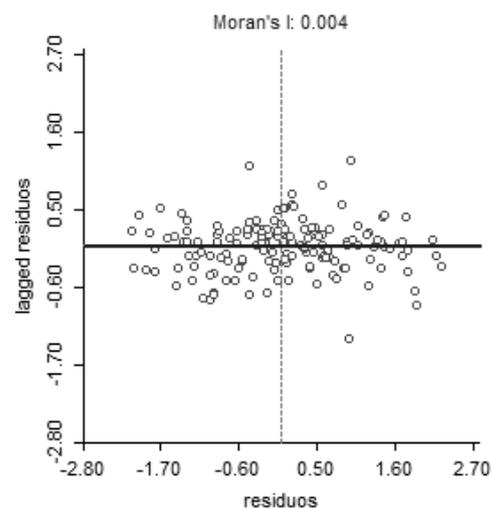
Além desses resultados, é importante investigar a existência de autocorrelação espacial. Isso pode ser realizado ao plotar espacialmente os resíduos padronizados (ver figura 25), onde o ideal é que resíduos positivos (em azul) e negativos (em vermelho) estejam distribuídos de modo aleatório.

Figura 25 – Verificação gráfica da autocorrelação espacial



Na figura anterior, é observado que existem resíduos negativos e positivos em todos os locais do mapa, indicando não haver autocorrelação espacial no modelo. No entanto, a análise puramente visual pode não favorecer a melhor interpretação, por isso, tal análise pode ser ratificada usando o índice de Moran (figura 26). O valor 0,004 indica ausência de dependência espacial, pois quanto mais próximo de 0 for o índice, menor é a correlação obtida; e o p-valor³³ referente ao índice é 0,3690, o que indica que não é possível rejeitar a hipótese nula de distribuição aleatória.

Figura 26 – Índice de Moran - RLC



³³ Obtido no *software* Geoda usando 999 permutações, *seed* igual a 10 e distância euclidiana de 212,707 metros na análise univariada de Moran.

Além disso, é realizada a verificação de dependência espacial no *software* Geoda usando as estatísticas “*LM-lag*” e “*LM-error*”, onde é notado que nenhum dos valores é significativo (tabela 14). Isso indica que não há dependência espacial, por isso o modelo RLC tradicional pode ser utilizado.

Tabela 14 – Testes para autocorrelação espacial

Estatística/Significância	Teste			
	Lagrange Multiplier (lag)	Robust LM (lag)	Lagrange Multiplier (error)	Robust LM (error)
Estatística	0,1489	0,2617	0,1743	0,2871
P-valor (α)	0,6996	0,6089	0,6763	0,5920

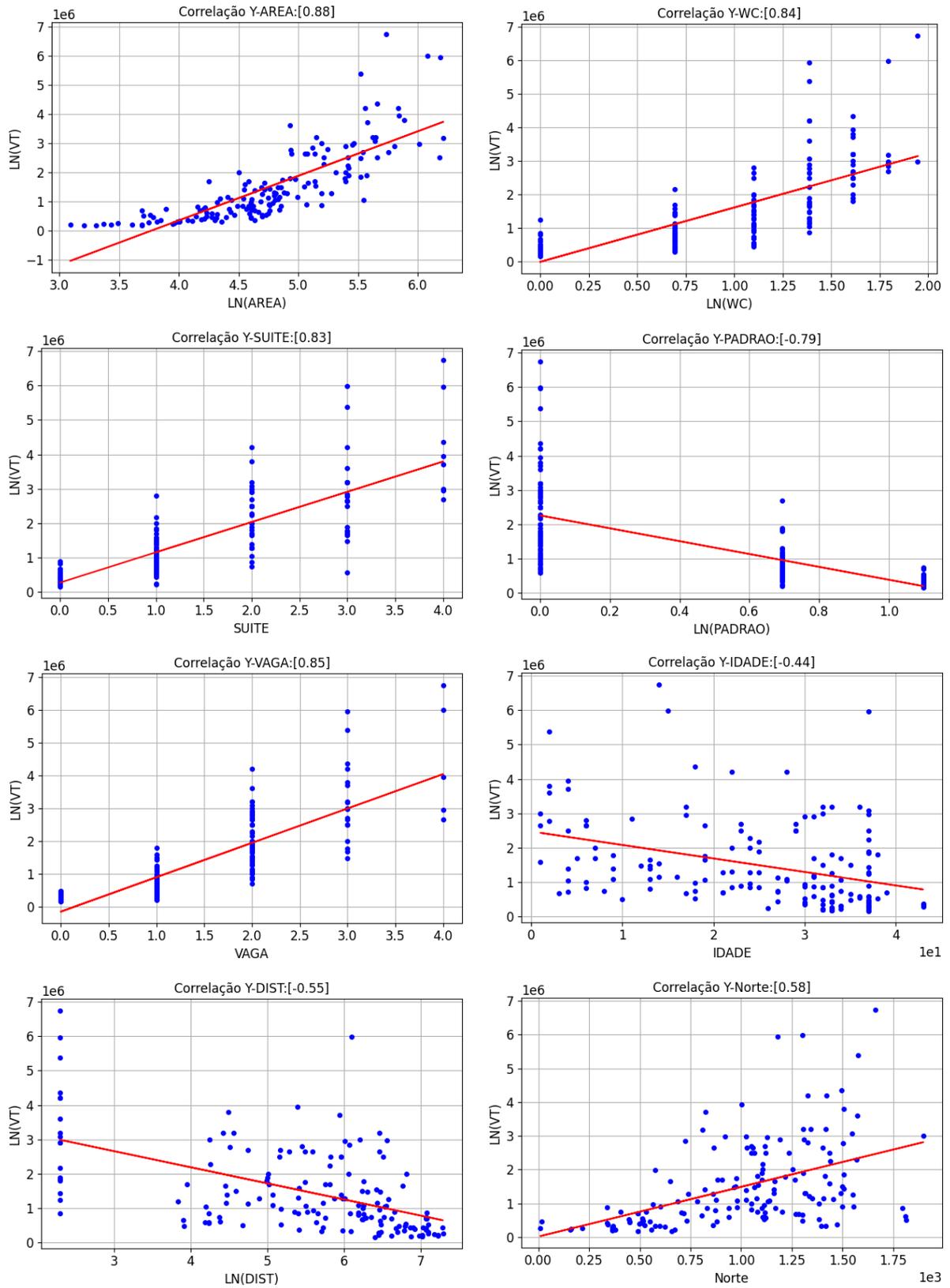
4.4.13 Verificação da Linearidade

Graficamente, a linearidade pode ser verificada pelos gráficos apresentados na figura 27, que apesar de apresentar gráficos com pontos indicando comportamento linear, é difícil garantir a linearidade apenas por verificação visual. Por isso, são apresentados testes de linearidade na tabela 15 que mostram que não se pode negar a hipótese nula da linearidade.

Tabela 15 – Testes de Linearidade

Estatística/Significância	Teste	
	LM (x^2)	Rainbow
Estatística	5,7644	0,9642
P-valor (α)	0,6736	0,5637

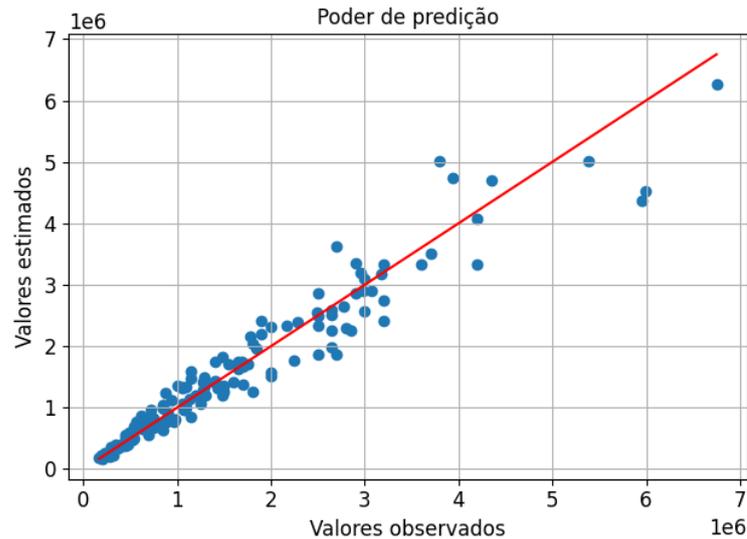
Figura 27 – Gráficos de linearidade



4.4.14 Gráfico de Predição

Na figura 28, o eixo “x” representa os valores coletados no mercado e o eixo “y” representa os valores estimados pelo modelo. Essa figura apresenta pontos aderentes à bissetriz, que é o cenário ideal, indicando assim que os valores observados foram bem estimados pelo modelo.

Figura 28 – Gráficos de Predição



4.4.15 Valores para os imóveis avaliados

Os valores médios, máximos e mínimos conforme o intervalo de confiança (IC), intervalo de predição (IP) e campo de arbítrio (CA) podem ser consultados nas tabelas 16, 17 e 18.

Tabela 16 – Valores estimados *versus* valores observados no mercado (amostra de validação)

Imóvel	Valor estimado (R\$)	Valor de Mercado (R\$)	Diferença (R\$)	Diferença (%)
I_005	2.261.638,53	2.650.000,00	-388.361,47	-17,17%
I_006	813.235,96	980.000,00	-166.764,04	-20,51%
I_008	1.617.935,88	2.099.000,00	-481.064,12	-29,73%
I_013	1.774.238,27	1.350.000,00	424.238,27	23,91%
I_015	395.752,73	425.000,00	-29.247,27	-7,39%
I_019	2.312.270,73	1.920.000,00	392.270,73	16,96%
I_023	676.354,44	550.000,00	126.354,44	18,68%
I_024	676.324,02	730.000,00	-53.675,98	-7,94%
I_026	859.792,83	849.900,00	9.892,83	1,15%
I_036	1.152.779,88	1.150.000,00	2.779,88	0,24%
I_041	2.148.463,02	1.800.000,00	348.463,02	16,22%
I_049	1.198.570,42	1.170.000,00	28.570,42	2,38%
I_064	263.332,66	230.000,00	33.332,66	12,66%

Tabela 16 – Valores estimados *versus* valores observados no mercado (amostra de validação)

Imóvel	Valor estimado (R\$)	Valor de Mercado (R\$)	Diferença (R\$)	Diferença (%)
I_072	2.762.381,77	2.400.000,00	362.381,77	13,12%
I_090	1.622.430,29	1.850.000,00	-227.569,71	-14,03%
I_092	1.483.195,20	1.480.000,00	3.195,20	0,22%
I_096	315.121,45	400.000,00	-84.878,55	-26,94%
I_099	206.625,53	259.000,00	-52.374,47	-25,35%
I_102	601.849,38	530.000,00	71.849,38	11,94%
I_107	1.919.840,03	1.725.000,00	194.840,03	10,15%
I_114	466.299,91	529.000,00	-62.700,09	-13,45%
I_122	985.099,70	1.200.000,00	-214.900,30	-21,82%
I_123	2.437.812,78	1.880.000,00	557.812,78	22,88%
I_126	782.238,05	799.000,00	-16.761,95	-2,14%
I_130	591.986,99	650.000,00	-58.013,01	-9,80%
I_131	530.834,51	459.000,00	71.834,51	13,53%
I_137	1.042.698,64	889.900,00	152.798,64	14,65%
I_138	212.516,75	220.000,00	-7.483,25	-3,52%
I_142	1.866.964,60	3.100.000,00	-1.233.035,40	-66,04%
I_157	2.232.322,35	2.099.700,00	132.622,35	5,94%
I_158	1.282.203,82	2.100.000,00	-817.796,18	-63,78%
I_159	2.014.738,83	1.890.000,00	124.738,83	6,19%
I_160	851.891,61	750.000,00	101.891,61	11,96%
I_170	1.213.930,96	950.000,00	263.930,96	21,74%
I_171	744.870,19	630.000,00	114.870,19	15,42%
I_173	1.844.553,79	2.100.000,00	-255.446,21	-13,85%
I_174	418.003,14	530.000,00	-111.996,86	-26,79%
I_176	1.579.521,55	1.500.000,00	79.521,55	5,03%
I_182	3.594.278,08	3.650.000,00	-55.721,92	-1,55%
I_183	2.998.199,39	4.200.000,00	-1.201.800,61	-40,08%
I_193	399.780,94	350.000,00	49.780,94	12,45%
I_195	574.645,09	450.000,00	124.645,09	21,69%
I_196	702.526,90	750.000,00	-47.473,10	-6,76%
I_197	664.193,16	600.000,00	64.193,16	9,66%
I_211	3.964.339,81	3.850.000,00	114.339,81	2,88%
I_212	3.803.634,80	3.900.000,00	-96.365,20	-2,53%
I_214	474.269,11	371.000,00	103.269,11	21,77%
I_219	5.326.161,85	5.900.000,00	-573.838,15	-10,77%
I_222	4.544.624,33	5.000.000,00	-455.375,67	-10,02%
I_223	3.848.466,68	5.500.000,00	-1.651.533,32	-42,91%
I_225	525.151,33	530.000,00	-4.848,67	-0,92%
I_226	317.145,55	400.000,00	-82.854,45	-26,13%

As amplitudes dos intervalos de confiança apresentadas na tabela 17, por serem menores que 30%, são enquadradas no grau III de fundamentação da NBR 14653-2 (2011) para todos os imóveis avaliados.

Tabela 17 – Intervalos de confiança e intervalos de predição

Imóvel	Intervalo de Confiança (IC)			Intervalo de Predição (IP)		
	Amplitude (%)	Valor mín. (R\$)	Valor máx. (R\$)	Amplitude (%)	Valor mín. (R\$)	Valor máx. (R\$)
I_005	9,90%	2.152.420,88	2.376.398,08	44,69%	1.812.012,13	2.822.833,66
I_006	8,08%	781.046,16	846.752,42	44,31%	652.769,70	1.013.148,63
I_008	7,27%	1.560.208,24	1.677.799,45	44,17%	1.299.608,59	2.014.234,54
I_013	10,15%	1.686.461,81	1.866.583,30	44,75%	1.421.110,85	2.215.113,22
I_015	9,45%	377.492,47	414.896,28	44,59%	317.231,27	493.709,90
I_019	6,62%	2.237.006,47	2.390.067,25	44,06%	1.858.288,67	2.877.161,12
I_023	6,78%	653.815,61	699.670,24	44,09%	543.495,05	841.691,80
I_024	6,78%	653.781,18	699.644,15	44,09%	543.469,96	841.654,95
I_026	6,97%	830.342,72	890.287,47	44,12%	690.795,49	1.070.133,97
I_036	8,64%	1.104.028,18	1.203.684,36	44,42%	924.819,27	1.436.930,98
I_041	10,70%	2.036.594,74	2.266.476,12	44,88%	1.719.750,38	2.684.048,44
I_049	10,82%	1.135.483,42	1.265.162,52	44,91%	959.264,60	1.497.575,40
I_064	10,13%	250.336,58	277.003,43	44,75%	210.927,55	328.757,87
I_072	13,81%	2.578.240,79	2.959.674,31	45,75%	2.201.837,86	3.465.628,95
I_090	10,24%	1.541.495,63	1.707.614,34	44,77%	1.299.387,94	2.025.784,57
I_092	8,34%	1.422.661,25	1.546.304,87	44,36%	1.190.248,99	1.848.241,86
I_096	8,67%	301.756,42	329.078,43	44,43%	252.800,31	392.806,20
I_099	9,25%	197.291,08	216.401,63	44,55%	165.664,29	257.714,64
I_102	9,23%	574.711,35	630.268,87	44,55%	482.548,28	750.645,46
I_107	8,60%	1.839.094,62	2.004.130,56	44,41%	1.540.267,36	2.392.951,92
I_114	10,06%	443.429,70	490.349,66	44,73%	373.530,19	582.109,85
I_122	11,88%	928.335,38	1.045.334,94	45,19%	787.359,47	1.232.501,10
I_123	9,84%	2.320.777,81	2.560.749,73	44,68%	1.953.291,36	3.042.521,61
I_126	9,61%	745.549,08	820.732,51	44,63%	626.926,00	976.026,47
I_130	8,14%	568.390,97	616.562,56	44,33%	475.151,94	737.550,59
I_131	11,46%	501.278,78	562.132,87	45,08%	424.507,51	663.793,40
I_137	9,80%	992.860,58	1.095.038,40	44,67%	835.500,68	1.301.280,15
I_138	8,06%	204.123,01	221.255,64	44,31%	170.586,11	264.754,07
I_142	27,17%	1.630.507,37	2.137.713,02	51,55%	1.446.759,07	2.409.217,20
I_157	10,09%	2.122.495,38	2.347.832,24	44,74%	1.788.139,86	2.786.841,89
I_158	23,02%	1.143.073,39	1.438.268,67	49,43%	1.003.899,69	1.637.660,28
I_159	9,91%	1.917.401,82	2.117.017,17	44,70%	1.614.189,49	2.514.681,58
I_160	13,13%	797.801,20	909.649,31	45,54%	679.709,39	1.067.690,57

Tabela 17 – Intervalos de confiança e intervalos de predição

Imóvel	Intervalo de Confiança (IC)			Intervalo de Predição (IP)		
	Amplitude (%)	Valor mín. (R\$)	Valor máx. (R\$)	Amplitude (%)	Valor mín. (R\$)	Valor máx. (R\$)
I_170	9,77%	1.156.054,50	1.274.704,92	44,66%	972.734,01	1.514.934,55
I_171	6,01%	722.807,63	767.606,17	43,97%	598.887,24	926.437,49
I_173	11,63%	1.740.445,29	1.954.889,76	45,12%	1.474.779,05	2.307.042,99
I_174	8,78%	400.051,59	436.760,22	44,45%	335.298,24	521.108,08
I_176	7,39%	1.522.242,44	1.638.955,96	44,19%	1.268.623,81	1.966.610,04
I_182	12,58%	3.375.233,38	3.827.538,25	45,38%	2.870.036,42	4.501.279,08
I_183	11,23%	2.834.610,62	3.171.229,06	45,02%	2.398.373,83	3.748.039,38
I_193	12,56%	375.461,53	425.675,58	45,38%	319.236,19	500.647,50
I_195	11,13%	543.558,75	607.509,27	44,99%	459.737,16	718.273,41
I_196	7,43%	676.910,22	729.113,00	44,20%	564.228,51	874.723,69
I_197	13,45%	621.036,62	710.348,69	45,64%	529.701,71	832.832,03
I_211	12,77%	3.719.235,05	4.225.597,45	45,44%	3.164.691,80	4.966.041,29
I_212	18,28%	3.471.871,26	4.167.100,85	47,34%	3.008.429,12	4.809.033,92
I_214	11,04%	448.821,71	501.159,32	44,97%	379.476,56	592.740,65
I_219	11,74%	5.022.732,91	5.647.921,28	45,15%	4.257.812,45	6.662.576,24
I_222	9,53%	4.333.303,93	4.766.250,11	44,61%	3.642.630,10	5.669.971,91
I_223	11,67%	3.630.389,43	4.079.643,82	45,13%	3.076.779,85	4.813.700,21
I_225	8,14%	504.225,04	546.946,11	44,32%	421.507,91	654.279,36
I_226	9,91%	301.816,09	333.253,59	44,70%	254.092,59	395.845,06

Os intervalos de valores, conforme o campo de arbítrio, constam na tabela 18. Em uma avaliação intervalar, a faixa de variação permitida apresenta intervalos mais curtos do que o estabelecido no campo de arbítrio apenas nos casos em ocorra IC menor que o campo de arbítrio.

Tabela 18 – Valores máximos e mínimos conforme o campo de arbítrio

Imóvel	Campo de arbítrio	Valor mín. (R\$)	Valor máx. (R\$)
I_005	15,00%	1.922.392,75	2.600.884,31
I_006	15,00%	691.250,57	935.221,36
I_008	15,00%	1.375.245,50	1.860.626,27
I_013	15,00%	1.508.102,53	2.040.374,01
I_015	15,00%	336.389,82	455.115,64
I_019	15,00%	1.965.430,12	2.659.111,34
I_023	15,00%	574.901,27	777.807,60
I_024	15,00%	574.875,42	777.772,62
I_026	15,00%	730.823,91	988.761,76
I_036	15,00%	979.862,90	1.325.696,87
I_041	15,00%	1.826.193,56	2.470.732,47
I_049	15,00%	1.018.784,86	1.378.355,99

Tabela 18 – Valores máximos e mínimos conforme o campo de arbítrio

Imóvel	Campo de arbítrio	Valor mín. (R\$)	Valor máx. (R\$)
I_064	15,00%	223.832,76	302.832,56
I_072	15,00%	2.348.024,51	3.176.739,04
I_090	15,00%	1.379.065,75	1.865.794,83
I_092	15,00%	1.260.715,92	1.705.674,48
I_096	15,00%	267.853,23	362.389,67
I_099	15,00%	175.631,70	237.619,36
I_102	15,00%	511.571,97	692.126,78
I_107	15,00%	1.631.864,02	2.207.816,03
I_114	15,00%	396.354,92	536.244,89
I_122	15,00%	837.334,74	1.132.864,65
I_123	15,00%	2.072.140,87	2.803.484,70
I_126	15,00%	664.902,34	899.573,76
I_130	15,00%	503.188,94	680.785,03
I_131	15,00%	451.209,34	610.459,69
I_137	15,00%	886.293,85	1.199.103,44
I_138	15,00%	180.639,23	244.394,26
I_142	15,00%	1.586.919,91	2.147.009,29
I_157	15,00%	1.897.474,00	2.567.170,70
I_158	15,00%	1.089.873,25	1.474.534,40
I_159	15,00%	1.712.528,01	2.316.949,66
I_160	15,00%	724.107,87	979.675,35
I_170	15,00%	1.031.841,31	1.396.020,60
I_171	15,00%	633.139,66	856.600,72
I_173	15,00%	1.567.870,72	2.121.236,85
I_174	15,00%	355.302,67	480.703,61
I_176	15,00%	1.342.593,32	1.816.449,78
I_182	15,00%	3.055.136,37	4.133.419,79
I_183	15,00%	2.548.469,48	3.447.929,30
I_193	15,00%	339.813,80	459.748,09
I_195	15,00%	488.448,33	660.841,85
I_196	15,00%	597.147,86	807.905,93
I_197	15,00%	564.564,18	763.822,13
I_211	15,00%	3.369.688,84	4.558.990,78
I_212	15,00%	3.233.089,58	4.374.180,02
I_214	15,00%	403.128,74	545.409,47
I_219	15,00%	4.527.237,58	6.125.086,13
I_222	15,00%	3.862.930,68	5.226.317,98
I_223	15,00%	3.271.196,68	4.425.736,69
I_225	15,00%	446.378,63	603.924,03
I_226	15,00%	269.573,71	364.717,38

4.5.CONCLUSÕES PARCIAIS

As análises e resultados apresentados no item 4.4 evidenciam que o modelo de RLC selecionado com auxílio de força-bruta atende aos pressupostos básicos exigidos pela NBR 14.653-2 (ABNT, 2011) em seu item A.2.1: linearidade, normalidade, homocedasticidade, não autocorrelação, não multicolinearidade e ausência de pontos influentes/*outliers*.

Desse modo, é possível afirmar que as avaliações realizadas usando o modelo de RLC podem ser utilizadas como uma referência comparativa para os modelos de RNA a serem desenvolvidos no capítulo 5 desta pesquisa. Essa abordagem visa garantir uma melhor avaliação e interpretação dos testes que serão realizados usando RNA, pois serão comparados à um modelo de RLC com resultados confiáveis.

5. PRODUÇÃO DE MODELO POR REDES NEURAIS ARTIFICIAIS (RNA)

Neste capítulo, os itens iniciais (5.1 até 5.9) apresentam definições importantes de RNA, aplicações, arquiteturas de redes, número de neurônios, *overfitting/underfitting* e estratégias para contorná-los. De posse desses conceitos, é possível avançar na descrição do método utilizado, no item 5.10. No item 5.11 são realizadas simulações considerando RNA com transformações. No item 5.12 são realizadas simulações considerando RNA sem transformações. No item 5.13 são realizadas verificações de qualidade nas RNA. Por fim, no item 5.14 são apresentadas as conclusões parciais para fechamento do capítulo.

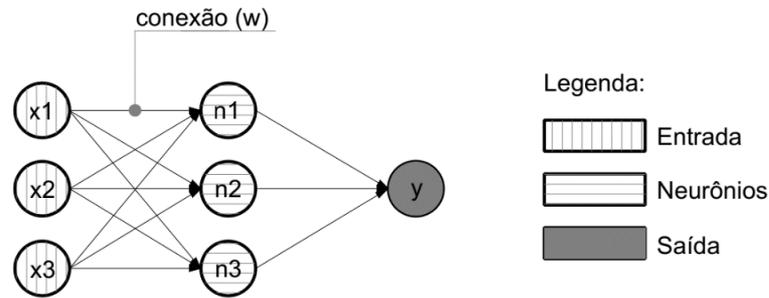
5.1.DEFINIÇÃO

O método de RNA consta no Anexo E da NBR 14653-2 (ABNT, 2011) como uma alternativa para o tratamento científico dos dados coletados. A sigla RNA (Rede Neural Artificial) é autoexplicativa: “Rede” denota que o modelo é composto por um conjunto de estruturas menores conectadas, os neurônios, que justificam o termo “Neural”, e “Artificial” porque são elementos abstratos, ou seja, são modelos matemáticos inspirados no neurônio biológico. RNA podem ser definidas como:

“Redes neurais artificiais são um método para solucionar problemas através da simulação do cérebro humano, inclusive em seu comportamento, ou seja, aprendendo, errando e fazendo descobertas. São técnicas computacionais que apresentam um modelo inspirado na estrutura neural de organismos inteligentes e que adquirem conhecimento por meio de algum tipo de regra de treinamento. O grande apelo desses modelos está em sua capacidade de aprender, generalizar ou extrair regras automaticamente de conjuntos de dados complexos com capacidade de gerar funções não-lineares” (OLIVEIRA E GRANDISKI, 2014, p. 109).

De forma análoga ao cérebro humano, RNA têm muitos neurônios interligados entre si, e se não fosse assim, isto é, se o neurônio artificial estivesse isolado, haveria pequeno poder de cálculo (ABUNAHMAN, 2008). Nesse sentido RNA são “caracterizadas por neurônios artificiais, que são interligados por um grande número de interconexões (sinapses artificiais), sendo representadas aqui por vetores/matrizes de pesos sinápticos” (SILVA; SPATTI; FLAUZINO, 2010, p. 24). Os pesos sinápticos representam matematicamente as RNA, funcionando como os “coeficientes” da equação. A representação gráfica dessa estrutura com neurônios e conexões pode ser observada na figura 29, que representa uma RNA com três entradas, três neurônios e uma saída.

Figura 29 – Representação de uma Rede Neural Artificial

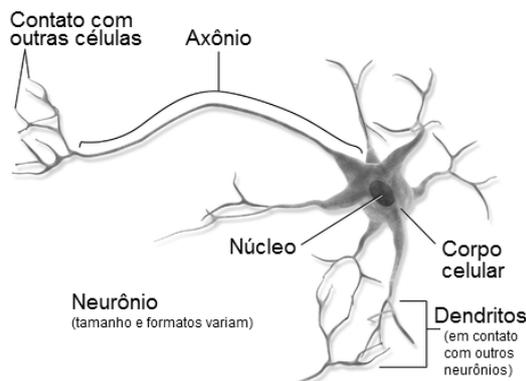


Fonte: Autor (2023)

5.2.NEURÔNIO BIOLÓGICO

Já que a inspiração das RNA procede dos neurônios biológicos, é importante entender o que são e quais funções desempenham. O neurônio é uma célula processadora de informações e formada pelo corpo celular (soma) e pelas ramificações axônio e dendritos (JAIN; MAO; MOHIUDDIN, 1996). Na figura 30 é representada a estrutura de um neurônio biológico.

Figura 30 – Neurônio biológico



Fonte: Adaptado de Blausen.com Staff (2014)

A função dos dendritos, soma e axônio é, respectivamente:

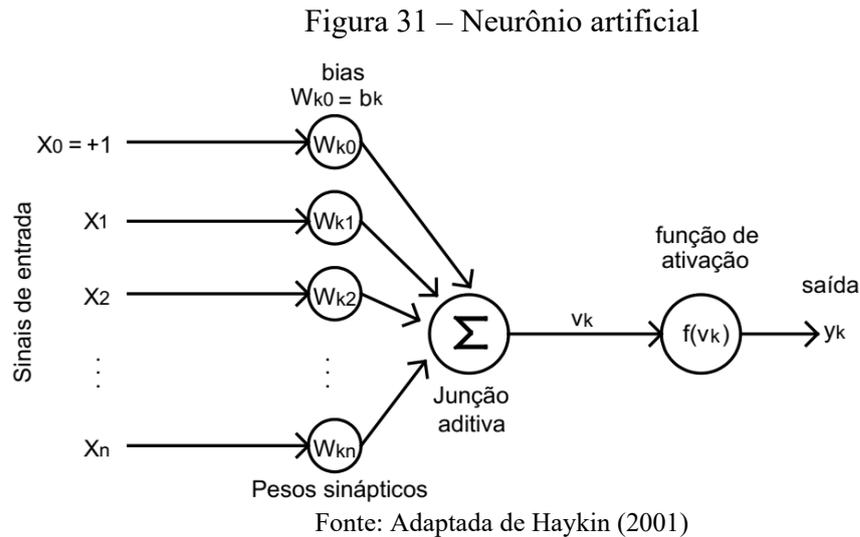
“**captar**, de forma contínua, os estímulos vindos de diversos outros neurônios (conectores) ou do próprio meio externo [...] **processar** todas as informações advindas dos dendritos a fim de produzir um potencial de ativação que indicará se o neurônio poderá disparar um impulso elétrico ao longo de seu axônio. [...] **conduzir** os impulsos elétricos para outros neurônios” (SILVA; SPATTI; FLAUZINO, 2010, p. 29-30).

5.3.NEURÔNIO ARTIFICIAL

“O neurônio matemático, similarmente ao natural, recebe um ou mais sinais de entrada e devolve um único sinal de saída, que pode ser distribuído como sinal de saída da rede, ou como sinal

de entrada para um ou vários outros neurônios da camada posterior” (LUDWIG JR.; MONTGOMERY, 2007, p. 11).

A figura 31 representa um neurônio artificial que, do mesmo modo que o neurônio biológico, pode ser dividido em três partes. A primeira delas, os sinais de entrada, representa os dendritos. A junção aditiva e a função de ativação³⁴ representam o corpo celular. Por fim, a saída representa o sinal conduzido para fora do neurônio pelo axônio.



As equações 9 e 10 apresentadas em Haykin (2001) ajudam a entender o cálculo dentro do neurônio. Os valores de X_1 a X_n são as entradas, ou seja, cada valor representa cada variável do problema e geralmente passam por um processo de normalização³⁵; os w_k são os pesos sinápticos, que servem para ponderar as entradas X e mostrar suas relevâncias para o resultado (SILVA; SPATTI; FLAUZINO, 2010). O valor “*bias*” é um valor externo à RNA, que pode majorar ou minorar o valor que entra na função de ativação, e se manifesta no modelo com uma entrada constante X_0 (+1) ponderada por um peso w igual ao b_k (HAYKIN, 2001). Assim, cada entrada é ponderada pelo seu respectivo peso e o somatório de tudo é realizado conforme a equação 9.

$$v_k = \sum_{j=0}^n w_{kj} \cdot X_j \quad (9)$$

34 Essa função modifica o valor do resultado proveniente do somatório do neurônio. “A função de ativação é também referida como função restritiva já que restringe (limita) o intervalo permissível de amplitude do sinal de saída a um valor finito” (HAYKIN, 2001, p. 37).

35 Processo onde as variáveis são escalonadas para um determinado intervalo de valores.

O y_k é a saída produzida (equação 10) e pode ainda ser usada para alimentar outros neurônios (SILVA; SPATTI; FLAUZINO, 2010):

$$y_k = f(v_k) \quad (10)$$

5.4.APLICAÇÕES

É possível aplicar RNA em diversos tipos de problema. Na literatura é possível encontrar exemplos como: ajuste de curvas, aprendizado por reforço, agrupamentos (GUYON, 1991); classificação de padrões e aproximação de funções (ABUNAHMAN, 2008); controle de processos, previsões, otimizações e memórias associativas (SILVA; SPATTI; FLAUZINO, 2010).

Para o caso específico de avaliações imobiliárias, a aplicação mais adequada é a aproximação de funções na qual, segundo Silva, Spatti e Flauzino (2010, p. 28), “o objetivo consiste em mapear o relacionamento funcional entre as variáveis (tipicamente reais) de um sistema a partir de um conjunto conhecido de seus valores representativos”. O valor do imóvel é função dos dados pesquisados e o objetivo é encontrar uma função que se aproxime da relação real entre variável dependente e variáveis independentes. A partir daí, é possível fazer inferência sobre o valor do imóvel.

Uma vez escolhido o tipo de aplicação, é necessário conhecer qual a arquitetura de RNA deve ser utilizada para esse tipo de aplicação.

5.5.ARQUITETURAS DE RNA

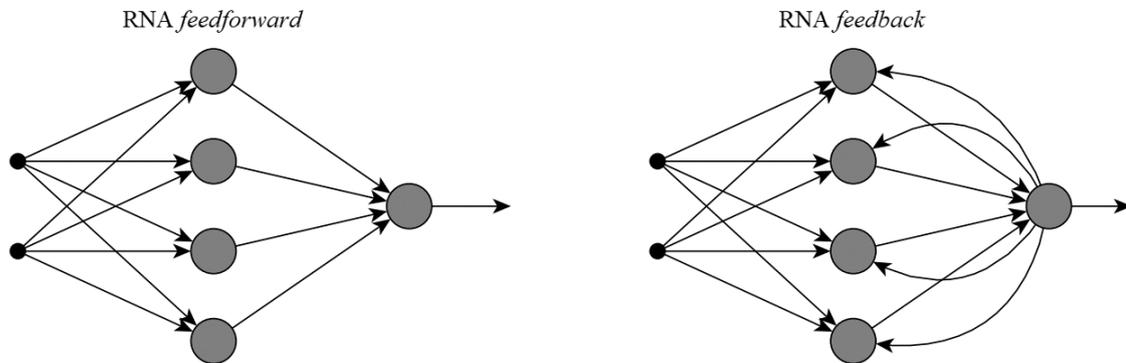
“Arquitetura de uma rede neural artificial define a forma como os seus diversos neurônios estão arranjos, ou dispostos, uns em relação aos outros. Esses arranjos são essencialmente estruturados através do direcionamento das conexões sinápticas dos neurônios” (SILVA; SPATTI; FLAUZINO, 2010, p. 45). Desse modo, a arquitetura caracteriza como as ligações são definidas entre os neurônios de diversas camadas.

As RNA têm três categorias de camadas: as de entrada, as ocultas e as de saída; a primeira recebe os *inputs* do sistema, a segunda, formada por neurônios, calcula a maior parte do que se passa na RNA, e a última, também formada por neurônios, finaliza o cálculo e gera o resultado (SILVA; SPATTI; FLAUZINO, 2010).

É necessário estabelecer um tipo de arquitetura para a RNA, e para isso, devem ser conhecidos os tipos e suas particularidades. As RNA podem assumir as seguintes arquiteturas: *feedforward* ou

Recurrent (ou *feedback*); a diferença é que nesta há presença de *loop*, naquela não há (JAIN; MAO; MOHIUDDIN, 1996). A figura 32 explica a diferença ao mostrar a RNA *feedforward* com um fluxo unidirecional (entrada→camada oculta→saída) e a *feedback* com fluxo de informações nas duas direções.

Figura 32 – Tipos de Arquiteturas de Redes Neurais



A arquitetura do tipo *feedforward*, conforme consta em Silva, Spatti e Flauzino (2010, p. 47) é utilizada comumente para “classificação de padrões e filtragem linear” no caso de possuir apenas uma camada de neurônios; no caso de múltiplas camadas, os autores citam aplicações em problemas relacionados a “aproximações de funções, classificação de padrões, identificação de sistemas, otimização, robótica, ao controle de processos, etc.”.

Já o tipo *feedback*, que “usa dados sequenciais ou dados de série temporal”, “são comumente usados para problemas ordinais ou temporais, como tradução de linguagem, processamento de linguagem natural (NLP), reconhecimento de fala e legendagem de imagens” (IBM, 2020).

Dentre as opções mencionadas, é possível perceber que a arquitetura adequada para avaliações imobiliárias é a *feedforward* de múltiplas camadas, já que é útil no caso de aproximações de funções, sendo essa a aplicação desejada. A utilização das redes *feedback* se justifica apenas se o objetivo for realizar predições com base em dados de séries temporais, que não é o objetivo usual de avaliações imobiliárias, tanto que a própria NBR 14653-2 (ABNT, 2011) afirma em seu item 8.2.1.3.2 que a amostra deve ser coletada em um período específico.

Conforme consta em Silva, Spatti e Flauzino (2010), as redes *feedforward* podem ter uma camada única de neurônios ou múltiplas camadas.

O termo “camada única” (de neurônios) faz referência apenas à camada de saída, já que os nós de entrada não são considerados neurônios, pois não efetuam cálculos (HAYKIN, 2001). As *feedforward* de camadas múltiplas têm, além da camada de saída, pelo menos uma camada oculta de neurônios (SILVA; SPATTI; FLAUZINO, 2010). As figuras 33 e 34 apresentam exemplos de RNA *feedforward* com uma única camada e com múltiplas camadas, respectivamente.

Figura 33 – RNA *feedforward* com única camada de neurônios (saída)

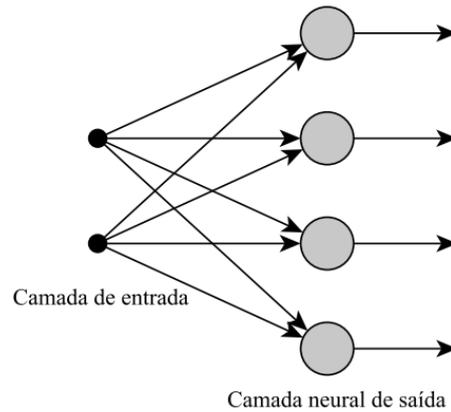
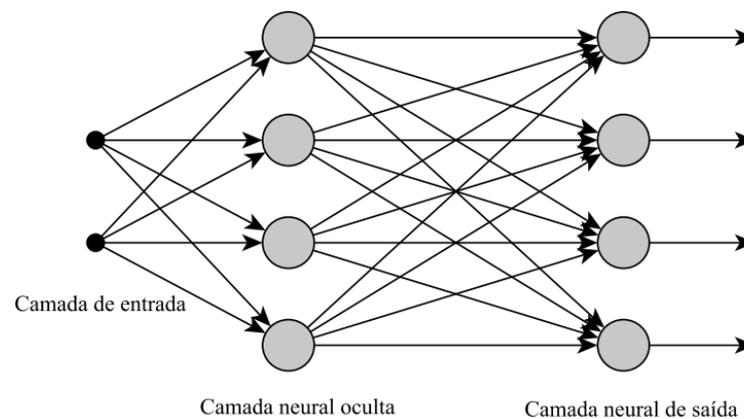


Figura 34 – RNA *feedforward* com camadas múltiplas (oculta + saída)



“Entre os principais tipos de redes com arquitetura *feedforward* de camadas múltiplas se encontra o Perceptron multicamadas (*multilayer Perceptron – MLP*) e as redes de base radial (*radial basis function – RBF*)” (SILVA; SPATTI; FLAUZINO, 2010, p. 48).

Ambos os tipos (MLP e RBF) podem ser usadas para problemas de aproximação de função, mas a principal diferença é que as RBF têm apenas uma camada neural oculta e sua função de ativação deve ser sempre a gaussiana (SILVA; SPATTI; FLAUZINO, 2010).

A NBR 14653-2 (ABNT, 2011) não indica explicitamente qual deve ser usada, se MLP ou RBF, no entanto, a norma usa o termo “RNA do tipo multicamadas”. Nesse sentido, é possível inferir que MLP é o tipo mais compatível. Além disso, a norma recomenda que o avaliador estabeleça o número de camadas ocultas e a função de ativação adotada; essas recomendações não fazem sentido para redes RBF, visto que esse tipo possui uma única camada oculta e usa apenas a função gaussiana. Assim, é possível concluir que as redes MLP são as mais compatíveis com as prescrições normativas.

Além da arquitetura da rede, também é importante definir o número de neurônios em suas camadas, que tem sido objeto de estudo de muitos pesquisadores.

5.6. PARÂMETROS COMUMENTE UTILIZADOS NA DEFINIÇÃO DAS RNA

5.6.1 Quantidade de neurônios

Um dos problemas no projeto de redes *feedforward* é a quantidade de neurônios que devem ser dispostos nas camadas (JAIN; MAO; MOHIUDDIN, 1996). Apesar de existirem várias pesquisas sobre esse tema, não há uma quantidade ideal definida e, segundo Kaastra e Boyd (1996), não existe uma fórmula padrão para a definição do número de neurônios, os quais costumam ser estabelecidos por meio de experimentação.

É importante explicar qual é o “número de neurônios” que tem sido objeto de estudo dos pesquisadores. Existem dois tipos de camadas neurais: as ocultas e as de saída. Para as camadas de saída, o número de neurônios sempre coincide com o próprio número de saídas do problema, ou seja, para x saídas, há x neurônios (SILVA; SPATTI; FLAUZINO, 2010). Assim, a definição da quantidade de neurônios de saída é simples. Porém, a questão se torna mais complexa no que diz respeito às camadas ocultas, pois não há uma regra estabelecida para determinar o número de neurônios. É, portanto, a quantidade de neurônios nessas camadas que tem sido alvo de estudos e investigações. Por exemplo, o quadro 5 apresenta algumas abordagens encontradas na literatura técnica para definir a quantidade de neurônios.

No quadro 5, N_n representa a quantidade de neurônios na camada; n representa a quantidade de dados da amostra; n_t representa a quantidade de dados da amostra de treinamento³⁶; n_i é o número de entradas; n_o é o número de saídas da rede neural; r representa a quantidade de camadas ocultas; N representa a ordem da função a ser aproximada (caso essa seja a finalidade da RNA).

³⁶ Amostra de treinamento é uma porcentagem da amostra total e é destinada exclusivamente para treinar a RNA. Já a amostra de validação é a outra parte da amostra total e é utilizada para validar se o modelo foi bem treinado. Se o modelo for bem treinado, ele é capaz de fazer boas previsões para as duas amostras.

Quadro 5 - Equações utilizadas para definir a quantidade de neurônios em RNA

Fonte	Descrição	Equação
Huang e Huang (1991); Tamura e Tateishi (1997)	São utilizadas RNA <i>feedforward</i> do tipo MLP com uma camada oculta para resolver problema de classificação. A equação 11 define o valor de N_n .	$N_n = n - 1 \quad (11)$
Wanas et. al (1998)	São utilizadas RNA <i>feedforward</i> com uma camada oculta para resolver problema de classificação. A equação 12 define o valor de N_n .	$N_n = \log_2(n_t) \quad (12)$
Cechin, Souto e González (2000)	São utilizadas RNA <i>feedforward</i> MLP aplicadas em avaliação imobiliária no município de Porto Alegre – RS. Os autores usam a equação 13 para definir a quantidade de neurônios.	$N_n = n_i(2^u) \quad (13)$ Onde u varia de 0 a 2.
Ke e Liu (2008)	São utilizadas RNA <i>feedforward</i> com variação no número de camadas ocultas (0 a 3) para resolver problema de predição de ações em bolsa de valores. A equação 14 define o valor de N_n .	$N_n = \frac{(n_i + \sqrt{n_i})}{r} \quad (14)^{37}$
Shibata e Ikeda (2009);	São utilizadas RNA <i>feedforward</i> com uma camada oculta para resolver problemas de classificação. A equação 15 define o valor de N_n . No entanto, segundo Kaastra e Boyd (1996), o número ideal de neurônios pode variar entre a metade e o dobro do resultado anterior (ver equação 16).	$N_n = \sqrt{n_i n_o} \quad (15)^{38}$ $\frac{\sqrt{n_i n_o}}{2} \leq N_n \leq 2\sqrt{n_i n_o} \quad (16)$
Trenn (2008)	As equações 17 e 18 são aplicadas para RNA <i>feedforward</i> MLP para uma e duas camadas ocultas respectivamente com a finalidade de resolver problemas de aproximação de função.	$N_n \geq \frac{(N+n_i)!}{(N!n_i!)} \cdot \frac{1}{n_i+2} \quad (17)$ $N_n \geq 2\sqrt{\frac{(N+n_i)!}{(N!n_i!)}} + 2(n_i + 1) - n_i - 3 \quad (18)$
Silva, Spatti e Flauzino (2010)	As equações 19 e 20 são denominadas de método de Kolmogorov e de Fletcher-Gloss respectivamente e são indicadas para RNA <i>feedforward</i> MLP com uma camada oculta. Os autores não indicam um problema específico no qual devem ser usadas as equações.	$N_n = 2n_i + 1 \quad (19)$ $2\sqrt{n_i} + n_o \leq N_n \leq 2n_i + 1 \quad (20)$

37 Segundo Ke e Liu (2008), não há vantagens em usar mais neurônios do que o número usado nos casos de treinamento. Por isso, nessa fórmula, o n pode ser usado o próprio n_t .

38 Essa equação é denominada “*geometric pyramid rule*” e foi proposta inicialmente por Masters (1993) apud Vega-Márquez et al. (2020), sendo aplicada para casos de RNAs com uma camada oculta.

Quadro 5 - Equações utilizadas para definir a quantidade de neurônios em RNA

Fonte	Descrição	Equação
Sheela e Deepa (2013a)	São utilizadas RNA <i>feedforward</i> com uma camada oculta para resolver problemas de predição da velocidade do vento em sistemas de energia renovável. Os autores testam várias equações e essa é a que produz os menores erros. A equação 21 define o valor de N_n .	$N_n = \frac{4^{n_i+18}}{n_i-1} \quad (21)$
Madhiarasan e Deepa (2016)	São utilizadas RNA <i>feedforward</i> com uma camada oculta para resolver problemas de predição da velocidade do vento em sistemas de energia renovável. Os autores testam várias equações e essa é a que produz os menores erros. A equação 22 define o valor de N_n .	$N_n = \frac{8n_i-4}{n_i-4} \quad (22)^{39}$
Yotov, Hadzhikolev e Hadzhikoleva (2020)	As equações 23 e 24 são aplicadas para RNA <i>feedforward</i> MLP com uma e múltiplas camadas ocultas respectivamente com a finalidade de resolver problemas de aproximação de função.	$N_n \leq \frac{n_t-1}{n_i+2} \quad (23)$ $N_n \leq \frac{\sqrt{(r+n_i+2)^2+4(r-1)(n_t-1)-(r+n_i+2)}}{2(r-1)} \quad (24)$
Torres-Pruñonosa, García-Estévez, Prado-Román (2021)	São utilizadas RNA <i>feedforward</i> MLP aplicadas para avaliação imobiliária em massa na região da Catalunha, Espanha. Os autores usam a equação 25 para definir a quantidade de neurônios.	$N_n = n_i \quad (25)^{40}$
Yasnitsky, Yasnitsky e Alekseev (2021)	É utilizada uma RNA <i>feedforward</i> MLP com duas camadas intermediárias para realizar avaliação imobiliária em massa capaz de se adaptar às informações de espaço e tempo. O número de neurônios usados pelos autores é baseado na proposição de Yasnitsky (2020) e pode ser observado na equação 28.	$N_n = N_{min} + \xi(N_{max} - N_{min}) \quad (26)$ $N_{min} \leq \frac{n_o n_t}{(1 + \log_2(n_t))(n_i + n_o)} \quad (27)$ $N_{max} \leq \frac{n_o}{(n_i + n_o)} \left(\frac{n_t}{n_i} + 1 \right) (n_i + n_o + 1) + n_o \quad (28)$ <p>Onde ξ é um coeficiente empírico que varia de 0 a 0,5 (os autores citados usam 0,1).</p>

É possível perceber que as equações apresentadas no quadro 5 utilizam parâmetros que podem ser claramente definidos pelo pesquisador como o número de entradas, saídas, quantidade de camadas intermediárias. No entanto, existem outros fatores citados por Ke e Liu (2008) que também interferem

39 Uma limitação dessa equação é que o número de entradas deve ser no mínimo 5 para que não ocorra inconsistência matemática. No trabalho de Madhiarasan e Deepa (2015) foram usadas 5 entradas.

40 A quantidade de neurônios adotada é igual ao número de variáveis independentes (DEMUTH et al., 2009 *apud* TORRES-PRUÑONOSA; GARCÍA-ESTÉVEZ; PRADO-ROMÁN, 2021).

na quantidade ideal dos neurônios, como o grau de ruído da amostra e a complexidade da função a ser aproximada, mas que são difíceis de serem quantificados e especificados como parâmetros.

Quanto ao grau de ruído, nem sempre o pesquisador consegue uma amostra homogênea, pois pode ser que a heterogeneidade seja uma característica da região de estudo. Com relação à complexidade da função, não há como mensurar, de antemão, a complexidade e não há como garantir que não haverá situações com alto grau de complexidade. Sendo assim, uma RNA que funciona muito bem em um caso pode não funcionar em outro, pois ambos podem ter características distintas que dificultam a convergência de bons resultados para o mesmo modelo.

Diante disso, fica claro os desafios na determinação do número ideal de neurônios em camadas ocultas de RNA, caso contrário o pesquisador pode incorrer até mesmo em *overfitting* e *underfitting*.

5.6.2 Quantidade de camadas intermediárias

De modo semelhante ao item 5.6.1, não há uma recomendação universal na literatura técnica capaz de sempre fornecer os melhores resultados.

Embora teoricamente uma única camada neural possa resolver até mesmo problemas complexos, redes neurais com múltiplas camadas podem oferecer melhor desempenho usando menos neurônios para resolver o mesmo problema (GÉRON, 2019).

Esse critério de complexidade torna a definição desse parâmetro algo subjetivo para o avaliador. Parece improvável que, previamente, o avaliador seja capaz de determinar se o problema a ser resolvido possui uma complexidade que justifique a utilização de redes neurais com muitas camadas intermediárias.

5.6.3 Taxa de aprendizagem e número de iterações

“A taxa de aprendizagem afeta a velocidade de treinamento da sua rede neural. [...] Taxas de aprendizagem mais altas podem ultrapassar as configurações de peso ideais. Uma taxa de treinamento mais baixa sempre produzirá melhores resultados. No entanto, diminuir a taxa de treinamento pode aumentar bastante o tempo de execução.” (HEATON, 2015).

Novamente, não há um valor padrão que deva sempre usado, pois isso depende do problema e da arquitetura da RNA. Valores comumente utilizados estão entre 0 e 1. (SILVA, SPATTI; FLAUZINO, 2010).

O tempo de execução do treinamento, por sua vez, guarda uma relação direta com o número de iterações envolvidas no processo. Esse número, apesar de ser um parâmetro do modelo, não precisa

necessariamente ser ajustado como os demais parâmetros, uma vez que o treinamento das RNA pode simplesmente ser interrompido como será visto no item 5.8.

Vale destacar que a quantidade de iterações tem relação direta com a ocorrência de *overfitting* e *underfitting* nas RNA, tema do item 5.7.

5.6.4 Funções de ativação

Outro parâmetro muito importante na definição de RNA é a função de ativação que será aplicada aos valores pós junção aditiva no neurônio como foi representado na figura 31. “Seu objetivo é limitar a saída do próprio neurônio dentro de um intervalo de valores razoáveis a serem assumidos pela sua própria imagem funcional” (SILVA, SPATTI; FLAUZINO, 2010).

Existem várias funções de ativação que podem ser utilizadas em redes neurais como: relu, sigmoid, softmax, softplus, softsign, tanh, selu, elu e exponencial.⁴¹

5.6.5 Otimizadores e inicializadores de pesos

“Um otimizador é uma função que altera o peso e a taxa de aprendizado de uma rede neural. Como resultado, contribui para diminuir a perda geral e, ao mesmo tempo, melhorar a precisão” (JAIN *et al.*, 2022). Vários são os otimizadores disponíveis para utilização em redes neurais como: SGD, RMSprop, Adam, AdamW, Adadelta, Adagrad, Adamax, Adafactor, Nadam e Ftrl.⁴¹

Segundo Moolayil (2018), o otimizador mais recomendado tem sido o Adam, mas suas variações como Adamax e Nadam também são opções a serem consideradas quando o Adam não estiver entregando o resultado desejado. E conforme Kingma e Ba (2015), Adam “é computacionalmente eficiente, tem poucos requisitos de memória, é invariante ao redimensionamento diagonal dos gradientes e é adequado para problemas grandes em termos de dados e/ou parâmetros”.

Já os inicializadores dos pesos são utilizados para a definição inicial (randomicamente) dos pesos sinápticos das camadas neurais⁴². Vale ressaltar que esses pesos serão alterados durante o treinamento, no entanto, a forma como são iniciados influenciam nos resultados. Exemplos de inicializadores são: Random Normal, Random Uniform, Truncated Normal, Glorot Normal e Glorot Uniform⁴².

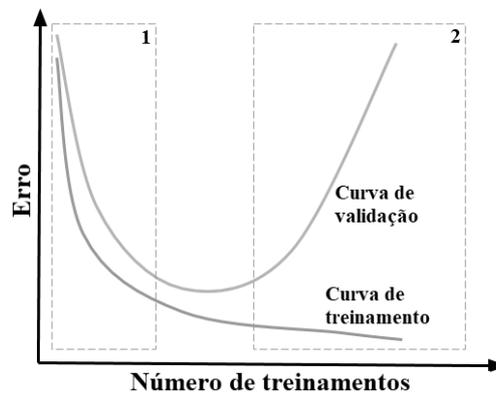
41 <https://keras.io/api/layers/activations/>. Acesso em 28 set. 23.

42 <https://keras.io/api/layers/initializers/>. Acesso em 28 set. 23.

5.7.OVERFITTING E UNDERFITTING

Overfitting e *underfitting* decorrem, respectivamente, do excesso e da insuficiência de treinamento da RNA. Graficamente, é possível identificar esses fenômenos como observado na figura 35, na qual o cenário 1 corresponde ao *underfitting*, onde nem mesmo a amostra de treinamento tem bons resultados, havendo elevados erros tanto na curva de treinamento como na de validação e o cenário 2 corresponde ao *overfitting*, caracterizado pelos grandes erros na curva de validação e pequenos erros na curva de treinamento, cuja amostra foi memorizada pela RNA (ZIGANTO, 2018).

Figura 35 – Cenários possíveis de uma RNA em treinamento



Fonte: adaptado de Ziganto (2018)

O problema dessa memorização é que o modelo consegue prever muito bem apenas os dados treinados, mas não consegue fazer previsões adequadas para valores fora da amostra de treinamento. O cenário desejável fica entre 1 e 2, local caracterizado por pequenos erros tanto na curva de treinamento como na de validação (ZIGANTO, 2018).

O número de neurônios influencia diretamente na ocorrência de *underfitting* ou *overfitting*. RNA com muitos neurônios tendem a convergir mais rápido, ou seja, menos repetições são necessárias para um treinamento adequado. Por outro lado, se for diminuído o número de neurônios, mais repetições são necessárias. O problema é que não é possível saber de antemão quantas repetições precisam ser realizadas para treinar adequadamente uma RNA. Por exemplo, se 500 repetições são necessárias para um treinamento adequado, mas a RNA é configurada para treinar 5.000 vezes, certamente haverá *overfitting*, devido ao excesso de treinamento. Por outro lado, se for configurada para treinar apenas 50 vezes, haverá *underfitting*, devido à insuficiência de treinamento.

No caso de *underfitting*, a solução é aumentar o número de treinamentos (mantendo fixo o número de neurônios). No entanto, isso pode ocasionar o surgimento do *overfitting*, caso o aumento seja feito sem controle/monitoramento. Para evitar o surgimento do *overfitting*, Braga, Carvalho e

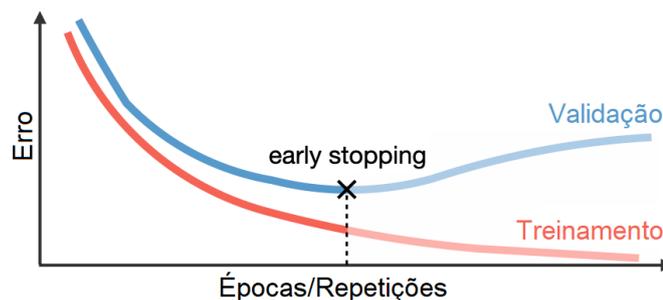
Ludermir (2000) citam duas soluções: a parada antecipada do treinamento (*early stopping*) e a poda (*pruning*) dos pesos da RNA.

5.8.EARLY STOPPING

O *early stopping* evita diretamente o surgimento do *overfitting*, pois não permite treinamentos excessivos, ou seja, se o modelo inicia o levantamento da curva de validação, automaticamente, o treinamento do modelo é interrompido.

Na prática, é realizado um monitoramento da acurácia dos resultados referentes à amostra de validação durante as repetições (épocas) e, quando houver perda de acurácia, o treinamento é interrompido (YING, 2019). Em outras palavras, quando o erro de validação atingir um platô (valor constante) ou começar a aumentar, o *early stopping* interrompe o processo de treinamento, conforme observado na figura 36 (AMIDI; AMIDI, 2018). O parâmetro comumente utilizado para acompanhar a acurácia é o erro quadrático médio (MSE).

Figura 36 – Interrupção do treinamento por meio do *Early Stopping*



Fonte: Adaptado de Amidi e Amidi (2018)

5.9.PRUNING

O termo *pruning* significa cortar partes indesejadas (MERRIAM-WEBSTER, 2014). No contexto de RNA, esse termo é aplicado quando há partes desnecessárias na rede neural e que precisam ser podadas/removidas.

