



Revista Brasileira de Geografia Física

Homepage: <https://periodicos.ufpe.br/revistas/rbgf>



Reamostragem em Redes Neurais: Uma Abordagem Alternativa aos Métodos Tradicionais de Interpolação Espacial para Modelagem de Superfícies

Wilker de Jesus Silva¹, Marcelo Tomio Matsuoka², Vinicius Francisco Rofatto³

¹Mestrando do Programa de Pós-Graduação em Agricultura e Informações Geoespaciais da Universidade Federal de Uberlândia. Rodovia LMG-746, km 1, Monte Carmelo - MG, CEP 38500-000. ORCID: <https://orcid.org/0009-0004-1529-7369>. E-mail: wilkerdejesus@gmail.com. ²Professor do Programa de Pós-Graduação em Agricultura e Informações Geoespaciais da Universidade Federal de Uberlândia. Rodovia LMG-746, km 1, Monte Carmelo - MG, CEP 38500-000. ORCID: <http://orcid.org/0000-0002-2630-522X>. E-mail: tomiomatsuoka@gmail.com. ³Professor do Programa de Pós-Graduação em Agricultura e Informações Geoespaciais da Universidade Federal de Uberlândia. Rodovia LMG-746, km 1, Monte Carmelo - MG, CEP 38500-000. ORCID: <https://orcid.org/0000-0003-1453-7530>. E-mail: vfrofatto@gmail.com.

Artigo recebido em 09/02/2024 e aceito em 12/08/2024

RESUMO

As Redes Neurais Artificiais (RNAs) têm sido empregadas em diversas aplicações, destacando-se como um poderoso recurso para analisar dados e resolver problemas em diversas áreas do conhecimento. Na modelagem de superfícies, as RNAs desempenham o papel de um método de interpolação espacial. Entretanto ao empregar as redes neurais, as previsões dos valores de altitude não vêm acompanhadas de suas correspondentes incertezas. Nesta contribuição, fornecemos o aprimoramento das estimativas de RNAs em uma abordagem inovadora, aplicando um método de reamostragem para prever intervalos de altitudes em vez de uma única estimativa, como é comumente realizado por técnicas convencionais. Uma rede Perceptron de Múltiplas Camadas (MLP) foi usada para prever os intervalos de altitudes com base nas coordenadas dos pontos coletados em campo por Posicionamento Cinemático em Tempo Real (RTK). A rede foi treinada e validada usando o método de reamostragem *Repeated Leave-One-Out Cross-Validation* (RLOOCV), uma extensão do clássico método *Leave-One-Out Cross-Validation* (LOOCV), que ao realizar diversas iterações permite a captura da aleatoriedade associada à rede neural, incluindo fatores como arquitetura, inicialização e procedimento de aprendizado. As métricas de desempenho revelaram resultados satisfatórios na estimativa de altitudes, apresentando valores de Raiz do Erro Quadrático Médio (RMSE) consistentes, com a média global de RMSE, RMSE máximo e RMSE mínimo de 0,081 m ($\pm 0,002$), 0,520 m e 0,027 m, respectivamente. Embora essa metodologia tenha apresentado um desempenho satisfatório, a análise espacial revelou desafios na generalização em áreas vertentes, de talvez e com variações abruptas na inclinação do terreno.

Palavras-chave: rede neural artificial, divisão de dados, reamostragem, leave-one-out, predição de intervalo.

Resampling in Neural Networks: An Alternative Approach to Traditional Methods of Spatial Interpolation for Surface Modeling

ABSTRACT

Artificial Neural Networks (ANNs) have been employed in a wide range of applications, representing a highly potential mechanism for data analysis and problem-solving across various fields of knowledge. In surface modeling, ANNs play the role of a spatial interpolation method. However, when employing neural networks, predictions of altitude values do not provide their corresponding uncertainties. In this contribution, this research provides an enhancement of ANN predictions through an innovative approach by applying a resampling method to predict altitude intervals instead of a single estimate, as commonly done by conventional techniques. A Multilayer Perceptron (MLP) network was used to predict altitude intervals based on the coordinates of field-collected points using Real Time Kinematic (RTK) positioning. The network was trained and validated using the *Repeated Leave-One-Out Cross-Validation* (RLOOCV) resampling method, an extension of the classic *Leave-One-Out Cross-Validation* (LOOCV) method, which, through multiple iterations, captures the randomness associated with the neural network, including factors like architecture, initialization, and learning procedure. The performance metrics demonstrated satisfactory results in altitude estimation, presenting consistent Root Mean Square Error (RMSE) values, with the global average RMSE, maximum RMSE and minimum RMSE of 0.081 m (± 0.002), 0.520 m e 0.027 m, respectively. Although this methodology demonstrated satisfactory performance, spatial analysis revealed challenges in generalizing to slope, valley, and areas with abrupt variations in terrain inclination.