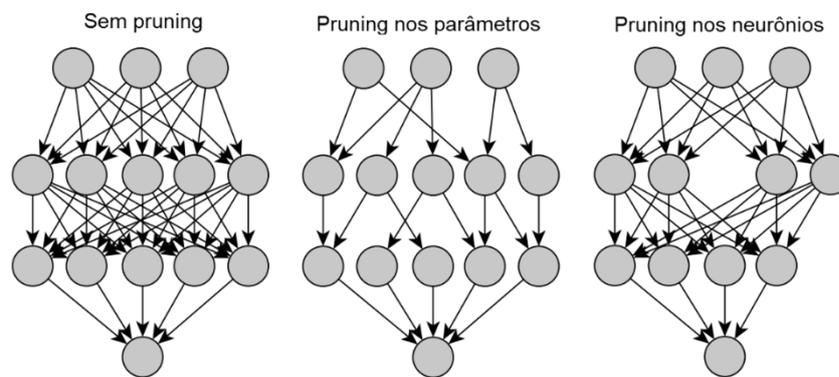
Empiricamente, é natural pensar que quanto maior a RNA, melhor será seu desempenho, no entanto, segundo Haykin (2001, p. 244-245), “é menos provável que uma rede neural com tamanho mínimo aprenda as idiossincrasias ou ruídos dos dados de treinamento, e pode assim generalizar melhor sobre novos dados”.

O significado de “aprender idiossincrasias/ruídos” é que a RNA deixou de aprender e passou a decorar os dados de treinamento e, nesse caso, a curva de validação tende a subir (muitos erros e

pouca capacidade de generalização). Esse tipo de situação é mais susceptível de ocorrer em RNA maiores, pois essas decoram os dados de treinamento muito mais rápido do que as menores. Desse modo, para simplificar a complexidade da RNA, pode ser utilizado a técnica do *pruning*.

O *pruning*, segundo Liang et al. (2021, p. 375), “remove parâmetros ou neurônios redundantes que não contribuem significativamente para a acurácia dos resultados”. Dessa forma, é obtida uma rede neural menor ou menos complexa que a original. O autor citado faz referência a dois tipos de *pruning*: o que poda parâmetros e o que poda neurônios. As duas formas diminuem a complexidade da rede neural, a diferença é que o *pruning* de parâmetros remove as conexões/sinapses entre neurônios e o *pruning* de neurônios remove a própria unidade neural (ver figura 37).

Figura 37 – Modelo original de uma RNA e seus modelos podados



Fonte: Autor (2023)

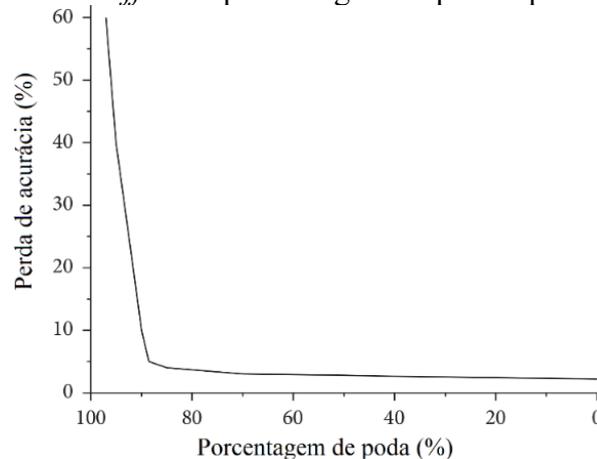
Além de ser usada para melhorar a capacidade de generalização da RNA, o *pruning*, segundo Pelli Neto (2006, p. 49), “permite obter modelos com pequena possibilidade de sobreajuste (*overfitting*)”. O autor não garante a ausência, mas sim a pequena possibilidade de *overfitting*. E a possibilidade ainda existe porque o *pruning*, em um eventual excesso de repetições, não interrompe definitivamente o treinamento da RNA.

Outra melhoria conferida pelo *pruning* é a redução do consumo de memória para execução de uma RNA (PIETRON; WIELGOSZ, 2020). Isso é mais perceptível em casos de modelos com um grande volume de dados (*big data*). Para o caso de avaliações imobiliárias, ainda que sejam avaliações em massa, o consumo de memória costuma ser pequeno.

Apesar das melhorias conferidas pelo *pruning*, essa técnica nem sempre garante maior capacidade de generalização (SHEELA; DEEPA, 2013b). Isso pode ser observado em trabalhos como os de Shan, Ye e Cui (2020) e Liebenwein *et al.* (2021), onde o aumento inadequado da porcentagem de poda provoca redução na acurácia dos resultados (ver figura 38). Os trabalhos citados mostram

que porcentagens de poda acima de 80% são prejudiciais aos modelos, por isso, é recomendável que a aplicação do *pruning* seja feita de modo cuidadoso.

Figura 38 – *Trade-off* entre porcentagem de poda e perda de acurácia



Fonte: Adaptado de Shan, Ye e Cui (2020).

5.10. DESCRIÇÃO DO MÉTODO

Neste item é descrito o método utilizado para a produção dos modelos de RNA e verificação de sua qualidade.

5.10.1 Definição dos parâmetros de cálculo

O primeiro parâmetro a ser definido é a porcentagem para a separação da amostra em validação e treinamento. Segundo Prakash *et al.* (2021), taxas comumente utilizadas na divisão de amostra de treinamento e validação são: 90:10, 80:20, 60:40. Naturalmente, quanto menor for a amostra total, menor deve ser a porcentagem dedicada à amostra de validação; e quanto maior for a amostra total, é possível dedicar uma porcentagem maior para o teste. Neste trabalho, 25% da amostra total é separada para compor a amostra de validação.

Outro parâmetro a ser definido é o número de neurônios. Como já foi visto na fundamentação teórica, não há unanimidade na literatura quanto à definição desse valor por meio de alguma equação, por isso, são realizadas simulações variando esse número. São realizadas simulações com no mínimo 5 neurônios e no máximo 615 neurônios.

É necessário definir também a quantidade de camadas ocultas (*hidden layers*). Segundo Kaastra e Boyd (1996), teoricamente, a utilização de apenas uma camada já é suficiente para realizar aproximações de funções e, na prática, os modelos de RNA mais utilizados possuem uma ou duas

camadas (menos comum), os quais têm bom funcionamento. Neste trabalho, serão realizadas simulações com até 3 camadas ocultas.

Com relação ao número máximo de treinamentos/iterações das RNA, neste estudo, alguns modelos foram treinados com até 2.000 iterações e outros foram submetidos a um máximo de 5.000 iterações.

Outro parâmetro muito importante é a taxa de aprendizagem (η):

“A taxa de aprendizagem $\{\eta\}$ exprime o quão rápido o processo de treinamento da rede estará sendo conduzido rumo à sua convergência (estabilização). A escolha de η deve ser realizada com cautela para evitar instabilidades no processo de treinamento, sendo que normalmente se adotam valores pertencentes ao intervalo compreendido em $0 < \eta < 1$ ” (SILVA, SPATTI; FLAUZINO, 2010, p. 64).

Valores como 0,1; 0,01; 0,001 são habituais para η , sendo que valores muito pequenos costumam convergir para bons resultados, porém com maior demanda de tempo para o treinamento ser processado (HEATON, 2022). As RNA utilizadas neste trabalho podem assumir valores iniciais de η igual a 0,001 ou 0,0005. Essa taxa pode ser reduzida ao longo do treinamento, o que ajuda a combater o *overfitting* e segundo You *et al.* (2011), “melhora a aprendizagem de padrões complexos”. Uma forma de forçar a redução de η é monitorando o desempenho do treinamento, por exemplo: se a métrica de desempenho não melhorar em uma determinada quantidade de iterações, isso significa que o valor de η deve ser alterado. O módulo Keras disponibiliza uma função⁴³ que faz esse monitoramento e no item 5.11.1 é descrito em quais simulações é usada tal funcionalidade.

Além desses parâmetros, nas simulações de RNA são testados⁴⁴ os seguintes:

- a) funções de ativação: *rectified linear unit*, *softsign* e tangente hiperbólica (*tanh*);
- b) otimizadores: *Adam*, *Nadam* e *RMSprop*;
- c) inicializadores dos pesos sinápticos: *Truncated normal*, *Glorot normal*, *Glorot uniform*.

43 <https://keras.io/api/callbacks/reduce_lr_on_plateau/>. Acesso em 10 de janeiro de 2023.

44 Nas RNA do item 5.11 as simulações consideram *tanh*, *Adam* e *Glorot uniform*. Nas RNA do item 5.12 é realizada uma busca mais abrangente, considerando mais parâmetros.

5.10.2 Transformações em RNA

Neste estudo, são desenvolvidos modelos de RNA aplicando ou não transformações nas variáveis.

As primeiras simulações de RNA utilizam transformações⁴⁵ em suas variáveis. Esse tipo de abordagem lineariza a RNA e torna mais fácil a convergência da RNA mesmo sem uma busca abrangente dos melhores parâmetros via *hypertuning*⁴⁶. Nesse caso será estudado especificamente a variação do número de neurônios, do número de camadas intermediárias e dos critérios para evitar *overfitting*.

As demais simulações são realizadas sem a utilização de quaisquer transformações nas variáveis. Nesse caso, é aplicado o *hypertuning* de forma mais abrangente para a busca da melhor combinação possível de parâmetros para lidar com modelos não lineares de RNA. Nesse caso são realizadas variações nos parâmetros: número de neurônios, camadas intermediárias, funções de ativação, taxa de aprendizagem, otimizadores, e inicializadores dos pesos sinápticos.

5.10.3 Escalonar/normalizar amostras

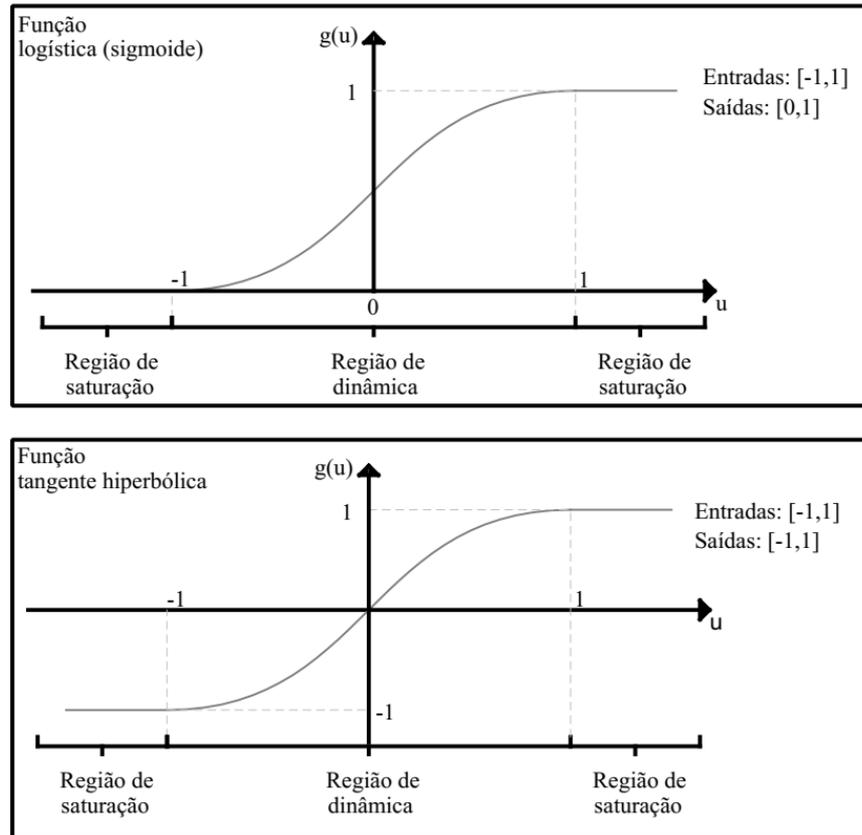
O escalonamento ou normalização das variáveis é um processo no qual são alteradas as escalas dos dados. A aplicação desse processo nos valores de entrada (variáveis independentes) e saída (variáveis dependentes) é uma prática recomendada para redes neurais MLP para que as variáveis fiquem fora da zona de saturação das funções de ativação (SILVA; SPATTI; FLAUZINO, 2010).

As zonas de saturação são regiões onde o gráfico da função de ativação já atingiu o comportamento assintótico, como observado na figura 39, sendo que os intervalos de normalização podem mudar conforme função utilizada (SILVA; SPATTI; FLAUZINO, 2010).

45 Será aplicado \ln nas variáveis VT, Área e Distância a fim de garantir que a RNA funcionará de modo linear.

46 É o ajuste fino onde deseja-se buscar a melhor combinação possível de parâmetros para uma RNA.

Figura 39 – Intervalos de normalização conforme a função usada.



Fonte: Adaptado de Silva, Spatti, Flauzino (2010)

Neste trabalho, o processo de normalização é realizado por meio da classe “*MinMaxScaler*” do módulo “*scikit-learn*”⁴⁷ e sua utilização consta no apêndice H. Essa classe recebe os dados juntamente com o intervalo de normalização e retorna os valores normalizados.

5.10.4 Treinar a RNA

O treinamento de um modelo de RNA é o processo iterativo em que a RNA tem seus pesos sinápticos calibrados à cada iteração visando minimizar os erros obtidos.

De posse dos modelos de RNA e das amostras escalonadas, pode ser utilizada a função “*fit*” do módulo “*TensorFlow.keras*”⁴⁸ para realizar o treinamento, como pode ser observado no apêndice H, onde são apresentados treinamentos de modelos diferentes usando *Early Stopping* e *Pruning*.

47 <<https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html>>. Acesso em 01 de maio de 2023.

48 <https://www.tensorflow.org/api_docs/python/tf/keras>. Acesso em 25 de outubro de 2021.

5.10.5 Verificações dos modelos

A NBR 14653-2 (ABNT, 2011) não estabelece nenhuma métrica a ser examinada, no entanto, há várias tradicionalmente usadas em modelos preditivos que podem ser consideradas: RMSE (*Root Mean Squared Error*), MAE (*Mean Absolut Error*) e MAPE (*Mean Absolute Percentage error*). No apêndice H consta o código para cálculo das métricas.

Vale ressaltar que existem algumas diferenças entre as áreas de estatística e *machine learning*, onde aqui é destacada a questão dos pressupostos do modelo: enquanto os modelos estatísticos precisam obedecer a suposições como normalidade, homocedasticidade, entre outros, os modelos de *machine learning* não precisam (LARRAÑAGA *et al.*, 2018). “Do ponto de vista estatístico, as RNA podem ser concebidas como métodos de regressão não paramétricos. [...] Como de costume em abordagens não paramétricas, a normalidade e a homocedasticidade não são suposições obrigatórias, tornando as RNA muito mais flexíveis do que os modelos RLC” (CHIAPPINI *et al.*, 2023). Por fim, segundo Tan *et al.* (2014) *apud* Rodríguez-Ardura e Meseguer-Artola (2020) a análise por RNA “não exige que os dados atendam aos principais pressupostos subjacentes de normalidade, homocedasticidade, linearidade e não multicolinearidade”.

Neste trabalho, portanto, não serão realizadas verificações de pressupostos em RNA como foi realizado para RLC, com exceção da verificação de dependência espacial. Serão analisados os seguintes gráficos para a verificação da qualidade dos modelos: curvas de convergência, gráficos de predição e mapa da distribuição espacial dos resíduos. Com relação às curvas de convergência, o cenário ideal é que o comportamento das amostras de treinamento e de validação sejam parecidos e que os erros obtidos sejam pequenos (ZIGANTO, 2018). E sobre o gráfico de predição, o que se espera é que ocorra boa aderência dos pontos na bissetriz do gráfico, ou seja, valores observados próximos aos preditos. O mapa da distribuição espacial dos resíduos é usado para verificar a ocorrência de dependência espacial.

5.11. SIMULAÇÕES DE RNA COM A UTILIZAÇÃO DE TRANSFORMAÇÕES

Neste item, são realizadas simulações considerando a aplicação de transformações nas variáveis. Nessas simulações, será estudado especificamente⁴⁹ a variação dos parâmetros: número de neurônios, número de camadas intermediárias e critérios para evitar *overfitting*. O objetivo é compreender de forma mais aprofundada como esses parâmetros impactam na capacidade preditiva das RNA.

5.11.1 Influência do número de neurônios (N_n) na capacidade preditiva de RNA

O objetivo nesse item é analisar a influência do N_n no desempenho das RNA. Para isso, é utilizado o coeficiente de determinação⁵⁰ das amostras de validação. Os resultados para cada RNA simulada constam no apêndice D. No item 5.11.1.1 são apresentados os cenários simulados. Nos itens 5.11.1.2, 5.11.1.3 e 5.11.1.4 são apresentadas e analisadas as simulações.

5.11.1.1 Cenários simulados

Com a finalidade de investigar a influência dos neurônios no comportamento das RNA, foram testados diversos cenários apresentados no quadro 6 e na figura 40.

Quadro 6 - Cenários simulados

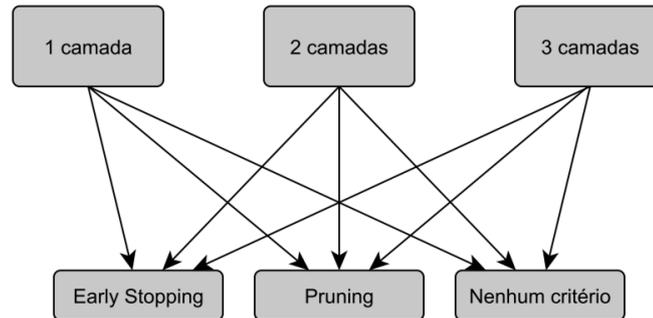
Cenário	Descrição ⁵¹
1HLES	Uma camada intermediária com <i>Early Stopping</i> ;
1HLPR	Uma camada intermediária com <i>Pruning</i> ;
1HL	Uma camada intermediária;
2HLES	Duas camadas intermediárias com <i>Early Stopping</i> ;
2HLPR	Duas camadas intermediárias com <i>Pruning</i> ;
2HL	Duas camadas intermediárias;
3HLES	Três camadas intermediárias com <i>Early Stopping</i> ;
3HLPR	Três camadas intermediárias com <i>Pruning</i> ;
3HL	Três camadas intermediárias;

49 Como trata-se de uma modelo já linearizado pelas transformações, será realizada a variação de apenas esses parâmetros. Nos cenários não linearizados (item 5.12) será realizada uma busca de parâmetros mais abrangente.

50 Modelos com desempenho inferior ao *baseline* horizontal médio serão considerados com R^2 igual 0.

51 Os cenários, com exceção de 1HL, 2HL e 3HL, consideram também a redução na taxa de aprendizagem (η) citada no item 5.10.1.

Figura 40 – Combinações possíveis para uma quantidade fixa de neurônios



Para cada um desses cenários são utilizadas diferentes quantidade de neurônios. Para efeitos comparativos, os resultados são categorizados conforme a quantidade total de neurônios (N_n):

- Categoria A: $5 \leq N_n \leq 54$;
- Categoria B: $55 \leq N_n \leq 104$;
- Categoria C: $105 \leq N_n \leq 154$;
- Categoria D: $155 \leq N_n \leq 204$;

Para comparar as RNA categorizadas por quantidade de neurônios, são utilizados gráficos de *box plot* e *scatter plot*, que permitem identificar diferenças e semelhanças entre as categorias, proporcionando uma melhor compreensão do comportamento das RNA. Para confirmar os resultados observados, são utilizados testes de hipóteses, como o testes-t e o teste de Mann-Whitney para realizar comparações entre as categorias das RNA simuladas.

Os resultados são separados nos três itens: 5.11.1.2, 5.11.1.3 e 5.11.1.4. O primeiro apresenta as simulações realizadas usando o critério do *Early Stopping* (ES), o segundo traz os modelos utilizando a técnica *Pruning* (PR) e o último é referente aos modelos sem *Early Stopping* ou *Pruning*.

5.11.1.2 Cenários com *Early Stopping*: 1HLES, 2HLES e 3HLES

As figuras (41 a 46) apresentam os gráficos *box-plot* e *scatter-plot* referentes aos R^2 obtidos. Nessas figuras, é possível observar que a categoria A é a que apresenta a maior dispersão. Essa observação é confirmada por meio da análise exploratória de dados (tabela 19), que apresenta as maiores medidas de dispersão (amplitude, desvio padrão e variância) na categoria A, onde também ocorrem os menores R^2 médios e mínimos.

Nessas figuras, é possível notar que a melhoria na capacidade preditiva das RNA ocorre principalmente na transição da categoria A para B e, a partir daí, os resultados tendem à estabilidade. Além das análises gráficas e exploratórias, são realizados testes para verificar a ocorrência de diferenças significativas entre as categorias. Conforme os testes (teste-t e Mann-Whitney) unilaterais apresentados⁵² na tabela 21 é possível afirmar a um nível de significância de 5% que:

- Em 1HLES, os testes de Mann-Whitney indicam probabilidade significativa de um valor de A ser menor que um valor de B, portanto é refutada a hipótese nula de igualdade estocástica. Já entre as categorias B e C, assim como C e D, é admitida igualdade estocástica entre as categorias, isto é, não foi percebida probabilidade significativa de uma categoria apresentar um valor maior ou menor que a outra;
- Em 2HLES, os testes de Mann-Whitney indicam probabilidade significativa de um valor de A ser menor que um valor de B. Entre as categorias B e C, assim como entre C e D, o teste de Mann-Whitney aponta igualdade estocástica.
- Em 3HLES, os testes de Mann-Whitney mostram probabilidade significativa de um valor de A ser menor que um valor de B, assim como um de B ser menor que um de C; por fim, entre C e D há igualdade conforme o teste-t de Welch.

Cumprido destacar, no entanto, que tais testes não indicam que modelos com um maior número de neurônios serão invariavelmente superiores, nem que os modelos com menos neurônios devam ser descartados. Por exemplo, nas figuras 41 até 46, também é possível notar bons modelos na categoria A.

⁵² São apresentados também os testes de Bartlett e Fligner-Killeen para a verificação da homogeneidade das variâncias, fundamental para decidir entre teste-t ou teste-t de Welch; e Shapiro-Wilk (tabela 20) para a verificação da normalidade, condição fundamental para a realização de testes paramétricos.

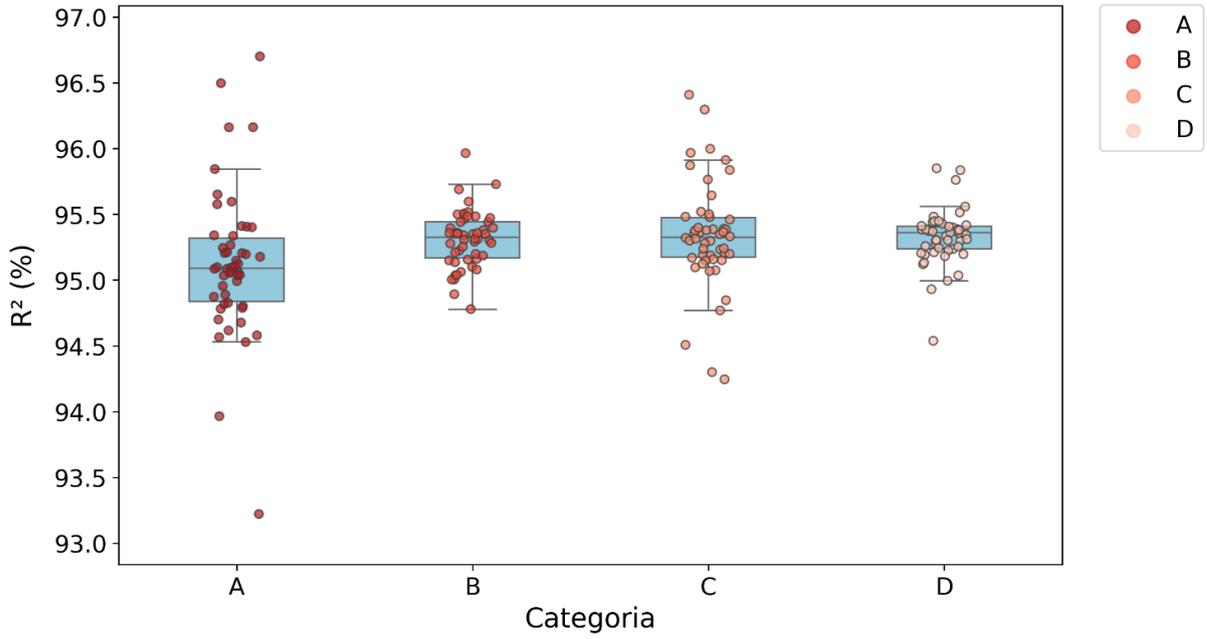
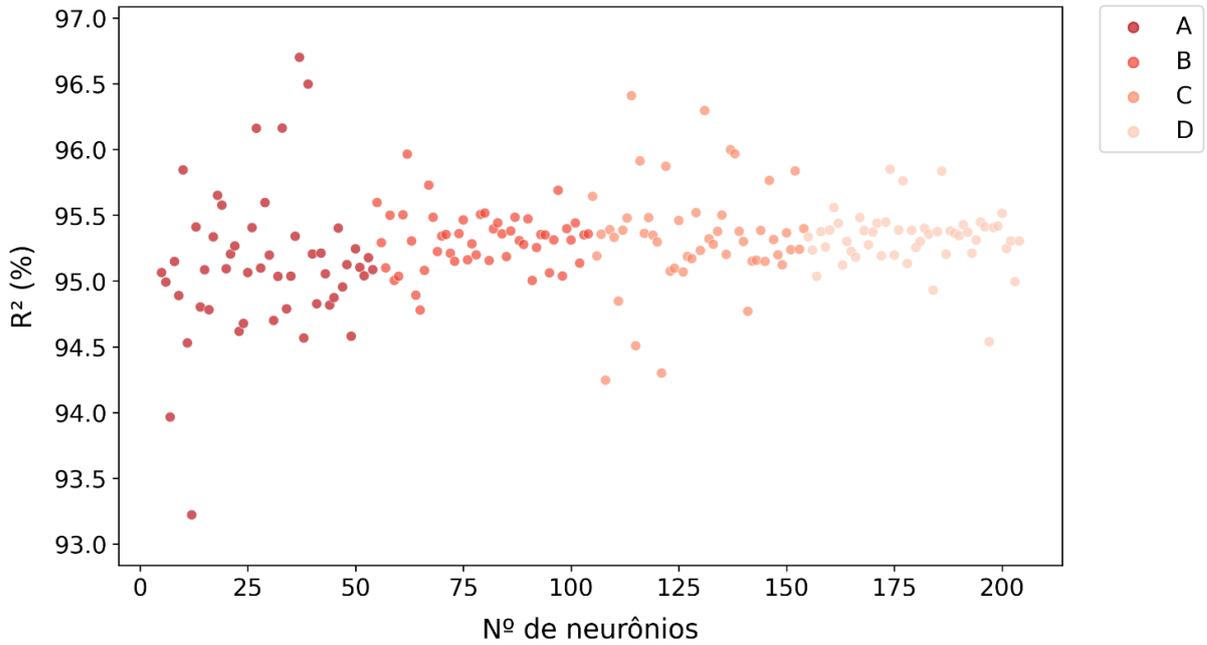
Figura 41 – $R^2 \times$ categorias (1HLES)Figura 42 – $R^2 \times$ N° de neurônios (1HLES)

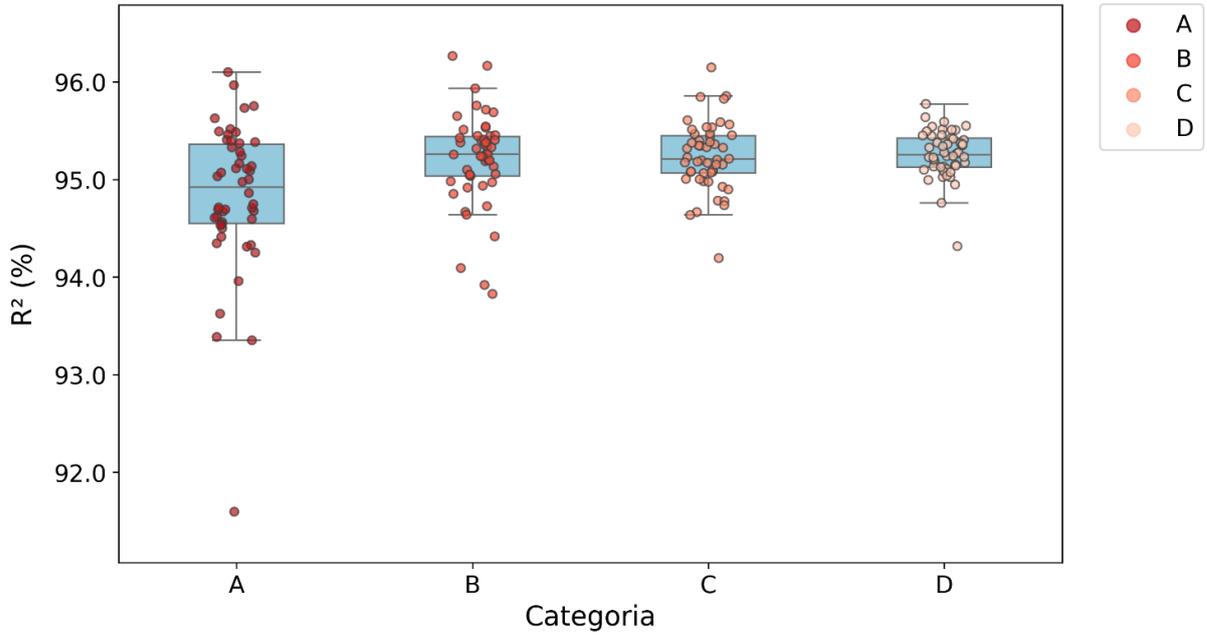
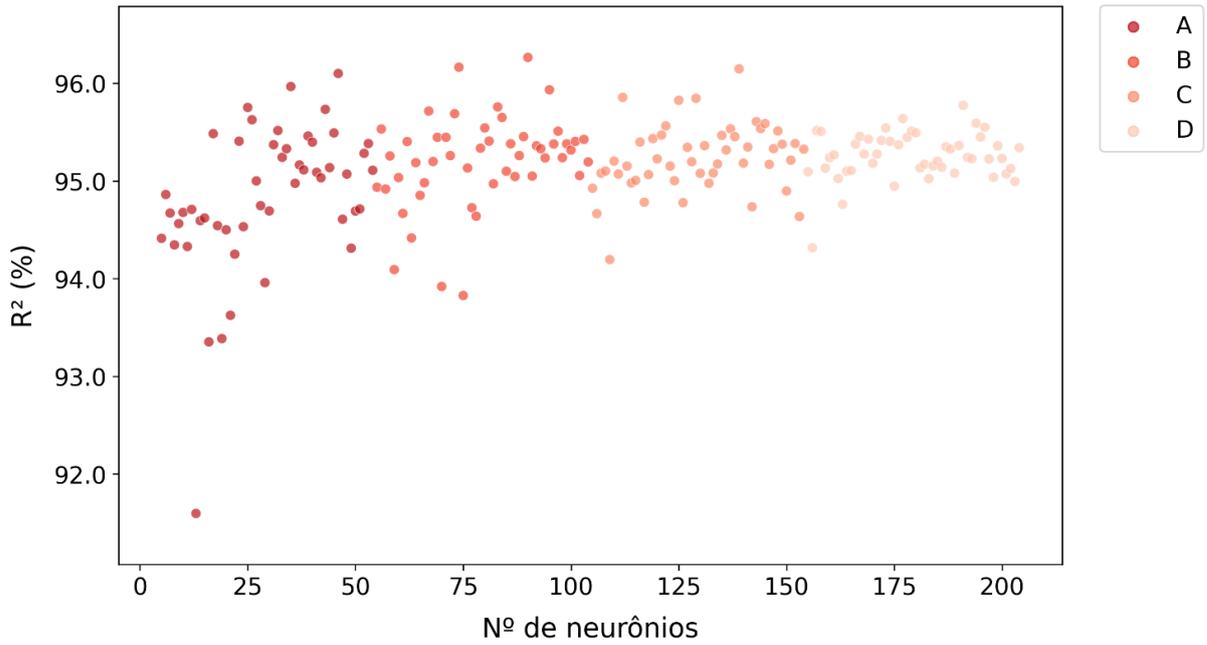
Figura 43 – $R^2 \times$ categorias (2HLES)Figura 44 – $R^2 \times$ N° de neurônios (2HLES)

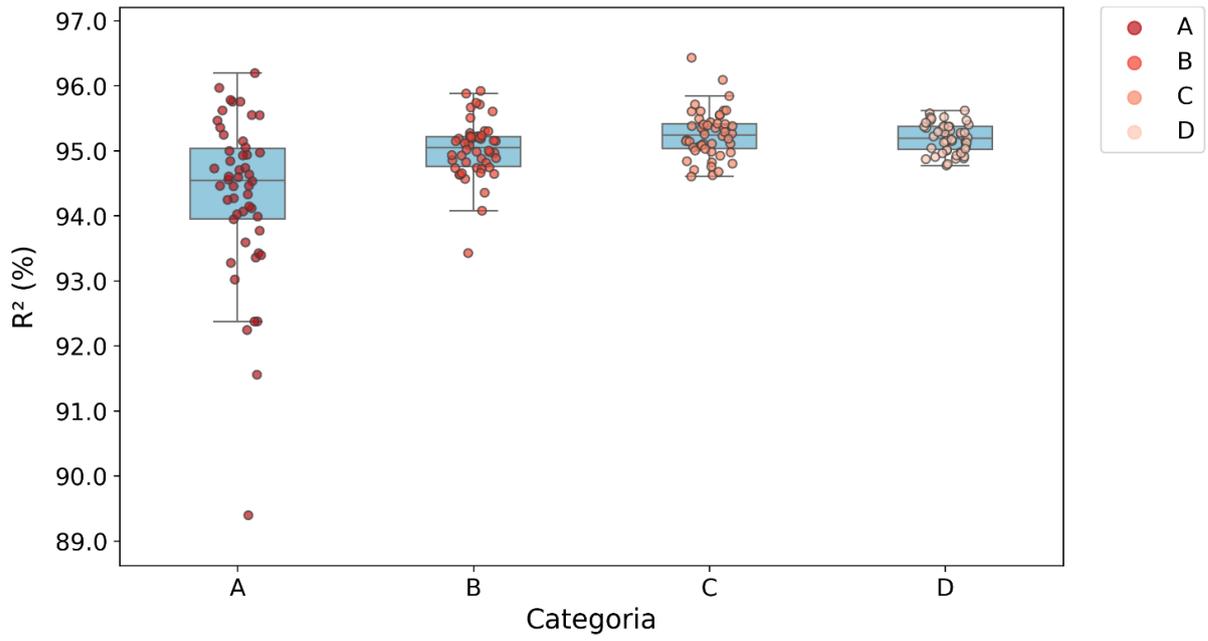
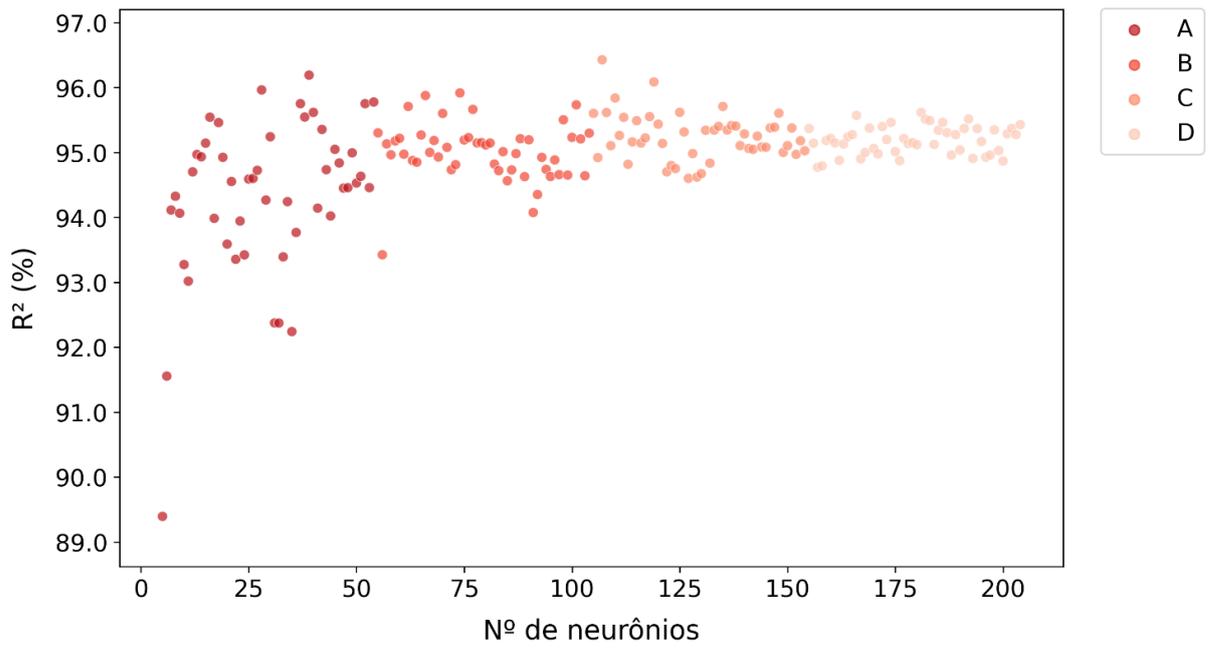
Figura 45 – R^2 x categorias (3HLES)Figura 46 – $R^2 \times N^\circ$ de neurônios (3HLES)

Tabela 19 – Análise exploratória de dados por categoria em HLES

Cenário	Cat.	Média	Mín.	25%	50%	75%	Máx.	A	(σ)	(σ^2)
1HLES	A	95,1239	93,2218	94,8380	95,0894	95,3180	96,7000	3,4782	0,5609	0,3147
	B	95,3108	94,7789	95,1668	95,3267	95,4417	95,9643	1,1855	0,2152	0,0463
	C	95,3374	94,2457	95,1739	95,3261	95,4740	96,4086	2,1629	0,4088	0,1671
	D	95,3280	94,5382	95,2375	95,3594	95,4101	95,8495	1,3113	0,2086	0,0435
2HLES	A	94,8304	91,5965	94,5476	94,9184	95,3605	96,0997	4,5032	0,7588	0,5757
	B	95,2081	93,8273	95,0370	95,2593	95,4408	96,2643	2,4370	0,4770	0,2275
	C	95,2346	94,1949	95,0664	95,2079	95,4479	96,1474	1,9525	0,3482	0,1213
	D	95,2595	94,3165	95,1275	95,2531	95,4241	95,7739	1,4574	0,2425	0,0588
3HLES	A	94,3270	89,3928	93,9541	94,5412	95,0352	96,1919	6,7990	1,2362	1,5282
	B	95,0218	93,4241	94,7582	95,0459	95,2169	95,9176	2,4935	0,4388	0,1926
	C	95,2441	94,6010	95,0323	95,2382	95,4139	96,4281	1,8271	0,3628	0,1317
	D	95,1916	94,7716	95,0168	95,1916	95,3695	95,6167	0,8451	0,2172	0,0472

Tabela 20 – Testes de normalidade por categorias em HLES

Categoria	Shapiro-Wilk (P-valor)		
	1HLES	2HLES	3HLES
A	0,0011	0,0001	0,0003
B	0,5365	0,0064	0,0111
C	0,0034	0,6520	0,1307
D	0,0003	0,0092	0,4399

Tabela 21 – Testes para comparação entre as categorias em HLES

Cenário	Cat.	Bartlett	Fligner-Killeen	Teste-t ou teste-t de Welch		Mann-Whitney	
				Unilateral (<)	Unilateral (>)	Unilateral (<)	Unilateral (>)
1HLES	A-B	-	0,0056	-	-	0,0004	0,9996
	B-C	-	0,0802	-	-	0,3691	0,6335
	C-D	-	0,0057	-	-	0,3434	0,6592
2HLES	A-B	-	0,0056	-	-	0,0020	0,9980
	B-C	-	0,4235	-	-	0,5535	0,4493
	C-D	-	0,0272	-	-	0,2641	0,7382
3HLES	A-B	-	0,0000	-	-	0,0002	0,9998
	B-C	-	0,5297	-	-	0,0044	0,9956
	C-D	0,0005	-	0,8089	0,1911	-	-

5.11.1.3 Cenários com *Pruning*: 1HLPR, 2HLPR e 3HLPR

Nas figuras (47 a 52), é possível notar que, em comparação ao *Early Stopping*, o *Pruning* apresenta capacidade preditiva inferior nas situações iniciais (poucos neurônios). Isso acontece porque a técnica do *Pruning* não é recomendada quando há poucas conexões neurais (não há motivos para podar algo que já é pequeno). Nelas é possível observar que a categoria com menos neurônios (A) é a que apresenta a maior dispersão. Essa observação é ratificada na análise exploratória de dados (tabela 22), que apresenta as maiores medidas de dispersão em A, onde também ocorrem os menores R^2 médios e mínimos.

Além disso, essas figuras indicam que a melhoria na capacidade preditiva das RNA ocorre essencialmente na transição da categoria A para B e, a partir daí, os resultados tendem à estabilidade. Por fim, são realizados testes para verificar a ocorrência de diferenças significativas entre as categorias. Conforme os testes (teste-t e Mann-Whitney) unilaterais apresentados na tabela 24 é possível afirmar a um nível de significância de 5% que:

- Em 1HLPR, os testes de Mann-Whitney apontam probabilidade significativa de um valor de A ser menor que um valor de B e igualdade estocástica entre as categorias B e C, assim como entre C e D;
- Em 2HLPR, entre todas as categorias analisadas, os testes de Mann-Whitney mostram que há probabilidade significativa de um valor da categoria anterior ser menor que um valor da categoria posterior;
- Em 3HLPR, os testes de Mann-Whitney indicam probabilidade significativa de um valor de A ser menor que um valor de B, assim como um de B ser menor que um de C. Por fim, há igualdade estocástica entre C e D.

Cumprir destacar, no entanto, que tais testes não indicam que modelos com um maior número de neurônios serão invariavelmente superiores, nem que os modelos com menos neurônios devam ser descartados. Por exemplo, nas figuras 47 até 52, também é possível notar bons modelos na categoria A.

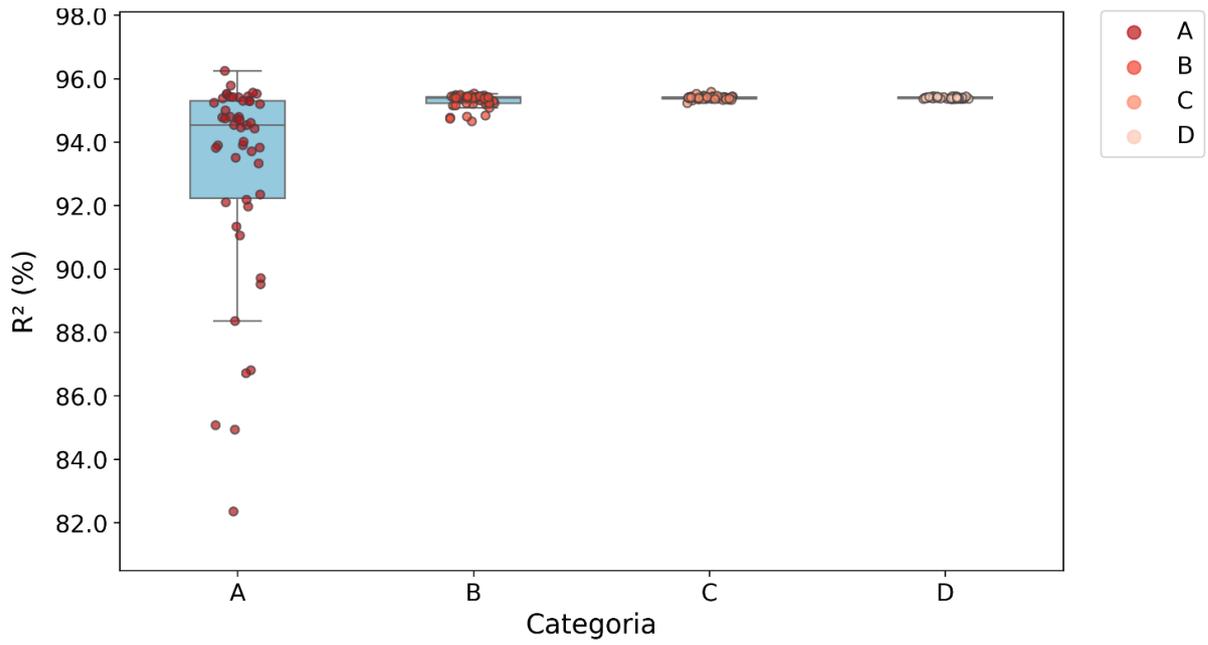
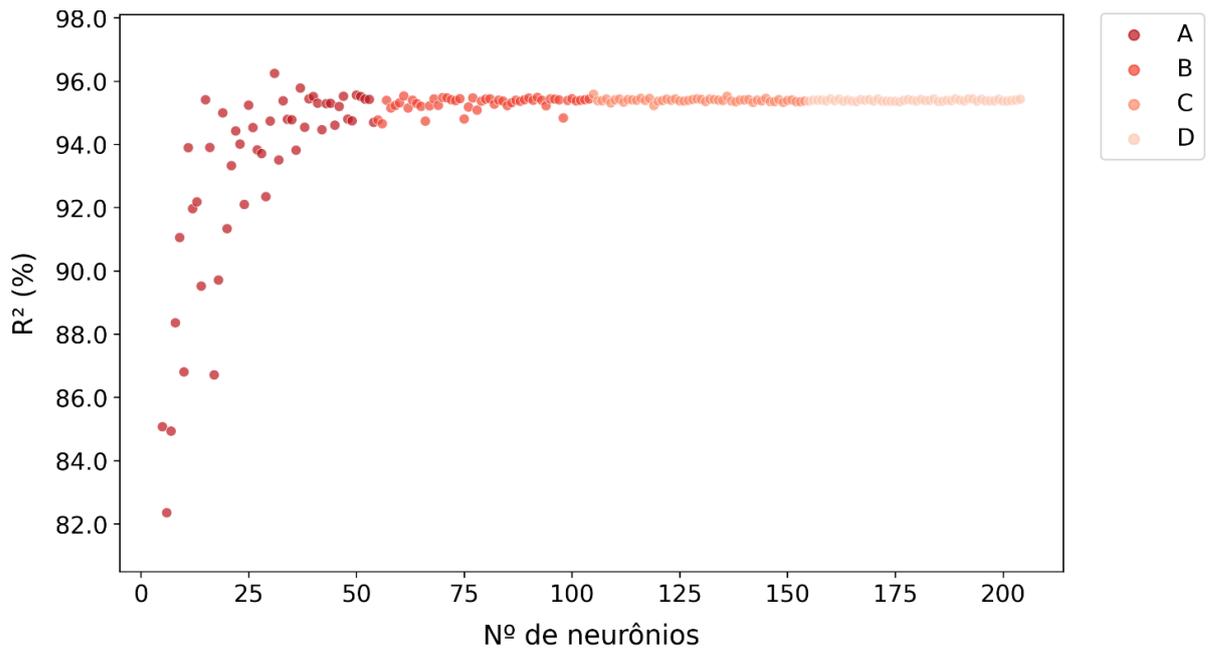
Figura 47 – $R^2 \times$ categorias (1HLPR)Figura 48 – $R^2 \times$ N° de neurônios (1HLPR)

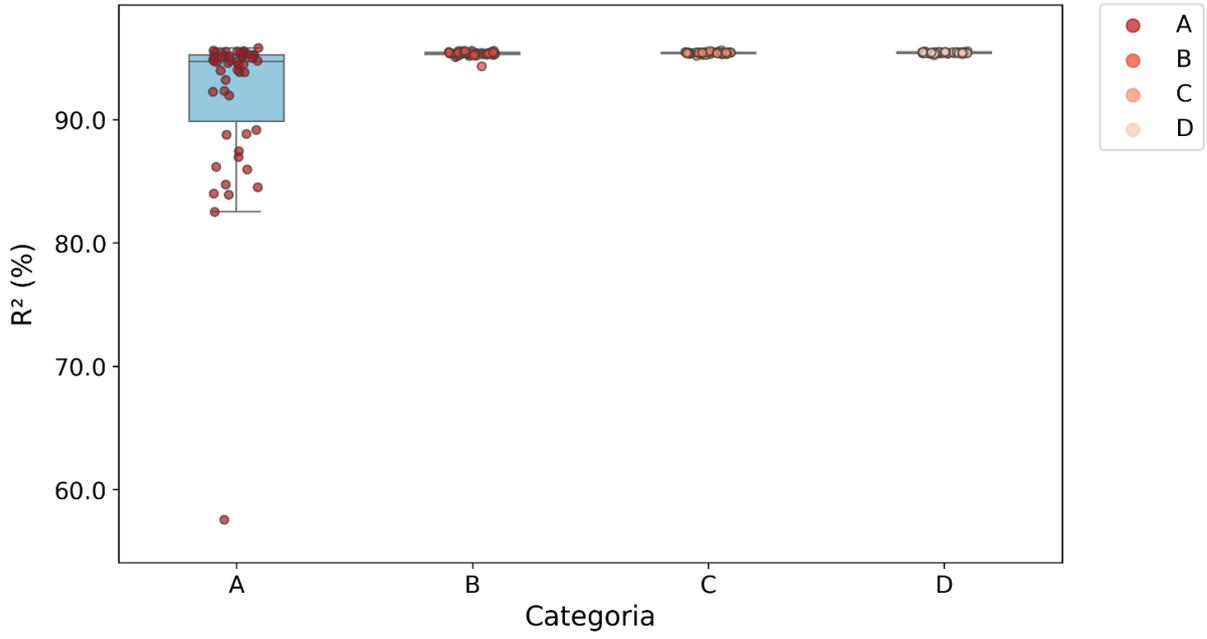
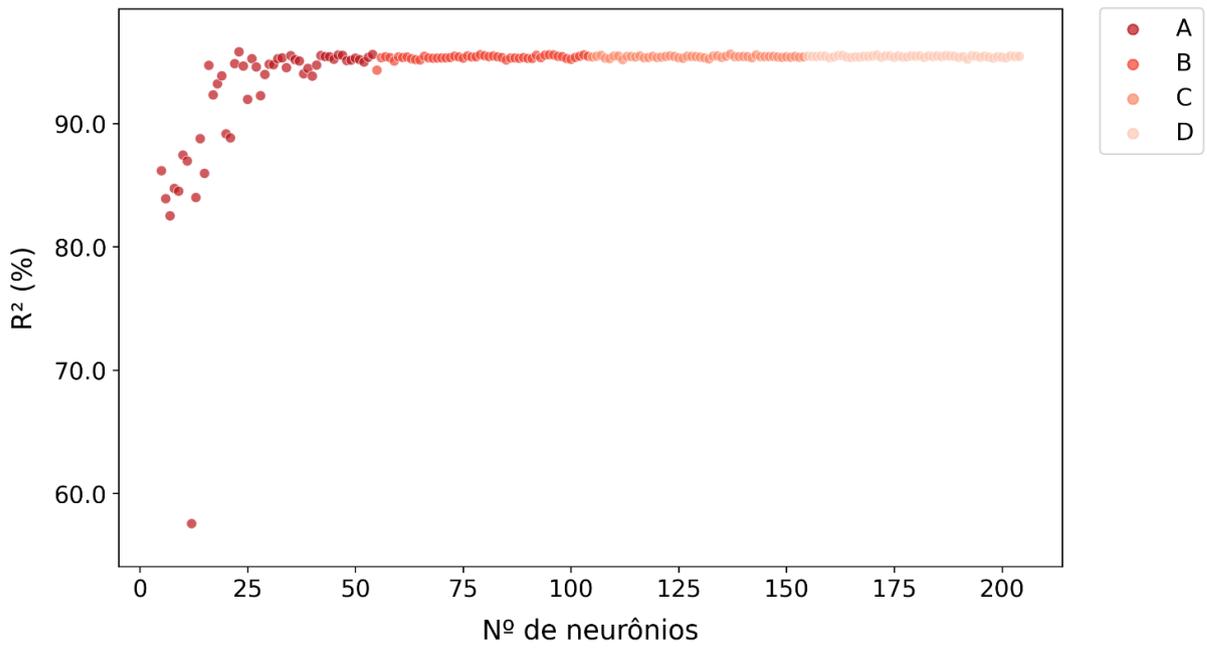
Figura 49 – $R^2 \times$ categorias (2HLPR)Figura 50 – $R^2 \times$ N° de neurônios (2HLPR)

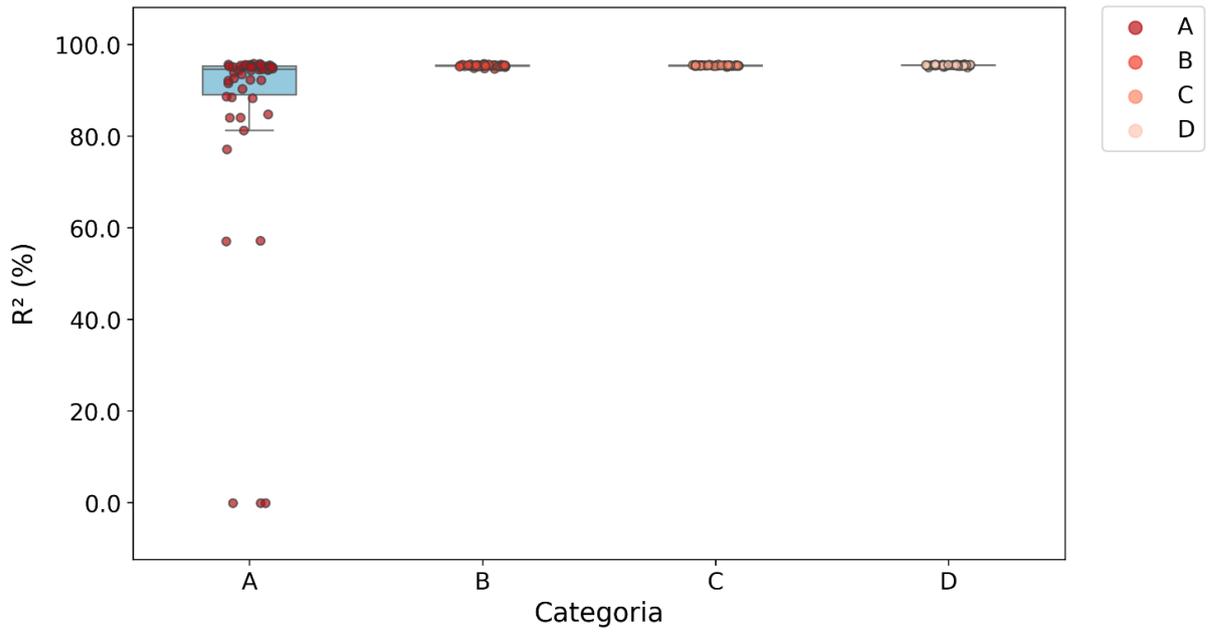
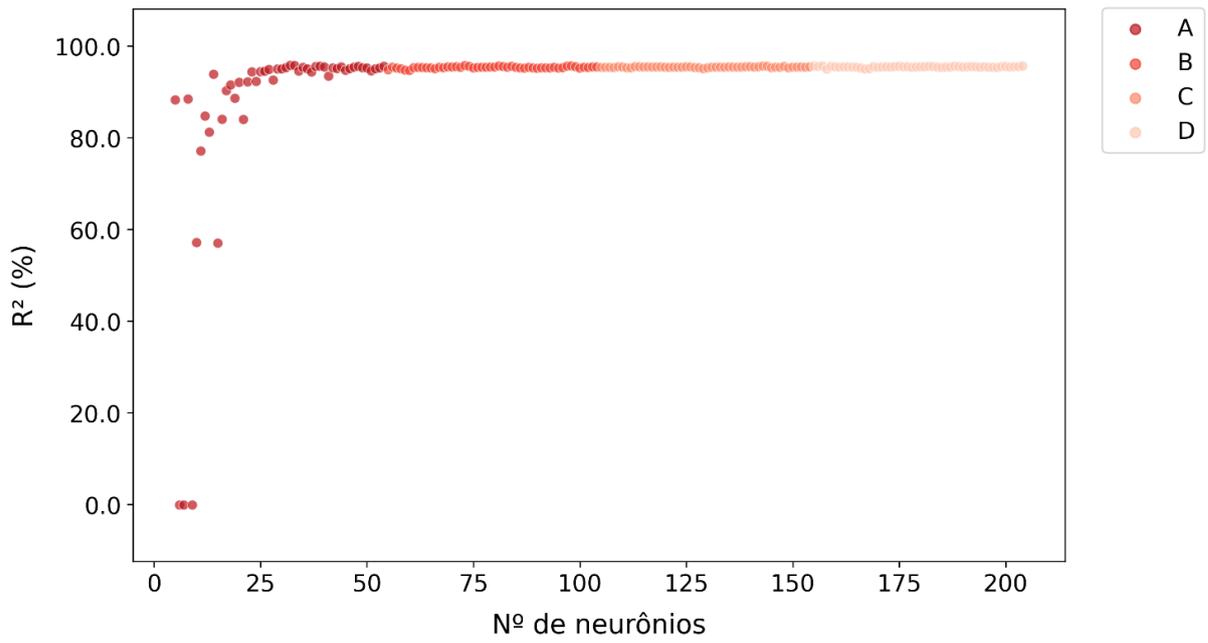
Figura 51 – $R^2 \times$ categorias (3HLPR)Figura 52 – $R^2 \times$ N° de neurônios (3HLPR)

Tabela 22 – Análise exploratória de dados por categoria em HLPR

Cenário	Cat.	Média	Mín.	25%	50%	75%	Máx.	A	(σ)	(σ^2)
1HLPR	A	93,1589	82,3503	92,2203	94,5332	95,2934	96,2410	13,8907	3,2243	10,3963
	B	95,2963	94,6502	95,2254	95,3881	95,4299	95,5269	0,8766	0,2083	0,0434
	C	95,3929	95,2247	95,3663	95,3923	95,4159	95,5808	0,3561	0,0534	0,0028
	D	95,3898	95,3453	95,3678	95,3874	95,4136	95,4369	0,0916	0,0268	0,0007
2HLPR	A	91,8983	57,5440	89,8602	94,6906	95,2471	95,8173	38,2732	6,3338	40,1172
	B	95,3532	94,3271	95,3073	95,3553	95,4515	95,5796	1,2525	0,1869	0,0349
	C	95,4114	95,1863	95,3707	95,4275	95,4509	95,6183	0,4320	0,0763	0,0058
	D	95,4307	95,2262	95,3897	95,4433	95,4774	95,5448	0,3186	0,0593	0,0035
3HLPR	A	85,7463	0,0000	89,0284	94,5290	95,2347	95,8024	95,8024	23,3556	545,4861
	B	95,3312	94,6712	95,2468	95,3553	95,4472	95,7574	1,0862	0,2098	0,0440
	C	95,4114	95,0877	95,3638	95,4225	95,4657	95,6401	0,5524	0,0950	0,0090
	D	95,4490	94,9870	95,4066	95,4770	95,5181	95,6560	0,6690	0,1404	0,0197

Tabela 23 – Testes de normalidade por categorias em HLPR

Métrica	Categoria	Shapiro-Wilk (P-valor)		
		RNA (1HLPR)	RNA (2HLPR)	RNA (3HLPR)
R ² (%)	A	0,0000	0,0000	0,0000
	B	0,0000	0,0000	0,0056
	C	0,0058	0,0838	0,0684
	D	0,0081	0,0309	0,0000

Tabela 24 – Testes para comparação entre as categorias em HLPR

Cenário	Cat.	Bartlett	Fligner-Killeen	Teste-t ou teste-t de Welch		Mann-Whitney	
				Unilateral (<)	Unilateral (>)	Unilateral (<)	Unilateral (>)
1HLPR	A-B	-	0,0000	-	-	0,0000	1,0000
	B-C	-	0,0003	-	-	0,0793	0,9217
	C-D	-	0,0240	-	-	0,6717	0,3308
2HLPR	A-B	-	0,0000	-	-	0,0000	1,0000
	B-C	-	0,0029	-	-	0,0195	0,9808
	C-D	-	0,5242	-	-	0,0480	0,9527
3HLPR	A-B	-	0,0000	-	-	0,0028	0,9972
	B-C	-	0,0005	-	-	0,0081	0,9919
	C-D	-	0,2591	-	-	0,0607	0,9393

5.11.1.4 Cenários sem Early Stopping ou Pruning: 1HL, 2HL e 3HL

Nas figuras 53 a 58 é possível notar que a categoria A é a que apresenta a maior dispersão, comportamento confirmado na análise exploratória de dados (tabela 25), que apresenta as maiores medidas de dispersão na categoria A.

Além disso, é observado um comportamento peculiar nos cenários HL, uma vez que os valores médios de R^2 (tabela 25) são de modo geral inferiores aos observados em HLES e HLPR. Isso sugere uma capacidade preditiva inferior dos modelos no cenário HL em relação aos demais.

Conforme os testes unilaterais apresentados na tabela 27 é possível afirmar, a um nível de significância de 5% que:

- Em 1HL, entre todas as categorias, os testes-t indicam que a média da categoria anterior é significativamente menor que a média da categoria posterior;
- Em 2HL, o teste-t de Welch indica que a média da categoria A é significativamente maior que a média de B, ou seja, há piora nos resultados. O teste-t entre B e C indica probabilidade significativa da média em B ser menor que a média em C; por fim, Mann-Whitney indica igualdade estocástica entre C e D, indicando estabilização;
- Em 3HL, os testes de Mann-Whitney demonstram haver probabilidade significativa de um valor de A ser maior que um valor de B; e probabilidade significativa de um valor de B ser menor que um de C, assim como entre C e D.

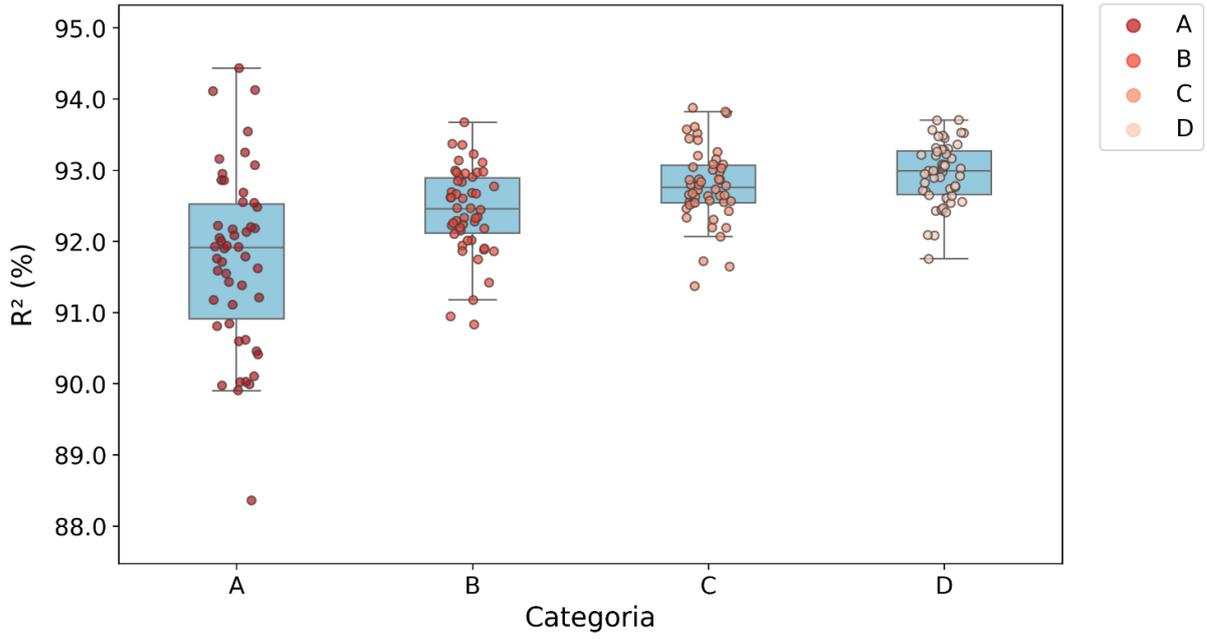
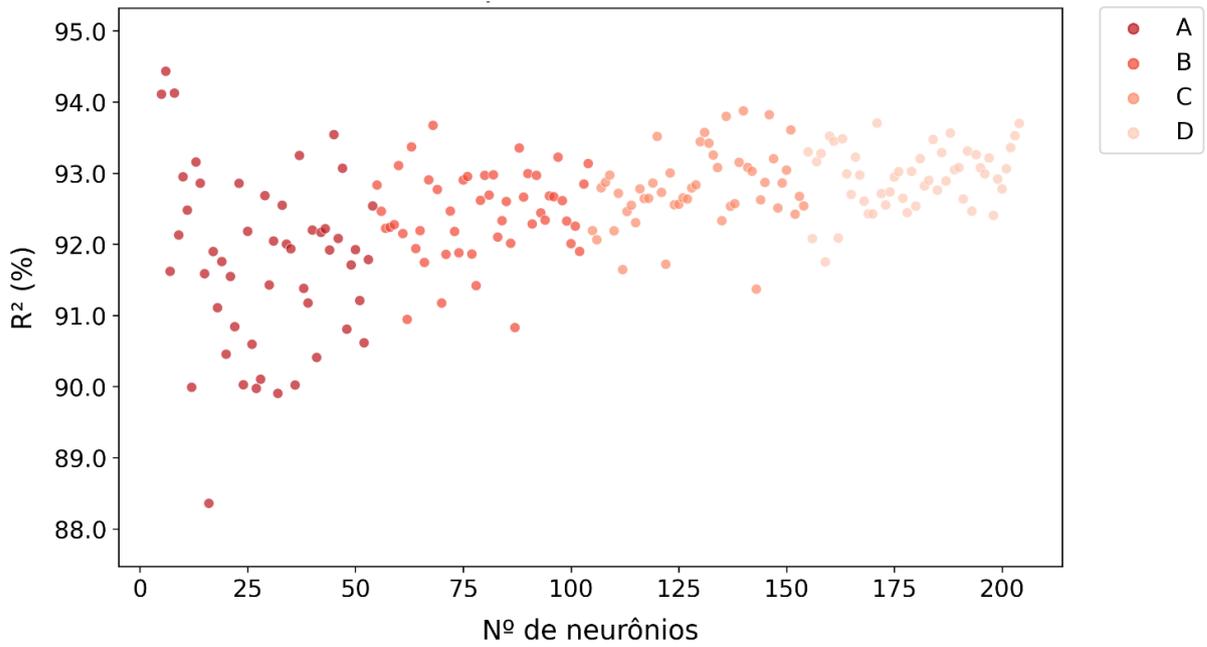
Figura 53 – $R^2 \times$ categorias (1HL)Figura 54 – $R^2 \times$ N° de neurônios (1HL)

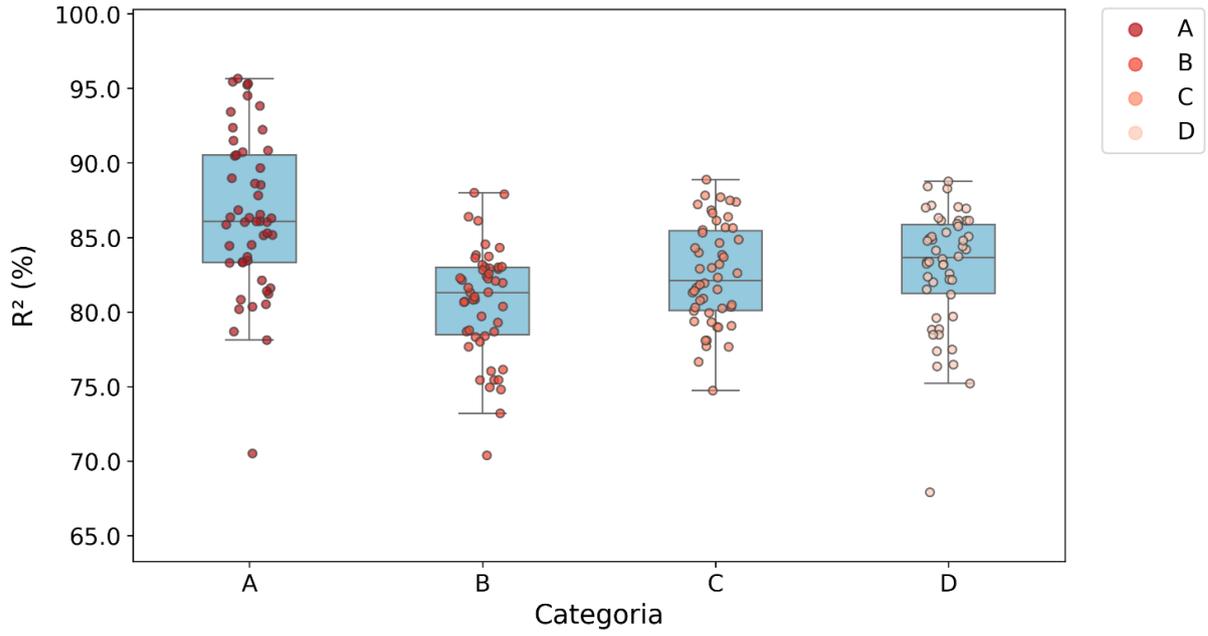
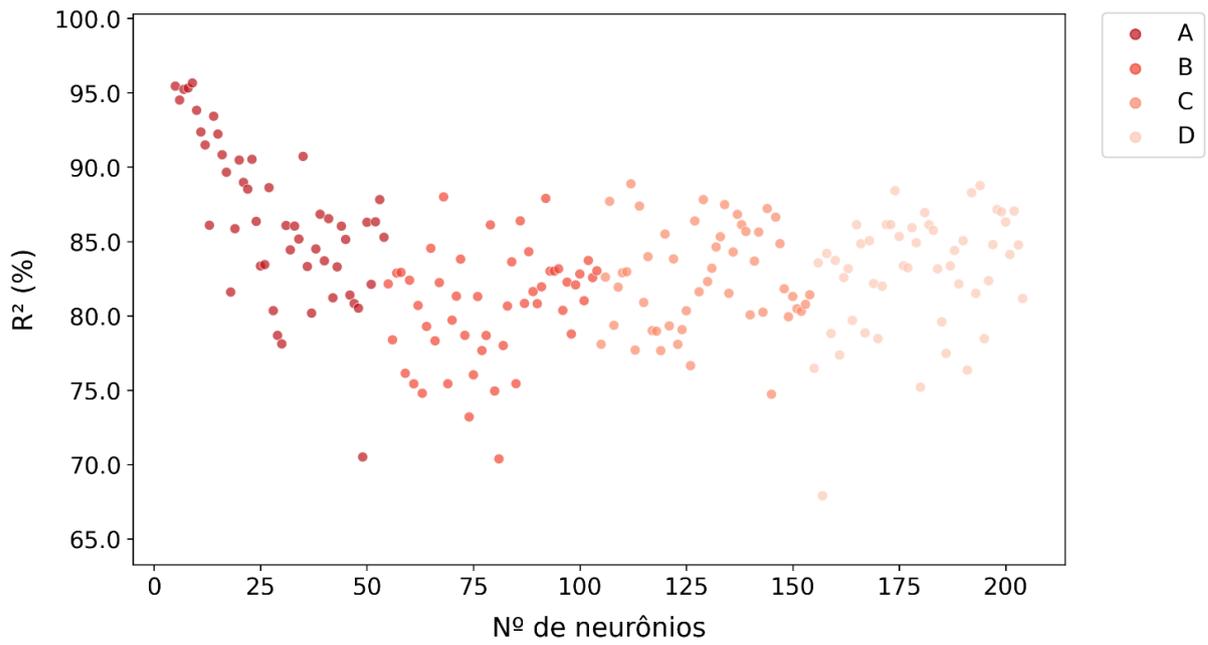
Figura 55 – $R^2 \times$ categorias (2HL)Figura 56 – $R^2 \times$ N° de neurônios (2HL)

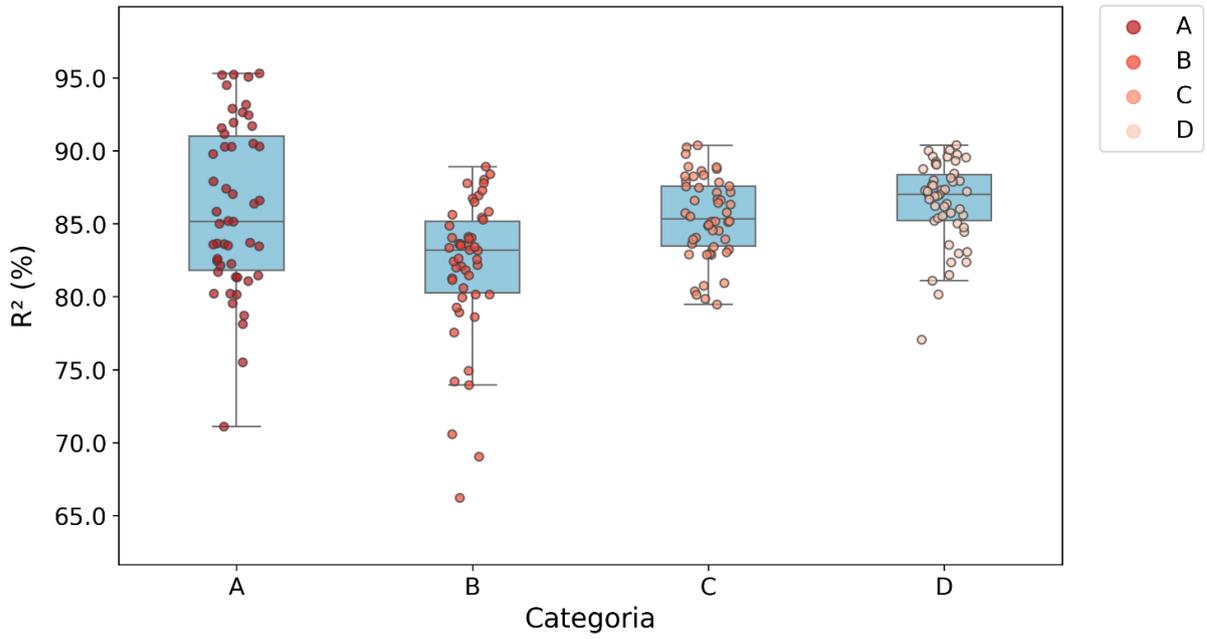
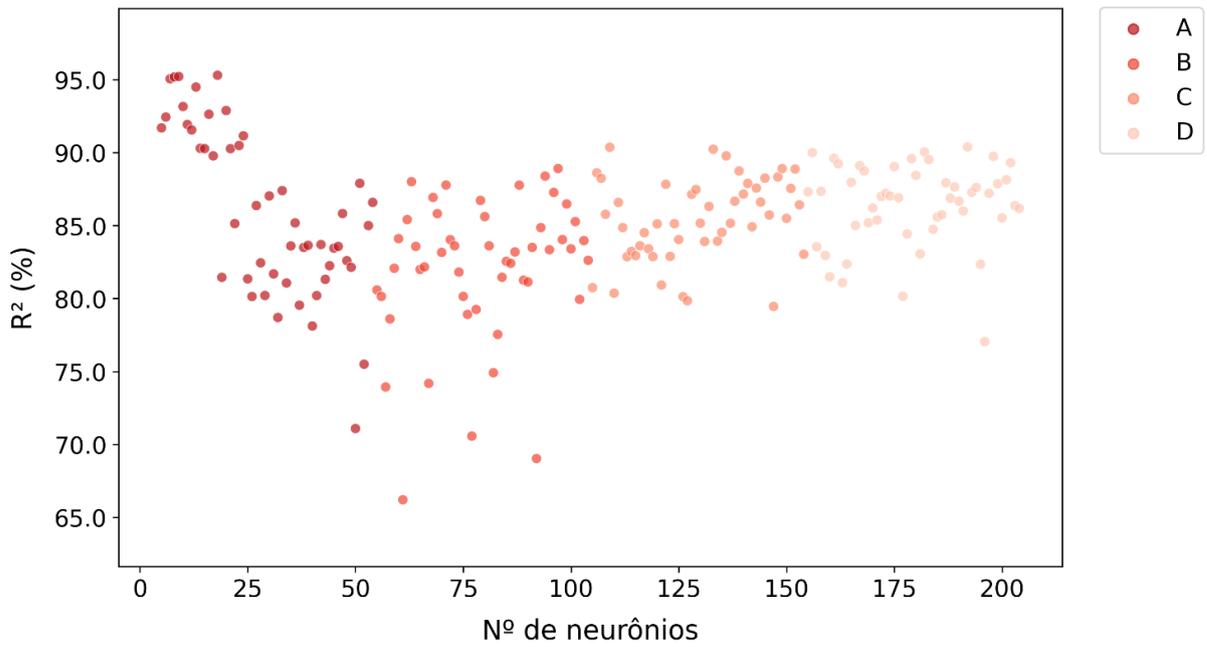
Figura 57 – $R^2 \times$ categorias (3HL)Figura 58 – $R^2 \times$ N° de neurônios (3HL)

Tabela 25 – Análise exploratória de dados por categoria em HL

Cenário	Cat.	Média	Mín.	25%	50%	75%	Máx.	A	(σ)	(σ^2)
1HL	A	91,7533	88,3577	90,9075	91,9079	92,5232	94,4313	6,0736	1,2325	1,5191
	B	92,4264	90,8286	92,1128	92,4517	92,8881	93,6709	2,8424	0,6015	0,3618
	C	92,7864	91,3687	92,5408	92,7535	93,0683	93,8747	2,5060	0,5299	0,2808
	D	92,9492	91,7505	92,6596	92,9892	93,2708	93,7024	1,9519	0,4330	0,1875
2HL	A	86,4989	70,4997	83,3206	86,0793	90,5096	95,6557	25,1560	5,2758	27,8340
	B	80,6501	70,3703	78,4524	81,3044	82,9772	87,9924	17,6221	3,7143	13,7957
	C	82,4954	74,7189	80,1018	82,1108	85,4491	88,8737	14,1548	3,3985	11,5498
	D	82,9087	67,8933	81,2468	83,6390	85,8777	88,7543	20,8610	4,0403	16,3237
3HL	A	86,0294	71,0847	81,7992	85,1584	90,9856	95,3042	24,2195	5,7249	32,7748
	B	82,0423	66,2001	80,2518	83,1750	85,1687	88,9085	22,7083	4,8789	23,8039
	C	85,4021	79,4550	83,4595	85,3297	87,5635	90,3718	10,9168	2,8169	7,9350
	D	86,4226	77,0428	85,2394	87,0125	88,3639	90,3884	13,3456	2,8509	8,1278

Tabela 26 – Teste de normalidade por categorias em HL

Métrica	Categoria	Shapiro-Wilk (P-valor)		
		RNA (1HL)	RNA (2HL)	RNA (3HL)
R ² (%)	A	0,6781	0,2121	0,1030
	B	0,2723	0,1896	0,0002
	C	0,2713	0,4247	0,2388
	D	0,4410	0,0009	0,0033

Tabela 27 – Testes para comparação entre as categorias em HL

Cenário	Cat.	Bartlett	Fligner-Killeen	Teste-t ou teste-t de Welch		Mann-Whitney	
				Unilateral (<)	Unilateral (>)	Unilateral (<)	Unilateral (>)
1HL	A-B	0,0000	-	0,0004	0,9996	-	-
	B-C	0,3786	-	0,0010	0,9990	-	-
	C-D	0,1608	-	0,0479	0,9521	-	-
2HL	A-B	0,0155	-	1,0000	0,0000	-	-
	B-C	0,5363	-	0,0055	0,9945	-	-
	C-D	-	0,9307	-	-	0,1769	0,8249
3HL	A-B	-	0,0195	-	-	0,9987	0,0013
	B-C	-	0,1698	-	-	0,0001	0,9999
	C-D	-	0,5297	-	-	0,0212	0,9792

Os cenários 1HL, 2HL e 3HL têm capacidade preditiva inferior quando comparados aos demais cenários (1HLES, 2HLES, 3HLES, 1HLPR, 2HLPR e 3HLPR). Essa capacidade preditiva inferior acontece porque os cenários HL são susceptíveis à ocorrência de *overfitting*, que degenera a capacidade da RNA de realizar boas predições.

Para mostrar graficamente o *overfitting*, são apresentadas as curvas de treinamento das RNA com 50 neurônios como exemplo (figuras 59, 60 e 61). É possível notar que a curva de validação se eleva durante as iterações, evidenciando um comportamento típico de *overfitting*.

Outro gráfico que pode ajudar a identificar o *overfitting* é o de predição (ver figuras 62, 63 e 64), onde são apresentados pontos pouco ajustados à bissetriz no caso da amostra de validação e pontos muito ajustados no caso da amostra de treinamento.

Figura 59 – Curva de treinamento × validação (1HL – $N_n = 50$)

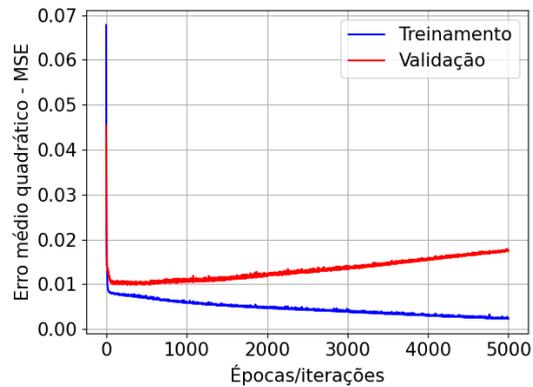


Figura 60 – Curva de treinamento × validação (2HL – $N_n = 50$)

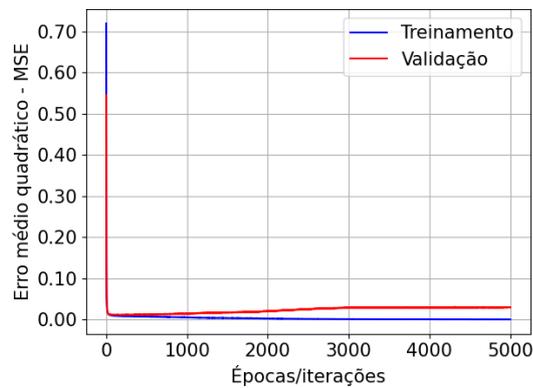


Figura 61 – Curva de treinamento × validação (3HL – $N_n = 50$)

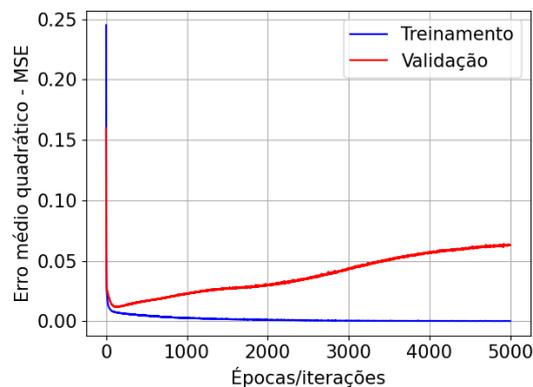


Figura 62 – Gráfico de predição da amostra de validação e treinamento (1HL – $N_n = 50$)

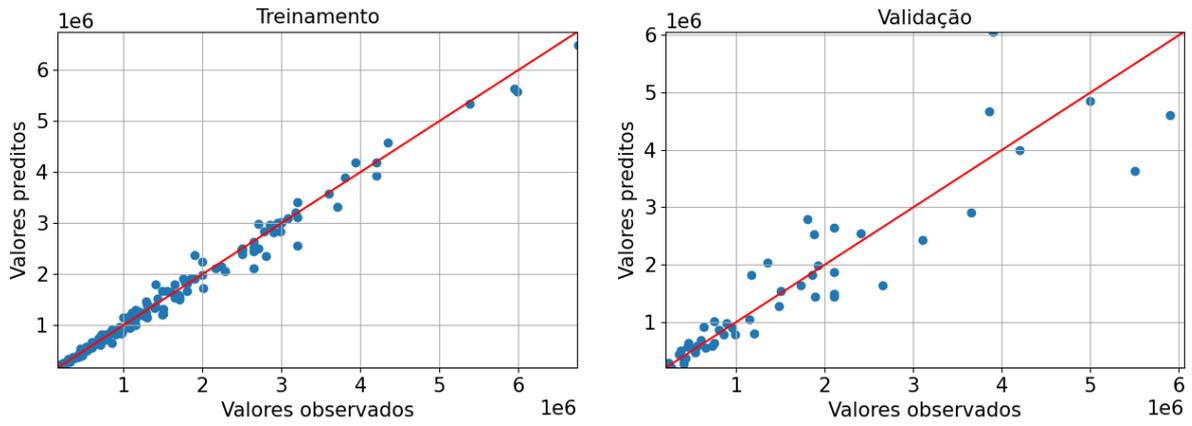


Figura 63 – Gráfico de predição da amostra de validação e treinamento (2HL – $N_n = 50$)

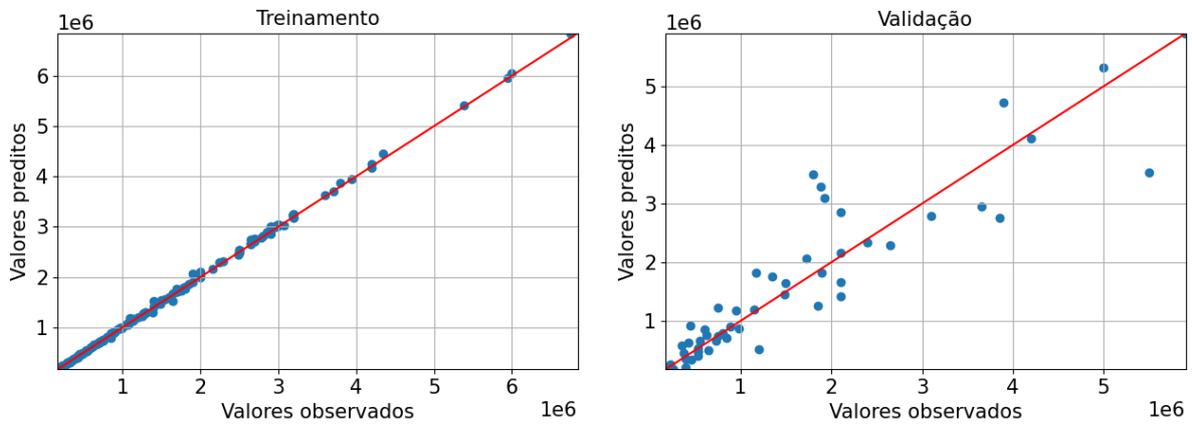
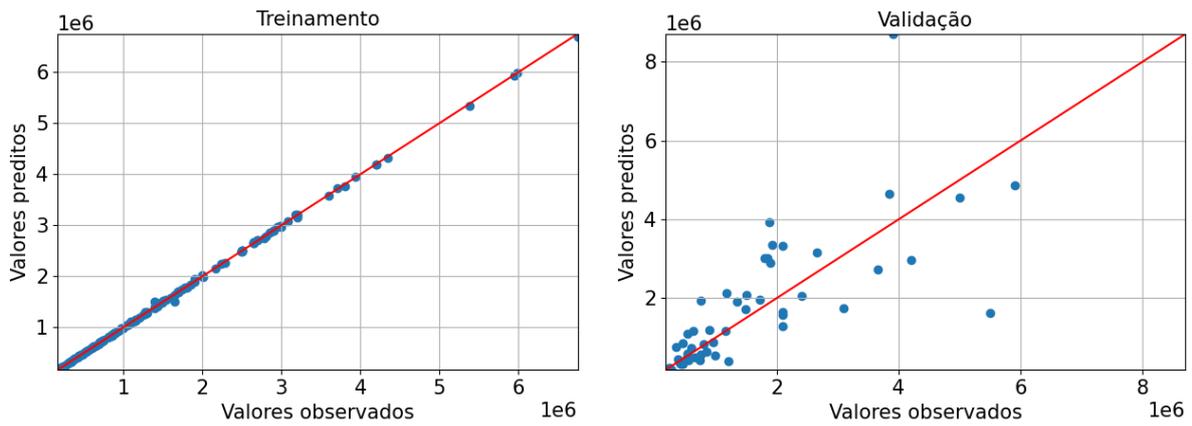


Figura 64 – Gráfico de predição da amostra de validação e treinamento (3HL – $N_n = 50$)



5.11.2 Influência do número de camadas ocultas na capacidade preditiva de RNA

Neste item são realizadas comparações usando a raiz do erro quadrático médio (RMSE⁵³) em escala monetária. Para isso, são utilizadas apenas RNA com *Early Stopping* e *Pruning* para evitar modelos com *overfitting*. São realizados testes de hipótese para verificação da variação do erro em função do incremento do número de camadas intermediárias. No item 5.11.2.1 são apresentados os casos com *Early Stopping* e no item 5.11.2.2 os casos com *Pruning*.

5.11.2.1 Comparações entre 1HLES, 2HLES e 3HLES

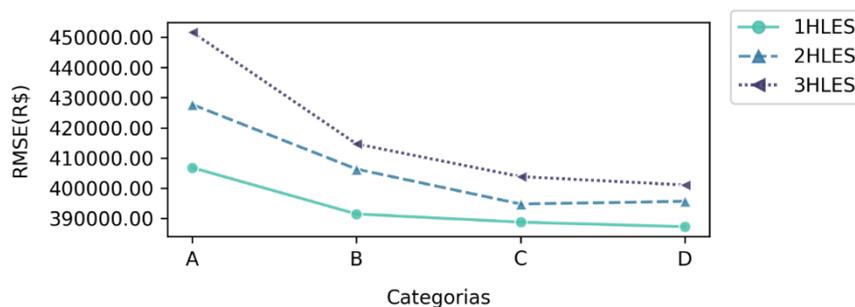
A tabela 28 apresenta os RMSE médios conforme a categoria e a quantidade de *hidden layers*. É possível notar que os maiores erros médios ocorrem na categoria com menos neurônios (A) e, à medida em que se aumenta o número de neurônios, há uma tendência de melhoria nos resultados médios.

Tabela 28 – RMSE médios nas RNA com *Early Stopping*

Métrica	Categoria	RNA (1HLES)	RNA (2HLES)	RNA (3HLES)
RMSE (R\$)	A	406.790,63	427.734,72	451.696,74
	B	391.432,08	406.322,47	414.662,96
	C	388.727,20	394.734,35	403.818,81
	D	387.206,67	395.626,95	401.055,26

Os resultados da tabela 28 estão sumarizados na figura 65, onde se percebe que a maior variação ocorre entre as categorias A e B. O cenário com uma camada 1HLES é o que apresenta os menores erros médios, seguido pelo cenário 2HLES e, por fim, o cenário com três *layers* 3HLES é o que apresenta os maiores erros médios.

Figura 65 – RMSE médios nas RNA com *Early Stopping*



53 Os resultados para cada modelo e cenário simulados constam no apêndice E.

Para ratificar tais observações são realizados os testes-t e de Mann-Whitney para comparações entre modelos com diferentes números de camadas. Conforme os resultados apresentados na tabela 30, é possível afirmar que os testes-t indicam diferenças significativas a um nível de significância de 5%, de modo que a média dos erros em RNA com menos camadas é menor do que em RNA com mais camadas e; os testes de Mann-Whitney indicam probabilidade significativa de que um modelo com menos camadas apresente erro menor do que um modelo com mais camadas.

Esses resultados não implicam que modelos com duas ou três camadas sejam inviáveis, nem que modelos de uma única camada são sempre os melhores. Por exemplo, entre os modelos que apresentam os cinco menores RMSE, constam os seguintes casos: $N_n=46$ em 2 camadas; $N_n=112$ em 2 camadas; $N_n=54$ em 3 camadas; $N_n=39$ em 1 camada; $N_n=119$ em 3 camadas. É possível observar que bons modelos foram obtidos com 1, 2 e 3 camadas, bem como com diferentes quantidades de neurônios. Isso sugere que uma abordagem interessante para encontrar bons modelos é explorar livremente diversas combinações de parâmetros, sem a necessidade de manter um parâmetro constante enquanto se varia o outro.

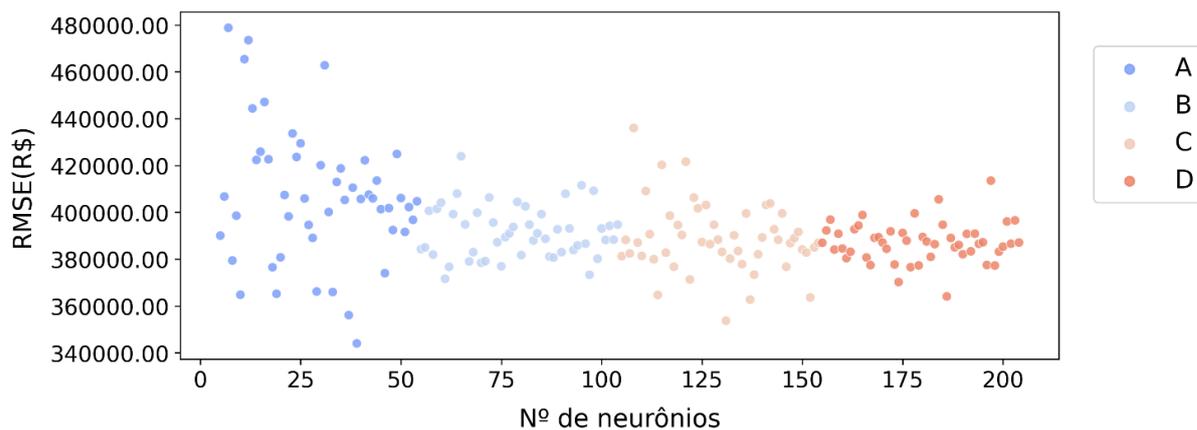
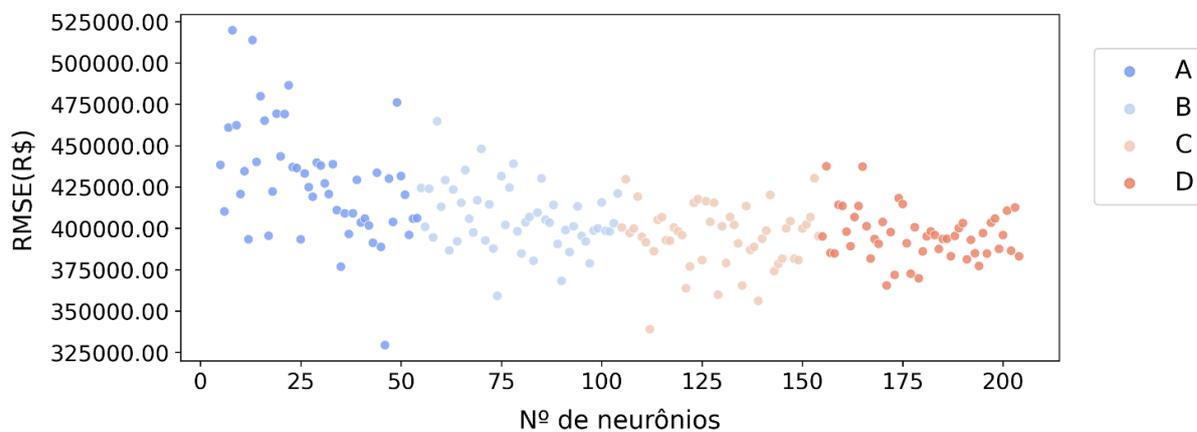
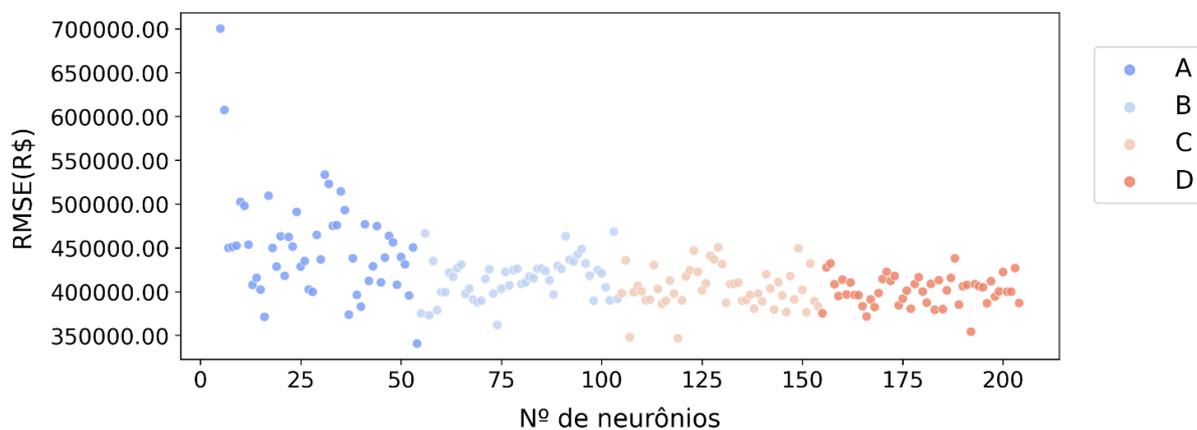
Tabela 29 – Teste de normalidade por categorias em HLES

Métrica	Categoria	Shapiro-Wilk (P-valor)		
		RNA (1HLES)	RNA (2HLES)	RNA (3HLES)
RMSE (R\$)	A	0,1895	0,0690	0,0001
	B	0,3078	0,6020	0,5279
	C	0,0491	0,1998	0,1521
	D	0,2379	0,1290	0,9738

Tabela 30 – Testes para comparação entre as camadas em HLES

Cat.	Camadas	Bartlett	Fligner-Killeen	Teste-t ou teste-t de Welch		Mann-Whitney	
				Unilateral (<)	Unilateral (>)	Unilateral (<)	Unilateral (>)
A	1-2	0,2244	-	0,0007	0,9993	-	-
	2-3	-	0,0449	-	-	0,0093	0,9909
B	1-2	0,0001	-	0,0000	1,0000	-	-
	2-3	0,2851	-	0,0275	0,9725	-	-
C	1-2	-	0,1457	-	-	0,0089	0,9912
	2-3	0,1450	-	0,0157	0,9843	-	-
D	1-2	0,0001	-	0,0004	0,9996	-	-
	2-3	0,4374	-	0,0453	0,9547	-	-

As figuras 66, 67 e 68 plotam os erros de cada RNA simulada. Os erros da categoria inicial são maiores e mais dispersos, porém à medida que se eleva a quantidade de neurônios, os erros diminuem, assim como a dispersão.

Figura 66 – RMSE \times N $^\circ$ de neur\u00f4nios (1HLES)Figura 67 – RMSE \times N $^\circ$ de neur\u00f4nios (2HLES)Figura 68 – RMSE \times N $^\circ$ de neur\u00f4nios (3HLES)

5.11.2.2 Comparações entre 1HLPR, 2HLPR e 3HLPR

A tabela 31 apresenta os RMSE médios conforme a categoria e a quantidade de *hidden layers*. É possível perceber que os erros na categoria A destoam das demais categorias. Isso acontece porque o *pruning* (poda) não é recomendado em redes neurais pequenas, pois não é sensato podar modelos que naturalmente já têm poucas conexões; além disso, os primeiros modelos dessa categoria possuem capacidade preditiva muito menor que os demais, funcionando como *outliers* que contribuem para elevar o erro médio.

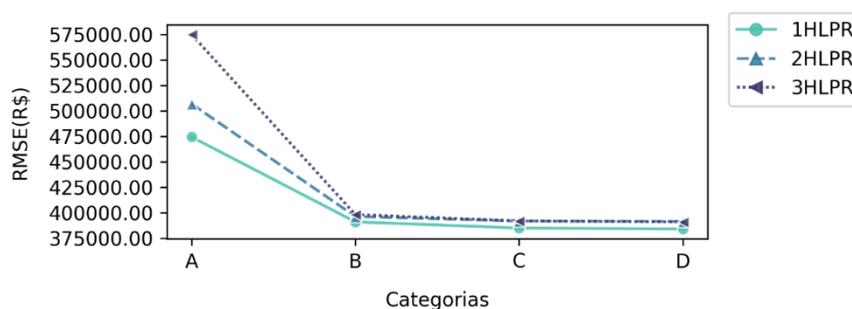
Os erros diminuem nas demais categorias, como observado na tabela 31 e figura 69, sendo que essa diminuição acontece de modo mais expressivo na transição de A para B e, a partir daí os resultados indicam variações menores entre as demais categorias.

Tabela 31 – RMSE médios nas RNA com *Pruning*

Métrica	Categoria	RNA (1HLPR)	RNA (2HLPR)	RNA (3HLPR)
RMSE (R\$)	A	474.226,95	506.681,85	574.799,48
	B	390.930,33	396.202,31	398.073,47
	C	384.900,74	391.774,84	391.852,50
	D	383.863,14	391.302,45	390.882,63

Os resultados da tabela 31 estão sumarizados na figura 69, onde o cenário com apenas uma camada (1HLPR) é o que alcança os menores erros médios. Além disso, essa figura indica que, o aumento no número de camadas intermediárias, não trouxe melhorias para os resultados.

Figura 69 – RMSE médios nas RNA com *Pruning*



Para confirmar tais observações são realizados os testes-t e de Mann-Whitney para comparações entre modelos com diferentes números de camadas. Conforme os resultados apresentados na tabela 33 é possível afirmar a um nível de significância de 5% que os testes-t indicam que a média dos erros em RNA com menos camadas é significativamente menor do que em RNA

com mais camadas ou que o aumento de camadas não altera de modo significativo os resultados e; os testes de Mann-Whitney apontam probabilidade significativa de que um modelo com menos camadas apresente erro menor do que um modelo com mais camadas ou que o aumento de camadas não resulta em diferenças significativas, ao contrário, em igualdade estocástica.

Esses resultados não implicam que modelos com duas ou três camadas sejam inviáveis, nem que modelos de uma única camada são sempre os melhores. Por exemplo, entre os modelos que apresentam os cinco menores RMSE, constam os seguintes casos: $N_n=98$ em 3 camadas; $N_n=35$ em 1 camada; $N_n=48$ em 3 camadas; $N_n=200$ em 3 camadas; $N_n=105$ em 1 camada. É possível observar que bons modelos foram obtidos com 1 e 3 camadas, bem como com diferentes quantidades de neurônios. Isso sugere que uma boa abordagem para encontrar modelos de com melhor capacidade preditiva é explorar livremente diversas combinações de parâmetros, sem a necessidade de manter um parâmetro constante enquanto se varia o outro.

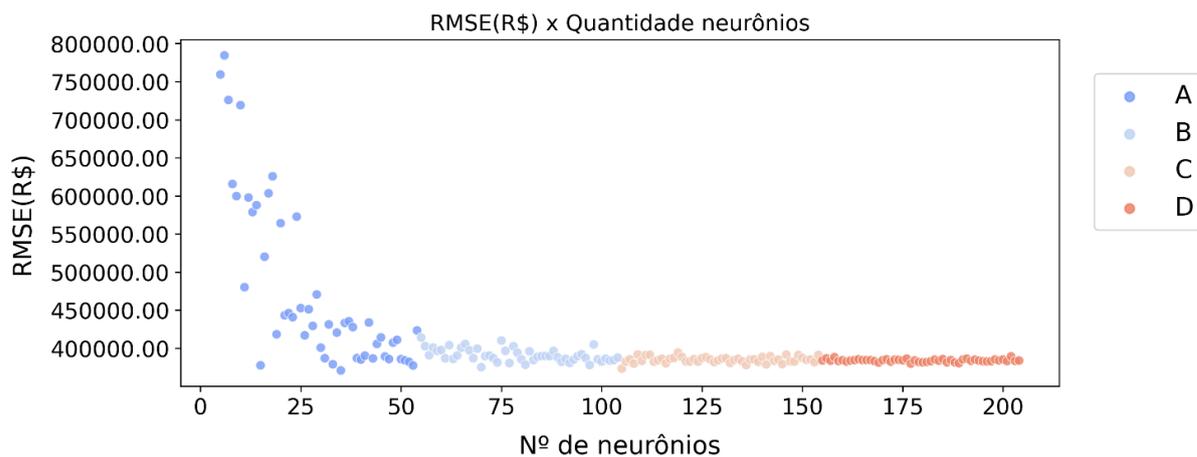
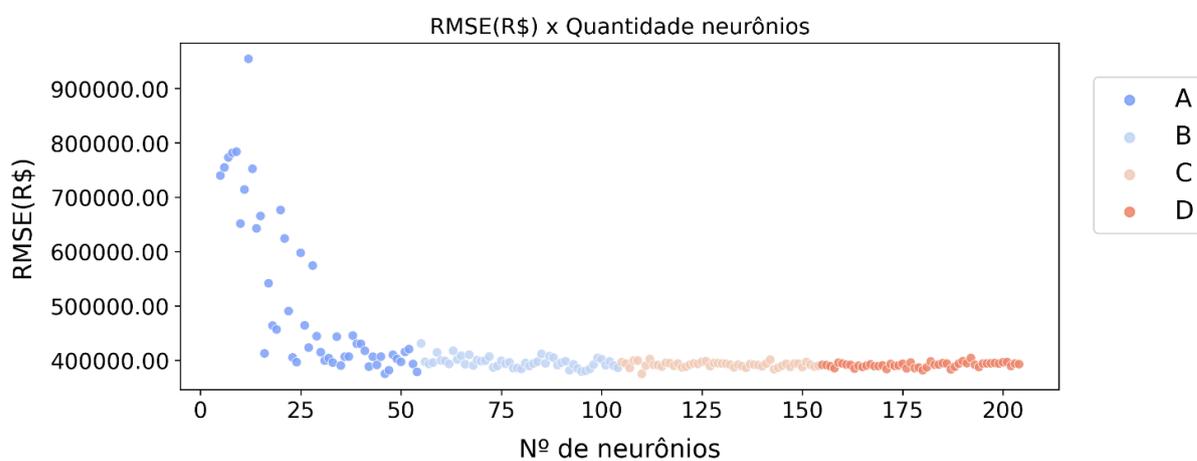
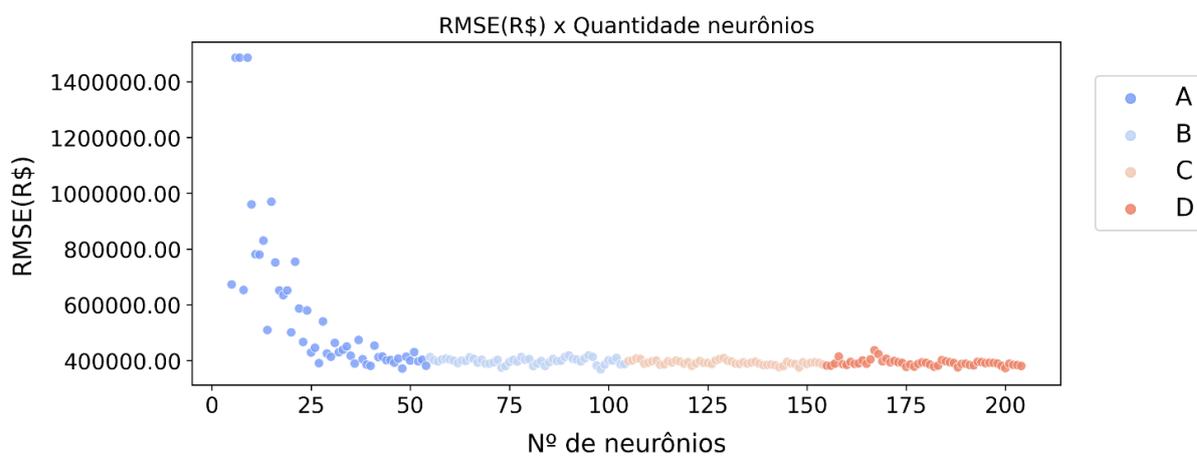
Tabela 32 – Teste de normalidade por categorias em HLPR

Métrica	Categoria	Shapiro-Wilk (P-valor)		
		RNA (1HLPR)	RNA (2HLPR)	RNA (3HLPR)
RMSE	A	0,0000	0,0000	0,0000
	B	0,0954	0,0350	0,3717
	C	0,6516	0,0826	0,9195
	D	0,5486	0,5801	0,0000

Tabela 33 – Testes para comparação entre as camadas em HLPR

Cat.	Camadas	Bartlett	Fligner-Killeen	Teste-t ou teste-t de Welch		Mann-Whitney	
				Unilateral (<)	Unilateral (>)	Unilateral (<)	Unilateral (>)
A	1-2	-	0,3967	-	-	0,1897	0,8122
	2-3	-	0,0837	-	-	0,19537	0,80653
B	1-2	-	0,6060	-	-	0,00355	0,99652
	2-3	-	0,3897	-	-	0,06338	0,93747
C	1-2	0,3031	-	0,0000	1,0000	-	-
	2-3	0,0020	-	0,4754	0,5246	-	-
D	1-2	0,0000	-	0,0000	1,0000	-	-
	2-3	-	0,0004	-	-	0,93747	0,06338

As figuras 70, 71 e 72 apresentam os erros para cada RNA calculada. Os erros da categoria inicial são maiores e mais dispersos, porém, à medida em que se eleva a quantidade de neurônios, os erros diminuem, assim como a dispersão.

Figura 70 – RMSE \times N $^\circ$ de neurônios (1HLPR)Figura 71 – RMSE \times N $^\circ$ de neurônios (2HLPR)Figura 72 – RMSE \times N $^\circ$ de neurônios (3HLPR)

5.11.3 Modelos selecionados

Nos itens 5.11.1 e 5.11.2, foi possível perceber que o número de neurônios (mantendo constante o número de camadas) e a quantidade de camadas (mantendo constante a faixa de neurônios) exercem influência sobre a capacidade preditiva das RNA e que a variação desses parâmetros tem o potencial de gerar impactos significativos nos resultados. Entretanto, na prática, é inviável investigar de forma isolada como cada parâmetro contribui para a melhoria da capacidade preditiva das RNA. A abordagem mais adequada é buscar uma combinação de parâmetros que alcance os melhores resultados.

Por isso, os modelos são selecionados sem a necessidade de estabelecer restrições quanto ao número de neurônios e/ou camadas. O foco da escolha reside na identificação de modelos que apresentem os melhores resultados nas métricas: RMSE, MAE e MAPE.

Tabela 34 – Melhores modelos de RNA (com aplicação de transformação)

ID	HL	N _n	Função	LR	Otimizador	Inicializador	RMSE	MAE	MAPE	Critério
RNA1	2	46	tanh	0,0010	Adam	glorot_uniform	329.338,34	212.364,91	14,00%	ES
RNA2	2	112	tanh	0,0010	Adam	glorot_uniform	338.948,56	210.085,93	13,66%	ES
RNA3	3	119	tanh	0,0010	Adam	glorot_uniform	346.648,75	201.999,19	13,67%	ES
RNA4	3	200	tanh	0,0010	Adam	glorot_uniform	372.654,33	225.611,25	14,22%	PR
RNA5	1	105	tanh	0,0010	Adam	glorot_uniform	373.105,79	229.036,19	14,25%	PR
RNA6	3	98	tanh	0,0010	Adam	glorot_uniform	368.151,41	229.138,71	14,41%	PR

5.12. SIMULAÇÕES DE RNA SEM A UTILIZAÇÃO DE TRANSFORMAÇÕES

No item 5.11 foram apresentados modelos de RNA utilizando transformações que linearizam os modelos. No entanto, é importante ressaltar que uma das principais vantagens dos RNAs é justamente sua capacidade de lidar com problemas não lineares sem a necessidade de artifícios preliminares para linearização.

Assim, neste item, serão conduzidas simulações sem a necessidade de transformações nas variáveis. Distingue-se do item 5.11 pelo emprego de um procedimento mais abrangente na busca por parâmetros (maior diversidade de cenários). Essa busca mais abrangente é uma estratégia na tentativa de encontrar modelos que lidem bem com a não linearidade do problema.

5.12.1 Cenários simulados

Para a busca pelos melhores parâmetros foram realizadas 3.564 simulações variando o número de camadas intermediárias, número de neurônios, funções de ativação, taxa de aprendizagem, otimizadores, inicializadores dos pesos sinápticos e critérios para evitar overfitting.

São conduzidas simulações com redes neurais contendo 1, 2 e 3 camadas neurais. No que tange ao número de neurônios, os modelos são testados abrangendo um mínimo de 5 neurônios e um máximo de 615 neurônios; mais especificamente: 5, 10, 15, 25, 45, 50, 65, 75, 85, 90, 105, 125, 130, 145, 165, 170, 185, 195, 205, 210, 250, 255, 290, 315, 330, 370, 375, 410, 435, 495, 555 e 615. Com relação às funções de ativação, são testadas as funções: *relu*, *tanh* e *softsign*; taxas de aprendizagem de 0,001 e 0,0005; são testados os otimizadores: *Adam*, *Nadam* e *RMSprop*; e os inicializadores de pesos sinápticos: *truncated_normal*, *glorot_normal* e *glorot_uniform*.

5.12.2 Modelos selecionados

As métricas RMSE, MAE e MAPE dos melhores modelos calculados estão disponíveis no Apêndice G. São apresentados na tabela 35 os seis melhores modelos, três com *Early Stopping* e três com *Pruning*, escolhidos considerando as métricas citadas anteriormente.