Keywords: artificial neural network, data splitting, resampling, leave-one-out, interval prediction.

Introdução

As Redes Neurais Artificiais (RNAs) são modelos computacionais amplamente usados que se inspiram na estrutura e função do sistema nervoso dos organismos vivos, replicando o processo de aprendizado observado em redes neurais biológicas (Aggarwal, 2018; Silva et al., 2016). As RNAs se encontram inseridas no conjunto dos sistemas inteligentes, também conhecidos como sistemas conexionistas ou de inteligência computacional (Jang et al., 1997; Yager e Zadeh, 1992).

Uma das principais características das Redes Neurais Artificiais reside na capacidade de delinear sistemas não-lineares, adquirindo conhecimento dos comportamentos associados a informações como medidas, amostras ou padrões (Silva et al., 2016).

As RNAs possuem uma gama diversificada de aplicações, sendo um mecanismo de alto potencial para análise de dados e resolução de problemas advindos de diferentes áreas do conhecimento, em especial, das geociências e engenharias, a saber:

- Transformação de Coordenadas entre sistemas de referência distintos (Abbas et al., 2022; Konakoglu et al., 2016);
- Interpolação de Dados, por exemplo, aprimoramento de modelos digitais de elevação (Y. Li et al., 2023; R. Zhang et al., 2021; Y. Zhang et al., 2022), modelagem da superfície geoidal (Cakir e Yilmaz, 2014; Kavzoglu e Saka, 2005; Stopar et al., 2006), determinação do campo de velocidade crustal (Farolfi e Del Ventisette, 2016; Moghtased-Azar e Zaletnyik, 2009; Özarpacı et al., 2024), aprimoramento das previsões do nível das marés (Jian et al., 2024);
- Monitoramento Geodésico de Estruturas (Kaloop e Hu, 2015; Kaloop e Kim, 2014; Lorenz e Petryna, 2024; Sun et al., 2016);
- Agricultura de Precisão, como mapeamento e classificação de solos (Akwensi et al., 2021; Coelho et al., 2021), predição de propriedades químicas do solo (Khaleedian e Miller, 2020; Pacci et al., 2024), avaliação da população de bactérias do solo (Ebrahimi et al., 2017); e
- Monitoramento de Desastres Ambientais, tais como avaliação de risco de queda de rochas (Fanos et al., 2020), avaliação de vulnerabilidade de áreas urbanas a terremotos (Afsari et al., 2023), identificação de áreas suscetíveis a deslizamentos de terra (Chen et

al., 2017), aprimoramento de sistemas de alerta precoce de tsunamis (Fuso et al., 2024).

No contexto da construção e avaliação de modelos de redes neurais, existem questões cruciais relacionadas à generalização, que se refere à capacidade de um modelo apresentar um bom desempenho com dados anteriormente não observados. O foco reside na busca pelo equilíbrio entre o viés e a variância como meio de compensação. Um ponto essencial é a necessidade de evitar sobreajustes (*overfitting*), que ocorre quando a rede apresenta um desempenho muito bom nos dados usados para treinamento, mas falha em generalizar com eficiência para dados de teste que não foram previamente observados (Aggarwal, 2018; Goodfellow et al., 2016). Tal problema pode ser identificado se for percebido um desempenho quase perfeito no conjunto de treinamento, porém um desempenho muito abaixo do esperado em outro conjunto de avaliação não visto (Jiang, 2021).

Ao examinar a existência de sobreajustes, é necessário introduzir um conceito fundamental na avaliação em ciência de dados: dados de retenção (*hold-out data*). Se faz necessário "reservar" alguns dados para os quais conhecemos o valor da variável alvo, mas que não serão usados para construir o modelo. A criação de dados de retenção pode ser compreendida como a criação de um "teste de laboratório" de desempenho de generalização (Provost e Fawcett, 2013).

Utilizando o método *Hold-out*, procedemos à segmentação do nosso conjunto de dados original em dois conjuntos distintos: um conjunto de treinamento e outro de teste. Uma abordagem mais aprimorada para aplicar o método *Hold-out* na seleção de modelos envolve a subdivisão dos dados em três partes: conjunto de treinamento, conjunto de validação e conjunto de teste. O conjunto de treinamento é geralmente empregado para a estimativa dos parâmetros desconhecidos da rede (valores internos), como pesos e vieses (*biases*) em uma rede neural de regressão *feedforward* (alimentação à frente). O conjunto de validação auxilia na tomada de decisões relacionadas a como treinar a rede e quando interromper o processo de treinamento. Ter um conjunto de teste que o modelo não tenha visto durante o treinamento e a seleção do modelo permite obter uma estimativa menos enviesada de sua capacidade de generalização para novos dados (Raschka e Mirjalili, 2017; Rodrigues et al., 2022; Rofatto et al., 2023).

Apesar de o método *Hold-out* ser uma ótima alternativa com a função de evitar o excesso de treinamento, existem alguns problemas, como mencionado por Reitermanová (2010): (1) este método depende da divisão adequada dos dados, e sendo a amostragem aleatória simples a mais comumente usada, pode não ser adequada para todos os tipos de conjuntos de dados; e (2) a divisão escolhida do conjunto de dados afeta fortemente a qualidade do modelo final, e estando o conjunto de dados mal dividido, os subconjuntos de dados podem não cobrir suficientemente os dados, levando ao aumento da variância. Nesses casos, a validação cruzada *K-fold* (*K-fold Cross-Validation*), que usa uma combinação de mais testes, pode ser uma alternativa mais confiável.

A validação cruzada *K-fold* é um método usado para estimativa de precisão e seleção de modelos em aprendizado de máquina que envolve a divisão do conjunto de dados em k subconjuntos ou dobras de tamanhos iguais, onde k é um número inteiro positivo. O modelo é treinado em $k - 1$ subconjuntos e testado no subconjunto restante, e esse processo é repetido k vezes, com cada subconjunto servindo como conjunto de teste uma vez. A média do desempenho do modelo é então calculada ao longo das k iterações para obter uma estimativa de sua precisão (Kohavi, 1995). O método *K-fold* fornece uma avaliação mais abrangente do desempenho do modelo, testando-o em diferentes subconjuntos de dados, capturando a capacidade do modelo de generalizar para dados não vistos (Yadav e Shukla, 2016).

Ambos os métodos *Hold-out* e *K-fold* devem ser utilizados com repetições para garantir estimativas de desempenho mais robustas e confiáveis. Conforme destacado por Yadav e Shukla (2016), para obter uma estimativa confiável de desempenho, uma grande quantidade de estimativas é usualmente preferível. Para aumentar ainda mais o número dessas estimativas, a validação cruzada *K-fold* é executada diversas vezes, de forma que os dados são reorganizados e reestruturados antes de cada rodada. Hilborn et al. (2012) discute que a validação cruzada repetida de retenção (*Repeated Hold-out Cross-Validation*) representa uma abordagem imparcial na validação de modelos para grandes conjuntos de dados. O uso de amostras consideráveis e a realização de numerosas iterações minimiza o viés associado à validação de retenção, incluindo a variação potencial nas medidas de desempenho, que dependem de como as divisões de treinamento e teste foram feitas.

Contudo, tanto para os métodos *Hold-out* quanto para o *K-fold*, a divisão de dados ainda representa um desafio. A questão central que surge é como particionar os dados de uma maneira que garanta a predição mais confiável possível. Embora a validação cruzada *K-fold* apresente vantagens sobre o método *Hold-out*, a incerteza persiste sobre como escolher os k subconjuntos distintos, e a possível existência de diferença de tamanhos entre esses subconjuntos pode comprometer o equilíbrio da validação cruzada. Diante deste cenário, a abordagem do método de reamostragem *Leave-One-Out Cross-Validation* (validação cruzada deixar um de fora) surge como uma promissora alternativa.

Por definição, entende-se métodos de reamostragem como técnicas estatísticas que reutilizam os dados da amostra para fins de análise estatística, dispensando a necessidade de assumir suposições específicas sobre a forma da distribuição dos dados. Ao contrário dos métodos não paramétricos que se baseiam apenas na ordem dos dados, os métodos de reamostragem levam em consideração informações sobre os próprios valores dos dados. Eles são úteis para estimar a variância e/ou viés de um estimador, criar intervalos de confiança ou predição e realizar testes estatísticos relacionados ao parâmetro que está sendo estimado (Chernick, 2012). Dentre os métodos de reamostragem mais comumente utilizados, podemos citar a validação cruzada (*Cross-Validation*), que segundo James et al. (2013) pode ser usada para estimar o erro de teste associado a um determinado método de aprendizagem estatística, a fim de avaliar o seu desempenho ou para selecionar o nível apropriado de flexibilidade.

O procedimento de reamostragem *Jackknife* (canivete), também conhecido como *Leave-One-Out Cross-Validation* (LOOCV) na linguagem de aprendizado de máquina, foi inicialmente proposto por Quenouille (1949) como uma técnica de validação cruzada não paramétrica destinada a estimar e reduzir o viés de um estimador associado a um parâmetro populacional. Posteriormente, Tukey (1958) expandiu seu uso para incluir a estimativa de variância. A principal motivação por trás da aplicação do método *Jackknife* é a redução do viés nas estimativas, a obtenção de uma predição imparcial (ou seja, um efeito aleatório) e a minimização de sobreajustes, como destacado por Abdi e Williams (2010).