Tabela 35 – Melhores modelos de RNA (sem aplicação de transformação)

ID	HL	N _n	Função	LR	Otimizador	Inicializador	RMSE	MAE	MAPE	Critério
RNA7	3	435	relu	0,0010	Adam	glorot_normal	349.705,71	218.253,56	14,28%	ES
RNA8	3	195	relu	0,0010	RMSprop	glorot_normal	350.807,34	214.986,72	14,07%	ES
RNA9	3	195	relu	0,0010	Nadam	glorot_normal	353.912,26	213.753,22	14,23%	ES
RNA10	2	170	relu	0,0005	RMSprop	glorot_normal	323.252,32	212.950,57	13,86%	PR
RNA11	3	135	relu	0,0005	Adam	truncated_normal	333.445,70	214.502,72	13,80%	PR
RNA12	3	195	relu	0,0010	Adam	glorot_normal	311.684,91	206.648,02	15,01%	PR

5.13. VERIFICAÇÕES DE QUALIDADE PARA MODELOS DE RNA

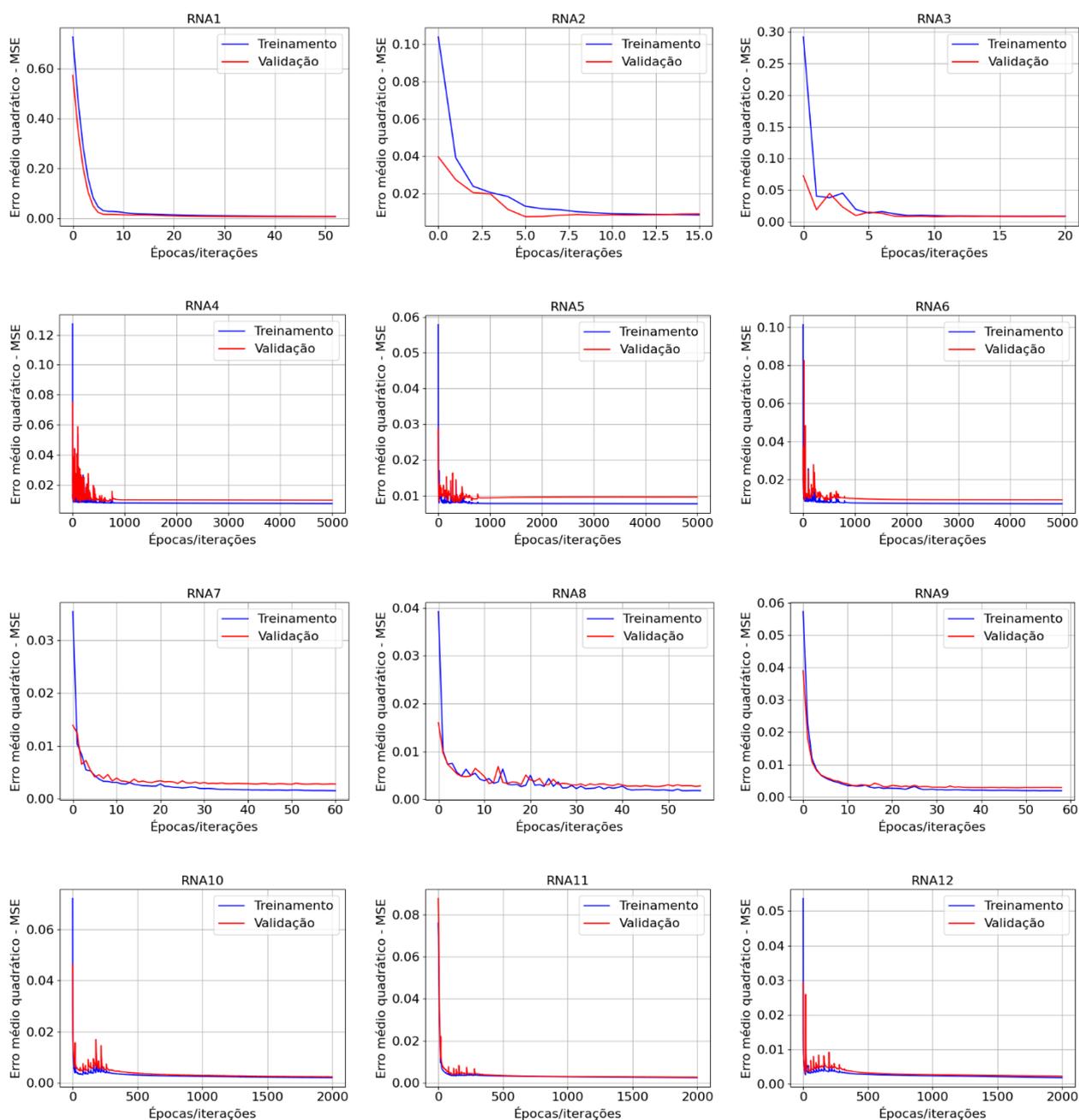
Um ponto de partida para verificar bons modelos de RNA são as métricas de qualidade já apresentadas como RMSE, MAE e MAPE. Além disso, existem outras abordagens que auxiliam nessa verificação e que são apresentados nos itens 5.13.1, 5.13.2 e 5.13.3: os gráficos de convergências dos modelos, os gráficos de predição dos valores observados *versus* valores ajustados e análise de dependência espacial.

5.13.1 Gráficos de convergência

Na figura 73, são apresentados gráficos de convergência das RNA escolhidas. Os gráficos de convergência mostram o desempenho das RNA ao longo do treinamento e são úteis para verificar se o modelo está sofrendo *overfitting*. Nessas figuras, é possível perceber que as curvas de validação

e treinamento se mantém próximas, o que indica que as RNA apresentam desempenho parecido nas duas amostras: treinamento e validação. Além disso, a ausência do levantamento das curvas de validação em relação às de treinamento denota ausência de *overfitting*. Esse é um resultado desejável, pois garante que os modelos não apresentam um desempenho enviesado, ou seja, um desempenho aparentemente ótimo apenas porque decoraram os dados de treinamento.

Figura 73 – Curvas de convergência das RNA escolhidas



5.13.2 Gráficos de Predição

O gráfico de predição é utilizado para verificar se os valores preditos são semelhantes aos observados no mercado. O objetivo nesse gráfico é que os pontos estejam aderentes à bissetriz. Quanto mais aderentes, melhor é a capacidade preditiva do modelo, pois indica que os valores preditos estão próximos aos observados no mercado.

Os gráficos de predição das RNA são apresentados na figura 74. Para efeito comparativo, é apresentado também o gráfico de predição referente à RLC na figura 75. São usados os dados da amostra de validação.

Figura 74 – Gráfico de predição da amostra de validação – RNA

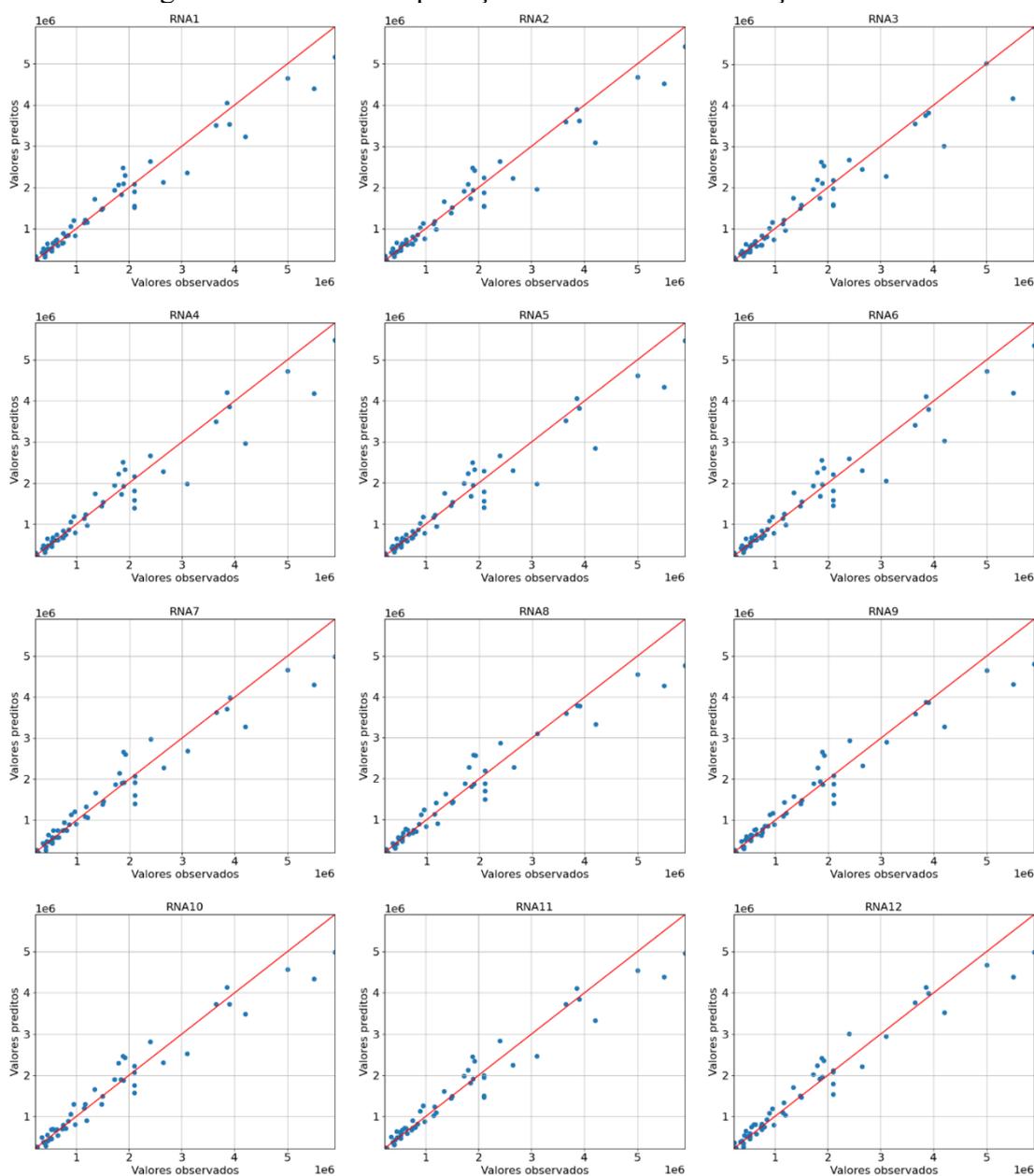
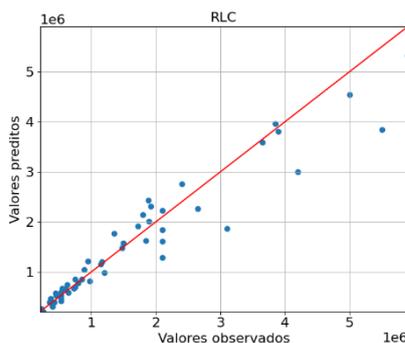


Figura 75 – Gráfico de predição da amostra de validação – RLC



5.13.3 Verificação da dependência espacial nos modelos RNA

De modo análogo à RLC pode ser verificada se a distribuição espacial dos resíduos padronizados ocorre sem dependência espacial. As figuras 76 a 88 apresentam a distribuição espacial dos resíduos, onde os pontos azuis e vermelhos representam, respectivamente, resíduos positivos e negativos; além disso, apresentam os índices de Moran⁵⁴ para cada caso plotado.

Para efeito comparativo, é apresentado os resultados de dependência espacial referente à RLC, que não apresenta dependência espacial. Os resultados apresentados na tabela 36 e nas figuras 76 a 88 indicam também que não há dependência espacial dos modelos de RNA. Desse modo, é possível afirmar que, nos modelos propostos, os erros estão dispostos aleatoriamente na área estudada.

Tabela 36 – Índices de Moran (RLC x RNA)

Modelo	Índice de Moran	P-valor
RLC	0,063	0,139
RNA1	0,044	0,211
RNA2	-0,032	0,463
RNA3	-0,001	0,399
RNA4	0,029	0,253
RNA5	0,042	0,207
RNA6	0,022	0,286
RNA7	0,033	0,235
RNA8	0,020	0,285
RNA9	0,043	0,195
RNA10	0,072	0,138
RNA11	0,062	0,159
RNA12	0,013	0,320

⁵⁴ Obtido no *software* Geoda usando 999 permutações, *seed* igual a 10 e distância euclidiana de 340,24 metros na análise univariada de Moran.

Figura 76 – Distribuição espacial dos resíduos e Moran (RLC)

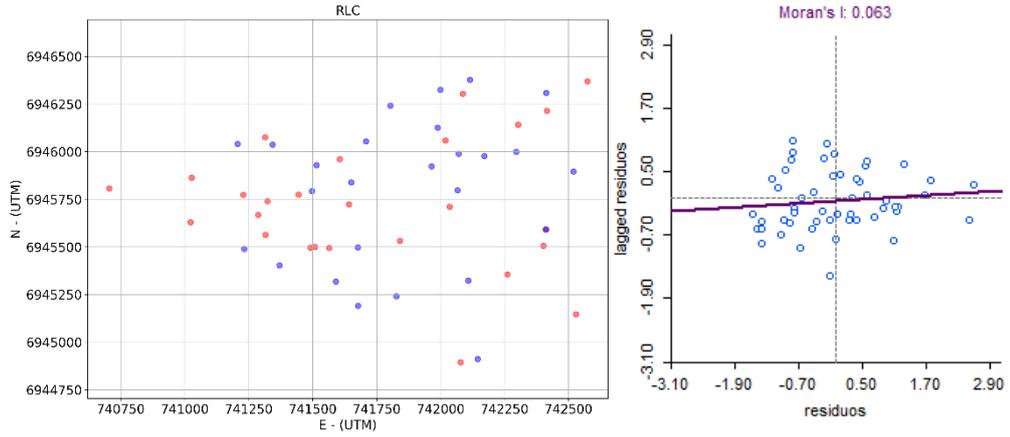


Figura 77 – Distribuição espacial dos resíduos e Moran (RNA1)

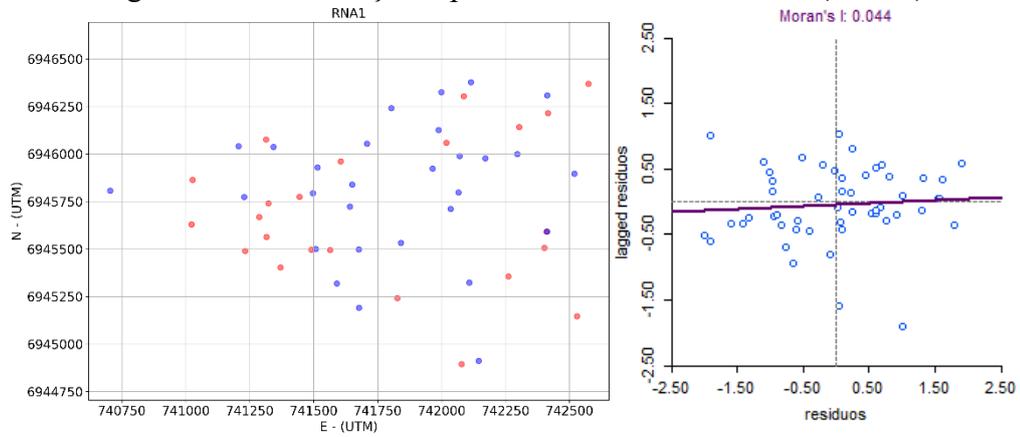


Figura 78 – Distribuição espacial dos resíduos e Moran (RNA2)

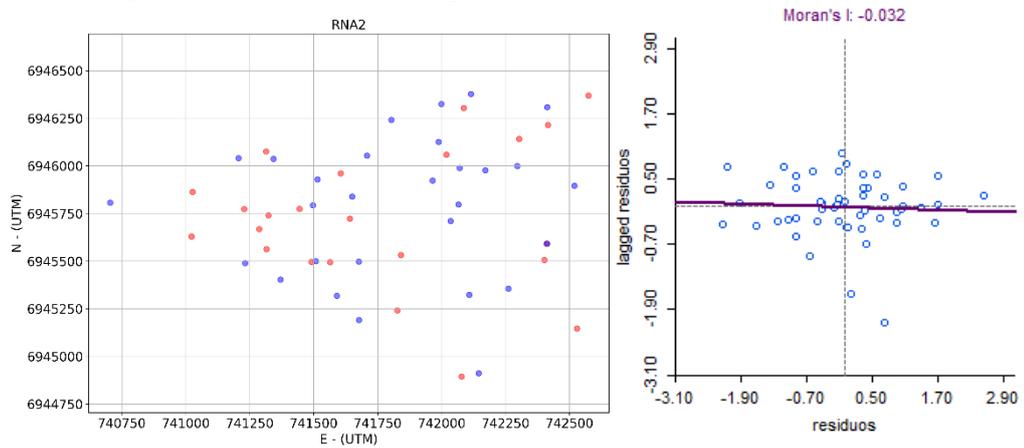


Figura 79 – Distribuição espacial dos resíduos e Moran (RNA3)

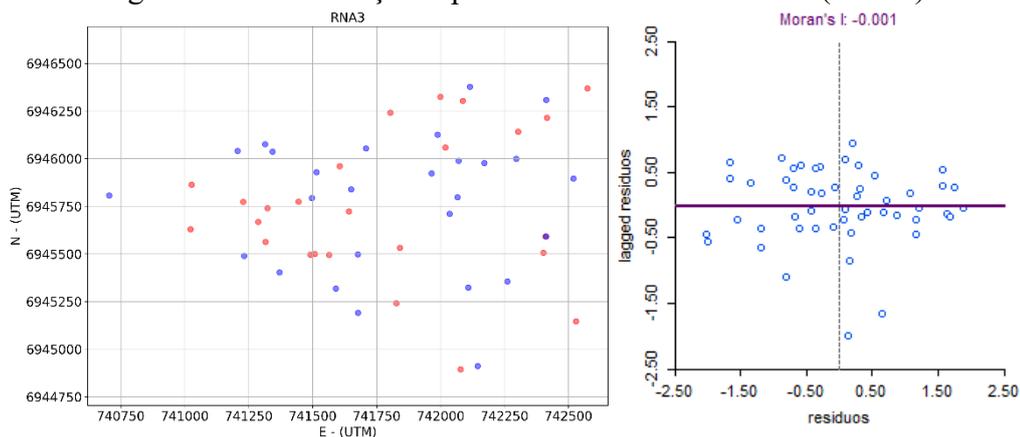


Figura 80 – Distribuição espacial dos resíduos e Moran (RNA4)

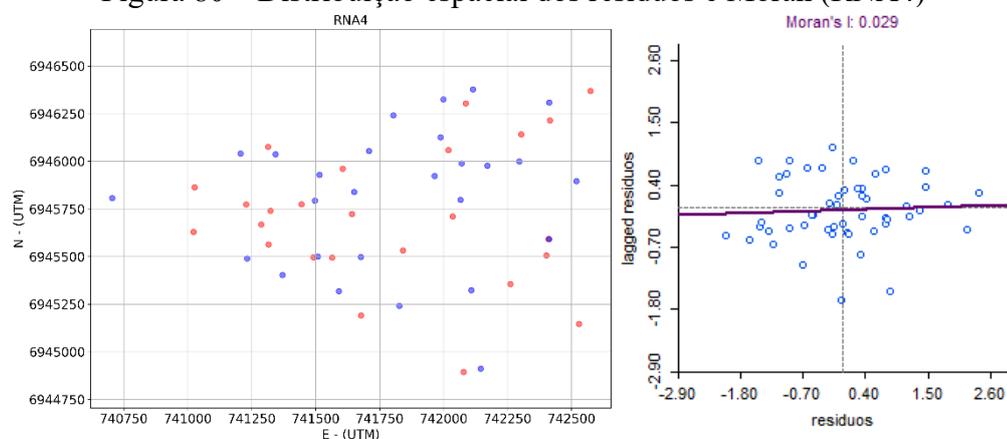


Figura 81 – Distribuição espacial dos resíduos e Moran (RNA5)

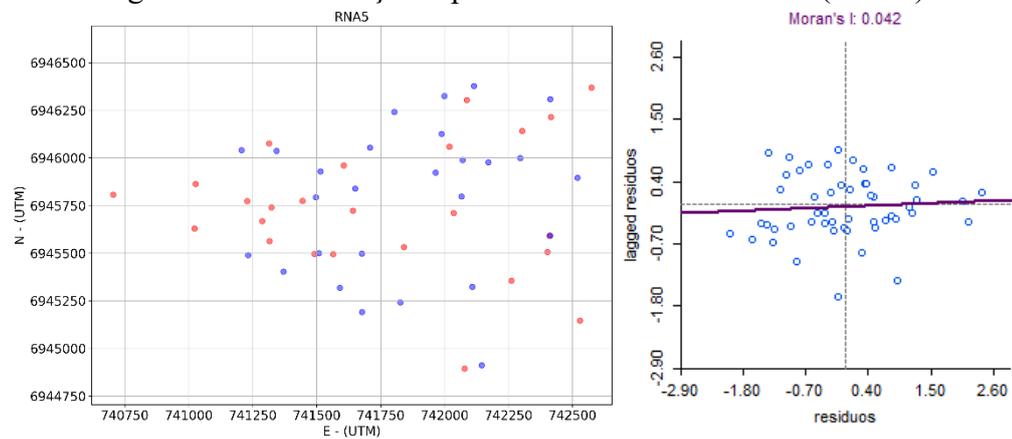


Figura 82 – Distribuição espacial dos resíduos e Moran (RNA6)

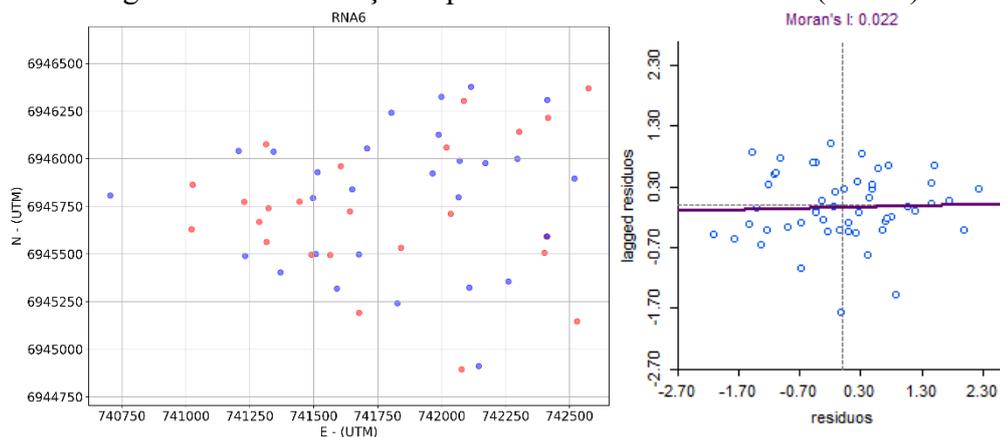


Figura 83 – Distribuição espacial dos resíduos e Moran (RNA7)

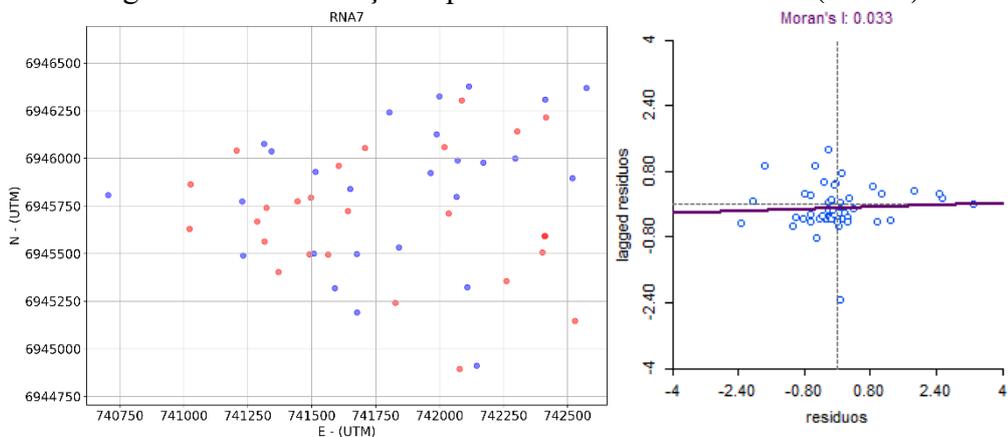


Figura 84 – Distribuição espacial dos resíduos e Moran (RNA8)

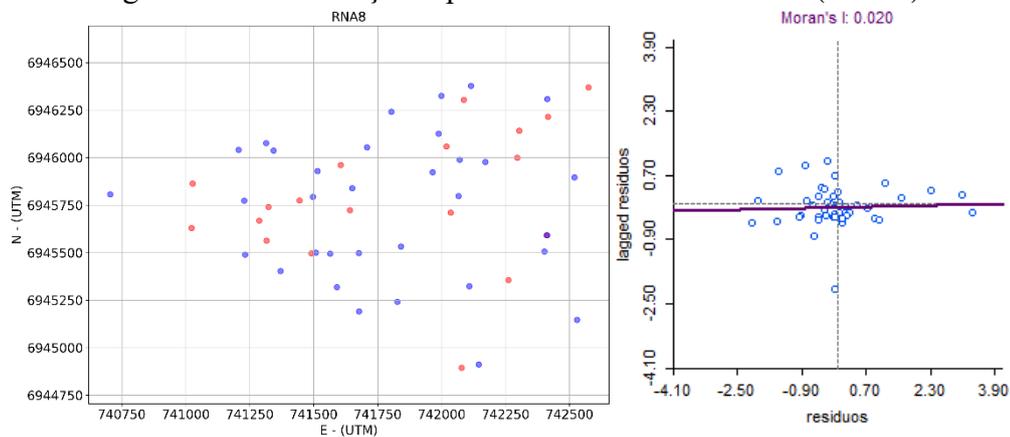


Figura 85 – Distribuição espacial dos resíduos e Moran (RNA9)

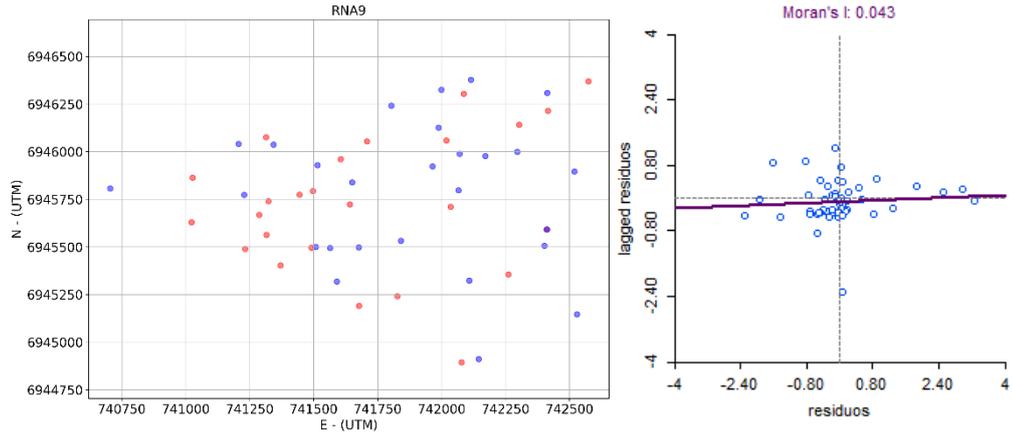


Figura 86 – Distribuição espacial dos resíduos e Moran (RNA10)

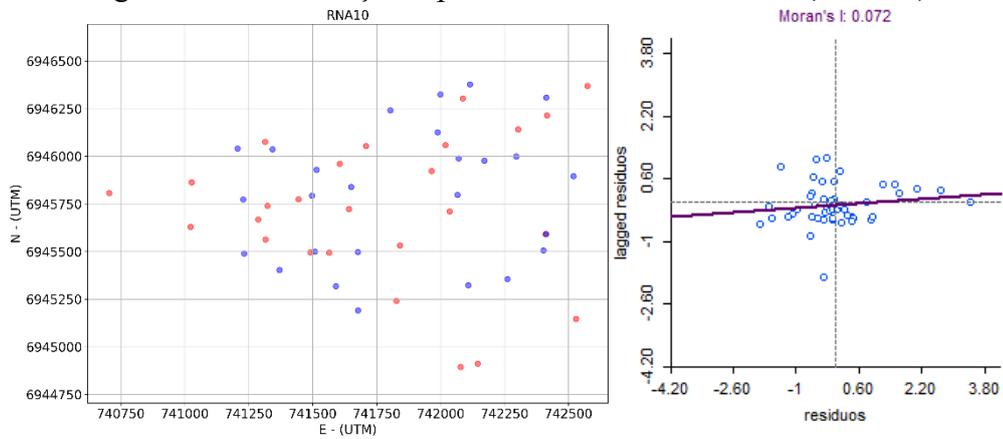


Figura 87 – Distribuição espacial dos resíduos e Moran (RNA11)

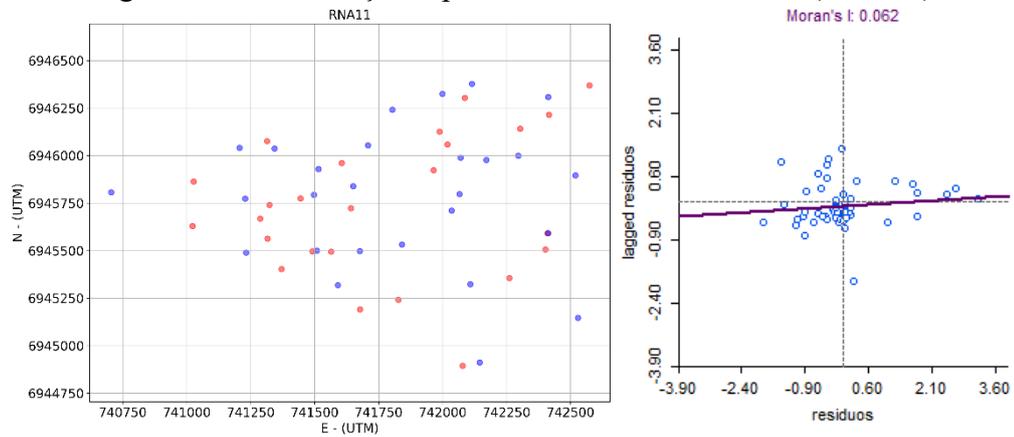
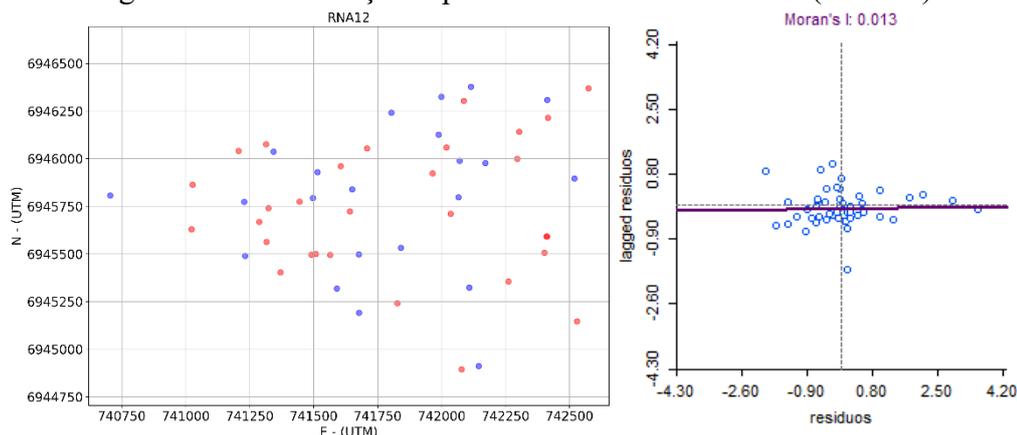


Figura 88 – Distribuição espacial dos resíduos e Moran (RNA12)



5.14. CONCLUSÕES PARCIAIS

É possível constatar no item 5.11 que, as RNA dos cenários 1HL, 2HL e 3HL, que não usam *Early Stopping* ou *Pruning*, são susceptíveis à ocorrência do *overfitting*, que degenera a capacidade do modelo de fazer previsões confiáveis. Ao contrário dos modelos tratados com *Early Stopping* ou *Pruning*, que tiveram desempenho notadamente superior.

Os resultados apresentados no item 5.11.1 indicam que, o número de neurônios tem papel fundamental no desempenho das RNA. Nas simulações realizadas, a categoria A, com até 54 neurônios ($5 \leq N_n \leq 54$), é a que apresenta resultados mais dispersos, com os maiores erros e menores coeficientes de determinação. Por outro lado, as demais categorias: B ($55 \leq N_n \leq 104$), C ($105 \leq N_n \leq 154$) e D ($155 \leq N_n \leq 204$) tendem a apresentar melhor desempenho. No entanto, foi percebido também que, o aumento indiscriminado do número de neurônios não significa, necessariamente, melhoria nos resultados, ou seja, após uma certa quantidade de neurônios pode não haver diferenças significativa nos resultados ou até mesmo piora do desempenho. Apesar dessa tendência de melhoria, foi possível observar que também são encontrados bons resultados na categoria A, o que mostra que modelos com menos neurônios não devem ser necessariamente descartados.

Com relação ao número de camadas intermediárias, os testes de hipótese realizados no item 5.11.2 mostram que, o aumento do número dessas camadas, fixando-se a categoria referente ao número de neurônios, não resulta em modelos melhores. Esses resultados não implicam necessariamente que modelos com duas ou três camadas são ruins, nem que modelos de apenas uma camada são os melhores. Isto fica evidenciado pelo fato de que, entre os resultados que apresentam os menores RMSE, encontram-se modelos compostos por 2 e 3 camadas.

Nos itens 5.11.1 e 5.11.2, pôde-se perceber que a importância do número de neurônios e a quantidade de camadas no desempenho das RNA. No entanto, a abordagem de buscar um bom modelo de RNA focando na variação de um parâmetro de cada vez parece pouco viável na prática. A abordagem mais interessante consiste em buscar uma arquitetura de RNA que apresenta métricas satisfatórias, sem a necessidade de pré-fixar determinados parâmetros enquanto variam-se outros.

Foi possível notar no item 5.12 que modelos de RNA não linearizados (sem transformações) foram capazes de fornecer bons resultados. Por exemplo, os modelos não linearizados selecionados (item 5.12) apresentaram RMSE menores do que os modelos linearizados (item 5.11).

Um dos entraves para a disseminação das RNA em avaliações imobiliárias é a verificação de qualidade dos modelos. Nesse sentido, é possível perceber que as métricas clássicas (RMSE, MAE e MAPE) e as verificações do item 5.13 como gráficos das curvas de convergência, de predição e de dependência espacial são abordagens importantes que auxiliam na verificação da qualidade das RNA.

6. COMPARAÇÃO ENTRE MODELOS DE RLC E RNA: RESULTADOS E ANÁLISES

Neste capítulo são realizadas comparações entre modelos de RLC e RNA. Os modelos separados para avaliação constam no item 6.1. No item 6.2 são apresentados os resultados das avaliações usando RNA e as comparações em relação ao modelo de RLC. Por fim, no item 6.3, é apresentada uma comparação entre RLC e RNA no que se refere à variação dos valores em função das variáveis explicativas. No item 6.4 são apresentadas as conclusões parciais para fechamento do capítulo.

6.1. MODELOS SEPARADOS PARA REALIZAR A AVALIAÇÃO

Para que sejam realizadas as avaliações dos imóveis, dentre as RNA verificadas no item 5.13, são escolhidos modelos com e sem a utilização de transformação, sendo dois com *Early Stopping* e dois com *Pruning*: RNA1, RNA6, RNA8 e RNA11.

Desse modo, esses modelos são utilizados para a avaliação dos 52 imóveis da amostra de validação e seus valores são comparados aos obtidos pela RLC e aos valores observados no Mercado.

6.2. VALORES OBTIDOS E COMPARAÇÕES

Na tabela 37, são apresentados os valores obtidos e as métricas (RMSE, MAE e MAPE)⁵⁵ correspondentes a cada modelo, calculadas tomando como referência os valores observados no mercado, que também são apresentados.

A RNA com o pior desempenho é a RNA-6, enquanto os modelos RNA-1 e RNA-11 se destacam como os melhores, com destaque para a RNA-1. Vale ressaltar que todos os modelos de RNA apresentados na tabela 37 possuem métricas melhores do que a RLC, uma vez que os valores de RMSE, MAE e MAPE são menores.

As métricas RMSE e MAE indicam que as previsões das RNA estão mais próximas dos valores de mercado do que ocorre no modelo RLC. De modo semelhante, o MAPE mostra que as previsões das RNA têm uma média de erro percentual menor em relação ao mercado do que ocorre na RLC.

⁵⁵ RMSE e MAE são os erros em escala monetária, com a diferença de que o RMSE penaliza de modo mais acentuado os maiores erros. Já o MAPE indica o erro percentual absoluto médio no modelo, oferecendo uma visão relativa do desempenho do modelo.

Tabela 37 – Valores obtidos por RNA (HLES), RLC e Mercado

Imóvel	RNA1	RNA6	RNA8	RNA11	RLC	Mercado
I_005	2.125.470,50	2.305.909,50	2.278.902,25	2.249.777,00	2.261.638,53	2.650.000,00
I_006	826.699,44	781.828,19	836.317,75	879.067,00	813.235,96	980.000,00
I_008	1.519.192,25	1.586.517,50	1.493.999,63	1.502.374,13	1.617.935,88	2.099.000,00
I_013	1.724.781,50	1.761.905,63	1.624.774,88	1.612.642,50	1.774.238,27	1.350.000,00
I_015	457.936,94	404.925,34	411.320,50	434.892,22	395.752,73	425.000,00
I_019	2.292.442,25	2.364.616,75	2.567.138,00	2.341.439,50	2.312.270,73	1.920.000,00
I_023	652.487,94	666.297,50	675.012,56	678.239,75	676.354,44	550.000,00
I_024	652.486,69	666.312,75	675.176,81	678.205,88	676.324,02	730.000,00
I_026	845.332,88	871.809,00	892.277,31	814.546,69	859.792,83	849.900,00
I_036	1.144.503,38	1.140.974,63	1.135.967,38	1.018.058,50	1.152.779,88	1.150.000,00
I_041	2.063.460,63	2.264.049,50	2.284.442,25	2.130.003,50	2.148.463,02	1.800.000,00
I_049	1.217.665,88	1.250.013,38	1.414.954,00	1.237.476,00	1.198.570,42	1.170.000,00
I_064	320.849,09	297.603,28	274.287,50	245.465,84	263.332,66	230.000,00
I_072	2.634.225,00	2.593.595,25	2.874.295,75	2.837.904,50	2.762.381,77	2.400.000,00
I_090	1.826.552,88	1.680.414,13	1.810.965,13	1.820.960,25	1.622.430,29	1.850.000,00
I_092	1.464.245,63	1.440.404,00	1.411.902,25	1.439.267,38	1.483.195,20	1.480.000,00
I_096	365.704,47	342.630,75	295.874,06	307.204,19	315.121,45	400.000,00
I_099	218.889,61	218.929,28	205.636,73	228.516,64	206.625,53	259.000,00
I_102	529.733,88	567.954,13	581.696,13	624.941,63	601.849,38	530.000,00
I_107	1.933.211,75	1.932.526,00	1.881.018,38	1.979.328,88	1.919.840,03	1.725.000,00
I_114	472.259,00	472.046,47	504.655,88	603.528,75	466.299,91	529.000,00
I_122	1.154.293,13	978.593,75	898.985,81	1.089.935,88	985.099,70	1.200.000,00
I_123	2.480.958,25	2.555.139,25	2.581.821,00	2.456.195,75	2.437.812,78	1.880.000,00
I_126	813.855,50	734.183,69	709.727,50	736.091,94	782.238,05	799.000,00
I_130	587.576,06	612.014,06	643.142,19	593.775,38	591.986,99	650.000,00
I_131	511.204,28	457.786,72	527.712,88	491.889,28	530.834,51	459.000,00
I_137	1.055.960,88	1.073.934,88	1.120.616,50	1.128.671,88	1.042.698,64	889.900,00
I_138	221.708,81	217.012,08	193.881,77	230.390,39	212.516,75	220.000,00
I_142	2.360.635,25	2.054.324,50	3.098.886,50	2.470.954,75	1.866.964,60	3.100.000,00
I_157	2.077.206,50	2.206.460,50	1.887.166,38	1.991.285,50	2.232.322,35	2.099.700,00
I_158	1.548.423,00	1.454.226,00	1.697.382,50	1.465.911,50	1.282.203,82	2.100.000,00
I_159	2.097.401,00	1.975.952,38	1.863.853,38	1.915.991,25	2.014.738,83	1.890.000,00
I_160	885.169,88	838.457,69	696.258,94	902.442,94	851.891,61	750.000,00
I_170	1.197.705,63	1.174.589,13	1.236.305,38	1.259.613,25	1.213.930,96	950.000,00
I_171	738.238,38	731.347,88	748.889,56	710.434,75	744.870,19	630.000,00
I_173	1.898.948,63	1.816.692,38	2.199.878,75	1.951.661,25	1.844.553,79	2.100.000,00
I_174	454.769,47	460.589,38	468.806,66	474.798,31	418.003,14	530.000,00
I_176	1.490.173,50	1.548.660,75	1.435.905,25	1.489.840,50	1.579.521,55	1.500.000,00
I_182	3.510.188,75	3.414.488,00	3.600.924,00	3.730.201,50	3.594.278,08	3.650.000,00
I_183	3.231.610,00	3.030.511,75	3.334.998,25	3.331.918,25	2.998.199,39	4.200.000,00
I_193	423.583,03	415.696,91	415.028,75	500.482,19	399.780,94	350.000,00
I_195	636.852,31	651.817,44	569.323,06	632.215,00	574.645,09	450.000,00
I_196	660.361,88	659.251,88	740.741,00	720.894,06	702.526,90	750.000,00
I_197	694.198,94	610.326,69	769.677,06	716.016,25	664.193,16	600.000,00
I_211	4.047.564,50	4.113.755,75	3.787.421,75	4.110.130,50	3.964.339,81	3.850.000,00
I_212	3.534.472,00	3.799.174,25	3.780.544,50	3.853.613,00	3.803.634,80	3.900.000,00

Tabela 37 – Valores obtidos por RNA (HLES), RLC e Mercado

Imóvel	RNA1	RNA6	RNA8	RNA11	RLC	Mercado
I_214	516.218,03	473.787,34	349.283,53	369.202,13	474.269,11	371.000,00
I_219	5.166.694,50	5.353.603,50	4.767.626,00	4.968.623,00	5.326.161,85	5.900.000,00
I_222	4.649.153,50	4.727.738,50	4.547.486,00	4.547.872,00	4.544.624,33	5.000.000,00
I_223	4.397.362,50	4.189.766,75	4.273.426,50	4.392.585,50	3.848.466,68	5.500.000,00
I_225	512.711,50	536.405,69	520.162,88	536.720,94	525.151,33	530.000,00
I_226	306.754,31	305.139,94	310.410,50	325.876,78	317.145,55	400.000,00
RMSE	329.338,34	368.151,41	350.807,34	333.445,70	408.853,13	-
MAE	212.364,91	229.138,71	214.986,72	214.502,72	240.121,12	-
MAPE	14,00%	14,41%	14,07%	13,80%	14,70%	-

As tabelas 38 e 39 apresentam as diferenças monetárias e percentuais dos valores observados no mercado em relação aos valores preditos. De forma geral, é possível perceber que as predições dos imóveis com valores mais elevados têm um erro percentual médio mais elevado. Essas diferenças mais acentuadas entre valores (observados x preditos) podem ter várias causas, tais como: alguma variável não considerada na pesquisa; erro na coleta; fatores de depreciação não indicados no anúncio; limitações dos modelos; a intenção do vendedor de se desfazer rapidamente ou tardiamente do imóvel, o que pode gerar ofertas com preço menor/maior do que se espera.

Tabela 38 – Diferenças monetárias do mercado em relação aos modelos RNA e RLC

Imóvel	RNA1	RNA6	RNA8	RNA11	RLC
I_005	524.529,50	344.090,50	371.097,75	400.223,00	388.361,47
I_006	153.300,56	198.171,81	143.682,25	100.933,00	166.764,04
I_008	579.807,75	512.482,50	605.000,38	596.625,88	481.064,12
I_013	-374.781,50	-411.905,63	-274.774,88	-262.642,50	-424.238,27
I_015	-32.936,94	20.074,66	13.679,50	-9.892,22	29.247,27
I_019	-372.442,25	-444.616,75	-647.138,00	-421.439,50	-392.270,73
I_023	-102.487,94	-116.297,50	-125.012,56	-128.239,75	-126.354,44
I_024	77.513,31	63.687,25	54.823,19	51.794,13	53.675,98
I_026	4.567,13	-21.909,00	-42.377,31	35.353,31	-9.892,83
I_036	5.496,63	9.025,38	14.032,63	131.941,50	-2.779,88
I_041	-263.460,63	-464.049,50	-484.442,25	-330.003,50	-348.463,02
I_049	-47.665,88	-80.013,38	-244.954,00	-67.476,00	-28.570,42
I_064	-90.849,09	-67.603,28	-44.287,50	-15.465,84	-33.332,66
I_072	-234.225,00	-193.595,25	-474.295,75	-437.904,50	-362.381,77
I_090	23.447,13	169.585,88	39.034,88	29.039,75	227.569,71
I_092	15.754,38	39.596,00	68.097,75	40.732,63	-3.195,20
I_096	34.295,53	57.369,25	104.125,94	92.795,81	84.878,55
I_099	40.110,39	40.070,72	53.363,27	30.483,36	52.374,47
I_102	266,13	-37.954,13	-51.696,13	-94.941,63	-71.849,38
I_107	-208.211,75	-207.526,00	-156.018,38	-254.328,88	-194.840,03
I_114	56.741,00	56.953,53	24.344,13	-74.528,75	62.700,09

Tabela 38 – Diferenças monetárias do mercado em relação aos modelos RNA e RLC

I_122	45.706,88	221.406,25	301.014,19	110.064,13	214.900,30
I_123	-600.958,25	-675.139,25	-701.821,00	-576.195,75	-557.812,78
I_126	-14.855,50	64.816,31	89.272,50	62.908,06	16.761,95
I_130	62.423,94	37.985,94	6.857,81	56.224,63	58.013,01
I_131	-52.204,28	1.213,28	-68.712,88	-32.889,28	-71.834,51
I_137	-166.060,88	-184.034,88	-230.716,50	-238.771,88	-152.798,64
I_138	-1.708,81	2.987,92	26.118,23	-10.390,39	7.483,25
I_142	739.364,75	1.045.675,50	1.113,50	629.045,25	1.233.035,40
I_157	22.493,50	-106.760,50	212.533,63	108.414,50	-132.622,35
I_158	551.577,00	645.774,00	402.617,50	634.088,50	817.796,18
I_159	-207.401,00	-85.952,38	26.146,63	-25.991,25	-124.738,83
I_160	-135.169,88	-88.457,69	53.741,06	-152.442,94	-101.891,61
I_170	-247.705,63	-224.589,13	-286.305,38	-309.613,25	-263.930,96
I_171	-108.238,38	-101.347,88	-118.889,56	-80.434,75	-114.870,19
I_173	201.051,38	283.307,63	-99.878,75	148.338,75	255.446,21
I_174	75.230,53	69.410,63	61.193,34	55.201,69	111.996,86
I_176	9.826,50	-48.660,75	64.094,75	10.159,50	-79.521,55
I_182	139.811,25	235.512,00	49.076,00	-80.201,50	55.721,92
I_183	968.390,00	1.169.488,25	865.001,75	868.081,75	1.201.800,61
I_193	-73.583,03	-65.696,91	-65.028,75	-150.482,19	-49.780,94
I_195	-186.852,31	-201.817,44	-119.323,06	-182.215,00	-124.645,09
I_196	89.638,13	90.748,13	9.259,00	29.105,94	47.473,10
I_197	-94.198,94	-10.326,69	-169.677,06	-116.016,25	-64.193,16
I_211	-197.564,50	-263.755,75	62.578,25	-260.130,50	-114.339,81
I_212	365.528,00	100.825,75	119.455,50	46.387,00	96.365,20
I_214	-145.218,03	-102.787,34	21.716,47	1.797,88	-103.269,11
I_219	733.305,50	546.396,50	1.132.374,00	931.377,00	573.838,15
I_222	350.846,50	272.261,50	452.514,00	452.128,00	455.375,67
I_223	1.102.637,50	1.310.233,25	1.226.573,50	1.107.414,50	1.651.533,32
I_225	17.288,50	-6.405,69	9.837,13	-6.720,94	4.848,67
I_226	93.245,69	94.860,06	89.589,50	74.123,22	82.854,45

Tabela 39 – Diferenças percentuais do mercado em relação aos modelos RNA e RLC

Imóvel	RNA1	RNA6	RNA8	RNA11	RLC
I_005	24,68%	14,92%	16,28%	17,79%	17,17%
I_006	18,54%	25,35%	17,18%	11,48%	20,51%
I_008	38,17%	32,30%	40,50%	39,71%	29,73%
I_013	-21,73%	-23,38%	-16,91%	-16,29%	-23,91%
I_015	-7,19%	4,96%	3,33%	-2,27%	7,39%
I_019	-16,25%	-18,80%	-25,21%	-18,00%	-16,96%
I_023	-15,71%	-17,45%	-18,52%	-18,91%	-18,68%
I_024	11,88%	9,56%	8,12%	7,64%	7,94%
I_026	0,54%	-2,51%	-4,75%	4,34%	-1,15%
I_036	0,48%	0,79%	1,24%	12,96%	-0,24%

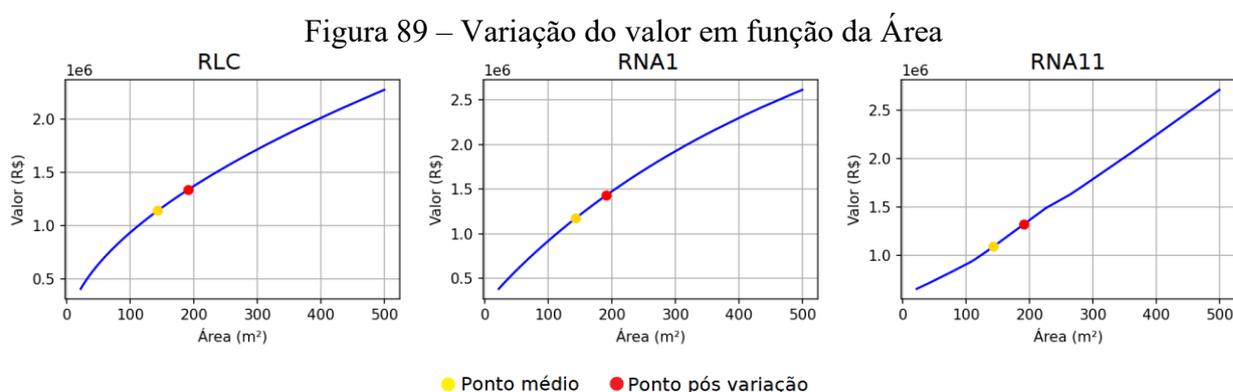
Tabela 39 – Diferenças percentuais do mercado em relação aos modelos RNA e RLC

I_041	-12,77%	-20,50%	-21,21%	-15,49%	-16,22%
I_049	-3,91%	-6,40%	-17,31%	-5,45%	-2,38%
I_064	-28,32%	-22,72%	-16,15%	-6,30%	-12,66%
I_072	-8,89%	-7,46%	-16,50%	-15,43%	-13,12%
I_090	1,28%	10,09%	2,16%	1,59%	14,03%
I_092	1,08%	2,75%	4,82%	2,83%	-0,22%
I_096	9,38%	16,74%	35,19%	30,21%	26,94%
I_099	18,32%	18,30%	25,95%	13,34%	25,35%
I_102	0,05%	-6,68%	-8,89%	-15,19%	-11,94%
I_107	-10,77%	-10,74%	-8,29%	-12,85%	-10,15%
I_114	12,01%	12,07%	4,82%	-12,35%	13,45%
I_122	3,96%	22,62%	33,48%	10,10%	21,82%
I_123	-24,22%	-26,42%	-27,18%	-23,46%	-22,88%
I_126	-1,83%	8,83%	12,58%	8,55%	2,14%
I_130	10,62%	6,21%	1,07%	9,47%	9,80%
I_131	-10,21%	0,27%	-13,02%	-6,69%	-13,53%
I_137	-15,73%	-17,14%	-20,59%	-21,16%	-14,65%
I_138	-0,77%	1,38%	13,47%	-4,51%	3,52%
I_142	31,32%	50,90%	0,04%	25,46%	66,04%
I_157	1,08%	-4,84%	11,26%	5,44%	-5,94%
I_158	35,62%	44,41%	23,72%	43,26%	63,78%
I_159	-9,89%	-4,35%	1,40%	-1,36%	-6,19%
I_160	-15,27%	-10,55%	7,72%	-16,89%	-11,96%
I_170	-20,68%	-19,12%	-23,16%	-24,58%	-21,74%
I_171	-14,66%	-13,86%	-15,88%	-11,32%	-15,42%
I_173	10,59%	15,59%	-4,54%	7,60%	13,85%
I_174	16,54%	15,07%	13,05%	11,63%	26,79%
I_176	0,66%	-3,14%	4,46%	0,68%	-5,03%
I_182	3,98%	6,90%	1,36%	-2,15%	1,55%
I_183	29,97%	38,59%	25,94%	26,05%	40,08%
I_193	-17,37%	-15,80%	-15,67%	-30,07%	-12,45%
I_195	-29,34%	-30,96%	-20,96%	-28,82%	-21,69%
I_196	13,57%	13,77%	1,25%	4,04%	6,76%
I_197	-13,57%	-1,69%	-22,05%	-16,20%	-9,66%
I_211	-4,88%	-6,41%	1,65%	-6,33%	-2,88%
I_212	10,34%	2,65%	3,16%	1,20%	2,53%
I_214	-28,13%	-21,69%	6,22%	0,49%	-21,77%
I_219	14,19%	10,21%	23,75%	18,75%	10,77%
I_222	7,55%	5,76%	9,95%	9,94%	10,02%
I_223	25,07%	31,27%	28,70%	25,21%	42,91%
I_225	3,37%	-1,19%	1,89%	-1,25%	0,92%
I_226	30,40%	31,09%	28,86%	22,75%	26,13%

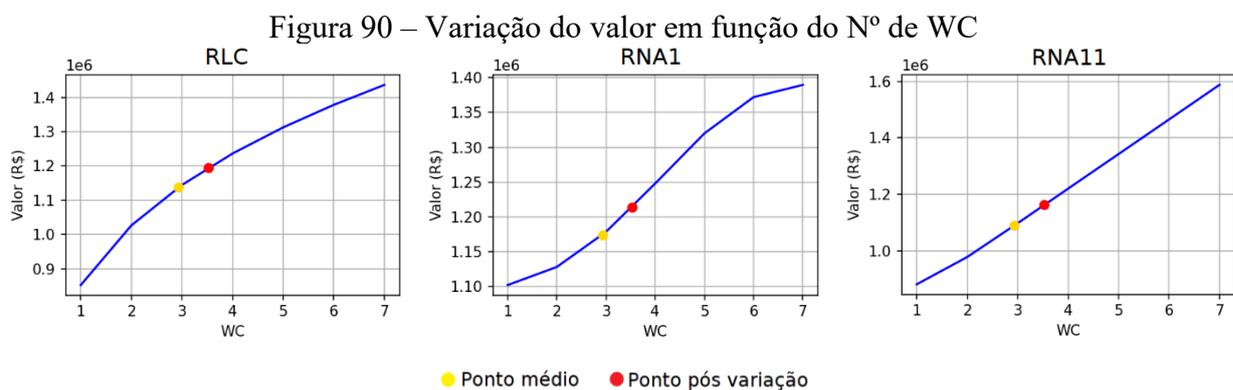
6.3. VARIACÃO DOS VALORES EM FUNÇÃO DAS VARIÁVEIS EXPLICATIVAS

Neste item são apresentadas as variações dos valores em função das variáveis explicativas. Para isso, como referência, é considerado o ponto médio da amostra. Assim, para cada variável é realizada uma variação dentro da sua amplitude disponível, deixando as demais variáveis explicativas constantes em seus respectivos pontos médios. Para isso, o modelo de RLC utilizado é aquele já definido no capítulo 4 e as RNA utilizadas são os modelos RNA-1 e RNA-11 apresentados no capítulo 5 e empregadas no item 6.2.

A figura 89 mostra a variação dos valores em função da variação da amplitude observada na Área (22 a 500). O comportamento é coerente, pois é esperado que o valor aumente à medida que se aumenta a área. Nos modelos da RLC, RNA-1 e RNA-11, se essa variável aumenta 10% da amplitude em sua média, o valor aumenta 17,30%, 21,65% e 21,18%, respectivamente.

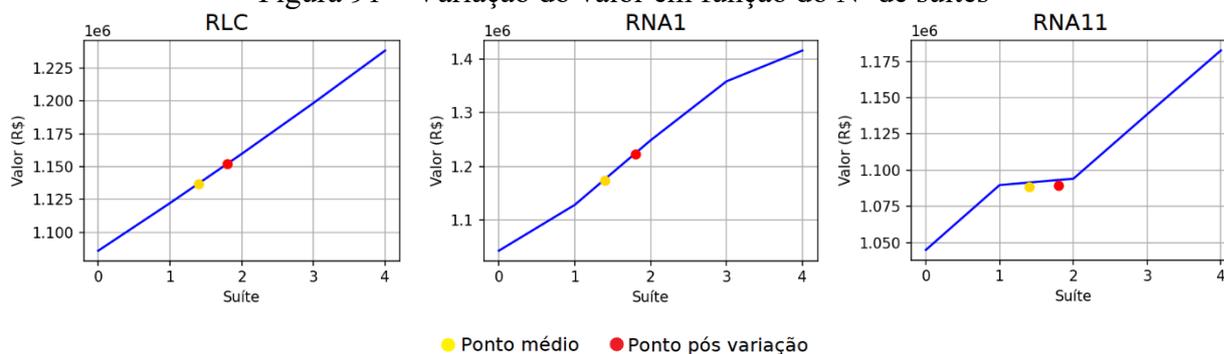


A figura 90 mostra a variação dos valores em função da variação da amplitude observada no número de banheiros (1 a 7). O comportamento é coerente, pois é esperado que o valor aumente quando houver incremento no número de banheiros. Nos modelos da RLC, RNA-1 e RNA-11, se essa variável aumenta 10% da amplitude em sua média, o valor aumenta 5,12%, 3,37% e 6,75%, respectivamente.



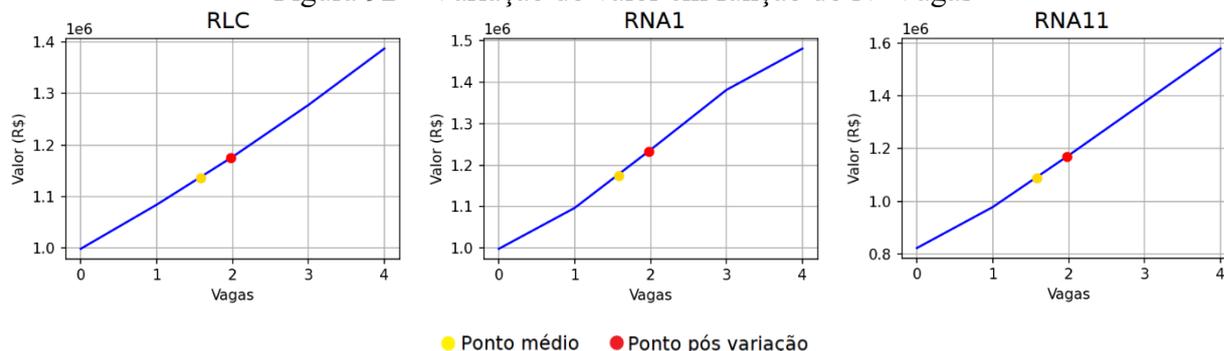
A figura 91 mostra a variação dos valores em função da variação da amplitude observada no número de suítes (0 a 4). O comportamento esperado é que o valor aumente com o incremento do número de suítes. Nos modelos da RLC, RNA-1 e RNA-11, se essa variável aumenta 10% da amplitude em sua média, o valor aumenta 1,32%, 4,20% e 0,07%, respectivamente. Era esperado que suítes exercessem pequena influência em RLC (pela significância apresentada no capítulo 4); e esse pequena influência também ocorre na RNA-11, só que de modo mais evidente. Vale destacar que gráfico da RNA-11 apresenta um comportamento diferente do esperado e destoa dos demais modelos, o que indica que a utilização dessa variável é dispensável no modelo.