Este método oferece a vantagem de fornecer estimativas imparciais ou com viés reduzido do parâmetro, utilizando apenas uma

amostra por vez. Além disso, métodos estatísticos de reamostragem, como o *Jackknife*, são baseados apenas no tamanho da amostra de dados observados e podem aprimorar a precisão das estimativas de parâmetros no processamento de dados geodésicos (Wang e Yu, 2020).

Os autores Rodrigues et al. (2022) e Rofatto et al. (2023) denominaram o método aplicado em suas pesquisas como *Delete-1 Jackknife*, porém há uma diferença sutil entre o LOOCV e o *Delete-1 Jackknife*: o LOOCV calcula as estatísticas a partir dos dados de teste, enquanto o *Delete-1 Jackknife* calcula uma estatística a partir dos dados de treinamento. Em outras palavras, o LOOCV calcula as estatísticas nas amostras que foram deixadas de fora, diferentemente do *Delete-1 Jackknife* que é uma técnica na qual uma estatística é calculada exclusivamente a partir das amostras salvas.

Segundo Abdi e Williams (2010), o termo "*jackknife*" refere-se a duas técnicas relacionadas, mas diferentes, e isso às vezes é fonte de confusão. O método de reamostragem *Jackknife* (ou *Delete-1 Jackknife*) está fortemente relacionado com o método *Bootstrap*, que é atualmente a principal técnica para a estimativa computacional de parâmetros populacionais. Ou seja, *Jackknife* é frequentemente uma aproximação linear do *Bootstrap*. Um método um pouco diferente, denominado *Leave-One-Out*, também chamado *Jackknife*, é utilizado para avaliar a qualidade da previsão de modelos computacionais construídos para prever o valor de variáveis dependentes a partir de um conjunto de variáveis independentes. A primeira técnica (*Delete-1 Jackknife*) estima parâmetros populacionais e seus erros padrão, enquanto a segunda técnica (*Leave-One-Out*) avalia o desempenho de generalização de modelos preditivos.

Extensão do Método de Reamostragem *Leave-One-Out Cross-Validation* (LOOCV) para Predição em RNAs

Em termos gerais, a quantidade de permutações potenciais para validação cruzada independente, denotada como C_n^p , e a quantidade de iterações para cada predição do ponto (R_n^p), são exibidas sequencialmente:

$$C_n^p = \binom{n}{p} = \frac{n!}{(n-p)!p!} \quad (1)$$

$$R_n^p = C_n^p \times \left(\frac{p}{n}\right) \quad (2)$$

onde n representa o tamanho do conjunto de dados com o qual se está trabalhando e p representa o número de pontos de amostra que foram removidos da amostra disponível para fins de validação. Isso implica que validamos o subconjunto de tamanho p , e em seguida, passamos a treinar as observações restantes ($n - p$) de uma vez. O tamanho do subconjunto d é selecionado entre todas as observações sem substituição. Para ilustrar, se $n = 20$ e $p = 2$, teríamos $C_{20}^2 = 190$ subconjuntos independentes de validação cruzada, com $R_{20}^2 = 19$ iterações independentes para cada ponto de predição. Essa técnica específica é comumente denominada de *Leave-P-Out Cross-Validation* (LPOCV), uma versão mais geral do *Leave-One-Out Cross-Validation* (Celisse, 2008; Efron, 1982; Shao e Tu, 1995; Wang e Yu, 2020; P. Zhang, 1993).

O *Leave-One-Out Cross-Validation* é um caso especial, tomando $p = 1$. No método LOOCV, um conjunto de dados é excluído da amostra a cada vez, e as amostras restantes de tamanho $n - 1$ são usadas para treinar o modelo. O modelo é validado testando-o no exemplo deixado de fora. O experimento é repetido por um total de n vezes, cada vez deixando de fora um exemplo diferente para validação (Haykin, 1998; Wang e Yu, 2020).

Por exemplo, quando $n = 100$, a utilização da técnica LOOCV resultaria em 100 validações cruzadas. Cada observação seria estimada apenas uma vez, o que significa $C_{100}^1 = 100$ e $R_{100}^1 = 1$.

Optamos por aproveitar a ideia de fazer várias realizações (tentativas) de *Leave-One-Out Cross-Validation* para avaliar a distribuição de probabilidade da predição de intervalo em vez da clássica predição pontual. Para ilustrar isso, vamos considerar um cenário em que o tamanho do conjunto de dados seja $n = 20$ e o número de tentativas (indicado como t) seja $t = 100$. Nesse contexto, o número de preditores da rede neural será equivalente a $n \times t$. Isso produziria um total de 2.000 preditores de rede neural, fornecendo 100 predições para cada ponto de teste individualmente. Essa abordagem nos permite capturar a aleatoriedade associada à própria rede neural, incluindo fatores como arquitetura, inicialização e procedimento de aprendizado. Nos referimos a esse método como *Repeated Leave-One-Out Cross-Validation* (RLOOCV).

De modo a tornar claro a metodologia empregada ao método RLOOCV, segue uma sequência resumida dos passos necessários a realização do procedimento:

1. O conjunto de dados disponível de tamanho n é particionado em dois componentes distintos: o conjunto de treinamento e o conjunto de testes.
2. O treinamento do modelo será realizado utilizando $(n - 1)$ observações e avaliado empregando a observação restante. Para ilustrar, suponha que x represente a entrada enquanto y represente a variável de saída dentro da rede neural. Assim, teríamos como conjunto de treinamento $\{(x_1, y_1), (x_2, y_2), \dots, (x_{k-1}, y_{k-1}), (x_{k+1}, y_{k+1}), \dots, (x_n, y_n)\}$ e o desempenho do modelo seria avaliado usando a observação restante (x_k, y_k) .
3. O procedimento acima mencionado (2) é repetido n vezes, cada vez excluindo uma observação diferente, até que todas as observações tenham sido excluídas.
4. O procedimento descrito nas etapas 2 e 3 é repetido em um número específico de vezes, indicado por t . Por exemplo, se n for igual a 20 e o número de tentativas (t) for definido como 100, isso resultará em $20 \times 100 = 2.000$ preditores de rede neural. Consequentemente, haverá 100 previsões para cada ponto de teste individual. Assim o número de preditores da rede neural será $n \times t$.

Neste estudo, aplicamos o método de reamostragem RLOOCV, proposto por Rofatto et al. (2023). Trata-se de uma nova abordagem para modelagem da superfície do terreno com o uso das RNAs para a predição de intervalos de altitudes em vez de fornecer apenas uma predição pontual. A

possibilidade da realização de diversas tentativas no procedimento de reamostragem *Leave-One-Out Cross-Validation* (LOOCV) nos permite mensurar o desempenho da predição para cada ponto de teste individual em suas diferentes localizações espaciais, bem como avaliar a precisão e a incerteza associadas a essa predição. As métricas de desempenho, tais como Raiz do Erro Quadrático Médio (RMSE), Erro Médio Absoluto (MAE) e Desvio Padrão Amostral (s), foram atribuídas a cada ponto levantado individualmente em campo, com coordenadas E , N e h conhecidas.

Material e métodos

Área de Estudo

Aplicamos uma área de testes de 9.896 m², situada na zona rural do município de Moiporá, estado de Goiás, Brasil (Figura 1). A altitude geométrica do terreno (h) varia entre 644,51 m e 667,79 m, e apresentam as seguintes classes de relevo, conforme Santos et al. (2018) (Figura 9):

- Plano, caracterizado por inclinações que vão de 0% a 3%;
- Suave ondulado, exibindo declives suaves, predominantemente variáveis de 3% a 8%;
- Ondulado, com declives moderados, principalmente na faixa de 8% a 20%;
- Fortemente ondulado, marcado por declives acentuados, predominantemente variando de 20% a 45%;
- Montanhoso, com desníveis relativamente grandes e declives fortes e muito fortes, predominantemente variando de 45% a 75%.