Figura 91 – Variação do valor em função do N° de suítes



A figura 92 mostra a variação dos valores em função da variação da amplitude observada no número de vagas (0 a 4). O comportamento é coerente, pois é esperado que o valor seja maior se houver mais vagas. Nos modelos da RLC, RNA-1 e RNA-11, se essa variável aumenta 10% da amplitude em sua média, o valor aumenta 3,33%, 5,04% e 7,43%, respectivamente.

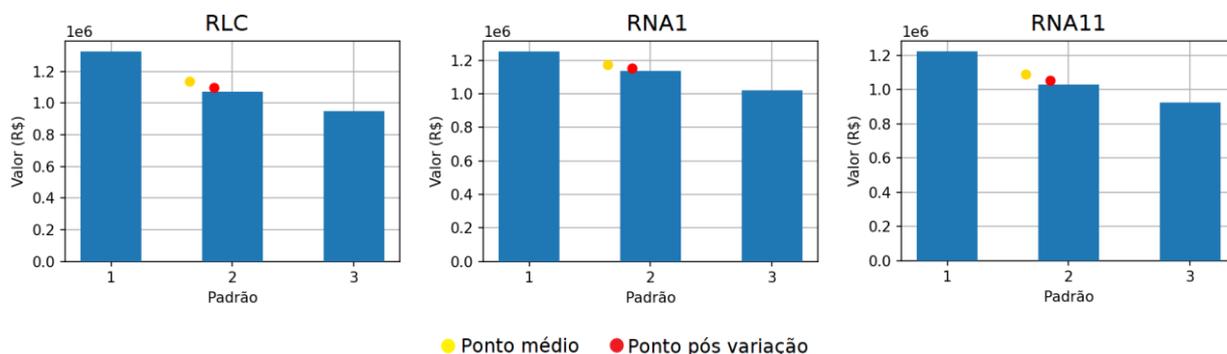
Figura 92 – Variação do valor em função do N° Vagas



A figura 93 mostra a variação dos valores em função da variação da amplitude observada no padrão construtivo (1 - alto, 2 - médio e 3 - simples). O comportamento é coerente, pois é esperado que os valores diminuam à medida que o padrão esteja mais próximo do nível "simples". Nos modelos

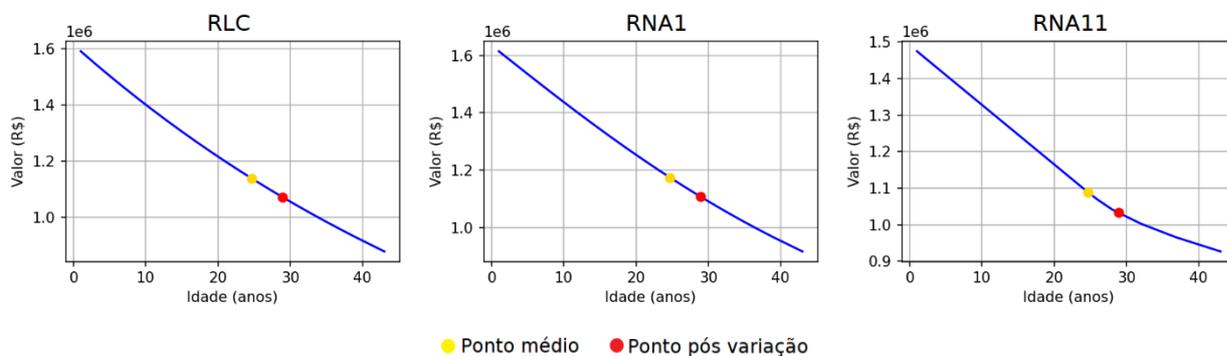
da RLC, RNA-1 e RNA-11, se essa variável aumenta 10% da amplitude em sua média, o valor diminui 3,43%, 1,71% e 3,19%, respectivamente. RLC e RNA-11 apresentam saltos diferentes entre os padrões 1-2 e 2-3, o que parece ser um comportamento mais coerente do que a RNA-1.

Figura 93 – Variação do valor em função do Padrão



A figura 94 mostra a variação dos valores em função da variação da amplitude observada na idade (1 a 43). O comportamento é coerente, pois é esperado que o valor diminua quando houver aumento na idade do imóvel. Nos modelos RLC, RNA-1 e RNA-11, se essa variável aumenta 10% da amplitude em sua média, o valor diminui 5,78%, 5,65% e 5,12%, respectivamente.

Figura 94 – Variação do valor em função da Idade

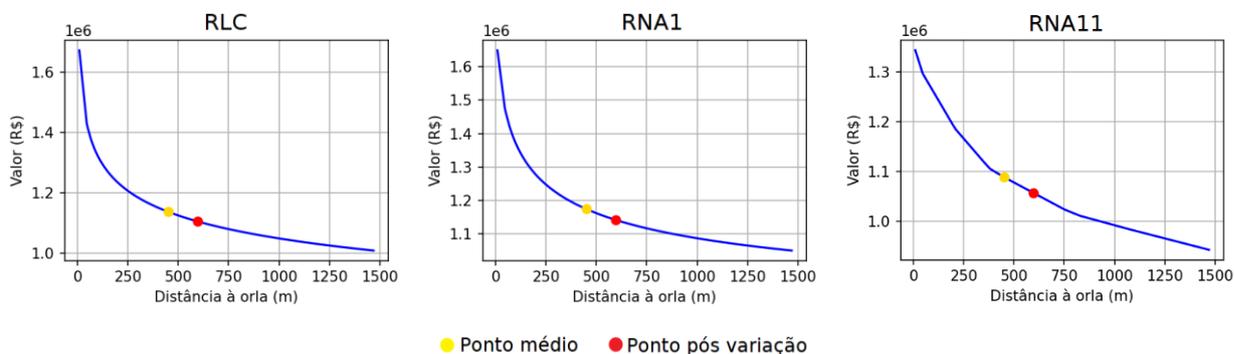


A figura 95 mostra a variação dos valores em função da variação da amplitude observada na distância à Beira-mar (10 a 1.467,91). O comportamento é coerente, pois é esperado que o valor diminua quando houver aumento da distância em relação à Beira-mar. Nos modelos RLC, RNA-1 e RNA-11, se essa variável aumenta 10% da amplitude em sua média, o valor diminui 2,80%, 2,76% e 2,90%, respectivamente.

É interessante notar que todos os modelos conseguiram mapear a relação não linear entre “distância à Beira Mar Norte *versus* valor”. Os mais adequados são RLC e RNA-1, que lembram mais

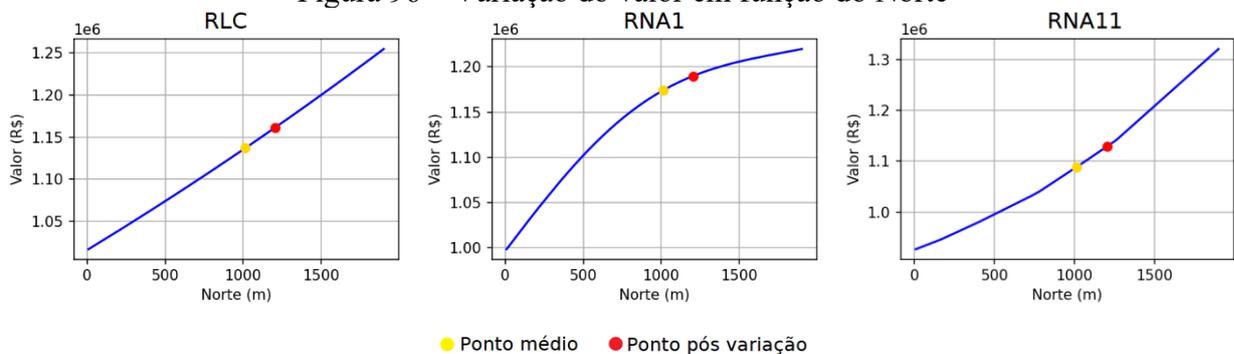
claramente o comportamento de uma assíntota, ou seja, após uma certa distância o valor passa a ser menos impactado pela variável.

Figura 95 – Variação do valor em função da Distância à Beira Mar



A figura 96 mostra a variação dos valores em função da variação da amplitude observada no Norte (10 a 1.897,92). O comportamento é coerente, pois é esperado que o valor seja maior nas regiões mais ao norte. Nos modelos RLC, RNA-1 e RNA-11, se essa variável aumenta 10% da amplitude em sua média, o valor aumenta 2,12%, 1,33% e 3,72%, respectivamente. Nesse caso, a RNA-1 parece ser o modelo mais coerente ao indicar que o Norte apresenta menor impacto em regiões mais próximas à Beira-Mar. Isso faz sentido porque à medida que a distância à Beira-mar passa a contribuir mais, o Norte deve contribuir menos. O único modelo que indica essa relação é a RNA-1.

Figura 96 – Variação do valor em função do Norte



6.4.CONCLUSÕES PARCIAIS

Em relação ao tempo, o uso de RNA demandou maior tempo do que a RLC para gerar modelos e estimar valores de imóveis. A busca pelo melhor modelo de RLC foi mais rápida do que a busca pelo melhor modelo de RNA, que exigiu a execução de várias simulações variando a quantidade

de diversos parâmetros. Portanto, se o fator tempo de cálculo for inegociável, a utilização da RLC tende a ser a melhor opção.

Em termos de verificação dos modelos, as RNA apresentam a vantagem de não requererem a observação de pressupostos clássicos, como já mencionado no item 5.10.5, ao contrário da RLC.

Com relação à capacidade preditiva, a análise dos resultados do item 6.2 é animadora pois mostra que modelos de RNA (com e sem transformações) fornecem métricas superiores em comparação à RLC, método já consagrado em avaliações imobiliárias.

Sobre a análise de elasticidade das variáveis (item 6.3), é possível notar que, em todos os modelos, a variável que apresenta maior impacto no valor é a “Área”, comportamento já esperado e coerente com o mercado. A variável que apresenta menor impacto no valor em RLC e RNA-11 é o número de suítes e na RNA-1 é o Norte.

Os modelos RLC e RNA-1 são os que apresentam um melhor comportamento na análise de elasticidade. Vale destacar que a RNA-1 foi o único modelo que mapeou o seguinte comportamento do mercado com relação às variáveis de localização: em regiões mais distantes da Beira-mar, a variável “Norte” possui maior impacto no valor e a “Distância à Beira-mar” possui menor impacto, mas em regiões mais próximas às Beira-mar essa relação é invertida. Vale destacar também que a RNA-11 destoa dos demais no que se refere à variável suítes, que apresenta influência muito pequena na variação do valor. Isso indica que a RNA-11 praticamente não utiliza essa variável para explicar o valor dos imóveis e poderia ser descartada. Isso levanta um questionamento: a RNA-11, mesmo apresentando métricas indiscutivelmente melhores do que a RLC, apresentou um comportamento não desejado em uma de suas variáveis (“suítes”). Isso mostra a importância de não se basear apenas nas métricas ao avaliar redes neurais.

7. CONCLUSÕES FINAIS

Considera-se que os objetivos foram cumpridos porque foi realizada uma análise comparativa entre o modelo clássico de regressão e modelos de redes neurais artificiais, utilizados para avaliações imobiliárias na área central do município de Florianópolis. Para tanto, foi selecionado um modelo de referência de RLC, obtido com auxílio de força bruta e tendo sido verificados seus pressupostos básicos. Depois, foram realizadas simulações usando RNA (com e sem transformação) testando diferentes parâmetros e foram escolhidos alguns modelos que apresentaram melhor capacidade preditiva. Por fim, foi realizada uma análise comparativa entre RNA e RLC.

Os modelos de RLC foram gerados com auxílio da aplicação *E-evaluation*, desenvolvida no âmbito dessa pesquisa, e foram realizadas verificações referentes à linearidade, normalidade, homocedasticidade, autocorrelação, colinearidade, pontos influenciantes e *outliers*. A melhor RLC, utilizada como referência, possui R^2 de 96,39%.

Com relação aos modelos de RNA, foram realizadas 5.364 simulações variando diversos parâmetros para encontrar modelos com boa capacidade preditiva.

No item 5.11, foram realizadas 1.800 simulações em RNA com transformações, considerando nove cenários (1HLES, 2HLES, 3HLES, 1HLPR, 2HLPR, 3HLPR, 1HL, 2HL, 3HL) agrupados em quatro categorias (A, B, C e D), conforme a quantidade de neurônios empregada. Nessas RNA, foi testada a influência do número de neurônios na capacidade preditiva das RNA, tendo-se fixado o número de camadas intermediárias; e testada a influência da quantidade de camadas, tendo sido fixado as categorias referentes à quantidade de neurônios. Com relação ao número de neurônios, pode-se concluir que, conforme as simulações realizadas no item 5.11.1, aumentar a quantidade de neurônios foi importante para melhorar os resultados dos modelos, no entanto, o aumento indiscriminado não significa, necessariamente, melhoria na capacidade preditiva, ou seja, após um certo número de neurônios, não foram percebidas diferenças significativas nos resultados e, em alguns casos, até mesmo uma piora de desempenho foi observada. Apesar dessa tendência de melhoria, foi possível observar que também são encontrados bons resultados na categoria A, o que mostra que modelos com menos neurônios não devem ser necessariamente descartados. Com relação ao número de camadas intermediárias, os testes de hipótese realizados no item 5.11.2 mostram que, o aumento do número dessas camadas não resulta em modelos melhores. Esses resultados não implicam necessariamente que modelos com duas ou três camadas são ruins, nem que modelos de apenas uma camada são sempre os melhores. Isso fica evidenciado pelo fato de que, entre os resultados que apresentam os menores RMSE, encontram-se modelos com 2 e 3 camadas.

Nos itens 5.11.1 e 5.11.2, foi possível notar a importância do número de neurônios e de camadas intermediárias nos resultados obtidos. No entanto, a estratégia de buscar uma boa RNA focando na variação de um parâmetro de cada vez se mostrou ineficiente. A abordagem mais interessante consiste em buscar uma configuração de RNA capaz de retornar métricas satisfatórias por meio da variação de parâmetros, processo que a literatura chama de ajuste fino de hiperparâmetros, sem a necessidade de fixar determinados parâmetros enquanto variam-se outros.

Outro ponto a ser destacado nessas simulações é a importância da utilização de critérios como o *Early Stopping* e *Pruning*. Tais critérios foram fundamentais para evitar o *overfitting* nos cenários 1HLES, 2HLES, 3HLES, 1HLPR, 2HLPR e 3HLPR. E quando comparados, o *Early Stopping* permitiu a realização de cálculos mais rapidamente, porém apresentou resultados mais dispersos do que os calculados usando *Pruning*.

As demais simulações de RNA reuniram um total de 3.564 modelos (item 5.12), onde foi realizada uma maior busca no que se refere aos parâmetros testados. Os resultados mostraram que RNA não linearizadas (sem transformações) são capazes de fornecer métricas muito boas, por exemplo, os modelos não linearizados selecionados apresentaram RMSE menores do que os modelos linearizados.

Conclui-se que não existe um número fixo para a quantidade de neurônios, camadas neurais ou qualquer outro parâmetro que deva ser empregado sempre, pois cada problema é único e deve ser tratado segundo suas peculiaridades. O melhor caminho é o de testar vários modelos de RNA com parâmetros diferentes e realizar uma análise cuidadosa dos resultados para encontrar uma configuração que consiga fazer boas previsões.

Com relação à verificação de qualidade das RNA, é possível entender que a utilização de gráficos como: curvas de convergência, gráficos de previsão e a distribuição espacial dos resíduos são pontos de partida importantes para garantir uma qualidade mínima das RNA, uma vez que tais modelos não precisam obedecer aos pressupostos básicos como os da RLC, conforme visto no item 5.10.5.

A comparação realizada no item 6.2 apresenta modelos de RNA (com e sem transformações) com capacidade preditiva superior ao modelo de RLC, sendo essa a principal conclusão da pesquisa, porque confirma a hipótese apresentada no capítulo 1. Para essa verificação, foram avaliados os 52 imóveis da amostra de validação. Nesse conjunto avaliado, todos os modelos de RNA apresentaram métricas de desempenho superiores em comparação à RLC. Para fins de comparação, são apresentados alguns resultados: A RNA-1 apresentou RMSE, MAE e MAPE, respectivamente, iguais a 329.338,34, 212.364,91 e 14,00%. A RNA-11 forneceu 333.445,70, 214.502,72 e 13,80%. E o modelo de RLC obteve 408.853,13, 240.121,12 e 14,70%. Vale ressaltar, no entanto, que capacidade

preditiva não significa necessariamente que esses modelos de RNA possuem comportamento superior à RLC no que se refere à representação do mercado. Para isso, foi realizada a análise de elasticidade das variáveis no item 6.3.

Para realizar a análise de elasticidade foi utilizado o modelo RLC e duas RNA dentre as selecionadas nos itens 5.12.1 (com transformação) e 5.12.2 (sem transformação). Nessa análise, em todos os modelos, a variável “Área” foi a que mais impactou no valor do imóvel, comportamento já esperado. A variável que apresenta menor impacto no valor em RLC e RNA-11 é o número de suítes e na RNA-1 é o Norte. De modo geral, os modelos RLC e RNA-1 são os que apresentam um melhor comportamento na análise de elasticidade, ressaltando que a RLC representa melhor a variável padrão, pois há um salto não linear entre os padrões, e a RNA-1 foi o único modelo que percebeu a seguinte relação no que se refere às variáveis de localização: em regiões mais afastadas da Beira-mar, a variável “Norte” possui maior impacto no valor e a “Distância à Beira-mar” possui menor impacto, e o contrário para regiões mais próximas. Destaca-se que RNA-11 apresentou comportamento bem diferente dos demais modelos na variável suítes, indicando que essa variável não precisaria ter sido utilizada no modelo.

Em suma, com relação à capacidade preditiva, todos os modelos de RNA utilizados na avaliação da amostra de 52 imóveis apresentaram métricas (RMSE, MAE e MAPE) melhores do que a RLC. Com relação à análise de elasticidade das variáveis, o modelo de RLC e o modelo de RNA-1 apresentaram um melhor comportamento, já o modelo RNA-11 apresentou um comportamento não desejado, pois uma de suas variáveis (“suítes”) poderia ser descartada. Isso suscita uma reflexão: apesar de métricas superiores, redes neurais podem, na verdade, apresentar comportamento indesejado em alguma variável, e isso pode ser verificado na análise de elasticidade. Isso reforça a necessidade de uma avaliação mais abrangente, que vá além das métricas.

Desse modo, a boa capacidade preditiva das RNA aponta para bons prognósticos na utilização desse método em avaliações imobiliárias. Por outro lado, é necessário reconhecer que a validação de modelos de RLC segue uma abordagem já muito bem estabelecida, ao contrário dos métodos de *machine learning*, o que tem dificultado a disseminação de métodos desta natureza, como é o caso das redes neurais.

Portanto, como propostas para trabalhos futuros sugere-se o aprofundamento do estudo sobre a análise de elasticidade das variáveis em RNA e sua coerência com o mercado. Sugere-se ainda o aprofundamento no que tange à verificação/validação da qualidade de modelos de RNA, uma vez que não precisam seguir pressupostos como a RLC. E, nesse sentido, que haja aprofundamento e investigação sobre o uso de RNA em casos em que a RLC apresente problemas de dependência espacial ou não linearidade. Por fim, como sugestão adicional, que outras técnicas de inteligência

artificial, não descritas na NBR 14653-2 (ABNT, 2011), possam ser estudadas em avaliações imobiliárias: como o *Random Forest* e a *Support Vector Machine*.

REFERÊNCIAS BIBLIOGRÁFICAS

ABUNAHMAN, S. A. **Curso básico de engenharia legal e de avaliações**. São Paulo: Pini, 2008.

AMIDI A.; AMIDI S. **Deep Learning Tips and Tricks cheatsheet**. Stanford's CS 230 Deep Learning course. 2018. Disponível em: <https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-deep-learning-tips-and-tricks>. Acesso em 14 fev. 23.

ANSELIN, L. **Exploring Spatial Data with GeoDa™: A Workbook**. Spatial Analysis Laboratory, Department of Geography, University of Illinois, 2005. Disponível em: <https://geodacenter.github.io/docs/geodaworkbook.pdf>. Acesso em 16 fev. 23.

AFSHARTOUS, D.; PRESTON, R. A. **Key Results of Interaction Models with Centering**. Journal of Statistics Education, Taylor & Francis, 2011, DOI: 10.1080/10691898.2011.11889620. Acesso em 15 fev. 23.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS (ABNT). **NBR 14653-1: Avaliação de bens Parte 1: Procedimentos gerais**. Rio de Janeiro, 2019.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS (ABNT). **NBR 14653-2: Avaliação de bens Parte 2: Imóveis Urbanos**. Rio de Janeiro, 2004.

BALTAGI, B. **Econometrics**. 5º ed. New York: Springer Science & Business Media, 2011. Disponível em: https://link.springer.com/chapter/10.1007/978-3-642-20059-5_1. Acesso em 16 fev. 23.

BAPTISTELLA, M; CUNICO, L. H. B; STEINER, M. T. A. **O uso de Redes Neurais na engenharia de avaliações: determinação dos valores venais de imóveis urbanos**. Revista de Ciências Exatas e Naturais, 9, 2, 215-229. Disponível em: <https://revistas.unicentro.br/index.php/RECEN/article/view/82>. Acesso em 09 jun. 23.

BERRINI, L. C. **Avaliações de imóveis**. São Paulo: Escolas Profissionais Salesianas, 1949.

BERTONCELLO, A. G.; ZANONI RAMOS, A.; TRESSINO CAMPOS, G.; PINEDA TORQUATO, H. **Loop econômico: mercado imobiliário influencia e é influenciado pelas condições socioeconômicas**. Colloquium Socialis. ISSN: 2526-7035, v. 3, n. 3, p. 35-44, 7 nov. 2019. Disponível em <http://journal.unoeste.br/index.php/cs/article/view/3324>. Acesso em 03/02/2021.

BLAUSEN.COM STAFF. **Medical gallery of Blausen Medical 2014**. WikiJournal of Medicine 1 (2). DOI:10.15347/wjm/2014.010. ISSN 2002-4436. 2014. Disponível em: <<https://doi.org/10.15347/wjm/2014.010>>. Acesso em 23 abr. 2021.

BRAGA, A. P.; LUDERMIR, T. B.; CARVALHO, A. C. P. de L. F. **Redes neurais artificiais: teoria e aplicações**. Rio de Janeiro: LTC, 2000.

- BRASIL, Ministério do Planejamento, Desenvolvimento e Gestão, Secretaria do Patrimônio da União – SPU. **Manual de Avaliações de Imóveis do Patrimônio da União**. Brasília: 2018. Disponível em: <<https://www.gov.br/economia/pt-br/centrais-de-conteudo/publicacoes/planejamento/publicacoes-sobre-patrimonio-da-uniao/manual-de-avaliacao-de-imoveis-2018-versao-impressa-1.pdf/view>>. Acesso em 17 nov. 2020
- BRUCE, P.; BRUCE, A. **Estatística Prática para Cientistas de Dados: 50 Conceitos Essenciais**. Rio de Janeiro: Alta Books, 2019.
- CAPELLANO, L. H. Princípios de Estatística. IBABE/SP. **Engenharia de Avaliações Volume 1**. São Paulo: Leud, 2014. p. 97-175.
- CECHIN, A.; SOUTO, A.; GONZÁLEZ, M. A. **Real estate value at Porto Alegre city using artificial neural networks**. Proceedings. Vol.1. Sixth Brazilian Symposium on Neural Networks, Rio de Janeiro, Brazil, 2000, pp. 237-242, doi: 10.1109/SBRN.2000.889745. Disponível em: <https://ieeexplore.ieee.org/abstract/document/889745>. Acesso em 02 mar. 23.
- CHANASIT, K. et al. **A Real Estate Valuation Model Using Boosted Feature Selection**. IEEE Access, v. 9, pp. 86938-86953, 2021. DOI: 10.1109/ACCESS.2021.3089198. Disponível em: <https://ieeexplore.ieee.org/document/9454508>. Acesso em 09 jun. 23.
- CHIAPPINI, F.A.; AZCARATE, S.M.; TEGLIA, C.M.; GOICOECHEA, H.C. **Fundamentals of Design of Experiments and Optimization: Data Modeling in Response Surface Methodology**. In: Breitzkreitz, M.C., Goicoechea, H. (eds) Introduction to Quality by Design in Pharmaceutical Manufacturing and Analytical Development. AAPS Introductions in the Pharmaceutical Sciences, vol 10. Springer, Cham, 2023.
- DALAL, D. K., & ZICKAR, M. J. (2012). **Some Common Myths About Centering Predictor Variables in Moderated Multiple Regression and Polynomial Regression**. Organizational Research Methods, p. 339–362. Disponível em: <https://doi.org/10.1177/1094428111430540>. Acesso em 15 fev. 23.
- DA SILVA, I. N.; SPATTI, D. H.; FLAUZINO, R. A. **Redes Neurais Artificiais para engenharia e ciências aplicadas: fundamentos teóricos e aspectos práticos**. São Paulo: Artliber, 2010.
- DANTAS, R. A. **Engenharia de avaliações: uma introdução à metodologia científica**. São Paulo: Pini, 2012.
- GAZOLA, S. **Construção de um modelo de regressão para avaliação de imóveis**. 2002. 110f. Dissertação (Mestrado). PPGEP, UFSC, Florianópolis. Disponível em: <<https://repositorio.ufsc.br/bitstream/handle/123456789/82455/193809.pdf>>. Acesso em 23 abr. 2021.
- GÉRON, A. **Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems**. Sebastopol, CA: O'Reilly Media, Incorporated, 2019.
- GOLDBERGER, A. S.; GOLDBERGER, A. S. G. **A Course in Econometrics**. Cambridge: Harvard University Press, 1191.

GONZÁLEZ, M. A. S.; FORMOSO, C. T. **Análise Conceitual das Dificuldades na Determinação de Modelos de Formação de Preços Através da Análise de Regressão.** Engenharia Civil–UM (Universidade do Milho), Portugal, v.8, p 65-75, 2000. Disponível em: https://www.civil.uminho.pt/revista/artigos/Num8/Pag_65-75.pdf. Acesso em: 09 jun. 2023.

GRUS, J. **Data science do zero: primeiras regras com o Python.** Rio de Janeiro: Alta Books, 2016. Tradução de Weligton Nascimento.

GUJARATI, D. N.; PORTER, D. C. **Econometria básica - 5ª ed.** Porto Alegre: AMGH, 2011. Traduzido por Denise Durante, Mônica Rosemberg, Maria Lúcia G. L. Rosa.

GUYON, I. **Neural networks and applications tutorial.** Physics Reports, Vol. 207. Nº. 3. p. 215-259, 1991. DOI: [doi.org/10.1016/0370-1573\(91\)90146-D](https://doi.org/10.1016/0370-1573(91)90146-D). Disponível em: [sciencedirect.com/science/article/pii/037015739190146D](https://www.sciencedirect.com/science/article/pii/037015739190146D). Acesso em 10 abr. 2021.

ASTERIOU, D.; HALL, S.G. **Applied Econometrics.** 2º Ed. New York, Palgrave Macmillan, 2011

HAYKIN, S. **Redes Neurais: Princípios e prática.** Porto Alegre: Bookman, 2001. Traduzido por Paulo Martins Engel.

HEATON, J. **Training Neural Networks.** Curso T81-558: Applications of Deep Neural Networks do Professor Jeff Heaton, Washington University in St. Louis, 2022. Disponível em: https://github.com/jeffheaton/t81_558_deep_learning/blob/master/t81_558_class_04_4_backprop.ipynb. Acesso em 16 fev. 23.

HEATON, J. **Artificial Intelligence for Humans, Volume 3: Deep Learning and Neural Networks.** Heaton Research, Inc. Edição Kindle. Chesterfield, 2015.

IBM. **What is recurrent neural networks?** Disponível em: https://www.ibm.com/topics/recurrent-neural-networks?mhsrc=ibmsearch_a&mhq=Recurrent%20Neural%20Networks. Acesso em 13 fevereiro 2023.

HLAVÁČ, V.; JEFFERY, K. G; WIEDERMANN, J. **SOFSEM 2000: Theory and Practice of Informatics: 27th Conference on Current Trends in Theory and Practice of Informatics Milovy, Czech Republic, November 25 - December 2, 2000 Proceedings.** Berlin: Springer Berlin Heidelberg, 2000. Disponível em: <https://link.springer.com/book/10.1007/3-540-44411-4>. Acesso em 15 fev. 23.

HARRELL, F. E. **Regression modeling strategies: with applications to linear models, logistic and ordinal regression, and survival analysis.** New York: Springer, 2001. Disponível em: <https://link.springer.com/book/10.1007/978-3-319-19425-7>. Acesso em 15 fev. 23.

HOFFMANN, R. **Análise estatística de relações lineares e não lineares.** Piracicaba: Portal de Livros Abertos da USP, 2016. Disponível em: www.livrosabertos.sibi.usp.br/portaldelivrosUSP/catalog/book/74. Acesso em 14 maio 2021.

HUANG, S. C.; HUANG, Y. F. **Bounds on the number of hidden neurons in multilayer perceptrons.** IEEE Transactions on Neural Networks, vol. 2, no. 1, pp. 47-55, 1991, doi: 10.1109/72.80290. Disponível em: <https://ieeexplore.ieee.org/document/80290>. Acesso em 13 fev. 23.

International Association of Assessing Officers - IAAO. **Standard on Automated Valuation Models (AVMS)**. Kansas City, Missouri, 2018. Disponível em https://www.iaao.org/media/standards/Standard_on_Automated_Valuation_Models.pdf. Acesso em 03/02/2021.

KOLMOGOROV–SMIRNOV TEST. **The Concise Encyclopedia of Statistics**; Springer: New York, 2008; pp 283– 287. Disponível em: https://link.springer.com/referenceworkentry/10.1007/978-0-387-32833-1_214. Acesso em 28 dez. 2022.

JAIN, A. K.; MAO, J.; MOHIUDDIN, K. M. **Artificial neural networks: a tutorial**," in *Computer*, vol. 29, no. 3, pp. 31-44, 1996, doi: 10.1109/2.485891. Disponível em: <https://ieeexplore.ieee.org/document/485891>. Acesso em 13 fevereiro 2023.

JAIN, G.; MAHARA, T.; SHARMA, S.C.; AGARWAL, S.; KIM, H. **TD-DNN: A Time Decay-Based Deep Neural Network for Recommendation System**. *Appl. Sci.* 2022, 12, 6398. Disponível em: <https://doi.org/10.3390/app12136398>. Acesso em 28 set. 2023

JUDD, C. M.; MCCLELLAND, G. H.; RYAN, C. S. **Data Analysis: A Model Comparison Approach To Regression, ANOVA, and Beyond**. New York: Taylor & Francis, 2017. Disponível em: bit.ly/3xry2n2.

KAASTRA, I.; BOYD, M. **Designing a neural network for forecasting financial and economic time series**. *Neurocomputing*, Volume 10, Issue 3, 1996, Pages 215-236, ISSN 0925-2312, [https://doi.org/10.1016/0925-2312\(95\)00039-9](https://doi.org/10.1016/0925-2312(95)00039-9). Disponível em: <https://www.sciencedirect.com/science/article/pii/0925231295000399>. Acesso em 13 fev. 2023.

KE, J.; Liu, X. **Empirical Analysis of Optimal Hidden Neurons in Neural Network Modeling for Stock Prediction**. *IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application*, Wuhan, China, 2008, pp. 828-832, doi: 10.1109/PACIIA.2008.363. Disponível em: <https://ieeexplore.ieee.org/document/4756892>. Acesso em 13 fev. 23.

KINGMA, D. P.; BA, J. **Adam: A method for stochastic optimization**. In *ICLR (Poster)*, 2015. Disponível em: <https://arxiv.org/abs/1412.6980>. Acesso em 28 set. 23.

KMENTA, J. **Elementos de econometria: teoria econométrica básica**. São Paulo: Atlas, 1988.

KOENKER, R. **A note on studentizing a test for heteroscedasticity**. *Journal of Econometrics*, Volume 17, Issue 1, 1981, p.107-112, ISSN 0304-4076. Disponível em: [https://doi.org/10.1016/0304-4076\(81\)90062-2](https://doi.org/10.1016/0304-4076(81)90062-2). Acesso em 15 fev. 2023.

KRÄMER, W.; SONNBERGER, H. **The Linear Regression Model Under Test**. Heidelberg: Physica-Verlag, 1986. Disponível em: <https://doi.org/10.1007/978-3-642-95876-2>. Acesso em 15 fev. 23.

LARRAÑAGA, P.; ATIENZA, D.; DIAZ-ROZO, J.; OGBECHIE, A.; PUERTO-SANTANA, C. E.; BIELZA, C. **Industrial Applications of Machine Learning**. Estados Unidos: CRC Press, 2018.

LIANG, T.; GLOSSNER, J.; WANG, L.; SHI, S.; ZHANG, X.. **Pruning and quantization for deep neural network acceleration: A survey**. Neurocomputing. Vol. 461. p. 370-403. DOI: doi.org/10.1016/j.neucom.2021.07.045. Disponível em: [sciencedirect.com/science/article/abs/pii/S0925231221010894](https://www.sciencedirect.com/science/article/abs/pii/S0925231221010894). Acesso em 29 abr. 2021.

LIEBENWEIN, L.; BAYKAL, C.; CARTER, B.; GIFFORD, D.; RUS, D. **Lost in pruning: the effects of pruning neural networks beyond test accuracy**. Disponível em: <https://arxiv.org/abs/2103.03014>. Acesso em 29 ago. 2021.

LIMA, I.; PINHEIRO, C. A. M.; SANTOS, F. A. O. **Inteligência Artificial**. Rio de Janeiro: Elsevier, 2014.

LUDWIG JR., O.; MONTGOMERY, E. **Redes Neurais: Fundamentos e Aplicações com Programas em C**. Rio de Janeiro: Editora Ciência Moderna Ltda. 2007.

MADHIARASAN, M.; DEEPA, S.N. **A novel criterion to select hidden neuron numbers in improved back propagation networks for wind speed forecasting**. Appl Intell 44, 878–893 (2016). Disponível em: <https://doi.org/10.1007/s10489-015-0737-z>. Acesso em 14 fev. 2023.

McCLUSKEY, W. J. *et al.* **Prediction accuracy in mass appraisal: a comparison of modern approaches**. Journal of Property Research, v. 30, n. 4, p. 239-265, 2013. Disponível em: <https://doi.org/10.1080/09599916.2013.781204>. Acesso em 09 jun. 2023.

MERRIAM-WEBSTER. **Merriam-webster's dictionary and thesaurus**. Martinsburg, West Virginia: Quad Graphics. 2014.

MOOLAYIL, J. **Learn Keras for Deep Neural Networks: A Fast-Track Approach to Modern Deep Learning with Python**. Alemanha: Apress, 2018.

OLIVEIRA, A. M. B. D.; GRANDISKI, P.. Métodos Científicos e a Engenharia de Avaliações (com ênfase em Inferência Estatística). IBABE/SP. **Engenharia de Avaliações Volume 2**. São Paulo: Leud, 2014. p. 13-118.

PELLI NETO, A. **Redes neurais artificiais aplicadas às avaliações em massa estudo de caso para a cidade de Belo Horizonte/MG**. Dissertação (Mestrado). UFMG, Belo Horizonte. Disponível em: repositorio.ufmg.br/handle/1843/AVFV-6W7R62. Acesso em 29 ago. 2021.

PENNSSTATE. **Stepwise Regression**. Material do curso STAT 501: Regression Methods. Disponível em: <https://online.stat.psu.edu/stat501/lesson/10/10.2> . Acesso em 15 fev. 23.

PIETRON, M.; WIELGOSZ, M. **Retrain or Not Retrain? - Efficient Pruning Methods of Deep CNN Networks**. In: Krzhizhanovskaya V.V. et al. (eds) Computational Science – ICCS 2020. ICCS 2020. Lecture Notes in Computer Science, vol. 12139. Springer, Cham. Disponível em: https://doi.org/10.1007/978-3-030-50420-5_34. Acesso em 29 ago. 2021.

PRAKASH, K.B.; KANNAN, R.; ALEXANDER, S.A.; Kanagachidambaresan, G.R. **Advanced Deep Learning for Engineers and Scientists: A Practical Approach**. EAI/Springer Innovations in Communication and Computing. Springer, 2021. Disponível em: https://books.google.com.br/books?id=o_05EAAAQBAJ&newbks. Acesso em 15 maio 2023.

RAO, C. R.; SHALABH, TOUTENBURG, H.; HEUMANN, C. **Linear Models and Generalizations: Least Squares and Alternatives**. Berlin, Heidelberg: Springer, 2008. Disponível em: <https://doi.org/10.1007/978-3-540-74227-2>. Acesso em 15 fev. 23.

ROBINSON, C.; SCHUMACKER, R.E. **Interaction effects: Centering, variance inflation factor, and interpretation issues**. *Mult. Linear Regres. Viewp.* 2009, 35, 6–11. Disponível em: <http://www.glmj.org/archives.html>. Acesso em 15 fev. 23.

RODRÍGUEZ-ARDURA, I.; MESEGUER-ARTOLA, A. **A PLS-Neural Network Analysis of Motivational Orientations Leading to Facebook Engagement and the Moderating Roles of Flow and Age**. *Front Psychol*, 2020.

SEM, A.; SRIVASTAVA, M. **Regression Analysis: Theory, Methods, and Applications**. New York, NY: Springer, 1990. Disponível em: <https://doi.org/10.1007/978-1-4612-4470-7>. Acesso em 15 fev. 23.

SHAN, N.; YE, Z.; CUI, X. **Collaborative Intelligence: Accelerating Deep Neural Network Inference via Device-Edge Synergy**. *Security and Communication Networks*. Volume 2020. Disponível em: <https://doi.org/10.1155/2020/8831341>. Acesso em 29 ago. 2021.

SHEELA, K. G.; DEEPA, S. N. **A New algorithm to find number of hidden neurons in Radial Basis Function Networks for Wind Speed Prediction in Renewable Energy Systems**. *CEAI*, Vol.15, No.3 pp. 30-37, 2013a. Disponível em: <http://www.ceai.srait.ro/index.php?journal=ceai&page=article&op=view&path%5B%5D=1590>. Acesso em 29 ago. 2021.

SHEELA, K. G.; DEEPA, S. N. **Review on Methods to Fix Number of Hidden Neurons in Neural Networks**. *Mathematical Problems in Engineering* Vol. 2013b. Disponível em: <http://dx.doi.org/10.1155/2013/425740>. Acesso em 29 ago. 2021.

SHIBATA, K.; IKEDA, Y. **Effect of number of hidden neurons on learning in large-scale layered neural networks**. 2009 ICCAS-SICE, Fukuoka, Japan, 2009, pp. 5008-5013. Disponível em: <https://ieeexplore.ieee.org/document/5334631>. Acesso em 14 de fev. 2023.

SHILIN, Q.; ZHIFENG, S.; HUIFANG, F.; KUN, L.. **BP Neural Network for the Prediction of Urban Building Energy Consumption Based on Matlab and its Application**. 2010 Second International Conference on Computer Modeling and Simulation, 2010, p. 263-267, DOI: 10.1109/ICCMS.2010.118. Disponível em: <https://ieeexplore.ieee.org/document/5421081>. Acesso em 20 set. 2022.

SMITH, G. **Step away from stepwise**. *J Big Data*, [S.l.], v. 5, n. 1, p. 32, 2018. Disponível em: <https://doi.org/10.1186/s40537-018-0143-6>. Acesso em 15 de fev. de 2023

STATSMODELS. **Regression Diagnostics and Specification Tests**. Disponível em: <https://www.statsmodels.org/stable/diagnostic.html>. Acesso em 17 fev. 23.

STATSMODELS. **Source code for statsmodels.stats.diagnostic**. Disponível em: https://www.statsmodels.org/stable/_modules/statsmodels/stats/diagnostic.html#linear_lm. Acesso em 17 fev. 23.

TAMURA, S.; TATEISHI, M. **Capabilities of a four-layered feedforward neural network: four layers versus three.** IEEE Transactions on Neural Networks, vol. 8, no. 2, pp. 251-255, 1997, doi: 10.1109/72.557662. Disponível em: <https://ieeexplore.ieee.org/document/557662>. Acesso em 13 fev. 23.

TORRES-PRUÑONOSA, J.; GARCÍA-ESTÉVEZ, P.; PRADO-ROMÁN, C. **Artificial Neural Network, Quantile and Semi-Log Regression Modelling of Mass Appraisal in Housing.** Mathematics. 2021; 9(7):783. Disponível em: <https://doi.org/10.3390/math9070783>. Acesso em 02 mar. 23.

TRENN, S. **Multilayer Perceptrons: Approximation Order and Necessary Number of Hidden Units.** in IEEE Transactions on Neural Networks, vol. 19, no. 5, pp. 836-844, May 2008, doi: 10.1109/TNN.2007.912306. Disponível em: <https://ieeexplore.ieee.org/document/4469950>. Acesso em 14 fev. 23.

UZAIR, M.; JAMIL, N. **Effects of Hidden Layers on the Efficiency of Neural Networks.** In: IEEE 23rd International Multitopic Conference (INMIC), 2020, Bahawalpur, Pakistan, 2020. p. 1-6. DOI: 10.1109/INMIC50486.2020.9318195. Disponível em <https://ieeexplore.ieee.org/document/9318195>. Acesso em 09 jun. 23.

VAN GAANS, P. M. F.; VRIEND, S. P. **Multiple linear regression with correlations among the predictor variables. Theory and computer algorithm ridge (fortran 77).** Computers & Geosciences, Vol. 16. Nº. 7. p. 933-952, 1990. DOI: [doi.org/10.1016/0098-3004\(90\)90104-2](https://doi.org/10.1016/0098-3004(90)90104-2). Disponível em: [sciencedirect.com/science/article/pii/0098300490901042](https://www.sciencedirect.com/science/article/pii/0098300490901042). Acesso em 08 abr. 2021.

VEGA-MÁRQUEZ, B.; NEPOMUCENO-CHAMORRO, I.; JURADO-CAMPOS N.; RUBIO-ESCUDE, C. **Deep Learning Techniques to Improve the Performance of Olive Oil Classification.** Frontiers in Chemistry. Vol. 7. 2020. DOI: 10.3389/fchem.2019.00929. Disponível em: <https://www.frontiersin.org/articles/10.3389/fchem.2019.00929>. Acesso em 14 fev. 23.

YASNITSKY, L.N. **Algorithm for searching and analyzing abnormal observations of statistical information based on the Arnold – Kolmogorov – Hecht-Nielsen theorem.** International Journal of Advanced Trends in Computer Science and Engineering, Volume 9, 2020. Disponível em: <https://www.warse.org/IJATCSE/static/pdf/file/ijatcse139922020.pdf>. Acesso em 28 fev. 23.

YASNITSKY, L.N.; YASNITSKY, V. L.; ALEKSEEV, A. O. **The Complex Neural Network Model for Mass Appraisal and Scenario Forecasting of the Urban Real Estate Market Value That Adapts Itself to Space and Time.** Complexity, vol. 2021, Article ID 5392170, 17 pages, 2021. Disponível em: <https://doi.org/10.1155/2021/5392170>. Acesso em 28 fev. 23.

YING, X. **An Overview of Overfitting and its Solutions.** Journal of Physics: Conference Series, Vol. 1168. DOI: 10.1088/1742-6596/1168/2/022022. Disponível em: <https://doi.org/10.1088/1742-6596/1168/2/022022>. Acesso em 29 ago. 2021.

YOTOV, K.; HADZHIKOLEV, E.; HADZHIKOLEVA, S. **Determining the Number of Neurons in Artificial Neural Networks for Approximation, Trained with Algorithms Using the Jacobi Matrix.** TEM Journal. Volume 9, p. 1320-1329, DOI: 10.18421/TEM94-02. Disponível em: https://www.temjournal.com/content/94/TEMJournalNovember2020_1320_1329.html. Acesso em 20 set. 2022.

WANAS, N.; AUDA, G.; KAMEL, M. S.; KARRAY, F. **On the optimal number of hidden nodes in a neural network**. Conference Proceedings. IEEE Canadian Conference on Electrical and Computer Engineering (Cat. No.98TH8341), 1998, pp. 918-921 vol.2, DOI: 10.1109/CCECE.1998.685648. Disponível em: <https://ieeexplore.ieee.org/document/685648>. Acesso em 20 set. 2022.

YOU, K.; LONG, M.; ANG, J.; ORDAN, M. I. **How Does Learning Rate Decay Help Modern Neural Networks**. DOI: 10.48550/arXiv.1908.01878. 2019. Disponível em: <https://ui.adsabs.harvard.edu/abs/2019arXiv190801878Y>. Acesso em 16 fev. 23.

ZIGANTO, D. **Model Tuning (Part 2 - Validation & Cross-Validation)**. Disponível em: <https://dziganto.github.io/cross-validation/data%20science/machine%20learning/model%20tuning/python/Model-Tuning-with-Validation-and-Cross-Validation/>. Acesso em 23 set. 23.

ZHANG, Z. **Variable selection with stepwise and best subset approaches**. Annals of Translational Medicine, North America, 4, mar. 2016. Disponível em: <https://atm.amegroups.org/article/view/9706>. Acesso em: 01 out. 2023.

APÊNDICE A – AMOSTRA COLETADA NO CENTRO DE FLORIANÓPOLIS

Os elementos com asterisco (*) foram retirados da amostra coletada.

Tabela 40 – Amostra coletada no Centro de Florianópolis

ID	VT (R\$)	Área (m ²)	WC (un.)	Suítes (un.)	Vagas (un.)	Padrão (classes)	Idade (anos)	Dist. (m)	Norte (m)
I 001	1.060.000,00	257,00	4,00	1,00	2,00	2,00	34,00	700,45	685,84
I 002	850.000,00	98,00	2,00	1,00	1,00	1,00	22,00	287,79	1.102,60
I 003	850.000,00	128,60	1,00	1,00	2,00	2,00	32,00	10,00	1.170,50
I 004	1.500.000,00	156,00	4,00	1,00	2,00	1,00	32,00	509,97	972,22
I 005	2.650.000,00	193,54	4,00	1,00	2,00	1,00	24,00	50,18	1.290,86
I 006	980.000,00	99,00	2,00	1,00	2,00	2,00	31,00	206,49	1.183,24
I 007	617.000,00	68,73	2,00	1,00	1,00	1,00	33,00	80,51	1.810,35
I 008	2.099.000,00	116,00	3,00	2,00	2,00	1,00	6,00	514,57	1.051,58
I 009	1.400.000,00	112,00	3,00	2,00	2,00	1,00	13,00	87,78	1.162,85
I 010	2.650.000,00	140,00	5,00	3,00	3,00	1,00	6,00	242,92	1.402,52
I 011	600.000,00	72,00	1,00	1,00	1,00	1,00	36,00	66,03	1.108,01
I 012*	790.000,00	100,00	2,00	1,00	2,00	1,00	14,00	386,56	1.268,46
I 013	1.350.000,00	117,00	4,00	3,00	2,00	1,00	4,00	610,17	750,03
I 014	600.000,00	98,64	2,00	1,00	1,00	2,00	37,00	69,36	1.118,74
I 015	425.000,00	42,00	1,00	1,00	1,00	2,00	17,00	660,26	657,15
I 016	210.000,00	37,00	1,00	0,00	0,00	3,00	32,00	748,81	556,35
I 017	255.000,00	32,82	1,00	1,00	0,00	2,00	26,00	1.308,70	217,82
I 018	190.000,00	40,11	1,00	0,00	0,00	3,00	33,00	1.123,69	363,54
I 019	1.920.000,00	237,00	4,00	2,00	2,00	1,00	22,00	205,06	1.214,82
I 020	420.000,00	67,65	2,00	0,00	1,00	3,00	30,00	1.181,64	339,73
I 021	215.000,00	40,11	1,00	1,00	0,00	3,00	33,00	1.122,17	364,08
I 022	450.000,00	72,00	2,00	0,00	1,00	2,00	33,00	1.462,45	14,64
I 023	550.000,00	96,00	3,00	1,00	1,00	2,00	32,00	840,81	845,42
I 024	730.000,00	96,00	3,00	1,00	1,00	2,00	32,00	841,46	845,72
I 025*	1.150.000,00	220,00	4,00	3,00	2,00	1,00	32,00	839,31	845,65
I 026	849.900,00	113,00	3,00	1,00	1,00	2,00	25,00	582,81	964,85
I 027	1.650.000,00	161,36	4,00	3,00	2,00	1,00	19,00	866,92	650,37
I 028	1.270.000,00	180,00	3,00	1,00	2,00	1,00	33,00	717,91	738,17
I 029	890.000,00	159,00	2,00	0,00	2,00	2,00	37,00	783,08	602,22
I 030*	450.000,00	131,00	3,00	1,00	1,00	2,00	30,00	1.179,84	340,57
I 031	880.000,00	180,00	4,00	2,00	1,00	1,00	30,00	1.183,62	338,68
I 032	2.990.000,00	180,00	6,00	4,00	3,00	1,00	1,00	498,14	1.022,98
I 033	1.484.000,00	125,00	4,00	3,00	3,00	1,00	12,00	648,75	1.066,32
I 034	1.280.000,00	196,00	3,00	2,00	2,00	1,00	37,00	354,64	862,41
I 035	1.070.000,00	100,25	3,00	1,00	1,00	2,00	19,00	244,32	978,36
I 036	1.150.000,00	130,00	3,00	1,00	1,00	2,00	27,00	58,26	1.060,78
I 037	850.000,00	93,00	2,00	1,00	1,00	2,00	32,00	436,77	832,48
I 038	630.000,00	106,00	2,00	1,00	1,00	2,00	33,00	645,72	807,07
I 039	450.000,00	100,00	3,00	0,00	1,00	3,00	33,00	991,57	449,97
I 040	940.000,00	120,00	3,00	1,00	1,00	1,00	30,00	270,79	1.114,11
I 041	1.800.000,00	125,00	5,00	3,00	2,00	1,00	2,00	382,25	922,43
I 042	2.850.000,00	167,00	6,00	3,00	2,00	1,00	11,00	432,28	725,49
I 043	2.800.000,00	189,00	3,00	1,00	2,00	1,00	6,00	234,44	1.330,44
I 044	2.778.569,79	139,00	4,00	3,00	2,00	1,00	2,00	91,19	1.502,07
I 045	2.650.000,00	170,00	4,00	3,00	4,00	1,00	6,00	642,58	863,34
I 046	2.650.000,00	157,00	3,00	3,00	2,00	1,00	1,00	267,53	1.028,92
I 047	2.500.000,00	184,00	4,00	3,00	2,00	1,00	4,00	387,56	1.120,28

Tabela 40 – Amostra coletada no Centro de Florianópolis

ID	VT (R\$)	Área (m ²)	WC (un.)	Suítes (un.)	Vagas (un.)	Padrão (classes)	Idade (anos)	Dist. (m)	Norte (m)
I 048	2.500.000,00	224,00	5,00	2,00	2,00	1,00	37,00	339,45	1.434,94
I 049	1.170.000,00	142,00	4,00	1,00	1,00	2,00	37,00	58,52	1.557,83
I 050	895.000,00	116,00	3,00	1,00	1,00	2,00	33,00	249,57	1.500,27
I 051	1.100.000,00	77,00	2,00	1,00	2,00	1,00	13,00	486,71	875,04
I 052	450.000,00	44,00	1,00	0,00	1,00	2,00	27,00	308,27	879,26
I 053	1.900.000,00	226,00	4,00	3,00	2,00	2,00	25,00	10,00	1.103,93
I 054	731.400,00	49,00	2,00	1,00	1,00	1,00	4,00	537,54	705,13
I 055	475.000,00	63,00	2,00	1,00	0,00	2,00	37,00	828,82	731,20
I 056	720.000,00	127,00	3,00	1,00	1,00	2,00	27,00	318,10	1.118,60
I 057	1.800.000,00	138,00	3,00	1,00	1,00	2,00	38,00	10,00	1.414,01
I 058	1.484.000,00	109,00	2,00	1,00	2,00	1,00	13,00	545,96	964,38
I 059	2.980.000,00	409,38	7,00	4,00	3,00	1,00	37,00	706,12	919,14
I 060	1.995.000,00	203,00	3,00	1,00	2,00	1,00	22,00	909,58	575,45
I 061	980.000,00	75,26	2,00	1,00	1,00	2,00	18,00	177,46	1.383,67
I 062	371.000,00	103,00	1,00	0,00	0,00	3,00	30,00	1.178,98	340,97
I 063	499.990,00	40,50	1,00	1,00	1,00	2,00	10,00	108,25	1.814,18
I 064	230.000,00	46,00	1,00	0,00	0,00	2,00	32,00	1.366,13	148,34
I 065*	280.000,00	57,00	1,00	0,00	0,00	3,00	38,00	1.210,61	443,12
I 066	2.005.000,00	90,00	3,00	2,00	2,00	1,00	7,00	150,42	1.104,42
I 067	2.162.000,00	226,00	2,00	1,00	2,00	1,00	25,00	10,00	1.106,78
I 068	1.650.000,00	112,00	3,00	2,00	2,00	1,00	13,00	85,72	1.164,76
I 069	750.000,00	83,00	2,00	1,00	1,00	2,00	18,00	681,54	532,09
I 070	2.490.000,00	250,00	5,00	2,00	3,00	1,00	23,00	389,82	1.234,57
I 071	2.500.000,00	287,26	5,00	2,00	3,00	1,00	29,00	674,32	1.063,70
I 072	2.400.000,00	177,77	4,00	2,00	4,00	1,00	9,00	255,49	1.313,07
I 073	2.900.000,00	331,00	4,00	2,00	2,00	1,00	31,00	10,00	1.311,42
I 074	2.000.000,00	220,00	5,00	2,00	3,00	1,00	24,00	429,42	1.252,89
I 075	2.700.000,00	254,50	5,00	2,00	2,00	2,00	29,00	175,57	1.051,29
I 076*	2.500.000,00	504,00	6,00	2,00	4,00	1,00	32,00	532,81	1.124,51
I 077*	3.300.000,00	133,00	5,00	3,00	2,00	1,00	13,00	89,29	1.161,32
I 078	1.702.000,00	70,00	3,00	2,00	2,00	1,00	7,00	151,70	1.103,27
I 079	1.150.000,00	127,00	2,00	1,00	2,00	1,00	16,00	87,25	1.408,34
I 080	680.000,00	40,00	1,00	0,00	1,00	1,00	3,00	510,30	1.285,46
I 081	320.000,00	54,00	1,00	0,00	1,00	3,00	34,00	654,78	599,47
I 082	370.000,00	47,00	1,00	0,00	1,00	2,00	43,00	225,14	1.471,70
I 083	1.498.000,00	128,00	3,00	1,00	1,00	1,00	35,00	98,77	1.499,34
I 084	2.289.900,00	184,00	5,00	2,00	2,00	1,00	24,00	70,56	1.568,07
I 085	2.900.000,00	221,00	5,00	2,00	2,00	1,00	30,00	10,00	1.193,24
I 086	1.545.000,00	171,00	3,00	1,00	2,00	1,00	14,00	619,15	1.083,38
I 087*	3.381.000,00	131,00	3,00	1,00	2,00	1,00	5,00	222,67	1.210,00
I 088	3.200.000,00	283,00	5,00	3,00	2,00	1,00	32,00	10,00	1.306,08
I 089*	3.494.798,00	210,00	3,00	2,00	2,00	1,00	4,00	385,13	821,19
I 090	1.850.000,00	264,00	4,00	1,00	2,00	2,00	37,00	149,05	1.307,91
I 091	1.700.000,00	98,00	2,00	1,00	2,00	1,00	5,00	52,03	1.266,10
I 092	1.480.000,00	184,00	4,00	2,00	2,00	1,00	32,00	612,04	753,79
I 093*	1.170.000,00	115,00	2,00	1,00	1,00	2,00	35,00	317,70	938,97
I 094	1.100.000,00	94,00	2,00	1,00	2,00	1,00	9,00	159,25	1.024,30
I 095	1.300.000,00	131,87	3,00	1,00	2,00	2,00	22,00	222,72	1.163,35
I 096	400.000,00	55,89	1,00	0,00	0,00	2,00	34,00	809,98	572,48
I 097*	160.000,00	58,00	2,00	0,00	0,00	3,00	32,00	1.060,72	405,82
I 098	205.000,00	22,00	1,00	0,00	1,00	3,00	32,00	1.379,91	156,78
I 099	259.000,00	37,00	1,00	0,00	0,00	3,00	32,00	1.370,86	165,56
I 100	3.180.000,00	500,00	6,00	3,00	3,00	1,00	33,00	641,25	809,32
I 101*	3.000.000,00	530,00	4,00	3,00	3,00	1,00	20,00	10,00	1.401,08

Tabela 40 – Amostra coletada no Centro de Florianópolis

ID	VT (R\$)	Área (m ²)	WC (un.)	Suítes (un.)	Vagas (un.)	Padrão (classes)	Idade (anos)	Dist. (m)	Norte (m)
I 102	530.000,00	108,64	2,00	0,00	1,00	3,00	32,00	441,99	977,25
I 103	1.700.000,00	220,00	4,00	2,00	2,00	1,00	34,00	336,86	883,21
I 104*	1.774.000,00	76,00	3,00	2,00	1,00	1,00	7,00	775,99	686,45
I 105	1.900.000,00	263,00	5,00	2,00	2,00	1,00	37,00	148,70	1.307,39
I 106	3.200.000,00	275,93	5,00	2,00	2,00	1,00	36,00	96,04	1.417,15
I 107	1.725.000,00	242,00	5,00	2,00	2,00	1,00	34,00	337,02	883,56
I 108	1.289.000,00	123,00	4,00	2,00	2,00	2,00	24,00	409,60	1.312,13
I 109	1.150.000,00	82,00	3,00	1,00	1,00	2,00	14,00	622,99	1.078,01
I 110	820.000,00	64,00	1,00	0,00	1,00	1,00	6,00	519,50	1.050,19
I 111	998.000,00	108,58	2,00	1,00	2,00	1,00	6,00	517,73	1.050,27
I 112	275.000,00	42,30	1,00	0,00	0,00	3,00	37,00	1.215,23	364,43
I 113	320.000,00	45,52	1,00	0,00	0,00	3,00	37,00	1.213,76	367,31
I 114	529.000,00	42,00	1,00	0,00	1,00	2,00	18,00	181,95	1.380,08
I 115*	3.500.000,00	202,00	5,00	3,00	3,00	1,00	20,00	10,00	1.397,89
I 116	3.078.000,00	282,00	4,00	2,00	2,00	1,00	37,00	10,00	1.547,00
I 117	1.430.000,00	117,00	2,00	1,00	1,00	1,00	37,00	10,00	1.507,74
I 118	1.400.000,00	94,29	3,00	1,00	2,00	1,00	4,00	387,77	819,25
I 119	1.390.000,00	102,72	2,00	1,00	2,00	1,00	9,00	158,78	1.025,50
I 120	850.000,00	88,90	2,00	1,00	1,00	2,00	24,00	221,55	938,74
I 121	530.000,00	113,00	3,00	1,00	1,00	3,00	38,00	1.209,01	446,27
I 122	1.200.000,00	100,00	2,00	2,00	2,00	2,00	23,00	210,99	1.562,42
I 123	1.880.000,00	190,00	5,00	2,00	3,00	1,00	23,00	115,26	1.117,59
I 124	1.050.000,00	122,00	2,00	1,00	2,00	2,00	28,00	66,37	1.131,84
I 125	880.000,00	117,00	2,00	1,00	2,00	2,00	31,00	488,91	978,73
I 126	799.000,00	100,00	2,00	1,00	1,00	1,00	33,00	534,61	743,66
I 127	1.150.000,00	147,00	4,00	1,00	2,00	1,00	31,00	308,58	1.351,85
I 128	1.290.000,00	135,00	4,00	2,00	2,00	2,00	21,00	133,28	1.493,14
I 129	1.080.000,00	125,00	3,00	1,00	2,00	2,00	28,00	523,91	1.140,89
I 130	650.000,00	120,00	2,00	0,00	1,00	2,00	37,00	970,90	577,28
I 131	459.000,00	67,73	3,00	2,00	1,00	3,00	26,00	997,30	610,03
I 132	800.000,00	68,00	2,00	1,00	1,00	2,00	13,00	486,76	873,94
I 133	597.000,00	115,00	2,00	0,00	1,00	2,00	31,00	282,68	940,20
I 134	749.000,00	67,00	2,00	1,00	1,00	1,00	8,00	1.172,85	405,48
I 135	1.850.000,00	250,00	3,00	1,00	2,00	2,00	37,00	10,00	1.506,14
I 136	3.000.000,00	237,00	5,00	2,00	2,00	1,00	32,00	69,76	1.897,92
I 137	889.900,00	110,00	3,00	1,00	2,00	2,00	25,00	292,64	1.468,43
I 138	220.000,00	35,00	1,00	0,00	0,00	3,00	33,00	957,37	495,68
I 139	335.000,00	55,00	1,00	0,00	0,00	2,00	43,00	302,18	1.393,83
I 140	1.295.000,00	121,00	3,00	2,00	2,00	1,00	25,00	398,19	1.316,03
I 141	2.950.000,00	233,50	6,00	4,00	4,00	1,00	17,00	406,67	1.128,40
I 142	3.100.000,00	306,00	1,00	2,00	4,00	1,00	29,00	340,53	1.242,87
I 143	750.000,00	100,00	3,00	2,00	1,00	2,00	27,00	575,83	1.202,21
I 144	710.000,00	101,00	2,00	1,00	2,00	2,00	21,00	1.184,85	491,72
I 145	3.200.000,00	172,00	5,00	3,00	3,00	1,00	17,00	83,41	1.343,89
I 146	2.650.000,00	160,00	4,00	3,00	2,00	1,00	19,00	197,07	1.058,98
I 147	535.000,00	87,00	3,00	1,00	1,00	2,00	30,00	652,69	1.113,09
I 148	450.000,00	81,00	2,00	1,00	1,00	2,00	33,00	1.134,38	548,21
I 149	970.000,00	100,00	3,00	1,00	1,00	2,00	24,00	641,18	566,23
I 150	1.050.000,00	85,00	3,00	2,00	2,00	1,00	4,00	689,07	782,77
I 151	1.200.000,00	121,00	4,00	1,00	1,00	2,00	31,00	46,62	1.337,38
I 152	1.800.000,00	219,00	5,00	2,00	2,00	1,00	33,00	147,57	1.080,81
I 153	1.780.000,00	145,00	4,00	3,00	3,00	1,00	9,00	522,13	1.216,28
I 154	1.690.000,00	170,00	4,00	3,00	3,00	1,00	31,00	315,45	905,66
I 155*	2.300.000,00	204,00	6,00	4,00	4,00	1,00	14,00	304,77	1.454,65

Tabela 40 – Amostra coletada no Centro de Florianópolis

ID	VT (R\$)	Área (m ²)	WC (un.)	Suítes (un.)	Vagas (un.)	Padrão (classes)	Idade (anos)	Dist. (m)	Norte (m)
I 156	2.240.000,00	224,00	4,00	2,00	2,00	1,00	37,00	335,11	1.438,11
I 157	2.099.700,00	190,64	4,00	2,00	2,00	1,00	4,00	686,16	786,07
I 158	2.100.000,00	279,00	1,00	1,00	2,00	1,00	33,00	620,64	1.150,16
I 159	1.890.000,00	222,00	4,00	3,00	3,00	1,00	23,00	915,06	760,24
I 160	750.000,00	110,00	2,00	0,00	2,00	1,00	33,00	634,23	748,75
I 161	230.000,00	52,00	1,00	0,00	1,00	3,00	32,00	1.375,40	160,93
I 162	258.000,00	52,61	1,00	0,00	1,00	3,00	33,00	1.467,91	10,00
I 163	225.000,00	29,00	1,00	0,00	1,00	2,00	34,00	636,86	673,92
I 164	169.000,00	24,76	1,00	0,00	0,00	3,00	37,00	604,09	657,38
I 165	2.700.000,00	316,00	6,00	4,00	3,00	1,00	23,00	115,77	1.117,84
I 166	850.000,00	102,00	2,00	1,00	1,00	2,00	25,00	69,33	1.795,60
I 167*	690.000,00	107,00	4,00	2,00	1,00	1,00	28,00	637,81	1.130,14
I 168	2.500.000,00	483,00	3,00	2,00	2,00	1,00	29,00	174,49	1.050,97
I 169	1.750.000,00	129,95	4,00	3,00	2,00	1,00	19,00	243,61	977,43
I 170	950.000,00	74,82	2,00	1,00	2,00	1,00	2,00	325,14	994,59
I 171	630.000,00	81,00	2,00	1,00	1,00	2,00	19,00	330,90	1.028,76
I 172	1.130.000,00	115,00	3,00	1,00	2,00	2,00	27,00	115,54	1.454,50
I 173	2.100.000,00	272,12	3,00	2,00	2,00	1,00	33,00	423,74	1.253,18
I 174	530.000,00	71,00	1,00	0,00	1,00	2,00	32,00	666,67	752,01
I 175	320.000,00	55,00	1,00	0,00	0,00	3,00	34,00	807,50	576,28
I 176	1.500.000,00	150,00	3,00	1,00	2,00	1,00	20,00	263,93	1.027,63
I 177	210.000,00	31,00	1,00	0,00	0,00	2,00	37,00	1.080,51	379,63
I 178	294.800,00	78,00	2,00	0,00	0,00	3,00	43,00	632,85	625,04
I 179	1.250.000,00	110,00	1,00	1,00	1,00	1,00	37,00	10,00	1.550,18
I 180*	2.995.000,00	396,00	6,00	4,00	4,00	1,00	36,00	10,00	1.173,95
I 181	3.600.000,00	138,44	4,00	3,00	2,00	1,00	2,00	10,00	1.574,30
I 182	3.650.000,00	256,00	5,00	4,00	4,00	1,00	10,00	359,59	1.177,16
I 183	4.200.000,00	253,36	3,00	3,00	2,00	1,00	28,00	10,00	1.631,32
I 184*	4.150.000,00	423,07	6,00	4,00	4,00	1,00	14,00	10,00	1.659,58
I 185	6.750.000,00	309,70	7,00	4,00	4,00	1,00	14,00	10,00	1.662,58
I 186	4.350.000,00	287,00	5,00	4,00	3,00	1,00	18,00	10,00	1.496,85
I 187	4.200.000,00	260,00	4,00	2,00	3,00	1,00	28,00	10,00	1.328,88
I 188*	1.010.000,00	190,00	3,00	1,00	2,00	2,00	26,00	135,77	1.670,16
I 189	900.000,00	100,00	2,00	1,00	1,00	2,00	23,00	211,37	1.556,63
I 190	1.595.223,00	95,00	2,00	1,00	1,00	1,00	1,00	224,59	1.437,93
I 191	180.000,00	27,35	1,00	0,00	0,00	3,00	33,00	963,48	491,60
I 192	330.000,00	61,00	1,00	0,00	1,00	3,00	37,00	441,64	1.418,41
I 193	350.000,00	66,00	1,00	0,00	1,00	2,00	43,00	301,67	1.395,90
I 194	5.990.000,00	436,36	6,00	3,00	4,00	1,00	15,00	446,60	1.301,45
I 195	450.000,00	67,00	1,00	0,00	1,00	2,00	10,00	490,96	817,40
I 196	750.000,00	97,00	3,00	1,00	1,00	2,00	38,00	329,62	1.047,50
I 197	600.000,00	76,36	2,00	1,00	1,00	3,00	22,00	276,91	1.623,33
I 198	424.000,00	74,76	1,00	0,00	0,00	2,00	33,00	794,11	505,33
I 199*	710.000,00	117,00	2,00	1,00	1,00	2,00	32,00	715,90	755,63
I 200	696.000,00	117,00	3,00	1,00	1,00	2,00	21,00	1.181,94	491,95
I 201	490.000,00	110,00	3,00	0,00	1,00	3,00	32,00	895,08	518,29
I 202	530.000,00	43,00	1,00	0,00	1,00	2,00	18,00	177,40	1.384,80
I 203	3.710.000,00	265,00	5,00	4,00	3,00	1,00	4,00	381,60	823,96
I 204	3.940.000,00	343,74	5,00	4,00	4,00	1,00	4,00	220,86	1.003,02
I 205*	649.000,00	59,00	1,00	0,00	1,00	2,00	29,00	1.497,91	94,69
I 206	680.000,00	65,00	2,00	1,00	1,00	2,00	17,00	515,13	1.270,38
I 207	700.000,00	103,00	3,00	1,00	1,00	3,00	39,00	750,04	716,52
I 208	750.000,00	96,00	3,00	1,00	1,00	3,00	34,00	79,53	1.081,71
I 209*	3.919.000,00	108,00	4,00	2,00	2,00	1,00	5,00	215,47	1.203,72

Tabela 40 – Amostra coletada no Centro de Florianópolis

ID	VT (R\$)	Área (m ²)	WC (un.)	Suítes (un.)	Vagas (un.)	Padrão (classes)	Idade (anos)	Dist. (m)	Norte (m)
I 210	3.800.000,00	361,00	5,00	2,00	3,00	1,00	2,00	89,63	1.505,73
I 211	3.850.000,00	259,00	6,00	4,00	3,00	1,00	28,00	10,00	1.330,03
I 212	3.900.000,00	315,00	5,00	2,00	4,00	1,00	36,00	10,00	1.294,68
I 213	350.000,00	73,80	2,00	0,00	0,00	3,00	32,00	888,55	522,94
I 214	371.000,00	96,00	2,00	1,00	0,00	2,00	37,00	1.295,67	400,45
I 215	350.000,00	59,00	1,00	0,00	1,00	3,00	30,00	396,97	861,96
I 216	570.000,00	75,00	3,00	3,00	1,00	2,00	37,00	69,17	1.120,43
I 217*	450.000,00	51,00	1,00	0,00	0,00	3,00	33,00	660,33	620,12
I 218	5.950.000,00	485,77	4,00	4,00	3,00	1,00	37,00	10,00	1.179,08
I 219	5.900.000,00	250,00	5,00	3,00	3,00	1,00	2,00	10,00	1.578,79
I 220*	4.900.000,00	262,00	4,00	3,00	2,00	1,00	37,00	10,00	1.548,42
I 221	4.200.000,00	343,00	4,00	3,00	2,00	1,00	22,00	10,00	1.419,48
I 222	5.000.000,00	287,60	5,00	3,00	3,00	1,00	18,00	10,00	1.495,36
I 223	5.500.000,00	400,00	7,00	4,00	3,00	1,00	23,00	391,19	1.231,01
I 224	5.380.000,00	250,00	4,00	3,00	3,00	1,00	2,00	10,00	1.576,32
I 225	530.000,00	99,00	2,00	0,00	1,00	2,00	37,00	957,86	445,26
I 226	400.000,00	57,00	1,00	0,00	0,00	3,00	36,00	334,66	1.093,17
I 227	390.000,00	66,00	2,00	0,00	0,00	2,00	37,00	1.016,56	498,93
I 228	650.000,00	84,00	2,00	0,00	1,00	2,00	35,00	49,73	1.301,96
I 229	435.000,00	100,00	2,00	0,00	0,00	3,00	37,00	953,72	448,16
I 230	540.000,00	110,00	2,00	0,00	1,00	3,00	36,00	901,64	513,84
I 231	480.000,00	68,97	1,00	0,00	0,00	2,00	35,00	50,00	1.302,85

APÊNDICE B – LOCALIZAÇÕES DOS DADOS DA AMOSTRA COLETADA

Tabela 41 – Nome da edificação e coordenadas UTM dos dados pesquisados

ID	Condomínio/Edifício	E (m)	N (m)
I_001	Residencial Halley	741.585,23	6.945.431,10
I_002	Boulevard Saint Michel	741.530,44	6.945.847,86
I_003	Arpoador	740.804,73	6.945.915,76
I_004	Presidente Coutinho	741.843,00	6.945.717,48
I_005	Beverly Boulevard Residence	741.343,31	6.946.036,12
I_006	Esteves Junior	741.515,37	6.945.928,49
I_007	Solar Angra dos Reis	742.465,43	6.946.555,61
I_008	Carmelo Faraco	742.066,27	6.945.796,84
I_009	Residencial Náutico	741.076,91	6.945.908,11
I_010	Residencial Zélia Freitas	742.189,55	6.946.147,78
I_011	Cláudia	740.811,07	6.945.853,27
I_013	Residencial Villagio di Francesco	741.491,45	6.945.495,29
I_014	Edifício Fontainebleau	740.865,89	6.945.864,00
I_015	Hannover Residence	741.370,80	6.945.402,41
I_016	Rua Felipe Schmidt, 554	741.348,45	6.945.301,61
I_017	Cittadella	742.105,61	6.944.963,08
I_018	Anita Garibaldi	741.957,41	6.945.108,80
I_019	Viena	741.605,90	6.945.960,08
I_020	Daniela	742.069,55	6.945.084,99
I_021	Anita Garibaldi	741.954,20	6.945.109,34
I_022	Topázio	741.981,42	6.944.759,89
I_023	Estoril	742.410,56	6.945.590,67
I_024	Estoril	742.412,74	6.945.590,98
I_026	Atlanta Bloco A	742.035,60	6.945.710,11
I_027	Solar de Castela	742.005,57	6.945.395,63
I_028	Portal da Ilha	741.797,90	6.945.483,42
I_029	Cruzeiro do Sul 30	741.595,38	6.945.347,47
I_031	Daniela	742.072,65	6.945.083,94
I_032	Vila Loyola	741.950,77	6.945.768,24
I_033	Anabella Residence	742.424,19	6.945.811,58
I_034	Morada Tannenbaum	741.012,53	6.945.607,67
I_035	Everest Park	741.015,90	6.945.723,62
I_036	Saint Laurent	740.705,11	6.945.806,04
I_037	Ana Cláudia	741.189,48	6.945.577,74
I_038	Dom Jaime Câmara	741.776,36	6.945.552,33
I_039	Itamarati	741.798,29	6.945.195,23

Tabela 41 – Nome da edificação e coordenadas UTM dos dados pesquisados

ID	Condomínio/Edifício	E (m)	N (m)
I_040	Residencial Haroldo Pederneiras	741.511,00	6.945.859,37
I_041	Domus Madalena	741.287,54	6.945.667,69
I_042	Orla Marítima Residence	740.874,78	6.945.470,75
I_043	Residencial Dungenheim	742.006,18	6.946.075,70
I_044	Residencial Forte De Sant'Ana	742.042,19	6.946.247,32
I_045	D'Art Home Design	741.934,79	6.945.608,60
I_046	Villa Dei Fiori	741.244,12	6.945.774,18
I_047	Die Perle Residence 621	741.894,87	6.945.865,54
I_048	Geranium	742.451,58	6.946.180,19
I_049	Rosemary	742.087,03	6.946.303,09
I_050	Tivoli	742.385,16	6.946.245,53
I_051	Ritz Class	741.476,24	6.945.620,30
I_052	Lindacap Flat Residence	740.894,26	6.945.624,52
I_053	Eugênio Müller	740.670,62	6.945.849,19
I_054	D One Home Design	741.124,34	6.945.450,39
I_055	Eduardo	742.114,21	6.945.476,45
I_056	Coronel Ganzo	741.674,94	6.945.863,86
I_057	Saint Claude	741.566,71	6.946.159,27
I_058	Celina de Vincenzi 414	741.929,77	6.945.709,64
I_059	Casa Blanca	742.243,07	6.945.664,40
I_060	Residencial Heitor Ferrari	741.916,18	6.945.320,71
I_061	Trompowsky Class	741.985,76	6.946.128,93
I_062	Daniela	742.064,85	6.945.086,23
I_063	Metropolitan HB	742.497,72	6.946.559,44
I_064	Trabalhador Catarinense	742.078,64	6.944.893,60
I_066	Estúdio Oceano	741.099,78	6.945.849,68
I_067	Eugênio Müller	740.671,57	6.945.852,04
I_068	Náutico	741.076,15	6.945.910,01
I_069	Porto dos corais	741.042,08	6.945.277,35
I_070	Residencial Orlando Becker	742.176,14	6.945.979,82
I_071	Residencial Ferreira Lima	742.472,87	6.945.808,96
I_072	Residencial Plaza Danúbio Residence	742.019,13	6.946.058,32
I_073	Richard Gottsmann	741.256,45	6.946.056,68
I_074	Residencial Amadeus Mozart	742.308,10	6.945.998,15
I_075	Residencial Villas	741.018,29	6.945.796,55
I_078	Estúdio Oceano	741.100,40	6.945.848,53
I_079	Costa Marina	741.808,16	6.946.153,60
I_080	Jardim Milano	742.534,39	6.946.030,71
I_081	Antônio Apóstolo	741.178,36	6.945.344,73

Tabela 41 – Nome da edificação e coordenadas UTM dos dados pesquisados

ID	Condomínio/Edifício	E (m)	N (m)
I_082	Iracema	742.282,96	6.946.216,96
I_083	Solar de Bocaiuva 2086	742.055,07	6.946.244,60
I_084	Residencial Villa Adriana	742.136,12	6.946.313,33
I_085	Residencial Villa Cristina	740.891,55	6.945.938,50
I_086	Residencial Regueira	742.393,36	6.945.828,63
I_088	Ilha do Arvoredo	741.240,35	6.946.051,34
I_090	Atlântico Norte	741.708,68	6.946.053,17
I_091	Dalva Shultz	741.274,46	6.946.011,35
I_092	Classic	741.508,75	6.945.499,05
I_094	Palas Athena Residence	740.859,95	6.945.769,56
I_095	Walter Meyer	741.506,85	6.945.908,61
I_096	Goya	741.591,03	6.945.317,74
I_098	Valparaizo	742.147,71	6.944.902,04
I_099	Valparaizo	742.145,41	6.944.910,82
I_100	Dom Jaime Câmara	741.768,92	6.945.554,58
I_102	Edifício Presidente	741.642,26	6.945.722,51
I_103	Georgios	741.021,03	6.945.628,47
I_105	Atlântico Norte	741.706,02	6.946.052,65
I_106	Costa Azul	741.855,58	6.946.162,41
I_107	Georgios	741.022,82	6.945.628,82
I_108	Residencial Mario Cantição	742.375,66	6.946.057,39
I_109	Residencial Regueira	742.391,33	6.945.823,27
I_110	Residencial Carmelo Faraco	742.075,54	6.945.795,45
I_111	Residencial Carmelo Faraco	742.071,22	6.945.795,53
I_112	Cristina	742.237,41	6.945.109,68
I_113	Cristina	742.240,81	6.945.112,57
I_114	Trompowsky Class	741.988,79	6.946.125,34
I_116	Morada do Norte	741.925,09	6.946.292,26
I_117	Edifício Belvedere	741.834,21	6.946.253,00
I_118	Place Residence	740.979,28	6.945.564,51
I_119	Palas Athena	740.862,76	6.945.770,76
I_120	Residencial Maison Classic	740.813,60	6.945.684,00
I_121	Conjunto Residencial Minerva	742.416,84	6.945.191,53
I_122	Residencial Jardim da Ilha	742.413,26	6.946.307,68
I_123	Residencial Fortaleza da Barra	741.026,87	6.945.862,85
I_124	Alameda dos Almirantes	740.904,38	6.945.877,10
I_125	Residencial Rita de Cassia	741.795,77	6.945.723,99
I_126	Solar do Araguaia	741.232,66	6.945.488,92
I_127	Residencial Caminho das Chácaras	742.235,71	6.946.097,11

Tabela 41 – Nome da edificação e coordenadas UTM dos dados pesquisados

ID	Condomínio/Edifício	E (m)	N (m)
I_128	Residencial Ana Lúcia	742.125,01	6.946.238,40
I_129	Residencial Andorra	742.293,87	6.945.886,15
I_130	Velasques	742.108,49	6.945.322,54
I_131	Antônio Souza	742.261,42	6.945.355,28
I_132	Ritz Class	741.473,06	6.945.619,20
I_133	Residencial Spazio	741.022,27	6.945.685,46
I_134	Residencial Galápagos	742.226,09	6.945.150,74
I_135	Belvedere	741.830,13	6.946.251,40
I_136	Residencial Henrique Stodieck	742.486,84	6.946.643,18
I_137	Residencial Erckmann	742.416,07	6.946.213,69
I_138	Edifício Canadá	741.827,41	6.945.240,94
I_139	Marta Joanita	742.300,98	6.946.139,09
I_140	Rafael	742.358,92	6.946.061,29
I_141	Porto Ventura	741.970,17	6.945.873,66
I_142	Stella Moritz	742.070,88	6.945.988,13
I_143	Residencial Capri	742.520,75	6.945.947,47
I_144	Residenziale Francesco di Arcene	742.461,04	6.945.236,98
I_145	Residencial Benta barbato	741.608,68	6.946.089,14
I_146	Residencial Ilha da magia	741.110,91	6.945.804,23
I_147	Residencial Prof Eduardo Luz 1309	742.518,56	6.945.858,35
I_148	Solar do Faial Bloco B	742.464,79	6.945.293,47
I_149	Porto do Sol	741.015,17	6.945.311,49
I_150	Smart Hoepcke Miguel H. Daux	741.840,79	6.945.528,03
I_151	Katepisp	741.472,27	6.946.082,64
I_152	Paul Bernard	741.018,45	6.945.826,07
I_153	Elza Momm	742.434,97	6.945.961,54
I_154	Monte D Arius	741.021,44	6.945.650,91
I_156	Geranium	742.448,41	6.946.183,37
I_157	Smart Hoepcke Miguel H. Daux	741.841,51	6.945.531,33
I_158	Jane	742.519,55	6.945.895,42
I_159	Residencial Barriga Verde	742.402,22	6.945.505,50
I_160	Modigliani	741.564,56	6.945.494,01
I_161	Valparaizo	742.145,90	6.944.906,19
I_162	Topázio	741.984,80	6.944.755,26
I_163	Dallas	741.346,97	6.945.419,18
I_164	Residencial Arthur	741.192,26	6.945.402,64
I_165	Residencial Fortaleza da Barra	741.029,28	6.945.863,10
I_166	Residencial Lauro Muller	742.444,18	6.946.540,85
I_168	Residencial Villas	741.013,88	6.945.796,23

Tabela 41 – Nome da edificação e coordenadas UTM dos dados pesquisados

ID	Condomínio/Edifício	E (m)	N (m)
I_169	Everest Park	741.010,43	6.945.722,69
I_170	Residencial Herminio Jacques	741.323,58	6.945.739,85
I_171	Residencial Rodrik	741.444,95	6.945.774,02
I_172	Residencial Tatiana	741.993,30	6.946.199,76
I_173	Paulo Fontes	742.296,19	6.945.998,44
I_174	Solar das Alamandas	741.676,96	6.945.497,27
I_175	Goya	741.594,67	6.945.321,54
I_176	Residencial Hanna Julia	741.228,76	6.945.772,89
I_177	Praça XV	741.868,64	6.945.124,89
I_178	Ayrton Ramalho	741.185,92	6.945.370,30
I_179	Morada do Norte	741.932,65	6.946.295,44
I_181	Residencial Forte de Sant'ana	741.989,02	6.946.319,56
I_182	Celso Lopes	741.964,97	6.945.922,42
I_183	Residencial Jorge Lacerda	742.115,05	6.946.376,58
I_185	João Eduardo Moritz	742.170,49	6.946.407,83
I_186	Plaza Monet	741.807,17	6.946.242,11
I_187	Florença	741.309,67	6.946.074,14
I_189	Residencial Jardim da Ilha	742.405,45	6.946.301,88
I_190	Rua Germano Wendhausen, 29	742.219,69	6.946.183,19
I_191	Canadá	741.834,50	6.945.236,86
I_192	Rua Djalma Moellman 3	742.604,52	6.946.163,67
I_193	Rua Rafael Bandeira 162	742.303,73	6.946.141,16
I_194	Linda Koerich	742.433,51	6.946.046,71
I_195	Premier Office Center	741.316,15	6.945.562,66
I_196	Solar Dona Martha	741.497,46	6.945.792,76
I_197	Residencial Saint Peter	742.574,31	6.946.368,59
I_198	Onix	741.338,72	6.945.250,59
I_200	Residenziale Francesco di Arcene	742.454,18	6.945.237,21
I_201	Jayme Linhares	741.697,75	6.945.263,55
I_202	Trompowsky Class	741.988,26	6.946.130,06
I_203	Place Residence	740.972,79	6.945.569,22
I_204	DS Prime Residence	741.016,27	6.945.748,28
I_206	Classic Firenze	742.518,10	6.946.015,64
I_207	Simone	741.834,71	6.945.461,78
I_208	Ilhas do Sol	740.763,54	6.945.826,97
I_210	Residencial Forte de Sant'ana	742.046,79	6.946.250,99
I_211	Florença	741.314,33	6.946.075,29
I_212	Residencial Praia de Fora	741.206,80	6.946.039,94
I_213	Jayme Linhares	741.691,10	6.945.268,20

Tabela 41 – Nome da edificação e coordenadas UTM dos dados pesquisados

ID	Condomínio/Edifício	E (m)	N (m)
I_214	José Augusto e Dona Nair	742.529,44	6.945.145,71
I_215	Açores	741.151,58	6.945.607,22
I_216	Fontainebleau	740.871,57	6.945.865,69
I_218	Costa Do Sol	740.837,20	6.945.924,34
I_219	Residencial Forte de Sant'ana	741.999,51	6.946.324,05
I_221	The Palace	741.584,11	6.946.164,74
I_222	Plaza Monet	741.803,98	6.946.240,62
I_223	Residencial Orlando Becker	742.171,32	6.945.976,27
I_224	Residencial Forte de Sant'ana	741.994,22	6.946.321,58
I_225	Novo Rio	741.677,62	6.945.190,52
I_226	Algarve II	741.650,82	6.945.838,43
I_227	Visconde de Ouro Preto	742.025,11	6.945.244,19
I_228	Padre Rohr	741.375,74	6.946.047,22
I_229	Novo Rio	741.673,23	6.945.193,41
I_230	Jayme Linhares	741.705,09	6.945.259,10
I_231	Padre Rohr	741.379,34	6.946.048,11

APÊNDICE C – RESÍDUOS PADRONIZADOS E DISTÂNCIAS DE COOK

Tabela 42 – Métricas para verificação de *outliers* e pontos influenciantes.

Dado	Resíduo Padronizado	Distância de Cook
I 001	-1,3550	0,0127
I 002	-0,9797	0,0038
I 003	-1,2269	0,0298
I 004	0,5518	0,0021
I 007	-2,0263	0,0485
I 009	-1,2961	0,0047
I 010	0,1342	0,0001
I 011	-0,6395	0,0050
I 014	-1,4201	0,0052
I 016	-0,5273	0,0011
I 017	0,1480	0,0002
I 018	-0,9326	0,0038
I 020	0,2658	0,0004
I 021	-0,3922	0,0014
I 022	0,3328	0,0011
I 027	0,0530	0,0000
I 028	-0,0063	0,0000
I 029	0,8663	0,0071
I 031	-2,0453	0,0521
I 032	-0,1996	0,0004
I 033	-1,2673	0,0098
I 034	-0,6053	0,0020
I 035	0,6571	0,0017
I 037	1,7274	0,0041
I 038	-0,1514	0,0000
I 039	-1,1855	0,0120
I 040	-1,0642	0,0060
I 042	1,4022	0,0146
I 043	1,1934	0,0115
I 044	0,2897	0,0006
I 045	0,3061	0,0011
I 046	0,9441	0,0076
I 047	0,0306	0,0000
I 048	1,7211	0,0222
I 050	-0,0377	0,0000
I 051	0,5706	0,0016
I 052	1,0452	0,0047
I 053	-1,4468	0,0283
I 054	-0,4698	0,0022
I 055	0,9098	0,0051
I 056	-1,7409	0,0097
I 057	2,0900	0,0401
I 058	1,2185	0,0061
I 059	0,1461	0,0003
I 060	1,6089	0,0182
I 061	1,1814	0,0056
I 062	-0,2660	0,0012
I 063	-0,9365	0,0147
I 066	1,4221	0,0085
I 067	-0,4398	0,0026
I 068	-0,3327	0,0003
I 069	0,6410	0,0013

Tabela 42 – Métricas para verificação de *outliers* e pontos influenciantes.

Dado	Resíduo Padronizado	Distância de Cook
I 070	-0,1208	0,0001
I 071	0,3960	0,0010
I 073	-0,8795	0,0060
I 074	-0,8740	0,0041
I 075	2,1967	0,0235
I 078	1,2807	0,0114
I 079	-1,9525	0,0154
I 080	0,5679	0,0039
I 081	0,3062	0,0005
I 082	0,4318	0,0024
I 083	1,0582	0,0078
I 084	-0,2885	0,0003
I 085	0,0901	0,0001
I 086	-0,5931	0,0018
I 088	-0,2397	0,0005
I 091	0,0838	0,0000
I 094	-1,1989	0,0069
I 095	0,4995	0,0009
I 098	1,1890	0,0255
I 100	-0,0087	0,0000
I 103	-0,0537	0,0000
I 105	-0,8575	0,0036
I 106	1,6692	0,0131
I 108	0,4624	0,0012
I 109	1,8261	0,0188
I 110	0,5531	0,0036
I 111	-1,8133	0,0176
I 112	1,4799	0,0098
I 113	2,1383	0,0208
I 116	0,3550	0,0008
I 117	0,5064	0,0025
I 118	-0,1696	0,0002
I 119	-0,0999	0,0000
I 120	0,7197	0,0008
I 121	-0,2639	0,0004
I 124	-0,0714	0,0000
I 125	0,5749	0,0013
I 127	-1,4745	0,0159
I 128	-0,8914	0,0046
I 129	0,6098	0,0015
I 132	0,8309	0,0022
I 133	-1,3018	0,0055
I 134	-0,3526	0,0011
I 135	-0,3409	0,0012
I 136	0,9368	0,0069
I 139	0,0390	0,0000
I 140	-0,1897	0,0001
I 141	-0,4666	0,0025
I 143	-0,6012	0,0024
I 144	-0,2074	0,0002
I 145	0,9021	0,0035
I 146	1,7389	0,0107
I 147	-1,5714	0,0096
I 148	-0,7574	0,0013
I 149	1,3588	0,0068

Tabela 42 – Métricas para verificação de *outliers* e pontos influenciantes.

Dado	Resíduo Padronizado	Distância de Cook
I 150	-1,3653	0,0096
I 151	0,0365	0,0000
I 152	-0,7684	0,0024
I 153	-1,1572	0,0071
I 154	-0,2300	0,0004
I 156	1,4135	0,0147
I 161	-0,9676	0,0069
I 162	-0,0978	0,0001
I 163	-0,5439	0,0031
I 164	-0,2705	0,0006
I 165	-1,7481	0,0185
I 166	-0,9180	0,0068
I 168	-0,8209	0,0117
I 169	0,1347	0,0001
I 172	-0,0504	0,0000
I 175	0,8762	0,0037
I 177	0,0812	0,0000
I 178	-1,2913	0,0084
I 179	0,9879	0,0162
I 181	0,4638	0,0025
I 185	0,4429	0,0024
I 186	-0,4498	0,0017
I 187	1,3699	0,0185
I 189	0,1501	0,0001
I 190	0,7270	0,0057
I 191	-0,1694	0,0002
I 192	-0,4389	0,0026
I 194	1,6589	0,0305
I 198	0,7567	0,0043
I 200	-0,9721	0,0045
I 201	-1,1841	0,0116
I 202	0,6669	0,0030
I 203	0,3260	0,0010
I 204	-1,1111	0,0148
I 206	0,1212	0,0001
I 207	1,3168	0,0086
I 208	-0,0593	0,0000
I 210	-1,6562	0,0305
I 213	-0,7434	0,0031
I 215	-0,2660	0,0004
I 216	-1,8625	0,0563
I 218	1,8535	0,0645
I 221	0,1826	0,0003
I 224	0,4181	0,0017
I 227	0,0536	0,0000
I 228	-0,7095	0,0029
I 229	0,0648	0,0000
I 230	0,3882	0,0008
I 231	-0,2668	0,0005

APÊNDICE D – R² DAS RNA COM TRANSFORMAÇÃO

Tabela 43 – Coeficientes de determinação variando-se a quantidade de N_n para cada cenário

N_n	Cenário								
	1HLES	1HLPR	1HL	2HLES	2HLPR	2HL	3HLES	3HLPR	3HL
5	95,06%	85,07%	94,11%	94,41%	86,17%	95,44%	89,39%	88,26%	91,70%
6	94,99%	82,35%	94,43%	94,86%	83,90%	94,51%	91,55%	0,00%	92,44%
7	93,97%	84,93%	91,62%	94,67%	82,51%	95,23%	94,11%	0,00%	95,06%
8	95,15%	88,36%	94,12%	94,35%	84,74%	95,31%	94,33%	88,45%	95,19%
9	94,89%	91,05%	92,13%	94,56%	84,51%	95,66%	94,06%	0,00%	95,22%
10	95,84%	86,80%	92,95%	94,68%	87,44%	93,82%	93,27%	57,14%	93,16%
11	94,53%	93,89%	92,48%	94,33%	86,96%	92,36%	93,02%	77,10%	91,93%
12	93,22%	91,96%	89,99%	94,71%	57,54%	91,49%	94,70%	84,74%	91,56%
13	95,41%	92,18%	93,16%	91,60%	84,00%	86,09%	94,97%	81,22%	94,50%
14	94,80%	89,52%	92,86%	94,59%	88,77%	93,42%	94,93%	93,85%	90,30%
15	95,08%	95,41%	91,59%	94,62%	85,96%	92,22%	95,14%	57,02%	90,28%
16	94,78%	93,90%	88,36%	93,35%	94,72%	90,83%	95,54%	84,02%	92,63%
17	95,34%	86,71%	91,90%	95,48%	92,32%	89,65%	93,99%	90,28%	89,78%
18	95,65%	89,71%	91,11%	94,54%	93,22%	81,59%	95,46%	91,52%	95,30%
19	95,58%	94,99%	91,76%	93,39%	93,87%	85,85%	94,93%	88,61%	81,45%
20	95,09%	91,33%	90,45%	94,50%	89,16%	90,47%	93,59%	92,09%	92,89%
21	95,20%	93,32%	91,55%	93,62%	88,84%	88,97%	94,55%	83,99%	90,27%
22	95,27%	94,42%	90,84%	94,25%	94,86%	88,52%	93,35%	92,19%	85,13%
23	94,62%	94,01%	92,86%	95,41%	95,82%	90,52%	93,94%	94,38%	90,49%
24	94,68%	92,10%	90,02%	94,53%	94,66%	86,35%	93,42%	92,29%	91,15%
25	95,06%	95,24%	92,18%	95,75%	91,96%	83,35%	94,59%	94,41%	81,34%
26	95,41%	94,53%	90,59%	95,63%	95,26%	83,44%	94,60%	94,51%	80,14%
27	96,16%	93,82%	89,97%	95,00%	94,59%	88,62%	94,73%	94,90%	86,37%
28	95,10%	93,71%	90,10%	94,75%	92,26%	80,34%	95,96%	92,59%	82,45%
29	95,60%	92,34%	92,68%	93,96%	93,98%	78,68%	94,27%	95,00%	80,21%
30	95,20%	94,73%	91,43%	94,69%	94,81%	78,11%	95,24%	95,02%	87,03%
31	94,70%	96,24%	92,04%	95,37%	94,77%	86,07%	92,37%	95,30%	81,69%
32	95,03%	93,50%	89,90%	95,52%	95,26%	84,43%	92,37%	95,80%	78,70%
33	96,16%	95,37%	92,55%	95,24%	95,33%	86,03%	93,39%	95,76%	87,40%
34	94,79%	94,79%	92,00%	95,33%	94,53%	85,16%	94,24%	94,55%	81,06%
35	95,04%	94,78%	91,94%	95,97%	95,50%	90,72%	92,24%	95,35%	83,60%
36	95,34%	93,81%	90,02%	94,98%	95,18%	83,31%	93,77%	95,04%	85,18%
37	96,70%	95,78%	93,25%	95,16%	95,07%	80,18%	95,75%	94,34%	79,54%
38	94,57%	94,54%	91,38%	95,11%	94,04%	84,49%	95,55%	95,55%	83,50%
39	96,50%	95,44%	91,17%	95,46%	94,47%	86,83%	96,19%	95,61%	83,65%
40	95,21%	95,51%	92,20%	95,40%	93,85%	83,69%	95,62%	95,45%	78,11%
41	94,83%	95,30%	90,41%	95,09%	94,74%	86,53%	94,14%	93,45%	80,20%
42	95,21%	94,46%	92,17%	95,03%	95,51%	81,20%	95,35%	95,24%	83,70%
43	95,05%	95,29%	92,22%	95,73%	95,43%	83,29%	94,74%	95,09%	81,31%
44	94,82%	95,30%	91,92%	95,14%	95,42%	86,03%	94,02%	95,45%	82,25%
45	94,87%	94,60%	93,54%	95,49%	95,21%	85,13%	95,05%	94,72%	83,45%

Tabela 43 – Coeficientes de determinação variando-se a quantidade de N_n para cada cenário

N_n	Cenário								
	1HLES	1HLPR	1HL	2HLES	2HLPR	2HL	3HLES	3HLPR	3HL
46	95,40%	95,20%	92,08%	96,10%	95,56%	81,39%	94,84%	95,08%	83,57%
47	94,95%	95,52%	93,07%	94,61%	95,53%	80,82%	94,45%	95,41%	85,82%
48	95,12%	94,80%	90,81%	95,07%	95,11%	80,51%	94,46%	95,58%	82,59%
49	94,58%	94,74%	91,71%	94,31%	95,14%	70,50%	94,99%	95,27%	82,14%
50	95,24%	95,56%	91,92%	94,69%	95,30%	86,29%	94,53%	95,23%	71,08%
51	95,10%	95,51%	91,21%	94,71%	95,18%	82,11%	94,63%	94,59%	87,90%
52	95,04%	95,43%	90,61%	95,28%	94,99%	86,32%	95,75%	95,05%	75,49%
53	95,18%	95,42%	91,78%	95,38%	95,37%	87,81%	94,46%	95,23%	85,00%
54	95,09%	94,69%	92,54%	95,11%	95,60%	85,28%	95,78%	95,59%	86,59%
55	95,60%	94,77%	92,83%	94,94%	94,33%	82,14%	95,30%	94,84%	80,58%
56	95,29%	94,65%	92,46%	95,53%	95,34%	78,38%	93,42%	95,40%	80,14%
57	95,10%	95,39%	92,22%	94,92%	95,42%	82,87%	95,13%	95,20%	73,93%
58	95,50%	95,15%	92,24%	95,26%	95,35%	82,92%	94,96%	95,06%	78,60%
59	95,00%	95,23%	92,28%	94,09%	95,07%	76,13%	95,18%	94,76%	82,07%
60	95,04%	95,31%	93,11%	95,03%	95,41%	82,39%	95,22%	94,67%	84,10%
61	95,50%	95,53%	92,15%	94,67%	95,36%	75,42%	94,97%	95,30%	66,20%
62	95,96%	95,15%	90,94%	95,40%	95,40%	80,69%	95,71%	95,35%	85,41%
63	95,30%	95,39%	93,37%	94,42%	95,25%	74,79%	94,88%	95,33%	88,01%
64	94,89%	95,29%	91,94%	95,19%	95,19%	79,28%	94,85%	95,27%	83,57%
65	94,78%	95,20%	92,19%	94,85%	95,16%	84,54%	95,27%	95,25%	81,99%
66	95,08%	94,73%	91,74%	94,98%	95,46%	78,31%	95,88%	95,07%	82,16%
67	95,73%	95,22%	92,90%	95,72%	95,32%	82,23%	95,00%	95,38%	74,18%
68	95,48%	95,43%	93,67%	95,20%	95,30%	87,99%	95,18%	95,29%	86,93%
69	95,22%	95,23%	92,77%	95,45%	95,31%	75,42%	94,93%	95,46%	85,82%
70	95,34%	95,48%	91,17%	93,92%	95,32%	79,70%	95,60%	95,45%	83,16%
71	95,35%	95,47%	91,86%	95,45%	95,32%	81,32%	95,08%	95,50%	87,78%
72	95,21%	95,41%	92,47%	95,26%	95,35%	83,81%	94,73%	95,36%	84,03%
73	95,15%	95,38%	92,18%	95,69%	95,46%	78,68%	94,81%	95,76%	83,61%
74	95,36%	95,44%	91,88%	96,16%	95,41%	73,19%	95,92%	95,58%	81,81%
75	95,46%	94,80%	92,90%	93,83%	95,31%	76,02%	95,19%	95,25%	80,14%
76	95,16%	95,18%	92,95%	95,13%	95,49%	81,29%	95,23%	95,35%	78,91%
77	95,28%	95,47%	91,86%	94,73%	95,40%	77,66%	95,66%	95,39%	70,56%
78	95,20%	95,08%	91,42%	94,64%	95,41%	78,67%	95,15%	95,38%	79,25%
79	95,51%	95,36%	92,62%	95,34%	95,58%	86,11%	95,15%	95,44%	86,72%
80	95,52%	95,43%	92,97%	95,54%	95,50%	74,94%	95,12%	95,44%	85,61%
81	95,16%	95,44%	92,69%	95,41%	95,43%	70,37%	95,15%	95,66%	83,61%
82	95,40%	95,27%	92,98%	94,97%	95,48%	78,00%	94,82%	95,54%	74,91%
83	95,44%	95,40%	92,10%	95,76%	95,41%	80,65%	94,72%	95,39%	77,54%
84	95,36%	95,37%	92,33%	95,65%	95,35%	83,62%	95,01%	95,59%	81,45%
85	95,19%	95,22%	92,60%	95,10%	95,16%	75,43%	94,56%	95,36%	82,55%
86	95,38%	95,31%	92,01%	95,38%	95,32%	86,38%	94,73%	95,24%	82,41%
87	95,48%	95,39%	90,83%	95,05%	95,32%	80,83%	94,98%	95,19%	83,19%
88	95,31%	95,36%	93,35%	95,26%	95,29%	84,31%	95,21%	95,37%	87,77%
89	95,28%	95,41%	92,66%	95,45%	95,35%	81,63%	94,63%	95,25%	81,25%

Tabela 43 – Coeficientes de determinação variando-se a quantidade de N_n para cada cenário

N_n	Cenário								
	1HLES	1HLPR	1HL	2HLES	2HLPR	2HL	3HLES	3HLPR	3HL
90	95,47%	95,46%	92,99%	96,26%	95,28%	80,81%	95,20%	95,16%	81,13%
91	95,00%	95,39%	92,29%	95,05%	95,28%	81,95%	94,08%	95,27%	83,50%
92	95,25%	95,48%	92,97%	95,36%	95,54%	87,89%	94,35%	95,26%	69,02%
93	95,35%	95,39%	92,44%	95,33%	95,34%	83,00%	94,92%	95,24%	84,86%
94	95,35%	95,22%	92,34%	95,24%	95,55%	83,00%	94,74%	95,38%	88,39%
95	95,06%	95,43%	92,68%	95,93%	95,58%	83,16%	94,63%	95,20%	83,34%
96	95,31%	95,43%	92,67%	95,38%	95,56%	80,36%	94,89%	95,29%	87,27%
97	95,69%	95,40%	93,22%	95,51%	95,45%	82,26%	94,66%	95,64%	88,91%
98	95,04%	94,83%	92,61%	95,24%	95,43%	78,77%	95,50%	95,67%	84,04%
99	95,40%	95,39%	92,32%	95,38%	95,27%	82,07%	94,65%	95,50%	86,48%
100	95,31%	95,44%	92,01%	95,32%	95,21%	82,81%	95,23%	95,19%	83,40%
101	95,44%	95,37%	92,25%	95,40%	95,36%	81,01%	95,73%	95,40%	85,27%
102	95,14%	95,39%	91,90%	95,06%	95,46%	83,72%	95,21%	95,33%	79,94%
103	95,35%	95,41%	92,85%	95,42%	95,57%	82,56%	94,64%	95,47%	83,97%
104	95,36%	95,43%	93,13%	95,19%	95,45%	83,02%	95,30%	95,45%	82,62%
105	95,64%	95,58%	92,19%	94,93%	95,40%	78,08%	95,60%	95,34%	80,74%
106	95,19%	95,38%	92,06%	94,67%	95,46%	82,60%	94,92%	95,37%	88,61%
107	95,36%	95,38%	92,79%	95,08%	95,52%	87,70%	96,43%	95,36%	88,24%
108	94,25%	95,43%	92,87%	95,10%	95,30%	79,36%	95,62%	95,32%	85,77%
109	95,39%	95,31%	92,97%	94,19%	95,27%	81,92%	95,10%	95,49%	90,37%
110	95,33%	95,41%	92,19%	95,20%	95,47%	82,89%	95,84%	95,44%	80,36%
111	94,85%	95,43%	92,72%	95,07%	95,47%	82,96%	95,26%	95,29%	86,59%
112	95,39%	95,33%	91,64%	95,85%	95,19%	88,87%	95,54%	95,26%	84,86%
113	95,48%	95,42%	92,46%	95,15%	95,44%	77,69%	94,82%	95,52%	82,87%
114	96,41%	95,41%	92,55%	94,98%	95,44%	87,38%	95,16%	95,48%	83,23%
115	94,51%	95,39%	92,30%	95,01%	95,40%	80,89%	95,49%	95,43%	82,94%
116	95,91%	95,45%	92,78%	95,40%	95,48%	83,97%	95,15%	95,46%	83,61%
117	95,36%	95,39%	92,64%	94,78%	95,33%	79,00%	95,22%	95,43%	84,51%
118	95,48%	95,45%	92,64%	95,06%	95,36%	78,96%	95,55%	95,45%	83,41%
119	95,35%	95,22%	92,86%	95,43%	95,45%	77,65%	96,09%	95,47%	82,87%
120	95,30%	95,36%	93,51%	95,23%	95,35%	85,49%	95,44%	95,41%	85,12%
121	94,30%	95,38%	92,73%	95,47%	95,39%	79,31%	95,14%	95,35%	80,92%
122	95,87%	95,43%	91,72%	95,56%	95,44%	83,82%	94,70%	95,41%	87,83%
123	95,07%	95,39%	93,00%	95,15%	95,49%	78,07%	94,80%	95,38%	82,88%
124	95,10%	95,43%	92,55%	95,00%	95,44%	79,06%	94,75%	95,41%	85,13%
125	95,46%	95,37%	92,56%	95,83%	95,34%	80,33%	95,62%	95,48%	84,04%
126	95,07%	95,37%	92,65%	94,78%	95,30%	76,64%	95,32%	95,46%	80,12%
127	95,19%	95,39%	92,64%	95,34%	95,46%	86,37%	94,60%	95,33%	79,84%
128	95,17%	95,42%	92,79%	95,20%	95,44%	81,61%	94,98%	95,29%	87,14%
129	95,52%	95,44%	92,83%	95,85%	95,43%	87,81%	94,62%	95,09%	87,47%
130	95,23%	95,42%	93,44%	95,08%	95,39%	82,30%	94,67%	95,25%	85,16%
131	96,30%	95,35%	93,57%	95,36%	95,33%	83,19%	95,34%	95,39%	83,91%
132	95,32%	95,43%	93,42%	94,98%	95,24%	84,62%	94,84%	95,42%	86,31%
133	95,28%	95,42%	93,25%	95,08%	95,47%	85,32%	95,35%	95,42%	90,24%

Tabela 43 – Coeficientes de determinação variando-se a quantidade de N_n para cada cenário

N_n	Cenário								
	1HLES	1HLPR	1HL	2HLES	2HLPR	2HL	3HLES	3HLPR	3HL
134	95,38%	95,40%	93,08%	95,17%	95,49%	87,48%	95,40%	95,38%	83,93%
135	95,50%	95,38%	92,33%	95,47%	95,37%	81,51%	95,71%	95,47%	84,53%
136	95,20%	95,52%	93,80%	95,32%	95,46%	84,29%	95,35%	95,43%	89,78%
137	96,00%	95,37%	92,53%	95,53%	95,62%	86,82%	95,42%	95,40%	85,16%
138	95,97%	95,34%	92,57%	95,45%	95,43%	86,12%	95,41%	95,44%	86,67%
139	95,38%	95,40%	93,15%	96,15%	95,43%	85,67%	95,11%	95,49%	88,74%
140	95,30%	95,40%	93,87%	95,18%	95,44%	80,06%	95,29%	95,51%	87,16%
141	94,77%	95,42%	93,08%	95,35%	95,42%	83,67%	95,06%	95,43%	87,89%
142	95,15%	95,32%	93,03%	94,74%	95,34%	85,63%	95,05%	95,49%	84,92%
143	95,16%	95,40%	91,37%	95,61%	95,56%	80,24%	95,25%	95,64%	87,57%
144	95,38%	95,37%	92,62%	95,54%	95,43%	87,21%	95,08%	95,61%	86,61%
145	95,15%	95,45%	92,87%	95,59%	95,43%	74,72%	95,08%	95,29%	88,25%
146	95,76%	95,35%	93,82%	95,17%	95,44%	86,63%	95,38%	95,41%	85,72%
147	95,31%	95,35%	93,20%	95,33%	95,45%	84,85%	95,39%	95,36%	79,45%
148	95,20%	95,41%	92,51%	95,51%	95,40%	81,81%	95,61%	95,57%	88,33%
149	95,12%	95,33%	92,86%	95,38%	95,37%	79,93%	95,00%	95,27%	88,90%
150	95,37%	95,39%	93,04%	94,90%	95,42%	81,30%	95,10%	95,42%	85,49%
151	95,24%	95,40%	93,61%	95,21%	95,38%	80,47%	95,38%	95,42%	87,55%
152	95,84%	95,35%	92,42%	95,38%	95,44%	80,30%	94,97%	95,46%	88,88%
153	95,24%	95,35%	92,67%	94,64%	95,37%	80,76%	95,18%	95,40%	86,42%
154	95,40%	95,37%	92,54%	95,33%	95,40%	81,41%	95,03%	95,48%	83,03%
155	95,34%	95,36%	93,30%	95,09%	95,46%	76,46%	95,37%	95,65%	87,31%
156	95,23%	95,40%	92,08%	94,32%	95,44%	83,56%	95,14%	95,56%	90,00%
157	95,04%	95,39%	93,16%	95,51%	95,43%	67,89%	94,77%	95,66%	83,54%
158	95,38%	95,40%	93,27%	95,51%	95,46%	84,19%	94,79%	94,99%	87,35%
159	95,26%	95,37%	91,75%	95,13%	95,46%	78,80%	95,18%	95,59%	82,96%
160	95,39%	95,44%	93,52%	95,24%	95,34%	83,72%	95,22%	95,49%	81,49%
161	95,56%	95,38%	93,45%	95,27%	95,41%	77,36%	95,15%	95,43%	89,61%
162	95,44%	95,43%	92,09%	95,02%	95,51%	82,56%	94,88%	95,48%	89,24%
163	95,12%	95,36%	93,48%	94,76%	95,51%	83,17%	95,13%	95,48%	81,08%
164	95,30%	95,40%	92,99%	95,10%	95,39%	79,69%	95,24%	95,42%	82,36%
165	95,23%	95,36%	92,70%	95,11%	95,35%	86,12%	95,27%	95,40%	87,97%
166	95,18%	95,35%	93,22%	95,38%	95,41%	84,84%	95,57%	95,20%	85,01%
167	95,48%	95,41%	92,97%	95,46%	95,39%	78,84%	94,90%	95,00%	89,10%
168	95,38%	95,40%	92,61%	95,27%	95,45%	85,05%	95,00%	95,10%	88,75%
169	95,27%	95,41%	92,43%	95,43%	95,44%	82,17%	95,38%	95,52%	85,19%
170	95,37%	95,37%	92,43%	95,18%	95,48%	78,46%	95,06%	95,35%	86,20%
171	95,44%	95,43%	93,70%	95,28%	95,54%	81,99%	94,98%	95,46%	85,38%
172	95,19%	95,36%	92,71%	95,42%	95,38%	86,13%	95,40%	95,47%	87,00%
173	95,45%	95,37%	92,55%	95,54%	95,48%	86,13%	95,20%	95,44%	87,21%
174	95,85%	95,36%	92,73%	95,40%	95,48%	88,41%	95,46%	95,52%	87,03%
175	95,20%	95,36%	92,94%	94,95%	95,38%	85,33%	95,01%	95,59%	89,04%
176	95,39%	95,35%	93,02%	95,37%	95,48%	83,36%	94,87%	95,47%	86,90%
177	95,76%	95,39%	92,65%	95,64%	95,40%	83,23%	95,22%	95,50%	80,15%

Tabela 43 – Coeficientes de determinação variando-se a quantidade de N_n para cada cenário

N_n	Cenário								
	1HLES	1HLPR	1HL	2HLES	2HLPR	2HL	3HLES	3HLPR	3HL
178	95,13%	95,41%	92,44%	95,45%	95,41%	85,92%	95,14%	95,36%	84,43%
179	95,39%	95,39%	93,02%	95,51%	95,48%	84,91%	95,15%	95,46%	89,59%
180	95,25%	95,37%	92,54%	95,49%	95,46%	75,19%	95,12%	95,48%	88,44%
181	95,30%	95,42%	93,20%	95,13%	95,48%	86,94%	95,62%	95,51%	83,06%
182	95,40%	95,38%	92,82%	95,17%	95,37%	86,12%	95,51%	95,59%	90,05%
183	95,36%	95,39%	92,90%	95,02%	95,48%	85,75%	95,49%	95,52%	89,53%
184	94,93%	95,43%	93,47%	95,15%	95,43%	83,15%	95,12%	95,42%	84,74%
185	95,38%	95,35%	92,76%	95,20%	95,48%	79,59%	95,34%	95,32%	85,58%
186	95,84%	95,36%	93,29%	95,14%	95,43%	77,47%	95,46%	95,40%	85,74%
187	95,20%	95,39%	92,89%	95,35%	95,50%	83,35%	95,31%	95,42%	87,93%
188	95,38%	95,38%	93,56%	95,33%	95,48%	84,39%	94,96%	95,63%	86,88%
189	95,36%	95,44%	93,05%	95,08%	95,45%	82,14%	95,28%	95,51%	87,65%
190	95,34%	95,40%	93,08%	95,36%	95,37%	85,05%	95,04%	95,44%	86,67%
191	95,43%	95,37%	92,64%	95,77%	95,42%	76,34%	95,37%	95,48%	86,00%
192	95,37%	95,44%	93,31%	95,24%	95,23%	88,28%	95,52%	95,53%	90,39%
193	95,21%	95,43%	92,46%	95,23%	95,49%	81,51%	94,91%	95,50%	87,28%
194	95,31%	95,37%	93,26%	95,59%	95,46%	88,75%	95,37%	95,41%	87,61%
195	95,45%	95,42%	93,07%	95,45%	95,39%	78,46%	95,17%	95,48%	82,35%
196	95,41%	95,37%	92,99%	95,55%	95,46%	82,36%	94,93%	95,37%	77,04%
197	94,54%	95,38%	93,21%	95,22%	95,39%	84,78%	94,96%	95,38%	87,21%
198	95,41%	95,38%	92,41%	95,04%	95,31%	87,14%	95,35%	95,32%	89,74%
199	95,42%	95,42%	92,92%	95,36%	95,42%	86,99%	95,03%	95,51%	87,87%
200	95,51%	95,37%	92,78%	95,23%	95,38%	86,31%	94,87%	95,58%	85,52%
201	95,25%	95,37%	93,06%	95,07%	95,33%	84,12%	95,29%	95,41%	88,14%
202	95,30%	95,38%	93,36%	95,13%	95,48%	87,04%	95,37%	95,50%	89,31%
203	94,99%	95,40%	93,53%	94,99%	95,43%	84,76%	95,28%	95,52%	86,36%
204	95,30%	95,43%	93,70%	95,34%	95,44%	81,16%	95,43%	95,64%	86,17%