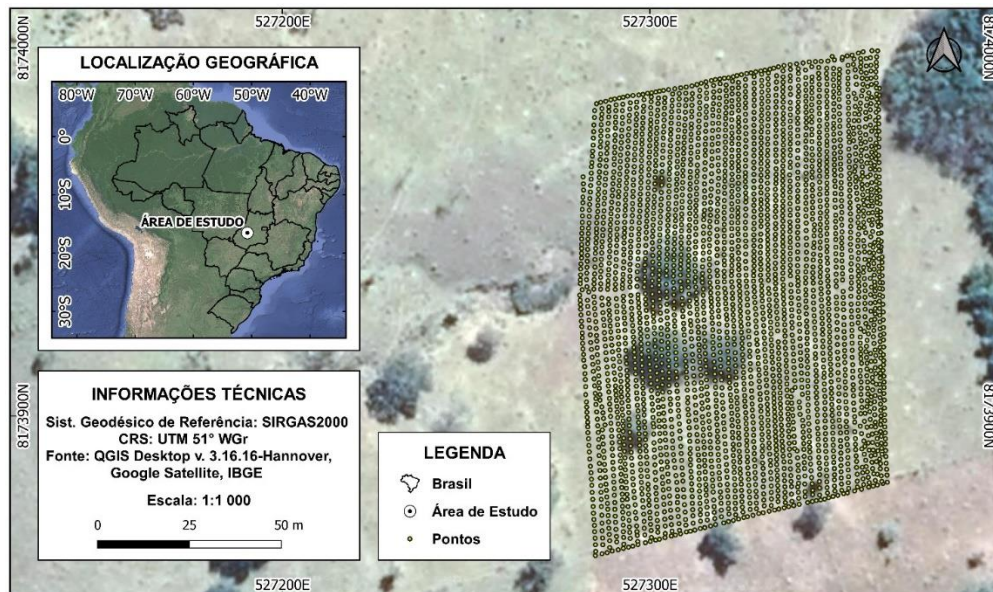


Figura 1. Localização geográfica da área de estudo e distribuição dos 3.224 pontos coletados.

Conjunto de dados experimentais

A área de estudo selecionada contém um conjunto de 3.224 pontos, com coordenadas leste (*E*), norte (*N*) e altitudes geométricas (*h*) previamente conhecidas (Figura 1). Os dados coletados em campo foram distribuídos espacialmente com espaçamento médio de 1,96 m.

As coordenadas e altitudes dos respectivos pontos foram adquiridas por meio do rastreamento dos sinais GNSS (*Global Navigation Satellite System*), utilizando dois receptores (*base e rover*) do modelo Topcon HiPer VR, que foram configurados especificamente para a técnica de Posicionamento Cinemático em Tempo Real - RTK (*Real Time Kinematic*). A coleta dos pontos, assim

como a ocupação do ponto-base de referência ao levantamento topográfico, ocorreu nos dias 29/04/2023, 30/04/2023 e 01/05/2023.

O sistema geodésico de referência adotado foi o SIRGAS2000, enquanto o sistema de coordenadas utilizado foi o UTM (*Universal Transversa de Mercator*), com o meridiano central da região de interesse definido em 51° WGr.

Foram obtidos 11 resultados possíveis de ajustamento das coordenadas do ponto-base (Tabela 1), e essa quantidade é justificada pela existência de dados provenientes de três dias distintos com diferentes períodos de rastreamento dos sinais GNSS.

Tabela 1: Resultados das 11 versões de pós-processamento do ponto-base.

<i>E</i> (m)	<i>N</i> (m)	<i>h</i> (m)	σ_E (m)	σ_N (m)	σ_h (m)	Tempo de Ocupação (h min s)	Data da Ocupação	Método de Pós-Processamento
527.522,362	8.174.197,788	618,907	0,007	0,007	0,022	01:40:22	29/04/2023	Relativo Estático
527.522,381	8.174.197,792	618,904	0,010	0,010	0,018	01:40:22	29/04/2023	Relativo Estático
527.522,395	8.174.197,764	618,983	0,004	0,003	0,009	02:24:52	01/05/2023	Relativo Estático
527.522,386	8.174.197,761	618,999	0,005	0,003	0,010	02:24:52	01/05/2023	Relativo Estático
527.522,392	8.174.197,763	619,000	0,004	0,004	0,013	02:24:52	01/05/2023	Relativo Estático
527.522,323	8.174.197,799	618,860	0,012	0,003	0,011	01:40:22	29/04/2023	IBGE-PPP
527.522,382	8.174.197,802	618,870	0,002	0,001	0,004	04:20:21	29/04/2023	IBGE-PPP
527.522,364	8.174.197,799	618,890	0,005	0,001	0,005	02:46:25	29/04/2023	IBGE-PPP
527.522,373	8.174.197,805	618,870	0,002	0,001	0,003	05:54:59	30/04/2023	IBGE-PPP
527.522,367	8.174.197,802	618,890	0,002	0,001	0,004	04:18:45	30/04/2023	IBGE-PPP
527.522,400	8.174.197,799	618,930	0,009	0,002	0,008	02:24:52	01/05/2023	IBGE-PPP

Para o conjunto total de 11 versões de pós-processamento do ponto-base, foi realizada a média ponderada das coordenadas *E*, *N* e *h*, de modo a torná-las mais bem ajustadas. Os resultados alcançados foram respectivamente: 527.522,376 m, 8.174.197,799 m e 618,890 m. O procedimento de cálculo pode ser verificado nas Equações (3), (4) e (5) que seguem:

$$\bar{x} = \frac{\sum_{i=1}^n x_i \cdot p_i}{\sum_{i=1}^n p_i} \quad (3)$$

onde x_i corresponde ao conjunto de dados referente às coordenadas atribuídas a cada eixo *E*, *N* e *h*.