APÊNDICE E – RMSE DAS RNA COM TRANSFORMAÇÃO

Tabela 44 – RMSE (R\$) variando-se a quantidade de N_n para cada cenário

N_n	RMSE - Raiz do erro quadrático médio (R\$)								
	1HLES	1HLPR	1HL	2HLES	2HLPR	2HL	3HLES	3HLPR	3HL
5	390.080	759.282	463.642	438.291	739.923	366.433	700.296	672.664	498.628
6	406.793	784.393	368.905	410.173	754.773	415.083	607.311	1.486.262	482.986
7	478.819	725.896	452.815	460.833	773.251	402.814	449.897	1.486.257	405.508
8	379.458	615.554	380.266	519.706	781.646	389.110	450.995	652.739	429.103
9	398.595	599.624	521.138	462.257	783.816	382.953	452.488	1.486.221	435.223
10	364.842	719.016	628.223	420.639	651.406	470.149	502.446	959.730	446.166
11	465.426	480.055	549.903	434.458	714.149	534.456	497.945	780.831	463.409
12	473.535	597.752	542.085	393.324	954.738	428.809	453.605	779.633	503.856
13	444.394	578.643	473.670	513.763	752.341	698.299	407.684	829.946	459.907
14	422.403	587.745	454.271	440.062	642.658	451.010	415.778	509.198	518.473
15	425.914	377.433	538.888	479.809	665.460	609.475	402.218	969.524	504.594
16	447.139	520.010	671.400	465.062	412.570	519.032	371.115	751.573	429.832
17	422.683	603.355	563.403	395.441	541.577	584.168	509.487	651.308	600.405
18	376.585	625.669	487.620	422.196	463.745	1.476.490	449.807	634.217	422.589
19	365.265	418.215	456.145	469.231	456.619	750.799	428.560	651.313	914.600
20	380.807	564.090	500.144	443.439	676.469	587.560	463.083	500.880	478.604
21	407.427	443.136	494.668	469.046	624.019	925.966	417.983	754.224	491.149
22	398.296	446.017	667.835	486.445	490.431	572.587	462.393	586.243	691.733
23	433.696	440.634	500.794	436.944	405.058	527.576	451.438	466.391	619.153
24	423.654	572.671	729.388	436.442	396.387	744.122	491.053	579.421	580.830
25	429.461	452.635	712.673	393.296	597.604	789.605	428.628	428.697	1.098.317
26	405.931	416.827	554.250	433.116	464.148	716.765	434.972	445.810	868.508
27	394.640	451.054	692.833	424.864	423.207	535.662	402.359	390.320	789.721
28	389.098	429.193	585.106	419.067	574.180	931.148	399.716	540.236	972.727
29	366.213	470.511	464.442	439.674	444.243	1.018.201	464.685	425.020	833.856
30	420.146	400.693	656.847	437.907	414.755	770.993	436.739	413.336	706.417
31	462.810	386.777	541.948	427.105	399.336	690.310	533.490	463.201	706.528
32	400.164	431.153	647.106	420.615	403.929	784.294	522.940	429.589	1.058.771
33	366.001	378.882	446.174	438.734	395.720	635.421	475.120	438.247	563.506
34	412.986	420.393	562.963	410.952	443.304	791.378	475.884	450.306	748.098
35	418.768	370.692	566.148	376.767	390.222	492.542	514.361	417.052	1.102.533
36	405.328	433.107	851.234	409.018	406.459	662.416	493.040	389.052	754.612
37	356.137	435.288	468.103	396.504	406.691	778.758	373.638	473.500	576.295
38	410.568	427.673	550.224	409.025	445.367	628.504	437.952	404.608	706.648
39	344.061	386.911	643.459	429.262	430.310	627.930	396.238	385.240	894.724
40	405.733	385.029	446.504	403.492	430.102	596.170	382.826	380.821	662.160
41	422.257	390.200	601.606	405.837	417.343	798.910	476.867	453.275	734.732
42	407.592	433.701	521.376	401.669	388.308	1.108.556	412.192	412.031	769.384
43	405.976	386.567	577.145	391.194	406.395	758.263	428.843	414.295	881.500
44	413.583	405.566	574.398	433.545	391.373	716.347	474.715	400.987	862.507
45	401.357	414.117	435.814	388.740	406.629	707.581	410.513	401.750	693.254

Tabela 44 – RMSE (R\$) variando-se a quantidade de N_n para cada cenário

N_n	RMSE - Raiz do erro quadrático médio (R\$)								
	1HLES	1HLPR	1HL	2HLES	2HLPR	2HL	3HLES	3HLPR	3HL
46	374.048	389.050	516.130	329.338	375.070	730.555	438.840	392.224	969.803
47	401.823	385.742	541.448	430.007	381.504	711.189	463.520	406.777	837.691
48	392.464	407.319	541.328	403.811	409.996	735.603	456.183	371.140	671.117
49	424.937	410.906	548.096	476.051	402.436	982.615	407.864	413.629	812.373
50	406.134	385.604	569.930	431.527	397.034	562.537	439.455	399.610	1.096.319
51	391.673	383.974	696.406	420.244	415.327	687.283	431.138	430.050	640.695
52	402.260	382.096	630.566	395.940	420.431	795.058	395.557	397.068	796.511
53	396.820	377.241	579.367	405.761	393.082	511.942	450.329	402.426	685.689
54	404.752	423.254	631.037	406.107	378.548	761.644	340.649	381.103	630.945
55	384.323	414.013	456.887	424.341	430.871	607.970	375.134	412.289	958.373
56	385.068	402.707	609.662	400.842	397.093	634.127	466.341	401.137	701.442
57	400.682	390.765	551.696	423.900	393.131	614.221	372.944	396.777	842.033
58	382.004	400.771	604.766	394.481	395.841	867.466	434.917	403.291	692.616
59	401.556	396.427	526.507	464.698	414.302	674.400	378.960	406.736	679.166
60	404.169	397.546	433.770	413.011	399.601	694.238	399.332	403.637	709.886
61	371.654	386.719	568.160	429.089	398.602	927.525	399.375	398.999	825.623
62	376.754	403.958	751.313	386.566	393.151	1.073.642	421.786	390.751	551.611
63	399.249	385.936	504.473	423.438	417.340	788.530	416.885	399.692	505.501
64	408.036	390.396	584.500	392.073	401.349	711.938	426.715	398.513	711.655
65	423.988	400.694	579.943	415.368	408.233	601.066	430.822	411.066	717.807
66	394.901	405.434	559.026	435.152	392.644	908.103	397.374	405.777	818.006
67	379.076	397.353	542.652	405.742	410.045	708.815	403.291	392.649	760.642
68	383.014	386.832	497.466	397.415	390.282	497.515	390.967	402.626	644.365
69	399.862	399.433	543.192	416.942	400.318	729.131	387.757	388.453	702.133
70	378.501	375.008	643.110	448.016	398.269	795.150	389.606	388.900	751.642
71	379.165	389.595	563.903	392.595	399.188	866.086	414.519	393.366	597.573
72	406.356	390.324	568.129	414.413	406.674	709.079	425.578	402.116	631.860
73	395.669	386.939	544.680	387.721	386.657	722.697	397.837	374.835	639.152
74	387.254	381.273	592.196	359.096	389.298	827.577	361.835	379.850	918.595
75	376.994	410.020	504.778	431.475	398.898	845.510	403.220	394.764	689.192
76	389.446	396.504	457.766	401.996	393.577	790.125	422.437	402.189	796.658
77	391.029	380.419	616.587	424.591	396.009	1.072.964	407.083	396.324	855.204
78	393.803	402.551	649.146	439.014	385.428	872.505	424.583	412.630	783.524
79	404.473	394.053	554.729	398.132	385.747	734.785	425.946	402.741	576.780
80	381.770	385.311	550.092	384.727	383.978	1.243.983	408.774	405.179	626.433
81	402.594	378.194	537.159	403.665	394.811	858.963	410.380	380.177	657.501
82	394.749	395.945	543.174	406.785	389.323	766.740	417.257	389.146	867.568
83	388.018	384.422	519.027	380.362	394.085	799.728	415.228	397.874	723.795
84	391.062	388.641	555.976	409.475	397.338	797.380	425.965	380.654	772.488
85	399.252	389.593	676.268	430.176	411.570	789.580	426.276	393.609	721.020
86	388.714	389.782	560.230	405.122	394.400	607.842	423.407	405.361	676.647
87	381.123	389.470	669.078	403.355	407.496	798.754	412.792	398.238	769.080
88	380.714	396.430	530.906	414.117	404.775	690.445	396.689	399.137	736.270
89	392.781	388.071	537.967	390.456	390.973	756.744	429.460	412.816	733.157

Tabela 44 – RMSE (R\$) variando-se a quantidade de N_n para cada cenário

N_n	RMSE - Raiz do erro quadrático médio (R\$)								
	1HLES	1HLPR	1HL	2HLES	2HLPR	2HL	3HLES	3HLPR	3HL
90	383.040	382.023	498.068	368.202	395.622	1.492.938	425.848	418.218	731.761
91	407.970	385.898	549.066	398.974	398.249	647.403	463.211	405.793	700.452
92	393.134	380.605	565.156	385.591	381.555	542.169	436.510	403.276	1.564.460
93	383.940	385.036	723.088	401.131	392.370	997.852	434.324	396.550	717.192
94	385.906	389.599	552.213	413.228	384.800	791.204	442.533	407.627	590.408
95	411.567	392.090	593.517	395.334	379.357	622.189	448.550	418.166	660.055
96	386.629	387.061	580.138	391.993	380.294	968.833	432.259	413.099	622.455
97	373.353	377.899	450.412	378.760	383.795	780.311	418.219	381.877	650.492
98	409.240	404.832	558.537	398.576	392.055	722.742	389.517	368.151	598.352
99	380.228	385.491	588.453	399.885	404.450	866.822	424.665	384.733	696.366
100	393.137	382.535	524.011	415.628	401.876	838.772	420.909	401.210	700.030
101	388.235	386.297	555.817	398.401	390.795	692.911	404.898	399.936	534.774
102	394.263	383.729	533.122	398.064	397.648	799.659	389.896	409.076	762.065
103	388.349	384.258	538.720	403.005	389.988	745.889	468.552	386.351	632.744
104	394.811	387.635	489.041	421.003	385.966	752.571	391.785	387.308	614.398
105	381.314	373.106	547.235	400.480	396.178	925.960	398.041	397.676	697.022
106	388.293	382.378	662.586	429.627	393.943	780.959	435.554	400.234	664.256
107	382.453	385.457	544.666	397.189	385.670	581.183	347.984	407.316	523.815
108	436.034	379.691	553.907	399.832	399.475	828.025	399.423	405.267	669.126
109	387.113	391.759	549.158	419.094	399.327	780.125	406.648	388.518	474.452
110	381.414	383.300	543.264	394.764	374.840	681.315	400.315	392.290	792.694
111	409.102	390.961	505.954	391.511	389.946	783.516	389.996	397.157	748.702
112	390.707	391.211	679.438	338.949	402.124	542.795	390.909	399.787	564.642
113	380.003	382.163	635.188	386.106	391.491	1.102.469	430.311	384.679	631.304
114	364.754	384.190	630.474	405.139	391.162	653.891	402.891	385.874	722.989
115	420.288	386.170	560.220	406.665	388.532	812.612	386.105	396.994	725.644
116	382.786	380.144	547.959	392.765	395.161	603.499	389.470	392.868	791.594
117	398.618	386.013	608.247	392.421	394.485	724.437	412.631	399.794	551.953
118	376.801	387.159	718.335	400.466	389.790	1.237.096	397.744	395.892	763.597
119	394.577	394.245	543.793	398.188	393.960	1.043.907	346.649	388.729	637.293
120	390.356	388.513	464.469	395.815	386.906	708.971	389.901	393.871	643.218
121	421.630	382.687	574.374	363.759	388.094	912.079	417.484	380.842	630.694
122	371.378	382.629	596.307	376.892	391.863	725.708	424.206	389.669	590.456
123	406.326	385.615	502.538	415.410	394.174	857.183	446.838	398.068	791.509
124	401.735	382.346	517.145	417.598	393.334	835.333	422.619	391.342	637.727
125	387.331	386.957	518.127	380.732	397.221	1.034.694	401.268	391.081	759.807
126	403.138	388.211	546.095	416.333	398.172	1.001.850	409.392	387.567	1.274.504
127	386.478	385.350	647.371	403.872	389.049	728.847	440.976	400.484	708.007
128	394.735	381.752	486.057	415.437	394.791	917.581	436.880	405.336	721.500
129	388.362	384.095	595.599	359.800	394.577	625.077	450.678	408.575	588.391
130	383.128	386.433	503.401	401.167	394.005	771.727	431.449	399.041	623.896
131	353.771	386.789	509.845	378.976	393.556	715.306	387.267	396.485	741.026
132	380.272	380.531	496.594	406.797	391.767	788.564	408.503	388.767	542.885
133	390.214	384.502	557.539	402.050	387.014	715.159	408.996	387.860	471.559

Tabela 44 – RMSE (R\$) variando-se a quantidade de N_n para cada cenário

N_n	RMSE - Raiz do erro quadrático médio (R\$)								
	1HLES	1HLPR	1HL	2HLES	2HLPR	2HL	3HLES	3HLPR	3HL
134	383.519	386.035	563.553	390.867	391.747	648.908	410.265	393.750	775.298
135	377.913	382.789	673.679	365.403	388.981	787.347	389.363	388.808	678.074
136	399.546	377.802	506.632	413.471	386.127	762.996	390.322	392.831	438.842
137	362.771	385.022	500.780	386.879	392.767	521.362	396.257	395.362	630.031
138	373.424	385.251	561.489	388.752	391.892	574.667	380.586	388.483	606.336
139	382.107	382.150	500.480	356.127	390.933	809.187	397.770	383.733	497.883
140	389.308	388.452	494.951	393.659	388.884	816.215	388.695	384.197	720.284
141	403.209	378.687	597.540	398.562	393.742	621.598	419.729	385.244	573.983
142	403.814	389.570	567.919	420.134	400.984	738.571	403.456	382.966	774.402
143	392.793	383.514	817.983	374.156	383.380	673.458	379.675	375.909	644.428
144	388.313	384.963	548.831	378.425	386.399	635.897	410.859	379.640	642.012
145	399.566	379.209	549.654	381.593	389.946	1.002.940	395.656	395.270	533.976
146	376.782	391.506	514.255	399.916	393.320	533.590	376.711	389.395	543.603
147	386.868	382.665	588.897	404.261	387.827	659.507	417.769	387.391	759.577
148	388.984	382.426	556.293	381.507	393.606	787.983	391.092	374.872	555.061
149	391.654	391.036	574.706	380.798	393.681	781.439	449.419	393.690	555.255
150	384.145	387.298	578.956	399.855	387.334	658.868	401.877	387.487	560.537
151	382.856	384.634	557.399	402.307	396.595	834.981	376.383	392.100	493.562
152	363.682	385.011	608.466	406.701	391.770	733.886	431.980	393.395	560.410
153	385.032	381.597	586.233	430.274	388.444	831.047	389.053	390.865	733.594
154	386.963	391.062	596.747	395.238	389.776	834.694	382.897	385.173	627.815
155	387.026	383.813	534.850	395.030	391.463	841.935	375.101	381.687	604.632
156	392.288	386.653	593.799	437.459	391.061	850.745	427.680	381.400	556.053
157	396.920	383.562	524.610	385.075	388.869	998.008	432.243	388.474	980.358
158	384.117	388.237	543.482	384.783	385.347	739.909	408.271	414.154	753.984
159	390.892	383.109	594.982	414.124	395.515	902.466	394.921	387.057	573.796
160	384.567	384.172	525.312	413.502	393.826	795.870	413.650	384.205	814.667
161	380.448	382.457	552.829	397.945	391.876	979.999	396.634	395.006	578.560
162	383.201	383.739	656.277	389.166	391.395	623.684	410.439	387.911	572.897
163	392.811	384.329	537.194	406.757	384.596	625.197	396.153	389.772	797.271
164	394.401	385.466	556.929	413.470	389.908	774.052	395.821	400.212	681.466
165	398.892	384.422	570.929	437.274	387.712	590.052	383.361	388.372	590.344
166	380.772	384.435	538.197	401.237	390.070	659.505	371.525	404.145	719.302
167	377.473	384.339	567.607	381.679	392.380	749.896	391.037	436.569	568.589
168	389.139	383.018	624.010	393.493	388.900	793.939	382.257	422.763	597.819
169	389.395	381.041	533.572	390.570	389.233	826.189	398.374	397.081	592.536
170	387.141	384.323	635.494	403.807	390.426	985.397	414.451	406.193	737.127
171	384.460	385.446	513.680	365.433	383.751	681.446	422.478	393.481	618.706
172	391.895	382.310	587.738	397.666	393.080	623.218	412.200	398.568	549.111
173	377.745	384.911	545.467	371.770	389.637	707.031	417.916	393.497	602.723
174	370.263	384.716	537.825	418.202	390.418	591.724	384.339	391.281	590.776
175	391.226	384.040	575.590	414.620	393.425	702.406	391.826	377.205	576.111
176	387.954	386.252	552.634	390.938	385.065	750.382	401.048	385.379	639.127
177	376.626	379.653	645.060	372.528	395.723	749.828	380.457	378.132	872.218

Tabela 44 – RMSE (R\$) variando-se a quantidade de N_n para cada cenário

N_n	RMSE - Raiz do erro quadrático médio (R\$)								
	1HLES	1HLPR	1HL	2HLES	2HLPR	2HL	3HLES	3HLPR	3HL
178	399.579	383.913	603.443	400.589	385.437	507.144	408.669	387.717	688.670
179	377.354	381.749	568.465	369.724	386.558	799.820	416.488	393.182	605.557
180	389.511	381.384	584.899	385.992	381.450	1.036.472	400.071	391.790	521.624
181	387.579	381.837	571.930	395.108	386.975	605.360	387.424	385.212	647.639
182	381.011	382.895	566.740	398.102	397.731	676.533	408.883	378.095	582.102
183	386.452	385.136	615.894	395.868	391.705	674.520	379.183	381.366	483.085
184	405.556	383.561	534.438	387.435	392.108	830.341	413.083	401.516	701.972
185	394.736	385.678	588.740	393.565	394.658	908.683	379.974	397.090	729.638
186	364.125	381.359	605.391	393.654	393.779	788.160	401.351	393.587	667.255
187	389.074	384.454	604.933	383.062	383.169	782.261	415.651	391.087	542.509
188	385.081	381.258	543.547	395.340	388.156	753.223	437.975	375.447	672.443
189	386.130	380.325	629.415	400.070	393.750	773.364	385.153	387.565	621.385
190	382.121	385.147	567.479	403.124	398.269	650.123	406.077	388.282	650.076
191	390.833	386.271	554.290	381.175	394.627	842.912	407.579	383.869	585.162
192	383.373	383.405	564.057	392.970	404.085	496.503	354.221	382.479	486.985
193	390.923	384.979	616.839	384.880	391.609	746.615	408.612	395.396	535.336
194	386.585	383.658	517.053	377.171	387.450	533.634	406.174	394.335	598.472
195	387.221	382.776	578.289	397.084	393.881	790.118	404.940	390.353	998.708
196	377.516	382.546	524.938	384.705	394.090	614.895	386.706	392.152	800.195
197	413.565	382.790	676.344	403.361	394.392	683.083	411.859	391.203	616.210
198	377.318	385.263	616.749	405.843	394.858	712.930	394.307	389.432	621.939
199	383.208	383.905	601.182	387.498	393.867	621.282	400.274	380.974	542.235
200	385.374	385.109	565.356	395.862	396.177	673.914	422.313	372.654	702.172
201	396.112	383.019	605.937	410.632	396.694	615.367	399.861	388.405	675.838
202	386.613	389.285	569.592	386.397	389.331	608.475	399.940	384.280	642.993
203	396.599	383.256	538.611	412.565	393.924	571.870	426.812	383.623	589.426
204	387.128	383.754	563.233	383.045	392.714	875.775	387.003	380.495	695.731

APÊNDICE F – MÉTRICAS DAS MELHORES RNA COM TRANSFORMAÇÃO

Tabela 45 – Métricas das RNA com transformação e usando *Early Stopping*.

HL	Nn	Função	LR	Otimizador	Inicializador	RMSE	MAE	MAPE	Critério
2	46	tanh	0,001	Adam	glorot_uniform	329.338,34	212.364,91	14,00%	ES
2	112	tanh	0,001	Adam	glorot_uniform	338.948,56	210.085,93	13,66%	ES
3	54	tanh	0,001	Adam	glorot_uniform	340.649,04	211.275,32	14,28%	ES
1	39	tanh	0,001	Adam	glorot_uniform	344.061,29	217.732,88	13,71%	ES
3	119	tanh	0,001	Adam	glorot_uniform	346.648,75	201.999,19	13,67%	ES
3	107	tanh	0,001	Adam	glorot_uniform	347.984,13	208.701,13	13,67%	ES
1	131	tanh	0,001	Adam	glorot_uniform	353.771,10	223.702,05	13,67%	ES
3	192	tanh	0,001	Adam	glorot_uniform	354.221,42	221.721,65	14,40%	ES
2	139	tanh	0,001	Adam	glorot_uniform	356.126,80	221.498,07	13,79%	ES
1	37	tanh	0,001	Adam	glorot_uniform	356.136,94	209.367,56	13,12%	ES
2	74	tanh	0,001	Adam	glorot_uniform	359.096,00	227.388,76	14,14%	ES
2	129	tanh	0,001	Adam	glorot_uniform	359.799,85	225.873,02	14,11%	ES
3	74	tanh	0,001	Adam	glorot_uniform	361.834,56	229.375,80	14,49%	ES
1	137	tanh	0,001	Adam	glorot_uniform	362.770,54	223.083,08	13,79%	ES
1	152	tanh	0,001	Adam	glorot_uniform	363.682,43	224.090,91	13,88%	ES
2	121	tanh	0,001	Adam	glorot_uniform	363.758,78	230.304,11	14,79%	ES
1	186	tanh	0,001	Adam	glorot_uniform	364.124,93	224.798,82	14,03%	ES
1	114	tanh	0,001	Adam	glorot_uniform	364.754,06	223.987,38	13,54%	ES
1	10	tanh	0,001	Adam	glorot_uniform	364.841,64	220.101,90	14,75%	ES
1	19	tanh	0,001	Adam	glorot_uniform	365.265,04	232.399,38	14,57%	ES
2	135	tanh	0,001	Adam	glorot_uniform	365.402,71	219.860,83	14,34%	ES
2	171	tanh	0,001	Adam	glorot_uniform	365.433,02	225.206,20	14,59%	ES
1	33	tanh	0,001	Adam	glorot_uniform	366.000,52	226.258,52	13,96%	ES
1	29	tanh	0,001	Adam	glorot_uniform	366.212,71	226.882,75	14,36%	ES
2	90	tanh	0,001	Adam	glorot_uniform	368.202,09	224.595,44	13,62%	ES
2	205	tanh	0,001	Adam	glorot_uniform	368.623,32	224.818,77	14,34%	ES
2	179	tanh	0,001	Adam	glorot_uniform	369.723,87	227.387,86	14,12%	ES
1	174	tanh	0,001	Adam	glorot_uniform	370.263,25	228.754,61	14,02%	ES
3	16	tanh	0,001	Adam	glorot_uniform	371.115,31	235.785,53	15,12%	ES
1	122	tanh	0,001	Adam	glorot_uniform	371.377,54	228.625,83	14,07%	ES
3	166	tanh	0,001	Adam	glorot_uniform	371.524,93	233.684,54	14,68%	ES
1	61	tanh	0,001	Adam	glorot_uniform	371.654,02	226.806,81	14,17%	ES
2	173	tanh	0,001	Adam	glorot_uniform	371.769,97	225.880,14	14,45%	ES
2	177	tanh	0,001	Adam	glorot_uniform	372.528,25	229.522,40	14,34%	ES
3	57	tanh	0,001	Adam	glorot_uniform	372.943,66	232.318,96	14,87%	ES
1	97	tanh	0,001	Adam	glorot_uniform	373.353,30	229.261,00	14,20%	ES
1	138	tanh	0,001	Adam	glorot_uniform	373.424,05	230.665,24	13,99%	ES
3	37	tanh	0,001	Adam	glorot_uniform	373.638,04	240.809,79	14,98%	ES
1	46	tanh	0,001	Adam	glorot_uniform	374.048,48	236.695,02	14,76%	ES
2	143	tanh	0,001	Adam	glorot_uniform	374.155,79	231.287,10	14,32%	ES
3	155	tanh	0,001	Adam	glorot_uniform	375.100,73	226.543,41	14,50%	ES
3	55	tanh	0,001	Adam	glorot_uniform	375.133,67	237.116,72	14,80%	ES

Tabela 45 – Métricas das RNA com transformação e usando *Early Stopping*.

HL	Nn	Função	LR	Otimizador	Inicializador	RMSE	MAE	MAPE	Critério
3	151	tanh	0,001	Adam	glorot_uniform	376.383,31	236.231,32	15,09%	ES
1	18	tanh	0,001	Adam	glorot_uniform	376.585,37	229.265,44	14,57%	ES
1	177	tanh	0,001	Adam	glorot_uniform	376.625,64	231.234,42	14,06%	ES
3	146	tanh	0,001	Adam	glorot_uniform	376.710,52	230.798,71	14,71%	ES
1	62	tanh	0,001	Adam	glorot_uniform	376.753,78	231.961,83	14,00%	ES
2	35	tanh	0,001	Adam	glorot_uniform	376.766,79	228.980,43	13,94%	ES
1	146	tanh	0,001	Adam	glorot_uniform	376.781,89	234.335,06	14,14%	ES
1	118	tanh	0,001	Adam	glorot_uniform	376.800,69	229.876,96	14,11%	ES
2	122	tanh	0,001	Adam	glorot_uniform	376.891,81	237.084,85	14,65%	ES
1	75	tanh	0,001	Adam	glorot_uniform	376.994,48	227.041,18	14,14%	ES
2	194	tanh	0,001	Adam	glorot_uniform	377.170,83	231.375,89	14,32%	ES
1	198	tanh	0,001	Adam	glorot_uniform	377.318,17	229.078,54	14,19%	ES
1	179	tanh	0,001	Adam	glorot_uniform	377.354,23	229.310,56	14,19%	ES
1	167	tanh	0,001	Adam	glorot_uniform	377.472,88	228.265,59	14,17%	ES
1	196	tanh	0,001	Adam	glorot_uniform	377.515,98	230.149,06	14,26%	ES
1	173	tanh	0,001	Adam	glorot_uniform	377.744,81	230.707,43	14,26%	ES
1	135	tanh	0,001	Adam	glorot_uniform	377.912,68	230.789,03	14,20%	ES
2	144	tanh	0,001	Adam	glorot_uniform	378.425,17	238.572,97	14,77%	ES
1	70	tanh	0,001	Adam	glorot_uniform	378.500,78	229.918,31	14,31%	ES
2	97	tanh	0,001	Adam	glorot_uniform	378.760,21	223.976,34	14,18%	ES
3	59	tanh	0,001	Adam	glorot_uniform	378.960,41	231.899,78	14,72%	ES
2	131	tanh	0,001	Adam	glorot_uniform	378.975,63	231.264,13	14,46%	ES
1	67	tanh	0,001	Adam	glorot_uniform	379.076,47	239.186,55	14,38%	ES
1	71	tanh	0,001	Adam	glorot_uniform	379.164,87	231.944,50	14,19%	ES
3	183	tanh	0,001	Adam	glorot_uniform	379.183,26	232.766,36	14,42%	ES
1	8	tanh	0,001	Adam	glorot_uniform	379.458,21	240.621,61	15,12%	ES
3	143	tanh	0,001	Adam	glorot_uniform	379.674,94	220.714,06	14,19%	ES
3	185	tanh	0,001	Adam	glorot_uniform	379.974,10	234.390,55	14,95%	ES
1	113	tanh	0,001	Adam	glorot_uniform	380.003,30	231.659,58	14,27%	ES
1	99	tanh	0,001	Adam	glorot_uniform	380.227,53	235.097,98	14,35%	ES
1	132	tanh	0,001	Adam	glorot_uniform	380.272,22	227.854,38	14,27%	ES
2	83	tanh	0,001	Adam	glorot_uniform	380.361,74	241.406,83	14,82%	ES
1	161	tanh	0,001	Adam	glorot_uniform	380.448,21	229.274,55	14,06%	ES
3	177	tanh	0,001	Adam	glorot_uniform	380.456,70	236.369,78	14,84%	ES
3	138	tanh	0,001	Adam	glorot_uniform	380.586,10	239.128,87	14,79%	ES
1	88	tanh	0,001	Adam	glorot_uniform	380.713,76	229.752,60	14,24%	ES
2	125	tanh	0,001	Adam	glorot_uniform	380.732,34	235.386,33	14,34%	ES
1	166	tanh	0,001	Adam	glorot_uniform	380.771,72	233.110,59	14,67%	ES
2	149	tanh	0,001	Adam	glorot_uniform	380.797,51	229.651,69	14,37%	ES
1	20	tanh	0,001	Adam	glorot_uniform	380.806,89	223.274,84	14,13%	ES
1	182	tanh	0,001	Adam	glorot_uniform	381.011,27	231.524,58	14,19%	ES
1	87	tanh	0,001	Adam	glorot_uniform	381.122,63	229.518,31	14,16%	ES
2	191	tanh	0,001	Adam	glorot_uniform	381.175,14	233.100,06	14,08%	ES
1	105	tanh	0,001	Adam	glorot_uniform	381.314,33	229.717,76	14,13%	ES
1	110	tanh	0,001	Adam	glorot_uniform	381.414,25	233.440,47	14,37%	ES

Tabela 45 – Métricas das RNA com transformação e usando *Early Stopping*.

HL	Nn	Função	LR	Otimizador	Inicializador	RMSE	MAE	MAPE	Critério
2	148	tanh	0,001	Adam	glorot_uniform	381.507,26	235.769,70	14,61%	ES
2	145	tanh	0,001	Adam	glorot_uniform	381.592,85	232.883,68	14,19%	ES
2	167	tanh	0,001	Adam	glorot_uniform	381.678,77	237.191,65	14,64%	ES
1	80	tanh	0,001	Adam	glorot_uniform	381.770,17	233.552,10	14,32%	ES
1	58	tanh	0,001	Adam	glorot_uniform	382.004,17	234.417,01	14,40%	ES
1	139	tanh	0,001	Adam	glorot_uniform	382.106,51	232.206,37	14,21%	ES
1	190	tanh	0,001	Adam	glorot_uniform	382.120,98	232.548,01	14,25%	ES
3	168	tanh	0,001	Adam	glorot_uniform	382.257,22	233.784,63	14,60%	ES
1	107	tanh	0,001	Adam	glorot_uniform	382.452,69	234.666,93	14,45%	ES
1	116	tanh	0,001	Adam	glorot_uniform	382.785,73	238.487,05	14,24%	ES
3	40	tanh	0,001	Adam	glorot_uniform	382.825,62	234.187,81	14,49%	ES
1	151	tanh	0,001	Adam	glorot_uniform	382.856,24	231.156,50	14,43%	ES
3	154	tanh	0,001	Adam	glorot_uniform	382.896,75	241.853,25	15,53%	ES

Tabela 46 – Melhores das RNA com transformação e usando *Pruning*

HL	Nn	Função	LR	Otimizador	Inicializador	RMSE	MAE	MAPE	Critério
3	98	tanh	0,001	Adam	glorot_uniform	368.151,41	229.138,71	14,41%	PR
1	35	tanh	0,001	Adam	glorot_uniform	370.691,96	230.743,10	15,77%	PR
3	48	tanh	0,001	Adam	glorot_uniform	371.140,39	229.716,01	14,49%	PR
3	200	tanh	0,001	Adam	glorot_uniform	372.654,33	225.611,25	14,22%	PR
1	105	tanh	0,001	Adam	glorot_uniform	373.105,79	229.036,19	14,25%	PR
3	73	tanh	0,001	Adam	glorot_uniform	374.834,87	229.212,56	14,14%	PR
2	110	tanh	0,001	Adam	glorot_uniform	374.839,83	236.911,84	14,66%	PR
3	148	tanh	0,001	Adam	glorot_uniform	374.871,56	227.473,97	14,29%	PR
1	70	tanh	0,001	Adam	glorot_uniform	375.008,36	229.560,93	14,22%	PR
2	46	tanh	0,001	Adam	glorot_uniform	375.069,95	234.508,34	14,46%	PR
3	188	tanh	0,001	Adam	glorot_uniform	375.447,48	225.748,28	14,05%	PR
3	143	tanh	0,001	Adam	glorot_uniform	375.908,82	229.148,93	14,32%	PR
3	175	tanh	0,001	Adam	glorot_uniform	377.204,60	227.884,91	14,31%	PR
1	53	tanh	0,001	Adam	glorot_uniform	377.240,54	233.322,81	14,37%	PR
1	15	tanh	0,001	Adam	glorot_uniform	377.433,02	242.286,86	15,71%	PR
1	136	tanh	0,001	Adam	glorot_uniform	377.802,44	228.721,86	14,09%	PR
1	97	tanh	0,001	Adam	glorot_uniform	377.898,97	232.326,29	14,28%	PR
3	182	tanh	0,001	Adam	glorot_uniform	378.095,42	229.131,93	14,23%	PR
3	177	tanh	0,001	Adam	glorot_uniform	378.132,14	229.656,93	14,38%	PR
1	81	tanh	0,001	Adam	glorot_uniform	378.194,07	229.010,15	14,16%	PR
2	54	tanh	0,001	Adam	glorot_uniform	378.548,46	232.991,55	14,35%	PR
1	141	tanh	0,001	Adam	glorot_uniform	378.687,31	231.066,61	14,20%	PR
1	33	tanh	0,001	Adam	glorot_uniform	378.882,20	227.480,90	14,14%	PR
1	145	tanh	0,001	Adam	glorot_uniform	379.209,45	231.933,55	14,23%	PR
2	95	tanh	0,001	Adam	glorot_uniform	379.357,08	230.348,82	14,21%	PR
3	144	tanh	0,001	Adam	glorot_uniform	379.640,11	227.170,63	14,13%	PR
1	177	tanh	0,001	Adam	glorot_uniform	379.653,26	230.656,53	14,24%	PR
1	108	tanh	0,001	Adam	glorot_uniform	379.691,27	230.616,95	14,19%	PR
3	74	tanh	0,001	Adam	glorot_uniform	379.850,01	232.470,30	14,37%	PR

Tabela 46 – Melhores das RNA com transformação e usando *Pruning*

HL	Nn	Função	LR	Otimizador	Inicializador	RMSE	MAE	MAPE	Critério
1	116	tanh	0,001	Adam	glorot_uniform	380.143,83	234.156,86	14,33%	PR
3	81	tanh	0,001	Adam	glorot_uniform	380.176,96	235.455,74	14,29%	PR
2	96	tanh	0,001	Adam	glorot_uniform	380.293,92	234.314,89	14,41%	PR
1	189	tanh	0,001	Adam	glorot_uniform	380.324,86	229.207,43	14,10%	PR
1	77	tanh	0,001	Adam	glorot_uniform	380.418,63	231.173,11	14,20%	PR
3	204	tanh	0,001	Adam	glorot_uniform	380.495,39	232.818,13	14,39%	PR
1	132	tanh	0,001	Adam	glorot_uniform	380.530,52	231.066,73	14,22%	PR
1	92	tanh	0,001	Adam	glorot_uniform	380.605,50	233.828,46	14,32%	PR
3	84	tanh	0,001	Adam	glorot_uniform	380.654,05	232.259,19	14,19%	PR
3	40	tanh	0,001	Adam	glorot_uniform	380.820,57	237.354,75	14,72%	PR
3	121	tanh	0,001	Adam	glorot_uniform	380.841,79	229.221,59	14,31%	PR
3	199	tanh	0,001	Adam	glorot_uniform	380.973,96	241.349,04	14,87%	PR
1	169	tanh	0,001	Adam	glorot_uniform	381.040,89	231.009,99	14,24%	PR
3	54	tanh	0,001	Adam	glorot_uniform	381.102,80	238.487,93	14,55%	PR
1	188	tanh	0,001	Adam	glorot_uniform	381.257,64	231.560,87	14,26%	PR
1	74	tanh	0,001	Adam	glorot_uniform	381.272,67	233.657,89	14,31%	PR
1	186	tanh	0,001	Adam	glorot_uniform	381.358,58	232.033,82	14,30%	PR
3	183	tanh	0,001	Adam	glorot_uniform	381.365,71	232.491,71	14,37%	PR
1	180	tanh	0,001	Adam	glorot_uniform	381.383,75	232.243,85	14,27%	PR
3	156	tanh	0,001	Adam	glorot_uniform	381.399,53	229.576,19	14,30%	PR
2	180	tanh	0,001	Adam	glorot_uniform	381.450,48	230.503,23	14,26%	PR
2	47	tanh	0,001	Adam	glorot_uniform	381.503,53	234.126,84	14,43%	PR
2	92	tanh	0,001	Adam	glorot_uniform	381.554,53	229.979,30	14,20%	PR
1	153	tanh	0,001	Adam	glorot_uniform	381.597,41	231.766,75	14,27%	PR
3	155	tanh	0,001	Adam	glorot_uniform	381.686,56	226.751,67	14,09%	PR
1	179	tanh	0,001	Adam	glorot_uniform	381.748,81	232.392,15	14,28%	PR
1	128	tanh	0,001	Adam	glorot_uniform	381.752,39	233.350,51	14,31%	PR
1	181	tanh	0,001	Adam	glorot_uniform	381.837,16	232.451,85	14,27%	PR
3	97	tanh	0,001	Adam	glorot_uniform	381.876,74	231.237,23	14,29%	PR
1	90	tanh	0,001	Adam	glorot_uniform	382.023,05	233.624,99	14,29%	PR
1	52	tanh	0,001	Adam	glorot_uniform	382.096,14	232.646,10	14,24%	PR
1	139	tanh	0,001	Adam	glorot_uniform	382.149,94	234.524,02	14,33%	PR
1	113	tanh	0,001	Adam	glorot_uniform	382.162,90	233.588,05	14,29%	PR
1	172	tanh	0,001	Adam	glorot_uniform	382.310,50	233.585,59	14,32%	PR
1	124	tanh	0,001	Adam	glorot_uniform	382.346,50	234.278,37	14,35%	PR
1	106	tanh	0,001	Adam	glorot_uniform	382.378,25	233.192,44	14,30%	PR
1	148	tanh	0,001	Adam	glorot_uniform	382.425,64	234.058,32	14,35%	PR
1	161	tanh	0,001	Adam	glorot_uniform	382.457,34	232.387,38	14,27%	PR
3	192	tanh	0,001	Adam	glorot_uniform	382.479,44	232.135,63	14,32%	PR
1	100	tanh	0,001	Adam	glorot_uniform	382.535,39	232.065,13	14,24%	PR
1	196	tanh	0,001	Adam	glorot_uniform	382.546,45	233.130,35	14,32%	PR
1	122	tanh	0,001	Adam	glorot_uniform	382.629,20	233.372,60	14,25%	PR
1	147	tanh	0,001	Adam	glorot_uniform	382.664,55	231.493,21	14,27%	PR
1	121	tanh	0,001	Adam	glorot_uniform	382.687,03	232.860,51	14,29%	PR
1	195	tanh	0,001	Adam	glorot_uniform	382.775,68	232.921,37	14,24%	PR

Tabela 46 – Melhores das RNA com transformação e usando *Pruning*

HL	Nn	Função	LR	Otimizador	Inicializador	RMSE	MAE	MAPE	Critério
1	135	tanh	0,001	Adam	glorot_uniform	382.789,04	232.461,71	14,31%	PR
1	197	tanh	0,001	Adam	glorot_uniform	382.790,38	232.359,73	14,29%	PR
1	182	tanh	0,001	Adam	glorot_uniform	382.895,35	233.608,83	14,35%	PR
3	142	tanh	0,001	Adam	glorot_uniform	382.966,12	230.989,84	14,37%	PR
1	168	tanh	0,001	Adam	glorot_uniform	383.018,26	231.955,20	14,26%	PR
1	201	tanh	0,001	Adam	glorot_uniform	383.018,69	233.136,31	14,33%	PR
1	159	tanh	0,001	Adam	glorot_uniform	383.108,60	232.823,67	14,32%	PR
2	187	tanh	0,001	Adam	glorot_uniform	383.168,79	233.888,55	14,38%	PR
1	203	tanh	0,001	Adam	glorot_uniform	383.255,92	231.638,63	14,21%	PR
1	110	tanh	0,001	Adam	glorot_uniform	383.299,76	231.793,61	14,27%	PR
2	143	tanh	0,001	Adam	glorot_uniform	383.380,02	231.913,38	14,34%	PR
1	192	tanh	0,001	Adam	glorot_uniform	383.405,14	233.743,23	14,29%	PR
1	143	tanh	0,001	Adam	glorot_uniform	383.514,42	233.468,35	14,29%	PR
1	184	tanh	0,001	Adam	glorot_uniform	383.560,57	231.721,70	14,21%	PR
1	157	tanh	0,001	Adam	glorot_uniform	383.562,34	231.619,05	14,27%	PR
3	203	tanh	0,001	Adam	glorot_uniform	383.622,74	234.570,32	14,39%	PR
1	194	tanh	0,001	Adam	glorot_uniform	383.658,44	232.774,95	14,28%	PR
1	102	tanh	0,001	Adam	glorot_uniform	383.728,78	233.010,07	14,31%	PR
3	139	tanh	0,001	Adam	glorot_uniform	383.733,33	233.496,01	14,36%	PR
1	162	tanh	0,001	Adam	glorot_uniform	383.738,59	234.167,61	14,32%	PR
2	171	tanh	0,001	Adam	glorot_uniform	383.751,43	232.200,46	14,31%	PR
1	204	tanh	0,001	Adam	glorot_uniform	383.753,78	234.881,11	14,32%	PR
2	97	tanh	0,001	Adam	glorot_uniform	383.794,89	233.174,83	14,40%	PR
1	155	tanh	0,001	Adam	glorot_uniform	383.813,24	232.433,61	14,28%	PR
3	191	tanh	0,001	Adam	glorot_uniform	383.869,02	234.010,80	14,43%	PR
1	199	tanh	0,001	Adam	glorot_uniform	383.904,95	233.440,84	14,26%	PR

APÊNDICE G – MÉTRICAS DAS MELHORES RNA SEM TRANSFORMAÇÃO

Tabela 47 – Melhores das RNA sem transformação e usando *Early Stopping*

HL	Nn	Função	LR	Otimizador	Inicializador	RMSE	MAE	MAPE	Critério
2	370	relu	0,00050	RMSprop	glorot_normal	341.323,88	227.017,93	16,86%	ES
2	170	relu	0,00050	Adam	glorot_uniform	342.837,67	220.250,81	15,54%	ES
3	135	relu	0,00100	Nadam	glorot_uniform	343.990,54	235.675,94	17,46%	ES
2	370	relu	0,00100	Nadam	glorot_normal	346.583,71	225.531,95	16,01%	ES
3	255	relu	0,00100	Adam	glorot_uniform	349.659,84	220.523,22	16,67%	ES
3	435	relu	0,00100	Adam	glorot_normal	349.705,71	218.253,56	14,28%	ES
3	195	relu	0,00100	RMSprop	glorot_normal	350.807,34	214.986,72	14,07%	ES
2	170	relu	0,00100	Nadam	glorot_uniform	350.945,29	222.559,30	15,41%	ES
2	170	relu	0,00100	Adam	glorot_uniform	352.137,01	230.970,30	16,74%	ES
3	195	relu	0,00050	Nadam	glorot_normal	352.915,13	219.707,11	15,12%	ES
3	195	relu	0,00100	Nadam	glorot_normal	353.912,26	213.753,22	14,23%	ES
2	370	relu	0,00100	Adam	glorot_normal	354.281,24	221.723,53	14,83%	ES
3	375	relu	0,00100	Adam	glorot_normal	355.083,49	223.478,93	14,64%	ES
3	195	relu	0,00050	RMSprop	glorot_normal	355.097,60	227.391,10	14,27%	ES
3	435	relu	0,00050	Adam	glorot_uniform	355.981,55	226.017,86	15,68%	ES
3	555	relu	0,00050	Nadam	glorot_normal	356.097,43	225.583,66	16,04%	ES
3	555	relu	0,00050	RMSprop	glorot_normal	356.674,54	223.825,57	16,21%	ES
3	435	relu	0,00050	Adam	glorot_normal	357.755,85	219.751,09	13,69%	ES
2	370	relu	0,00050	Adam	glorot_normal	358.046,06	231.892,37	15,80%	ES
1	185	relu	0,00100	Adam	glorot_normal	358.187,56	222.833,12	15,21%	ES
2	370	relu	0,00050	Nadam	glorot_normal	359.127,57	237.151,40	16,54%	ES
2	170	relu	0,00100	RMSprop	glorot_uniform	360.412,94	228.698,59	17,08%	ES
3	135	relu	0,00050	Nadam	glorot_uniform	360.528,49	240.682,57	18,16%	ES
3	255	relu	0,00050	Adam	glorot_uniform	360.612,36	232.333,31	17,64%	ES
2	370	relu	0,00100	RMSprop	glorot_normal	360.989,40	232.339,17	16,22%	ES
2	410	relu	0,00100	Adam	glorot_normal	361.166,17	226.043,66	15,25%	ES
3	255	relu	0,00050	Nadam	glorot_uniform	361.326,07	230.159,49	17,14%	ES
3	495	relu	0,00050	Adam	glorot_normal	361.842,89	241.408,95	17,91%	ES
3	435	relu	0,00050	RMSprop	glorot_uniform	362.125,46	233.287,06	16,52%	ES
2	90	relu	0,00100	Nadam	glorot_uniform	362.424,46	236.613,62	17,12%	ES
3	195	relu	0,00050	Adam	glorot_normal	362.535,60	232.938,40	15,50%	ES
3	255	relu	0,00100	Nadam	glorot_normal	362.887,16	240.568,81	18,43%	ES
3	435	relu	0,00100	Adam	glorot_uniform	363.056,46	228.429,45	15,26%	ES
1	185	relu	0,00100	RMSprop	glorot_normal	363.553,30	226.781,34	15,45%	ES
3	135	relu	0,00100	Adam	glorot_uniform	363.602,00	244.497,04	17,25%	ES
3	375	relu	0,00050	Adam	glorot_normal	364.557,28	230.176,36	14,92%	ES
3	615	relu	0,00100	Nadam	truncated_normal	366.335,16	225.808,56	14,60%	ES
3	315	relu	0,00050	RMSprop	glorot_normal	366.377,38	240.469,70	16,43%	ES
2	90	relu	0,00100	Adam	glorot_uniform	366.673,13	241.898,07	18,19%	ES
2	130	relu	0,00100	Adam	glorot_uniform	366.890,11	246.709,27	18,26%	ES
3	435	relu	0,00100	Nadam	glorot_uniform	367.310,73	232.824,75	16,68%	ES
3	315	relu	0,00100	Adam	glorot_normal	367.508,59	238.120,37	16,19%	ES

Tabela 47 – Melhores das RNA sem transformação e usando *Early Stopping*

HL	Nn	Função	LR	Otimizador	Inicializador	RMSE	MAE	MAPE	Critério
2	410	relu	0,00100	RMSprop	glorot_uniform	367.543,35	241.717,46	16,94%	ES
3	555	relu	0,00050	Adam	glorot_normal	367.692,18	232.651,66	16,72%	ES
2	250	relu	0,00100	RMSprop	glorot_normal	368.367,06	232.466,25	15,72%	ES
3	255	relu	0,00100	RMSprop	glorot_uniform	368.573,92	238.308,33	18,09%	ES
2	250	relu	0,00100	Nadam	glorot_uniform	369.758,21	230.970,56	15,81%	ES
3	195	relu	0,00100	Adam	glorot_normal	369.928,79	225.996,59	14,86%	ES
3	495	relu	0,00100	Adam	glorot_normal	370.040,02	240.442,99	16,23%	ES
1	185	relu	0,00100	RMSprop	truncated_normal	370.538,75	232.232,77	15,21%	ES
3	135	relu	0,00050	Adam	glorot_uniform	370.638,97	248.175,51	18,48%	ES
2	330	relu	0,00100	Adam	glorot_normal	370.778,12	234.283,17	16,12%	ES
3	255	relu	0,00100	Adam	glorot_normal	371.233,09	249.840,51	19,13%	ES
2	290	relu	0,00100	Adam	glorot_normal	371.580,14	237.051,76	15,81%	ES
2	290	relu	0,00100	Adam	glorot_uniform	372.004,15	236.483,94	16,84%	ES
2	90	relu	0,00050	Adam	glorot_uniform	372.204,66	244.106,64	17,30%	ES
2	410	relu	0,00050	RMSprop	glorot_normal	372.374,05	238.762,04	16,77%	ES
3	495	relu	0,00050	Nadam	glorot_uniform	372.463,51	248.197,80	17,09%	ES
2	170	relu	0,00050	RMSprop	glorot_uniform	372.496,26	239.333,93	17,11%	ES
2	130	relu	0,00100	Nadam	glorot_uniform	372.846,22	246.456,12	17,18%	ES
3	135	relu	0,00100	RMSprop	glorot_normal	373.436,18	250.258,59	16,77%	ES
1	185	relu	0,00100	Nadam	truncated_normal	373.617,89	234.082,13	15,48%	ES
3	555	relu	0,00100	Adam	glorot_normal	373.900,35	254.942,32	20,79%	ES
3	615	relu	0,00100	Adam	truncated_normal	374.727,08	236.623,41	14,67%	ES
3	315	relu	0,00050	Nadam	glorot_normal	374.806,34	244.034,78	16,85%	ES
2	410	relu	0,00050	Adam	glorot_normal	374.842,91	233.772,03	15,76%	ES
2	410	relu	0,00050	Adam	truncated_normal	375.067,47	237.820,97	15,63%	ES
3	615	relu	0,00050	RMSprop	truncated_normal	375.249,31	228.753,40	14,57%	ES
3	435	relu	0,00050	Nadam	glorot_normal	375.542,04	239.039,60	15,79%	ES
3	255	relu	0,00100	RMSprop	glorot_normal	375.560,47	253.540,09	17,82%	ES
2	370	relu	0,00100	RMSprop	truncated_normal	375.957,34	235.771,17	14,83%	ES
2	330	relu	0,00100	Nadam	glorot_uniform	376.287,08	240.529,56	16,47%	ES
2	410	relu	0,00050	Nadam	glorot_normal	376.372,17	234.778,43	15,97%	ES
2	370	relu	0,00100	Nadam	truncated_normal	376.405,53	235.987,97	14,73%	ES
2	370	relu	0,00050	Adam	truncated_normal	376.681,86	236.997,50	14,83%	ES
1	145	relu	0,00100	RMSprop	glorot_normal	376.711,14	233.175,61	16,39%	ES
3	255	relu	0,00050	Adam	glorot_normal	376.753,29	244.385,96	17,80%	ES
3	255	relu	0,00100	Nadam	glorot_uniform	376.796,48	242.173,73	18,16%	ES
2	290	relu	0,00100	RMSprop	glorot_uniform	376.868,42	238.842,17	17,67%	ES
2	410	relu	0,00050	Nadam	truncated_normal	376.901,45	234.097,29	14,88%	ES
2	370	relu	0,00050	Nadam	truncated_normal	376.979,11	236.230,05	14,68%	ES
3	615	relu	0,00050	Nadam	truncated_normal	377.176,29	230.627,47	14,42%	ES
2	330	relu	0,00100	RMSprop	glorot_uniform	377.233,29	239.800,38	16,50%	ES
2	410	relu	0,00100	Nadam	truncated_normal	377.370,40	236.940,28	15,30%	ES
3	375	relu	0,00100	RMSprop	glorot_normal	377.428,46	250.140,85	18,04%	ES
1	185	relu	0,00050	Adam	glorot_normal	377.646,86	230.231,67	14,91%	ES
2	330	relu	0,00100	RMSprop	glorot_normal	377.648,20	245.445,14	16,68%	ES

Tabela 47 – Melhores das RNA sem transformação e usando *Early Stopping*

HL	Nn	Função	LR	Otimizador	Inicializador	RMSE	MAE	MAPE	Critério
2	170	relu	0,00050	Nadam	glorot_uniform	377.703,82	240.386,44	16,27%	ES
3	615	relu	0,00050	Adam	glorot_uniform	377.705,71	238.908,41	16,56%	ES
1	185	relu	0,00100	Nadam	glorot_normal	377.753,97	227.123,03	14,17%	ES
3	195	relu	0,00050	Nadam	glorot_uniform	377.797,70	244.164,83	17,16%	ES
3	495	relu	0,00050	RMSprop	glorot_uniform	377.847,96	245.336,64	16,68%	ES
3	435	relu	0,00050	Nadam	glorot_uniform	378.034,07	244.197,72	16,67%	ES
3	615	relu	0,00050	Adam	glorot_normal	378.075,29	239.749,28	15,94%	ES
1	85	relu	0,00100	RMSprop	glorot_normal	378.103,33	240.261,17	15,24%	ES
3	195	relu	0,00050	Adam	glorot_uniform	378.211,19	239.724,21	16,69%	ES
3	375	relu	0,00050	RMSprop	glorot_normal	378.256,16	242.050,36	15,65%	ES
3	555	relu	0,00100	Nadam	truncated_normal	378.325,09	243.698,42	16,38%	ES
2	290	relu	0,00050	Adam	glorot_normal	378.377,46	241.340,63	15,66%	ES
3	315	relu	0,00050	Adam	glorot_normal	378.574,74	243.277,14	16,23%	ES

Tabela 48 – Melhores das RNA sem transformação e usando *Pruning*

HL	Nn	Função	LR	Otimizador	Inicializador	RMSE	MAE	MAPE	Critério
3	195	relu	0,00	Adam	glorot_normal	311.684,91	206.648,02	15,01%	PR
2	210	relu	0,00	Nadam	glorot_normal	314.839,20	211.991,75	15,85%	PR
2	210	relu	0,00	Adam	glorot_normal	323.050,22	219.680,58	16,18%	PR
2	170	relu	0,00	RMSprop	glorot_normal	323.252,32	212.950,57	13,86%	PR
2	170	relu	0,00	Nadam	glorot_normal	330.369,46	224.268,27	16,65%	PR
2	370	relu	0,00	Nadam	glorot_normal	330.639,48	228.380,84	16,89%	PR
2	370	relu	0,00	RMSprop	glorot_normal	330.786,04	226.684,53	16,42%	PR
3	135	relu	0,00	Adam	truncated_normal	333.445,70	214.502,72	13,80%	PR
2	290	relu	0,00	Adam	truncated_normal	334.817,88	219.043,78	14,45%	PR
2	370	relu	0,00	RMSprop	glorot_normal	335.406,46	236.984,99	18,29%	PR
2	370	relu	0,00	Adam	glorot_normal	336.358,29	229.522,95	15,99%	PR
2	210	relu	0,00	Nadam	glorot_normal	337.819,64	232.827,10	17,64%	PR
3	135	relu	0,00	Adam	glorot_normal	337.991,03	221.865,84	14,28%	PR
2	290	relu	0,00	Nadam	glorot_uniform	338.220,61	226.029,03	14,98%	PR
2	370	relu	0,00	Adam	glorot_normal	338.636,04	225.569,12	15,11%	PR
2	330	relu	0,00	Adam	glorot_uniform	338.939,45	228.533,63	16,49%	PR
2	210	relu	0,00	Adam	glorot_normal	339.982,77	235.951,83	18,18%	PR
2	330	relu	0,00	Adam	glorot_normal	341.932,82	231.400,71	15,90%	PR
2	370	relu	0,00	Nadam	glorot_normal	341.998,36	240.066,45	17,24%	PR
3	255	relu	0,00	Adam	truncated_normal	342.023,91	218.206,91	14,53%	PR
3	135	relu	0,00	Adam	truncated_normal	343.253,48	221.203,89	14,19%	PR
2	170	relu	0,00	Adam	glorot_normal	343.806,54	226.826,11	15,71%	PR
3	135	relu	0,00	RMSprop	truncated_normal	344.219,79	221.503,89	14,20%	PR
2	170	relu	0,00	Adam	glorot_normal	344.997,11	223.539,18	15,50%	PR
2	210	relu	0,00	RMSprop	glorot_normal	345.235,68	225.843,99	16,25%	PR
2	290	relu	0,00	Adam	glorot_normal	345.316,88	221.785,62	15,61%	PR
1	145	relu	0,00	Adam	glorot_uniform	345.418,27	226.042,71	16,44%	PR
2	250	relu	0,00	RMSprop	glorot_normal	345.507,37	231.655,37	16,60%	PR
2	410	relu	0,00	RMSprop	truncated_normal	345.587,39	225.723,40	14,60%	PR

Tabela 48 – Melhores das RNA sem transformação e usando *Pruning*

HL	Nn	Função	LR	Otimizador	Inicializador	RMSE	MAE	MAPE	Critério
1	165	relu	0,00	Nadam	glorot_uniform	345.904,95	222.804,96	15,14%	PR
2	330	relu	0,00	Nadam	glorot_uniform	346.412,41	234.737,82	17,18%	PR
1	145	relu	0,00	Nadam	glorot_uniform	346.444,24	234.642,01	17,58%	PR
2	250	relu	0,00	Adam	glorot_normal	346.793,60	226.699,98	15,15%	PR
3	195	relu	0,00	Nadam	glorot_normal	346.986,75	219.128,07	14,97%	PR
3	135	relu	0,00	RMSprop	glorot_normal	347.304,41	235.266,71	15,89%	PR
2	290	relu	0,00	RMSprop	glorot_normal	347.529,15	227.339,53	16,38%	PR
2	290	relu	0,00	RMSprop	truncated_normal	347.938,26	223.839,65	14,51%	PR
3	135	relu	0,00	Nadam	glorot_normal	348.042,50	228.056,56	14,92%	PR
2	290	relu	0,00	Nadam	truncated_normal	348.475,78	228.040,47	15,07%	PR
3	195	relu	0,00	RMSprop	truncated_normal	348.672,06	225.363,91	14,67%	PR
3	195	relu	0,00	RMSprop	glorot_normal	348.679,56	245.706,37	18,32%	PR
1	145	relu	0,00	Nadam	glorot_uniform	348.687,62	232.524,51	16,46%	PR
2	370	relu	0,00	RMSprop	truncated_normal	349.015,05	228.868,04	14,85%	PR
3	255	relu	0,00	Adam	truncated_normal	349.171,60	222.571,32	14,12%	PR
2	250	relu	0,00	Nadam	glorot_normal	349.227,33	241.483,20	18,24%	PR
3	195	relu	0,00	Nadam	truncated_normal	349.271,72	223.338,80	14,70%	PR
3	135	relu	0,00	Adam	glorot_normal	349.430,72	234.120,65	17,13%	PR
1	145	relu	0,00	RMSprop	glorot_uniform	349.693,21	232.036,48	15,99%	PR
2	370	relu	0,00	RMSprop	glorot_uniform	349.784,80	225.285,63	14,96%	PR
2	330	relu	0,00	Nadam	glorot_uniform	349.786,96	238.934,35	16,74%	PR
2	290	relu	0,00	Adam	glorot_uniform	350.222,08	237.147,60	16,26%	PR
2	170	relu	0,00	RMSprop	glorot_normal	350.492,85	233.212,87	15,56%	PR
1	165	relu	0,00	RMSprop	glorot_uniform	350.650,34	235.957,69	17,03%	PR
3	435	relu	0,00	Adam	glorot_uniform	350.658,95	238.659,27	16,40%	PR
3	195	relu	0,00	Adam	glorot_normal	351.244,53	216.495,98	14,53%	PR
2	290	relu	0,00	RMSprop	glorot_uniform	351.522,00	230.700,77	15,14%	PR
2	290	relu	0,00	Nadam	truncated_normal	351.787,32	229.809,01	14,91%	PR
2	250	relu	0,00	RMSprop	truncated_normal	351.789,94	231.133,14	15,10%	PR
2	210	relu	0,00	RMSprop	glorot_normal	352.020,63	241.552,92	17,56%	PR
2	250	relu	0,00	RMSprop	glorot_uniform	352.133,30	226.627,77	14,59%	PR
2	410	relu	0,00	Nadam	truncated_normal	352.324,86	229.367,30	15,06%	PR
2	290	relu	0,00	Nadam	glorot_uniform	352.424,31	238.397,33	15,95%	PR
3	135	relu	0,00	Adam	glorot_uniform	352.582,75	220.648,30	14,76%	PR
1	145	relu	0,00	RMSprop	glorot_uniform	353.195,75	228.369,57	16,70%	PR
1	165	relu	0,00	Adam	glorot_uniform	353.533,32	225.231,48	15,42%	PR
2	210	relu	0,00	Nadam	glorot_uniform	353.670,08	227.363,75	14,15%	PR
2	250	relu	0,00	RMSprop	glorot_normal	354.164,77	230.057,91	16,41%	PR
3	195	relu	0,00	RMSprop	glorot_normal	354.191,71	231.749,86	15,93%	PR
2	330	relu	0,00	Adam	glorot_uniform	354.754,57	224.757,40	14,62%	PR
3	195	relu	0,00	RMSprop	truncated_normal	354.830,18	230.368,26	15,06%	PR
2	370	relu	0,00	Nadam	truncated_normal	355.503,83	235.545,46	15,46%	PR
2	210	relu	0,00	Nadam	truncated_normal	355.524,65	229.341,63	14,70%	PR
2	330	relu	0,00	RMSprop	glorot_uniform	355.581,04	232.850,00	16,59%	PR
1	145	relu	0,00	Nadam	truncated_normal	355.591,62	228.544,72	14,61%	PR

Tabela 48 – Melhores das RNA sem transformação e usando *Pruning*

HL	Nn	Função	LR	Otimizador	Inicializador	RMSE	MAE	MAPE	Critério
3	255	relu	0,00	Nadam	truncated_normal	355.647,93	229.865,92	14,91%	PR
2	130	relu	0,00	Nadam	truncated_normal	356.009,45	234.812,63	15,14%	PR
2	210	relu	0,00	Nadam	truncated_normal	356.062,86	228.442,61	14,43%	PR
2	330	relu	0,00	Nadam	glorot_normal	356.118,62	241.327,13	15,87%	PR
3	195	relu	0,00	Nadam	truncated_normal	356.131,64	232.184,80	15,26%	PR
1	165	relu	0,00	Adam	glorot_uniform	356.478,06	241.081,65	16,77%	PR
2	410	relu	0,00	Nadam	glorot_uniform	356.644,83	235.627,83	16,63%	PR
1	165	relu	0,00	RMSprop	glorot_uniform	356.852,28	235.284,94	16,57%	PR
2	370	relu	0,00	Nadam	glorot_uniform	357.068,42	240.027,84	16,42%	PR
1	145	relu	0,00	Adam	glorot_uniform	357.072,52	227.911,94	15,88%	PR
2	210	relu	0,00	RMSprop	truncated_normal	357.538,19	228.247,76	14,67%	PR
2	130	relu	0,00	RMSprop	glorot_normal	357.603,08	236.178,70	16,49%	PR
2	290	relu	0,00	Nadam	glorot_normal	357.752,56	239.480,48	17,00%	PR
3	135	relu	0,00	Nadam	glorot_uniform	357.799,76	219.059,34	13,72%	PR
3	195	relu	0,00	Adam	truncated_normal	358.052,57	230.726,90	14,79%	PR
2	410	relu	0,00	Nadam	glorot_uniform	358.174,92	238.032,16	16,83%	PR
2	250	relu	0,00	Adam	glorot_normal	358.237,63	241.195,89	17,07%	PR
2	130	relu	0,00	Adam	glorot_normal	358.313,31	232.049,93	16,13%	PR
2	370	relu	0,00	Adam	truncated_normal	358.320,94	229.053,11	14,25%	PR
2	290	relu	0,00	Nadam	glorot_normal	359.066,83	235.407,73	16,45%	PR
2	170	relu	0,00	RMSprop	truncated_normal	359.234,91	236.225,77	15,70%	PR
2	290	relu	0,00	Adam	truncated_normal	359.241,61	232.973,40	15,06%	PR
2	370	relu	0,00	RMSprop	truncated_normal	359.409,02	236.064,05	15,25%	PR
2	170	relu	0,00	Nadam	glorot_normal	359.674,95	226.165,56	14,55%	PR
2	410	relu	0,00	Adam	truncated_normal	359.683,83	233.934,00	15,77%	PR
3	195	relu	0,00	Nadam	glorot_normal	359.853,70	244.398,68	18,13%	PR

APÊNDICE H – CÓDIGOS EM PYTHON

O módulo “*settings*” usados neste apêndice consta no quadro 20. Esse módulo define um conjunto de variáveis utilizadas em vários códigos apresentados. Os códigos podem ser obtidos também no [github](#).