Os pesos (p_i) devem ser relacionados às precisões dos dados, que são quantificados por meio dos desvios padrões (σ_i), de forma a pesar mais os dados de melhor precisão, que são os de menor desvio padrão.

$$p_i = \frac{1}{\sigma_i^2} \quad (4)$$

Logo,

$$\bar{x} = \frac{\sum_{i=1}^n \frac{x_i}{\sigma_i^2}}{\sum_{i=1}^n \frac{1}{\sigma_i^2}} \quad (5)$$

Desenvolvimento da RNA

A modelagem da rede neural para a análise dos dados geoespaciais foi realizada por meio do ambiente de programação do software proprietário MATLAB, versão R2019b.

A rede neural utilizada foi baseada em um Perceptron de Múltiplas Camadas – MLP (*Multilayer Perceptron*) de arquitetura *feedforward* (Figura 2).

O processo de aprendizagem foi programado para parar conforme configurações

padrão da função de treinamento de rede *trainlm* (retropropagação Levenberg-Marquardt), que atualiza os valores de peso e viés de acordo com a otimização do algoritmo. Foram critérios estabelecidos para parada antecipada (*early stopping*) o alcance de alguma das seguintes condições:

- A não-melhoria ou permanência do desempenho da rede nos vetores de validação por seis épocas consecutivas.
- O alcance do valor máximo do número de épocas, estabelecido em 1.000.

A rede neural foi treinada por 1.000 épocas, o que implica que os dados de treinamento foram apresentados à rede 1.000 vezes, ajustando gradualmente os pesos e os parâmetros da rede em cada iteração.

Os dados de entrada utilizados para a rede neural foram as coordenadas E_i e N_i de cada ponto da área de teste, enquanto os dados de saída correspondiam às altitudes h_i associadas a essas coordenadas (Figura 2).

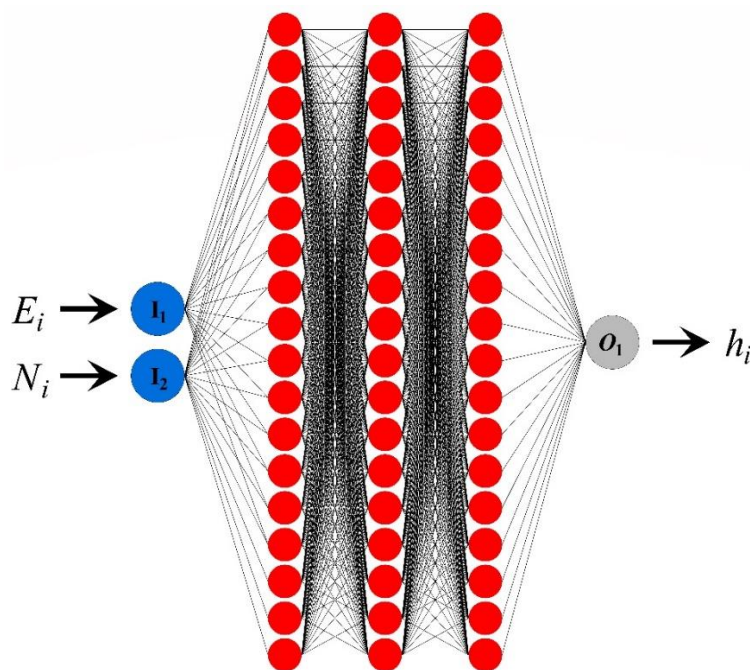


Figura 2. Ilustração da rede MLP (*Multilayer Perceptron*) de arquitetura *feedforward* para predição de altitudes (h_i).

Ao todo, foram realizados 33 testes para a determinação da melhor configuração de arquitetura da Rede Neural Artificial, com o uso do método *Repeated Leave-One-Out Cross-Validation* (RLOOCV).

O treinamento da RNA foi conduzido ao longo de 50 tentativas ($t = 50$), onde cada iteração treinou a rede usando diferentes conjuntos de dados de treinamento e teste. Esse loop que executa o treinamento da rede neural várias vezes, permitiu avaliar a capacidade de generalização do modelo em diferentes cenários.

A fim de validar o desempenho da rede neural e evitar o sobreajuste aos dados de treinamento, utilizamos um conjunto de validação composto por 1.000 amostras selecionadas aleatoriamente a partir do conjunto de dados do treinamento. A função utilizada para essa tarefa foi

a *randsample* (*random sample*), que seleciona aleatoriamente 1.000 elementos sem reposição. A seleção aleatória do conjunto de amostras desempenha um papel fundamental na criação de uma estratégia robusta de divisão de dados e validação cruzada ao treinar uma RNA.

Para garantir uma convergência eficaz durante o treinamento e obter parâmetros de entrada e saída adimensionais na rede neural, realizamos a normalização min-max dos dados utilizando a função *mapminmax*. A normalização min-max projeta a faixa original de dados em uma nova faixa. Intervalos de normalização muito comuns são $[0,1]$ e $[-1,1]$ (Sammut e Webb, 2017).