Quadro 7 – Definir as transformações possíveis para cada variável.

```

1 # --- DEFINIÇÃO DAS TRANSFORMAÇÕES QUE PODEM SER UTILIZADAS ---
2 import numpy as np
3 from settings import data_y, data_x, k, lista_selected_tr
4 # PADRÕES PARA CADA TRANSFORMAÇÃO (TR): 1=Original; 2=x²; 3=sqrt(x); 4=Ln; 5=1/x; 6=1/x²; 7=1/sqrt(x);
5 # CARREGARR "data_y" E "data_x" (REFERENTES À VARIÁVEL DEPENDENTE E INDEPENDENTES);
6 # "data_x_com_y" É UM ARRAY CONTENDO OS DADOS DAS VARIÁVEIS INDEPENDENTES E A DEPENDENTE;
7 data_x_com_y=np.hstack((data_y, np.delete(data_x, 0, 1)))
8 # "data_x_com_y_split" É UM ARRAY COM AS VARIÁVEIS SEPARADAS POR COLUNAS;
9 # "k" É O NÚMERO DE VARIÁVEIS INDEPENDENTES;
10 data_x_com_y_split = np.hsplit(data_x_com_y, k + 1)
11 # "lista_selected_tr" GUARDA AS TR QUE O USUÁRIO SELECIONOU NA INTERFACE GRÁFICA, ex.: [1,4,6];
12 # "lista_tr_nao_aceita_zero" É A LISTA COM AS TR QUE NÃO PODEM SER USADAS QUANDO HOUVER 0;
13 lista_tr_nao_aceita_zero = [i for i in lista_selected_tr if i >= 4]
14 # "lista_tr_indice" É A LISTA QUE MOSTRA QUAIS TR CADA VARIÁVEL ACEITA;
15 lista_tr_indice = []
16 for i in range(k + 1):
17     if 0 not in data_x_com_y_split[i]:
18         lista_tr_indice.append(lista_selected_tr)
19     else:
20         lista_tr_indice.append([i for i in lista_selected_tr if i not in lista_tr_nao_aceita_zero])
21
22 with open('results/transformacoes_aceitas.txt', 'w') as file:
23     file.write(f'A lista de transformações que cada variável aceita é: {lista_tr_indice}')
24     print(f'Lista de transformações exportadas com sucesso!')
```

Quadro 8 – Cálculo dos modelos RLC.

```

1 # --- BUSCA PELOS MELHORES MODELOS DE REGRESSÃO ---
2 import os, pathlib, statsmodels.api, numpy as np, pandas as pd
3 from itertools import product
4 from settings import r2min, maxcomb, lista_tr_indice, n, k, data_x, data_y, data_train, data_val
5
6 x_train = data_train[
7     ["AREA", "WC", "SUITE", "VAGA", "PADRAO", "IDADE", "DIST", "Norte"]
8 ]
9 y_train = data_train[["VT"]]
10
11 x_val = data_val[
12     ["AREA", "WC", "SUITE", "VAGA", "PADRAO", "IDADE", "DIST", "Norte"]
13 ]
14 y_val = data_val[["VT"]]
15
16 # É CRIADA UM ARRAY INICIAL ("array_incremental_variaveis") COM UMA COLUNA UNITÁRIA DE "n" DADOS;
17 # ESSA ARRAY RECEBERÁ AS VARIÁVEIS "x" E "y" DE MODO INCREMENTAL EM CADA ITERAÇÃO;
18 # PARA CADA COMBINAÇÃO, É GERADA UMA ARRAY DIFERENTE;
19 global array_incremental_variaveis, data_x_com_y_split
20 array_incremental_variaveis = np.ones(n)
21 # "data_y" REFERE-SE À VARIÁVEL DEPENDENTE;
22 # "data_x" REFEREM-SE ÀS VARIÁVEIS INDEPENDENTES (sem coluna unitária);
23 data_y = np.array(y_train)
24 data_x = np.array(x_train)
25 data_x = np.insert(data_x, 0, np.ones(n), axis=1)
26 data_x_com_y=np.hstack((data_y, np.delete(data_x, 0, 1))) # UM ARRAY CONTENDO AS VARIÁVEIS INDEPENDENTES E A DEPENDENTE;
27 data_x_com_y_split = np.hsplit(data_x_com_y, k + 1) # É UM ARRAY COM AS VARIÁVEIS SEPARADAS POR COLUNAS;
28
29 # CRIAR FUNÇÃO PARA APLICAR AS TRANSFORMAÇÕES (TR) NAS VARIÁVEIS;
30 # 1-Original; 2=x²; 3=sqrt(x); 4=Ln; 5=1/x; 6=1/x²; 7=1/sqrt(x);
31
32
33 def transformar_variavel(opcao_tr, contador_tr, data_x_com_y_split):
34     dicionario_transforcoes = {1: data_x_com_y_split[contador_tr],
35                               2: np.square(data_x_com_y_split[contador_tr]),
36                               3: np.sqrt(data_x_com_y_split[contador_tr]),
37                               4: np.log(data_x_com_y_split[contador_tr]),
38                               5: np.reciprocal(data_x_com_y_split[contador_tr], dtype=float),
39                               6: np.reciprocal(np.square(data_x_com_y_split[contador_tr]), dtype=float),
40                               7: np.reciprocal(np.sqrt(data_x_com_y_split[contador_tr]), dtype=float)}
41
```

Quadro 8 – Cálculo dos modelos RLC.

```

42     return dicionario_transforcoes[opcao_tr]
43
44 # FUNÇÃO QUE CHAMA AS TRANSFORMAÇÕES E JUNTA AS COLUNAS TRANSFORMADAS PARA CRIAR UMA ÚNICA ARRAY;
45 # "opcao_tr" E "contador_tr" SÃO A OPÇÃO DA TRANSFORMAÇÃO E O CONTADOR DA COLUNA REFERENTE À TRANSFORMAÇÃO;
46 # "coluna_temp" É A COLUNA DE DADOS QUE SERÁ ADICIONADA À array_incremental_variaveis;
47 def escolher_transformacao(opcao_tr, contador_tr):
48     global array_incremental_variaveis, data_x_com_y_split, coluna_temp
49     coluna_temp = transformar_variavel(opcao_tr, contador_tr, data_x_com_y_split)
50     array_incremental_variaveis = np.column_stack((array_incremental_variaveis, coluna_temp))
51
52 # FUNÇÃO QUE SALVA OS RESULTADOS DOS MODELOS. ISSO PERMITE UTILIZÁ-LOS POSTERIORMENTE;
53 def salvar_resultados(resultados, combinacao):
54     path = pathlib.Path(os.path.join(os.getcwd(), 'modelos_calculados'))
55     path.mkdir(parents=True, exist_ok=True)
56     resultados.save(os.path.join(path, str(combinacao)))
57
58 # "lista_combinacao_r2" É UMA LISTA DE LISTAS: A PRIMEIRA REPRESENTA A COMBINAÇÃO; A SEGUNDA REPRESENTA O R2;
59 lista_combinacao_r2 = [[], []]
60
61 # A SEGUIR, AS ITERAÇÕES, ONDE PARA CADA COMBINAÇÃO SERÁ CALCULADO UM R2;
62 # "lista_tr_indice" É A LISTA QUE MOSTRA QUAIS TR CADA VARIÁVEL ACEITA;
63 for combinacao in (product(*lista_tr_indice)):
64     for contador, opcao in enumerate(combinacao):
65         global coluna_temp
66         escolher_transformacao(opcao, contador)
67         array_data_x_com_y = np.delete(array_incremental_variaveis, 0, 1)
68         data_ones = np.ones((n, 1))
69         modelo = statsmodels.api.OLS(np.delete(array_data_x_com_y, np.s_[1:k + 1], 1),
70                                   np.hstack((data_ones, np.delete(array_data_x_com_y, 0, 1))))
71         resultados = modelo.fit()
72         salvar_resultados(resultados, combinacao)
73         # CASO O VALOR DE R2 SEJA MENOR QUE O MÍNIMO ADOTADO PELO USUÁRIO (r2min);
74         if resultados.rsquared < r2min:
75             array_incremental_variaveis = np.ones(n) # O ARRAY É REDEFINIDO;
76             continue # PASSA PRA PRÓXIMA ITERAÇÃO;
77
78         # CASO AINDA NÃO TENHA SIDO ATINGIDO O LIMITE DE COMBINAÇÕES DEFINIDO PELO USUÁRIO (maxcomb);
79         if len(lista_combinacao_r2[1]) <= maxcomb - 1:
80             lista_combinacao_r2[0].append(combinacao) # ADICIONA A COMBINAÇÃO USADA;
81             lista_combinacao_r2[1].append(resultados.rsquared) # ADICIONA O R2 DESSA COMBINAÇÃO;
82             array_incremental_variaveis = np.ones(n) # O ARRAY É REDEFINIDO;
83             continue # PASSA PRA PRÓXIMA ITERAÇÃO;
84
85         # CASO TENHA SIDO ATINGIDO O LIMITE DE COMBINAÇÕES DEFINIDO PELO USUÁRIO;
86         # EXCLUIR O MENOR R2 QUE JÁ CONSTA SE O NOVO R2 FOR MAIOR QUE O MENOR;
87         if len(lista_combinacao_r2[1]) == maxcomb and resultados.rsquared > min(lista_combinacao_r2[1]):
88             del lista_combinacao_r2[0][lista_combinacao_r2[1].index(min(lista_combinacao_r2[1]))]
89             del lista_combinacao_r2[1][lista_combinacao_r2[1].index(min(lista_combinacao_r2[1]))]
90             lista_combinacao_r2[0].append(combinacao) # ADICIONA A COMBINAÇÃO USADA;
91             lista_combinacao_r2[1].append(resultados.rsquared) # ADICIONA O R2 DESSA COMBINAÇÃO;
92             array_incremental_variaveis = np.ones(n) # O ARRAY É REDEFINIDO;
93             continue # PASSA PRA PRÓXIMA ITERAÇÃO;
94         array_incremental_variaveis = np.ones(n) # REDEFINIÇÃO CASO NÃO TENHA ENTRADO EM NENHUM IF;
95
96 # EXPORTAR TABELA DE ARRANJOS E OPÇÕES
97 pd.DataFrame(data=[[x, y] for x, y in zip(lista_combinacao_r2[0], lista_combinacao_r2[1])],
98             columns = ['Combinacao', 'R2']).sort_values(by='R2', ascending=False).to_csv('results/tabela_arranjos.csv')
99 print(f'Tabela exportada com sucesso!')

```

Quadro 9 – Cálculo de métricas de RLC

```

1 # ---- CALCULAR R2 E RMSE - metricas_rlc.py ----
2 import numpy as np, os, pathlib, statsmodels.api, statsmodels.stats.stattools
3 from sklearn.metrics import mean_squared_error, mean_absolute_error, mean_absolute_percentage_error, r2_score
4 from settings import combinacao, data_train, data_val, data_y
5
6 # CONSIDERANDO UM MODELO CALCULADO EM QUE:
7 path_resultados = pathlib.Path(os.path.join(os.getcwd(), 'modelos_calculados', str(combinacao)))
8 data_train['VT'] = np.log(data_train['VT'])
9 data_train['AREA'] = np.log(data_train['AREA'])
10 data_train['WC'] = np.log(data_train['WC'])
11 data_train['PADRAO'] = np.log(data_train['PADRAO'])
12 data_train['DIST'] = np.log(data_train['DIST'])
13 data_val['VT'] = np.log(data_val['VT'])
14 data_val['AREA'] = np.log(data_val['AREA'])
15 data_val['WC'] = np.log(data_val['WC'])
16 data_val['PADRAO'] = np.log(data_val['PADRAO'])
17 data_val['DIST'] = np.log(data_val['DIST'])
18 x_train = data_train[
19     ["AREA", "WC", "SUITE", "VAGA", "PADRAO", "IDADE", "DIST", "Norte"]
20 ]
21 y_train = data_train[["VT"]]
22

```

Quadro 9 – Cálculo de métricas de RLC

```

23 x_val = data_val[
24     ["AREA", "WC", "SUITE", "VAGA", "PADRAO", "IDADE", "DIST", "Norte"]
25 ]
26 y_val = data_val[["VT"]]
27
28 try:
29     resultados = statsmodels.api.load(str(path_resultados))
30     # TREINO
31     y_predicted = np.dot(resultados.model.exog, resultados.params)
32     r2 = r2_score(y_train, y_predicted)
33     rmse = np.sqrt(mean_squared_error(np.exp(y_train), np.exp(y_predicted)))
34     mae = mean_absolute_error(np.exp(y_train), np.exp(y_predicted))
35     mape = mean_absolute_percentage_error(np.exp(y_train), np.exp(y_predicted))
36
37     # VALIDACAO
38     y_predicted = np.dot(np.insert(np.array(x_val), 0, np.ones(52), axis=1), resultados.params)
39     r2_val = r2_score(y_val, y_predicted)
40     rmse_val = np.sqrt(mean_squared_error(np.exp(y_val), np.exp(y_predicted)))
41     mae_val = mean_absolute_error(np.exp(y_val), np.exp(y_predicted))
42     mape_val = mean_absolute_percentage_error(np.exp(y_val), np.exp(y_predicted))
43
44     with open('results/metricas_rlc.txt', 'w') as file:
45         file.write(f'TREINO:\n')
46         file.write(f'R² = {round(r2*100, 2)}%;\n')
47         file.write(f'RMSE = {round(rmse, 2)};\n')
48         file.write(f'MAE = {round(mae, 2)};\n')
49         file.write(f'MAPE = {round(mape, 4)}%.\n\n')
50         file.write(f'VALIDAÇÃO:\n')
51         file.write(f'R² = {round(r2_val * 100, 2)}%;\n')
52         file.write(f'RMSE = {round(rmse_val, 2)};\n')
53         file.write(f'MAE = {round(mae_val, 2)};\n')
54         file.write(f'MAPE = {round(mape_val, 4)}%.')
55     print(f'Métricas calculadas e exportadas com sucesso.')
56
57 except Exception:
58     print(f'Não existe o caminho "{path_resultados}" ou o modelo da combinação "{combinacao}" não foi calculado.')

```

Quadro 10 – Resíduos padronizados e distâncias de Cook

```

1 # --- CÁLCULO DOS RESÍDUOS PADRONIZADOS E DISTÂNCIAS DE COOK ---
2 import numpy as np, os, pathlib, statsmodels.api, statsmodels.stats.outliers_influence
3 from settings import n, k, combinacao
4
5 # CONSIDERANDO OS RESULTADOS DO MODELO SELECIONADO:
6 path_resultados = pathlib.Path(os.path.join(os.getcwd(), 'modelos_calculados', str(combinacao)))
7 try:
8     resultados = statsmodels.api.load(str(path_resultados))
9     # "n" É O NÚMERO DE DADOS DA AMOSTRA, "k" É O NÚMERO DE VARIÁVEIS EXPLICATIVAS;
10     sqe = resultados.mse_resid * (n - k - 1) # SOMA DOS QUADRADOS DOS ERROS;
11     s = (sqe / (n - k - 1)) ** 0.5 # DESVIO PADRÃO;
12     residuos_estimados = resultados.resid
13     residuos_padronizados = np.divide(residuos_estimados, s)
14     np.savetxt('results/residuos_padronizados.csv', residuos_padronizados)
15     distancias_cook = statsmodels.stats.outliers_influence.OLSInfluence(resultados).cooks_distance
16     np.savetxt('results/distancias_cook.csv', distancias_cook[0])
17     print(f'Resíduos padronizados e distâncias de cook calculados e exportados com sucesso.')
18 except Exception:
19     print(f'Não existe o caminho "{path_resultados}" ou o modelo da combinação "{combinacao}" não foi calculado.')

```

Quadro 11 – Teste de Kolmogorov-Smirnov

```

1 # --- TESTE DE NORMALIDADE DE KOLMOGOROV-SMIRNOV ---
2 import numpy as np, os, pathlib, statsmodels.api, statsmodels.stats.stattools
3 from scipy import stats
4 from settings import combinacao, n, k
5 # CONSIDERANDO UM MODELO CALCULADO EM QUE:
6 path_resultados = pathlib.Path(os.path.join(os.getcwd(), 'modelos_calculados', str(combinacao)))
7 try:
8     resultados = statsmodels.api.load(str(path_resultados))
9     # "n" É O NÚMERO DE DADOS DA AMOSTRA, "k" É O NÚMERO DE VARIÁVEIS EXPLICATIVAS;
10     sqe = resultados.mse_resid * (n - k - 1) # SOMA DOS QUADRADOS DOS ERROS;
11     s = (sqe / (n - k - 1)) ** 0.5 # DESVIO PADRÃO;
12     residuos_padronizados = np.divide(resultados.resid, s)
13     # "cdf" É UM PARÂMETRO QUE INDICA A FUNÇÃO DE REFERÊNCIA (CURVA NORMAL);
14     kolmogorov_smirnov = stats.kstest(residuos_padronizados, cdf='norm')
15     with open('results/kolmogorov_smirnov.txt', 'w') as file:
16         file.write(f'Kolmogorov-Smirnov - Estatística: {kolmogorov_smirnov[0]}; P-valor:{kolmogorov_smirnov[1]}')
17     print(f'Kolmogorov-Smirnov calculado e exportado com sucesso.')
18 except Exception:
19     print(f'Não existe o caminho "{path_resultados}" ou o modelo da combinação "{combinacao}" não foi calculado.')

```

Quadro 12 – Testes de White e Breusch-Pagan

```

1 # --- TESTES DE WHITE E BREUSCH-PAGAN ---
2 import statsmodels.api, statsmodels.stats.diagnostic, pathlib, os
3 from settings import combinacao
4 # CONSIDERANDO OS RESULTADOS DO MODELO SELECIONADO:
5 path_resultados = pathlib.Path(os.path.join(os.getcwd(), 'modelos_calculados', str(combinacao)))
6 try:
7     resultados = statsmodels.api.load(str(path_resultados))
8     # "resultados.resid" SÃO OS RESÍDUOS ESTIMADOS;
9     # "resultados.model.exog" SÃO AS VARIÁVEIS EXPLICATIVAS;
10    teste_white = statsmodels.stats.diagnostic.het_white(resultados.resid, resultados.model.exog)
11    with open('results/teste_white.txt', 'w') as file:
12        file.write(f'White - Estatística: {teste_white[0]}; P-valor:{teste_white[1]}')
13    teste_breusch_pagan = statsmodels.stats.diagnostic.het_breuschpagan(resultados.resid, resultados.model.exog)
14    with open('results/teste_breusch_pagan.txt', 'w') as file:
15        file.write(f'Breusch-Pagan - Estatística: {teste_breusch_pagan[0]}; P-valor:{teste_breusch_pagan[1]}')
16    print(f'White e Breusch-Pagan calculados e exportados com sucesso.')
17 except Exception:
18    print(f'Não existe o caminho "{path_resultados}" ou o modelo da combinação "{combinacao}" não foi calculado.')

```

Quadro 13 – Matriz de correlação, índice condicional e VIF.

```

1 # --- CÁLCULO DA CORRELAÇÃO, NÚMERO CONDICIONAL E VIF ---
2 import numpy as np, os, pandas as pd, pathlib, statsmodels.api, statsmodels.stats.outliers_influence
3 from sklearn.preprocessing import normalize
4 from settings import k, combinacao
5 # CONSIDERANDO UM MODELO CALCULADO EM QUE:
6 path_resultados = pathlib.Path(os.path.join(os.getcwd(), 'modelos_calculados', str(combinacao)))
7 try:
8     resultados = statsmodels.api.load(str(path_resultados))
9     # MATRIZ DE CORRELAÇÕES
10    # "resultados.model.exog" SÃO AS VARIÁVEIS EXPLICATIVAS;
11    correlacao = pd.DataFrame(resultados.model.exog[:, 1:]).corr().to_csv('results/correlacao.csv', header=False, index=False)
12    # ÍNDICE CONDICIONAL (IC)
13    dados_explicativas = resultados.model.exog
14    dados_explicativas = normalize(dados_explicativas, axis=0, norm="l2")
15    dados_transposta = dados_explicativas.transpose()
16    product = np.dot(dados_transposta, dados_explicativas)
17    IC = np.sqrt(np.linalg.eigvals(product).max() / np.linalg.eigvals(product).min())
18    with open('results/IC.txt', 'w') as file:
19        file.write(str(IC))
20    # VIF
21    # "k" É O NÚMERO DE VARIÁVEIS EXPLICATIVAS;
22    lista_vif = [statsmodels.stats.outliers_influence.variance_inflation_factor(resultados.model.exog, i + 1) for i in range(k)]
23    pd.DataFrame(lista_vif).to_csv('results/lista_vif.csv', header=False, index=False)
24    print(f'Correlação, numero condicional e VIFs calculados e exportados com sucesso.')
25 except Exception:
26    print(f'Não existe o caminho "{path_resultados}" ou o modelo da combinação "{combinacao}" não foi calculado.')

```

Quadro 14 – Testes de autocorrelação.

```

1 # --- TESTES DE DURBIN-WATSON E BREUSCH GODFREY ---
2 import os, pathlib, statsmodels.api, statsmodels.stats.diagnostic, statsmodels.stats.stattools
3 from settings import combinacao
4 # CONSIDERANDO UM MODELO CALCULADO EM QUE:
5 path_resultados = pathlib.Path(os.path.join(os.getcwd(), 'modelos_calculados', str(combinacao)))
6 try:
7     resultados = statsmodels.api.load(str(path_resultados))
8
9     # "resultados.resid" SÃO OS RESÍDUOS ESTIMADOS;
10    d_durbin_watson = statsmodels.stats.stattools.durbin_watson(resultados.resid)
11    with open('results/durbin_watson.txt', 'w') as file:
12        file.write(f'Durbin-Watson: {d_durbin_watson}')
13    teste_breusch_godfrey = statsmodels.stats.diagnostic.acorr_breusch_godfrey(resultados)
14    with open('results/breusch_godfrey.txt', 'w') as file:
15        file.write(f'Breusch-Godfrey-Watson - Estatística: {teste_breusch_godfrey[0]}; P-valor:{teste_breusch_godfrey[1]}')
16    print(f'Durbin-Watson e Breusch-Godfrey calculados e exportados com sucesso.')
17
18 except Exception:
19    print(f'Não existe o caminho "{path_resultados}" ou o modelo da combinação "{combinacao}" não foi calculado.')
20

```

Quadro 15 – Testes de linearidade.

```

1 # --- TESTES LINEARIDADE DE LAGRANGE E RAINBOW ---
2 import os, pathlib, statsmodels.api, statsmodels.stats.diagnostic, statsmodels.stats.stattools
3 from settings import combinacao
4 # CONSIDERANDO UM MODELO CALCULADO EM QUE:
5 path_resultados = pathlib.Path(os.path.join(os.getcwd(), 'modelos_calculados', str(combinacao)))
6 try:
7     resultados = statsmodels.api.load(str(path_resultados))
8     # "resultados.resid" SÃO OS RESÍDUOS ESTIMADOS;
9     # "resultados.model.exog" SÃO AS VARIÁVEIS EXPLICATIVAS;
10     teste_linear_lm = statsmodels.stats.diagnostic.linear_lm(resultados.resid, resultados.model.exog)
11     with open('results/linear_lm.txt', 'w') as file:
12         file.write(f'LM - Estatística: {teste_linear_lm[0]}, P-valor: {teste_linear_lm[1]}')
13     teste_linear_rainbow = statsmodels.stats.diagnostic.linear_rainbow(resultados)
14     with open('results/rainbow.txt', 'w') as file:
15         file.write(f'Rainbow - Estatística: {teste_linear_rainbow[0]}, P-valor: {teste_linear_rainbow[1]}')
16     print(f'Testes de Linearidades (LM e Rainbow) calculados e exportados com sucesso.')
17
18 except Exception:
19     print(f'Não existe o caminho "{path_resultados}" ou o modelo da combinação "{combinacao}" não foi calculado.')

```

Quadro 16 – Dividir uma amostra em treinamento e validação.

```

1 # --- SEPARAR AS AMOSTRAS EM TREINAMENTO E VALIDAÇÃO ---
2 import numpy as np, random
3 from settings import lista_id, data_y_rna_tr, data_x_rna_tr, k, pct_separacao, dataframe_identificadores
4 # GARANTIR QUE NA AMOSTRA DE TREINAMENTO HAVERÁ DADOS COM VALORES EXTREMOS (MÁX. E MÍN.)
5 # "lista_dados_extremos" É UMA LISTA COM OS ÍNDICES DOS DADOS QUE CONTENHAM VALORES EXTREMOS;
6 lista_dados_extremos = []
7 # BUSCAR MÁX. E MÍN. PARA Y
8 max_y, min_y = data_y_rna_tr.argmax(), data_y_rna_tr.argmin() # "data_y_rna_tr" SÃO DADOS DAS VARIÁVEL DEPENDENTE
9 lista_dados_extremos.append(max_y)
10 lista_dados_extremos.append(min_y)
11 # BUSCAR MÁX. E MÍN. PARA X
12 for i in range(k): # "k" É O NÚMERO DE VARIÁVEIS INDEPENDENTES.
13     # "data_x_rna_tr" SÃO DADOS DAS VARIÁVEIS INDEPENDENTES
14     max_x, min_x = np.array(data_x_rna_tr[:, i]).argmax(), np.array(data_x_rna_tr[:, i]).argmin()
15     lista_dados_extremos.append(max_x)
16     lista_dados_extremos.append(min_x)
17
18 # RETIRAR ELEMENTOS REPETIDOS (SE HOUVER)
19 lista_dados_extremos = list(set(lista_dados_extremos))
20 # CALCULAR A QUANTIDADE DE ELEMENTOS A SEREM TREINADOS
21 # "lista_id" É UMA LISTA COM IDENTIFICADORES DA AMOSTRA. EXEMPLO: [[0, 'AP_01'], [1, 'AP_02'], [2, 'AP_03'],...]
22 len_train = len(lista_id) - int(pct_separacao * len(lista_id)) # "pct_separacao" É A PORCENTAGEM DEFINIDA PELO USUÁRIO;
23
24 # CRIAR UMA LISTA COM OS ÍNDICES DOS ELEMENTOS PARA TREINAMENTO
25 # INCLUIR NESSA LISTA OS ELEMENTOS EXTREMOS
26 indices_random_train = list(lista_dados_extremos)
27 while len(indices_random_train) < len_train: # ENQUANTO A LISTA NÃO ATINGIR O TAMANHO DEFINIDO;
28     i = random.randint(0, len(lista_id) - 1) # GERAR ÍNDICES RANDOMICAMENTE
29     if i not in indices_random_train:
30         indices_random_train.append(i)
31
32 # CRIAR UMA LISTA COM OS ÍNDICES DOS ELEMENTOS PARA VALIDAÇÃO
33 # USAR APENAS ÍNDICES QUE NÃO ESTÃO NO TREINAMENTO
34 indices_random_val = [i for i in range(0, len(lista_id)) if i not in indices_random_train]
35 # AMOSTRAS DE TREINAMENTO E VALIDAÇÃO
36 amostra_treinamento, amostra_validacao = [], []
37 for contador, item in enumerate(dataframe_identificadores.iloc[:, 1]):
38     if contador in indices_random_train:
39         amostra_treinamento.append(item)
40     if contador in indices_random_val:
41         amostra_validacao.append(item)
42 print(f'Amostra de treinamento:\n{amostra_treinamento}')
43 print(f'Amostra de validação:\n{amostra_validacao}')

```

Quadro 17 – Cálculo das RNA com transformações.

```

1 import pandas as pd
2 import numpy as np
3 import tensorflow
4 from sklearn.metrics import r2_score, mean_squared_error, mean_absolute_error, mean_absolute_percentage_error
5 from sklearn.preprocessing import MinMaxScaler
6 from tensorflow.keras.models import Sequential
7 from tensorflow.keras.layers import Dense
8 from tensorflow.keras.callbacks import EarlyStopping, ReduceLRonPlateau
9 import math
10 import tensorflow_model_optimization as tfmot
11 from tensorflow.python.framework.errors_impl import InvalidArgumentError
12 from settings import lr, seed_value, repeticoes_max, k
13
14 def rna(nn, nh1, lr, epochs, k, otimizadores, criterio, seed_value, NAME):
15
16     seed_value = seed_value
17     np.random.seed(seed_value)
18     tensorflow.random.set_seed(seed_value)
19     modelo_rna = Sequential()
20
21     if otimizadores == 'adam':
22         optimizer = tensorflow.keras.optimizers.Adam(learning_rate=lr)
23
24     if nh1 == 1:
25         modelo_rna.add(Dense(nn, input_shape=(k,), activation='tanh'))
26
27     if nh1 == 2:
28         first_layer_qtd_neuronios = math.ceil(nn / 2)
29         second_layer_qtd_neuronios = nn - first_layer_qtd_neuronios
30         modelo_rna.add(Dense(first_layer_qtd_neuronios, input_shape=(k,),
31                             activation='tanh'))
32         modelo_rna.add(Dense(second_layer_qtd_neuronios, activation='tanh'))
33
34     if nh1 == 3:
35         first_layer_qtd_neuronios = math.ceil(nn / 3)
36         second_layer_qtd_neuronios = math.ceil((nn - first_layer_qtd_neuronios) / 2)
37         third_layer_qtd_neuronios = nn - second_layer_qtd_neuronios - first_layer_qtd_neuronios
38
39         modelo_rna.add(Dense(first_layer_qtd_neuronios, input_shape=(k,),
40                             activation='tanh'))
41         modelo_rna.add(Dense(second_layer_qtd_neuronios, activation='tanh'))
42         modelo_rna.add(Dense(third_layer_qtd_neuronios, activation='tanh'))
43
44     modelo_rna.add(Dense(1, activation='linear'))
45     modelo_rna.compile(loss='mse', optimizer=optimizer, metrics=['mse'])
46     monitoramento = None
47     reduce_lr = ReduceLRonPlateau(monitor='val_loss', factor=0.5,
48                                  patience=8, min_lr=0.0001)
49
50     if criterio == 0: # ES
51
52         monitoramento = [EarlyStopping(patience=10, verbose=1), reduce_lr]
53
54     if criterio == 2: # SEM ES, PR
55
56         monitoramento = None
57
58     if criterio == 1: # PR
59         pruning_schedule = tfmot.sparsity.keras.PolynomialDecay(initial_sparsity=0.50,
60                                                                final_sparsity=0.8,
61                                                                begin_step=0, end_step=epochs)
62
63         modelo_pruned = tfmot.sparsity.keras.prune_low_magnitude(modelo_rna,
64                                                                pruning_schedule=pruning_schedule)
65
66         modelo_pruned.compile(optimizer=optimizer,
67                              loss='mse',
68                              metrics=['mse'])
69
70     # DEFINIR DADOS
71     data_train = pd.read_csv('data/df_treinamento')
72     data_test = pd.read_csv('data/df_validacao')
73
74     X_train = data_train[['AREA', 'WC', 'SUITE', 'VAGA', 'PADRAO', 'IDADE', 'DIST', 'Norte']]
75     y_train = data_train[['VT']]
76
77     X_test = data_test[['AREA', 'WC', 'SUITE', 'VAGA', 'PADRAO', 'IDADE', 'DIST', 'Norte']]
78     y_test = data_test[['VT']]
79
80     # ESCALONAR OS DADOS
81     scaler_x = MinMaxScaler(feature_range=(-1, 1))
82     scaler_x.fit(X_train)

```

Quadro 17 – Cálculo das RNA com transformações.

```

83 X_train_scaled = scaler_x.transform(X_train)
84 X_test_scaled = scaler_x.transform(X_test)
85 scaler_y = MinMaxScaler(feature_range=(-1, 1))
86 scaler_y.fit(y_train)
87 y_train_scaled = scaler_y.transform(y_train)
88 y_test_scaled = scaler_y.transform(y_test)
89
90 np.random.seed(seed_value)
91 tensorflow.random.set_seed(seed_value)
92
93 if criterio != 1:
94     Hist = modelo_rna.fit(X_train_scaled, y_train_scaled, validation_data=(X_test_scaled, y_test_scaled),
95                          callbacks=monitoramento, verbose=2, epochs=epochs)
96
97 if criterio == 1:
98
99     callbacks = [tfmot.sparsity.keras.UpdatePruningStep(), reduce_lr]
100
101     try:
102         Hist = modelo_pruned.fit(X_train_scaled, y_train_scaled, validation_data=(X_test_scaled, y_test_scaled),
103                                 verbose=2, epochs=epochs, callbacks=callbacks)
104     except InvalidArgumentError:
105         return None
106
107     # VALIDAÇÃO
108     y_test_pred_scaled = modelo_rna.predict(X_test_scaled)
109     r2_rna_test = r2_score(np.array(y_test_scaled).flatten(), y_test_pred_scaled.flatten())
110     y_test_pred_inverse = scaler_y.inverse_transform(y_test_pred_scaled)
111     y_test_pred_moeda = np.exp(y_test_pred_inverse)
112     y_test_moeda = np.exp(y_test)
113     mse_test_moeda = mean_squared_error(y_test_moeda, y_test_pred_moeda)
114     mae_test_moeda = mean_absolute_error(y_test_moeda, y_test_pred_moeda)
115     mape_test_moeda = mean_absolute_percentage_error(y_test_moeda, y_test_pred_moeda)
116     with open(f'results/rna_{NAME}.txt', 'w') as f:
117         f.write(f'{NAME} - Validação - R²={r2_rna_test}; RMSE={mse_test_moeda**0.5};'
118               f'MAE={mae_test_moeda}; MAPE={mape_test_moeda}')
119
120 rna(nn=46, nh1=2, lr=lr, epochs=repeticoes_max, k=k, otimizadores='adam', criterio=0, seed_value=seed_value, NAME='RNA-1')
121 rna(nn=112, nh1=2, lr=lr, epochs=repeticoes_max, k=k, otimizadores='adam', criterio=0, seed_value=seed_value, NAME='RNA-2')
122 rna(nn=119, nh1=3, lr=lr, epochs=repeticoes_max, k=k, otimizadores='adam', criterio=0, seed_value=seed_value, NAME='RNA-3')
123 rna(nn=200, nh1=3, lr=lr, epochs=repeticoes_max, k=k, otimizadores='adam', criterio=1, seed_value=seed_value, NAME='RNA-4')
124 rna(nn=105, nh1=1, lr=lr, epochs=repeticoes_max, k=k, otimizadores='adam', criterio=1, seed_value=seed_value, NAME='RNA-5')
125 rna(nn=98, nh1=3, lr=lr, epochs=repeticoes_max, k=k, otimizadores='adam', criterio=1, seed_value=seed_value, NAME='RNA-6')

```

Quadro 18 – Cálculo das RNA sem transformações com Early Stopping.

```

1 import numpy as np
2 import pandas as pd
3 import tensorflow
4 from sklearn.metrics import mean_absolute_error, mean_absolute_percentage_error, mean_squared_error, r2_score,
5 from sklearn.preprocessing import MinMaxScaler
6 from settings import k, seed_value
7 from tensorflow import keras
8 from tensorflow.keras import layers
9
10 # DEFINIR DADOS
11 data_train = pd.read_csv("data/df_treinamento.csv")
12 data_val = pd.read_csv("data/df_validacao.csv")
13
14 x_train = data_train[
15     ["AREA", "WC", "VAGA", "SUITE", "PADRAO", "IDADE", "DIST", "Norte"]
16 ]
17 y_train = data_train[["VT"]]
18
19 x_val = data_val[
20     ["AREA", "WC", "VAGA", "SUITE", "PADRAO", "IDADE", "DIST", "Norte"]
21 ]
22 y_val = data_val[["VT"]]
23
24 # Escalonar
25 scaler_x = MinMaxScaler()
26 x_train_scaled = scaler_x.fit_transform(x_train)
27 x_val_scaled = scaler_x.transform(x_val)
28
29 scaler_y = MinMaxScaler()
30 y_train_scaled = scaler_y.fit_transform(y_train)
31 y_val_scaled = scaler_y.transform(y_val)
32
33 EPOCHS = 2000

```

Quadro 18 – Cálculo das RNA sem transformações com Early Stopping.

```

34
35 def build_and_fit_model_early_stopping(parameters):
36     np.random.seed(seed_value)
37     tensorflow.random.set_seed(seed_value)
38
39     model = keras.Sequential()
40
41     num_layers = parameters['num_layers']
42     units = parameters['units']
43     activation = parameters['activation']
44     learning_rate = parameters['learning_rate']
45     optimizer = parameters['optimizer']
46     kernel_initializer = parameters['kernel_initializer']
47     NAME = parameters['NAME']
48
49     model.add(layers.Dense(units=units, input_shape=(k,), activation=activation, kernel_initializer=kernel_initializer))
50
51     for i in range(num_layers - 1):
52         model.add(layers.Dense(units=units, activation=activation, kernel_initializer=kernel_initializer))
53
54     model.add(layers.Dense(units=1, activation='linear', kernel_initializer=kernel_initializer))
55
56     if optimizer == 'Adam':
57         optimizer = keras.optimizers.Adam(learning_rate=learning_rate)
58     elif optimizer == 'RMSprop':
59         optimizer = keras.optimizers.RMSprop(learning_rate=learning_rate)
60     elif optimizer == 'Nadam':
61         optimizer = keras.optimizers.Nadam(learning_rate=learning_rate)
62
63     model.compile(optimizer=optimizer,
64                   loss='mean_squared_error',
65                   metrics=['mean_squared_error'])
66
67     Hist = model.fit(x=x_train_scaled,
68                    y=y_train_scaled,
69                    epochs=EPOCHS,
70                    validation_data=(x_val_scaled, y_val_scaled),
71                    callbacks=[keras.callbacks.EarlyStopping(monitor='val_mean_squared_error', patience=10),
72                             tensorflow.keras.callbacks.ReduceLROnPlateau(monitor='val_mean_squared_error',
73                                 factor=0.5,
74                                 patience=8,
75                                 min_lr=0.0001)
76                    ])
77
78     # validacao-----
79     y_val_pred_scaled = model.predict(x_val_scaled)
80     y_val_pred = scaler_y.inverse_transform(y_val_pred_scaled)
81     y_val_inverse = scaler_y.inverse_transform(y_val_scaled)
82
83     r2 = r2_score(y_val_inverse, y_val_pred)
84     rmse = mean_squared_error(y_val_inverse, y_val_pred, squared=False)
85     mape = mean_absolute_percentage_error(y_val_inverse, y_val_pred)
86     mae = mean_absolute_error(y_val_inverse, y_val_pred)
87
88     with open(f'results/rna_{NAME}.txt', 'w') as f:
89         f.write(f'{NAME} - Validação - R²={r2}; RMSE={rmse}; '
90               f'MAPE={mape}; MAE={mae}')
91
92 combinations_generator = [{'num_layers': 3, 'units': 145, 'activation': 'relu', 'learning_rate': 0.001, 'optimizer': 'Adam',
93                           'kernel_initializer': 'glorot_normal', 'NAME': 'RNA-7'},
94                           {'num_layers': 3, 'units': 65, 'activation': 'relu', 'learning_rate': 0.001, 'optimizer': 'RMSprop',
95                           'kernel_initializer': 'glorot_normal', 'NAME': 'RNA-8'},
96                           {'num_layers': 3, 'units': 65, 'activation': 'relu', 'learning_rate': 0.001, 'optimizer': 'Nadam',
97                           'kernel_initializer': 'glorot_normal', 'NAME': 'RNA-9'}
98                           ]
99
100
101
102 for iteration, combination in enumerate(combinations_generator):
103     build_and_fit_model_early_stopping(combination)

```

Quadro 19 – Cálculo das RNA sem transformações com *Pruning*.

```

1 import numpy as np
2 import pandas as pd
3 import tensorflow
4 import tensorflow_model_optimization as tfmot
5 from sklearn.metrics import mean_absolute_error, mean_absolute_percentage_error, mean_squared_error, r2_score
6 from sklearn.preprocessing import MinMaxScaler
7 from settings import k, seed_value
8 from tensorflow import keras
9 from tensorflow.keras import layers
10
11 # DEFINIR DADOS
12 data_train = pd.read_csv("data/df_treinamento.csv")
13 data_val = pd.read_csv("data/df_validacao.csv")
14
15 x_train = data_train[
16     ["AREA", "WC", "VAGA", "SUITE", "PADRAO", "IDADE", "DIST", "Norte"]
17 ]
18 y_train = data_train[["VT"]]
19
20 x_val = data_val[
21     ["AREA", "WC", "VAGA", "SUITE", "PADRAO", "IDADE", "DIST", "Norte"]
22 ]
23 y_val = data_val[["VT"]]
24
25 # Escalonar
26 scaler_x = MinMaxScaler()
27 x_train_scaled = scaler_x.fit_transform(x_train)
28 x_val_scaled = scaler_x.transform(x_val)
29
30 scaler_y = MinMaxScaler()
31 y_train_scaled = scaler_y.fit_transform(y_train)
32 y_val_scaled = scaler_y.transform(y_val)
33
34 EPOCHS = 2000
35
36 def build_and_fit_model_early_stopping(parameters):
37     np.random.seed(seed_value)
38     tensorflow.random.set_seed(seed_value)
39
40     model = keras.Sequential()
41
42     num_layers = parameters['num_layers']
43     units = parameters['units']
44     activation = parameters['activation']
45     learning_rate = parameters['learning_rate']
46     optimizer = parameters['optimizer']
47     kernel_initializer = parameters['kernel_initializer']
48     NAME = parameters['NAME']
49
50     model.add(layers.Dense(units=units, input_shape=(k,), activation=activation, kernel_initializer=kernel_initializer))
51
52     for i in range(num_layers - 1):
53         model.add(layers.Dense(units=units, activation=activation, kernel_initializer=kernel_initializer))
54
55     model.add(layers.Dense(units=1, activation='linear', kernel_initializer=kernel_initializer))
56
57     if optimizer == 'Adam':
58         optimizer = keras.optimizers.Adam(learning_rate=learning_rate)
59     elif optimizer == 'RMSprop':
60         optimizer = keras.optimizers.RMSprop(learning_rate=learning_rate)
61     elif optimizer == 'Nadam':
62         optimizer = keras.optimizers.Nadam(learning_rate=learning_rate)
63
64     model.compile(optimizer=optimizer,
65                 loss='mean_squared_error',
66                 metrics=['mean_squared_error'])
67
68     pruning_schedule = tfmot.sparsity.keras.PolynomialDecay(initial_sparsity=0.50,
69                                                            final_sparsity=0.8,
70                                                            begin_step=0, end_step=EPOCHS)
71
72     modelo_pruned = tfmot.sparsity.keras.prune_low_magnitude(model,
73                                                             pruning_schedule=pruning_schedule)
74
75     modelo_pruned.compile(optimizer=optimizer,
76                          loss='mean_squared_error',
77                          metrics=['mean_squared_error'])
78
79     try:
80         Hist = modelo_pruned.fit(x=x_train_scaled,
81                                 y=y_train_scaled,
82                                 epochs=EPOCHS,

```

Quadro 19 – Cálculo das RNA sem transformações com *Pruning*.

```

31         validation_data=(x_val_scaled, y_val_scaled),
32         callbacks=[tfmot.sparsity.keras.UpdatePruningStep(),
33                   tensorflow.keras.callbacks.ReduceLROnPlateau(
34                       monitor='val_mean_squared_error',
35                       factor=0.5,
36                       patience=8,
37                       min_lr=0.0001)
38                   ]
39     )
40 except Exception:
41     return
42
43 # validacao-----
44 last_val_mean_squared_error = Hist.history['val_mean_squared_error'][-1]
45 y_val_pred_scaled = modelo_pruned.predict(x_val_scaled)
46 y_val_pred = scaler_y.inverse_transform(y_val_pred_scaled)
47 y_val_inverse = scaler_y.inverse_transform(y_val_scaled)
48
49 r2 = r2_score(y_val_inverse, y_val_pred)
50 rmse = mean_squared_error(y_val_inverse, y_val_pred, squared=False)
51 mape = mean_absolute_percentage_error(y_val_inverse, y_val_pred)
52 mae = mean_absolute_error(y_val_inverse, y_val_pred)
53
54 with open(f'results/rna_{NAME}.txt', 'w') as f:
55     f.write(f'{NAME} - Validação - R²={r2}; RMSE={rmse}; '
56           f'MAPE={mape}; MAE={mae}')
57
58 combinations_generator = [{'num_layers': 2, 'units': 85, 'activation': 'relu', 'learning_rate': 0.0005, 'optimizer': 'RMSprop',
59                           'kernel_initializer': 'glorot_normal', 'NAME': 'RNA-10'},
60
61                           {'num_layers': 3, 'units': 45, 'activation': 'relu', 'learning_rate': 0.0005, 'optimizer': 'Adam',
62                           'kernel_initializer': 'truncated_normal', 'NAME': 'RNA-11'},
63
64                           {'num_layers': 3, 'units': 65, 'activation': 'relu', 'learning_rate': 0.001, 'optimizer': 'Adam',
65                           'kernel_initializer': 'glorot_normal', 'NAME': 'RNA-12'}
66                          ]
67
68 for iteration, combination in enumerate(combinations_generator):
69     build_and_fit_model_early_stopping(combination)

```

Quadro 20 – Arquivo de configurações.

```

1 import numpy as np, pandas as pd
2
3 # --- VARIÁVEIS ---
4 # "data_y" REFERE-SE À VARIÁVEL DEPENDENTE;
5 # "data_x" REFERE-SE ÀS VARIÁVEIS INDEPENDENTES;
6 # "dataframe_identificadores" É A AMOSTRA COM OS RESPECTIVOS IDENTIFICADORES DOS IMÓVEIS;
7 # "n" É O NÚMERO DE DADOS DA AMOSTRA;
8 # "k" É O NÚMERO DE VARIÁVEIS EXPLICATIVAS;
9 # "r2min" É O MENOR R² ACEITO PELO USUÁRIO;
10 # "maxcomb" É O NÚMERO MÁXIMO DE COMBINAÇÕES GRAVADAS;
11 # "lista_selected_tr" GUARDA AS TR QUE O USUÁRIO SELECIONOU NA INTERFACE GRÁFICA;
12 # "lista_tr_indice" É A LISTA QUE MOSTRA QUAIS TR CADA VARIÁVEL ACEITA;
13 # "combinacao" É A COMBINAÇÃO ADOTADA;
14 # "data_y_rna_tr", "data_x_rna_tr" SÃO OS DADOS USADOS NA RNA;
15 # "pct_separacao" É A PORCENTAGEM DEFINIDA PELO USUÁRIO PARA SEPARAÇÃO DA AMOSTRA;
16 # "lista_id" É UMA LISTA COM OS PARES (ÍNDICE, IDENTIFICADOR). EXEMPLO: [[0, 'AP_01'], [1, 'AP_02'], ...];
17 # "indices_random_train" E "indices_random_val" SÃO OS ÍNDICES DAS AMOSTRAS DE TREINAMENTO E VALIDAÇÃO USADAS;
18 # "repeticoes_max" É A QUANTIDADE MÁXIMA DE REPETIÇÕES/TREINAMENTOS;
19 # "Lr" É A TAXA DE APRENDIZAGEM;
20 # "seed_value" É UMA CONSTANTE QUE PERMITE QUE A RNA SEJA TREINADA REPETIDAS VEZES DO MESMO MODO;
21
22 data_y = np.load('data/data_y.npy')
23 data_x = np.load('data/data_x.npy')
24 dataframe_identificadores = pd.read_csv('data/tabela_com_identificadores')
25 n = 156 # n do treino
26 k = 8
27 r2min = 0.90
28 maxcomb = 50
29 lista_selected_tr = [1, 4, 5]
30 lista_tr_indice = [[1, 4, 5], [1, 4, 5], [1, 4, 5], [1], [1], [1, 4, 5], [1, 4, 5], [1, 4, 5], [1, 4, 5]]
31 combinacao = (4, 4, 4, 1, 1, 4, 1, 4, 1)
32 data_y_rna_tr = np.load('data/data_y_rna_tr.npy')
33 data_x_rna_tr = np.load('data/data_x_rna_tr.npy')
34
35 # DEFINIR DADOS
36 data_train = pd.read_csv("data/df_treinamento.csv")
37 data_val = pd.read_csv("data/df_validacao.csv")

```

Quadro 20 – Arquivo de configurações.

```
38 pct_separacao = 0.25
39 lista_id = [[contador, identificador] for contador, identificador in enumerate(dataframe_identificadores.iloc[:, 1])]
40 indices_random_train = [1, 2, 167, 41, 75, 14, 147, 149, 87, 55, 89, 122, 28, 132, 153, 163, 18,
41                       113, 107, 100, 136, 133, 102, 194, 156, 79, 131, 118, 6, 58, 48, 139, 97,
42                       138, 103, 61, 204, 93, 82, 161, 59, 12, 24, 200, 185, 173, 151, 73, 43, 106,
43                       56, 33, 115, 31, 57, 10, 152, 195, 46, 207, 105, 54, 99, 66, 134, 34, 175,
44                       16, 85, 120, 3, 119, 72, 78, 159, 15, 130, 183, 171, 95, 64, 121, 53, 77, 26,
45                       168, 184, 8, 110, 135, 76, 125, 62, 74, 29, 71, 63, 150, 129, 127, 52, 170,
46                       179, 92, 80, 197, 51, 49, 38, 69, 25, 20, 141, 181, 50, 191, 146, 42, 39, 203,
47                       40, 172, 114, 91, 162, 182, 186, 169, 98, 35, 148, 111, 206, 126, 187, 0, 68,
48                       188, 9, 164, 137, 104, 36, 44, 70, 180, 27, 96, 30, 47, 205, 140, 19, 193, 84, 65]
49 indices_random_val = [4, 5, 7, 11, 13, 17, 21, 22, 23, 32, 37, 45, 60, 67, 81, 83, 86, 88, 90, 94,
50                       101, 108, 109, 112, 116, 117, 123, 124, 128, 142, 143, 144, 145, 154, 155, 157,
51                       158, 160, 165, 166, 174, 176, 177, 178, 189, 190, 192, 196, 198, 199, 201, 202]
52
53 repeticoes_max = 5000
54 lr = 0.001
55 seed_value = 10
```