Conforme apontado por Raschka e Mirjalili (2017), para normalização dos dados, é aplicada a escala min-max a cada coluna de recurso, onde o novo valor $x_{norm}^{(i)}$ de uma amostra

$x^{(i)}$ pode ser calculado conforme a Equação (6) a seguir:

$$x_{norm}^{(i)} = \frac{x^{(i)} - x_{min}}{x_{max} - x_{min}} \quad (6)$$

onde $x^{(i)}$ é uma amostra específica, x_{min} é o menor valor em uma coluna de recurso e x_{max} é o maior valor.

A utilização da função de transferência *satlins* (Função de Transferência por Saturação Linear Simétrica) mostrou-se eficaz ao ser aplicada em conjunto com o método de reamostragem RLOOCV, que é computacionalmente intensivo, especialmente se o conjunto de dados (n) for grande e o modelo for complexo (com muitas camadas ocultas e/ou muitos neurônios em cada camada, por exemplo), o que significa uma grande quantidade de parâmetros (pesos e vieses). A função de transferência (ou ativação) *satlins* pode ser vista como uma aproximação de um amplificador não linear. Esta função retorna valores que são iguais aos próprios valores dos potenciais de ativação no intervalo $[-1, 1]$, e restringe-se aos valores limites caso o potencial de ativação esteja fora desse intervalo (Silva et al., 2016). A representação da expressão matemática dessa função é dada na Equação (7), conforme segue:

$$\begin{aligned} satlins &= g(.) = f(n) \\ &= \begin{cases} -1, & n < -1 \\ n, & -1 \leq n \leq 1 \\ 1, & n > 1 \end{cases} \quad (7) \end{aligned}$$

onde n é a variável de entrada para a função, enquanto $g(.)$ é o valor de saída retornado pela função.

O comportamento desta função é ilustrado na Figura 3.

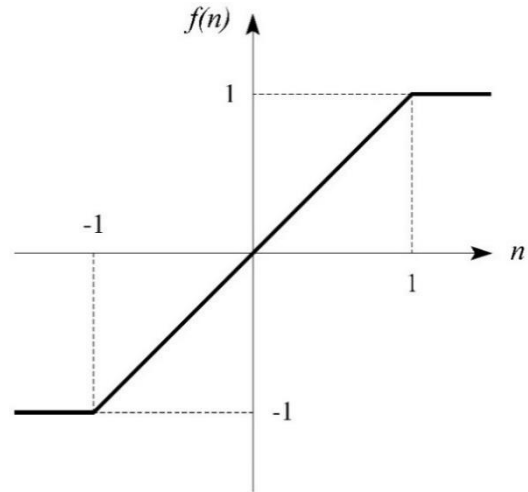


Figura 3. Função de Transferência por Saturação Linear Simétrica – *satlins*.

O método de reamostragem *Leave-One-Out Cross-Validation* (LOOCV), associado à função de ativação *satlins*, já foi testado em diferentes aplicações em contextos gerais, conforme evidenciado nos estudos de Cetó et al. (2012) e Cetó et al. (2013). Vale ressaltar que até o momento, não foi abordada uma metodologia que explore diversas tentativas para o método LOOCV, como é apresentado no estudo de Rofatto et al. (2023) com o método denominado RLOOCV.

Este é o primeiro estudo a sugerir que a função de ativação *satlins*, em conjunto com o método de reamostragem RLOOCV, demonstra ser apropriada em procedimentos de interpolação espacial no campo da modelagem do terreno utilizando Redes Neurais Artificiais.

Após vários testes, os melhores resultados foram obtidos com a arquitetura de rede neural estruturada por três camadas ocultas (ou intermediárias), com cada camada abrigando 18 neurônios (Figura 2).

Experimentos

Todos os 3.224 pontos de dados disponíveis foram usados para aplicação do método RLOOCV (Figura 4). Neste processo, a validação cruzada ocorre exclusivamente no conjunto de teste, que é uma forma de validar internamente o procedimento de reamostragem na rede neural.

No método RLOOCV, em cada iteração, um ponto de dados ($p = 1$) foi retirado do conjunto total para servir como conjunto de teste, enquanto os pontos restantes foram empregados para treinamento e validação. Esse processo foi repetido diversas vezes, garantindo que cada um dos 3.224 pontos fosse testado. No código desenvolvido para

a rede neural, esse processo foi controlado por um loop externo com 50 iterações ($t = 50$).

Portanto, dentro do contexto desse método de reamostragem, com um conjunto de dados de tamanho $n = 3.224$ e um total de $t = 50$

tentativas, obtivemos um conjunto de 161.200 preditores de rede neural, os quais geraram 50 predições para cada amostra de ponto de teste único.

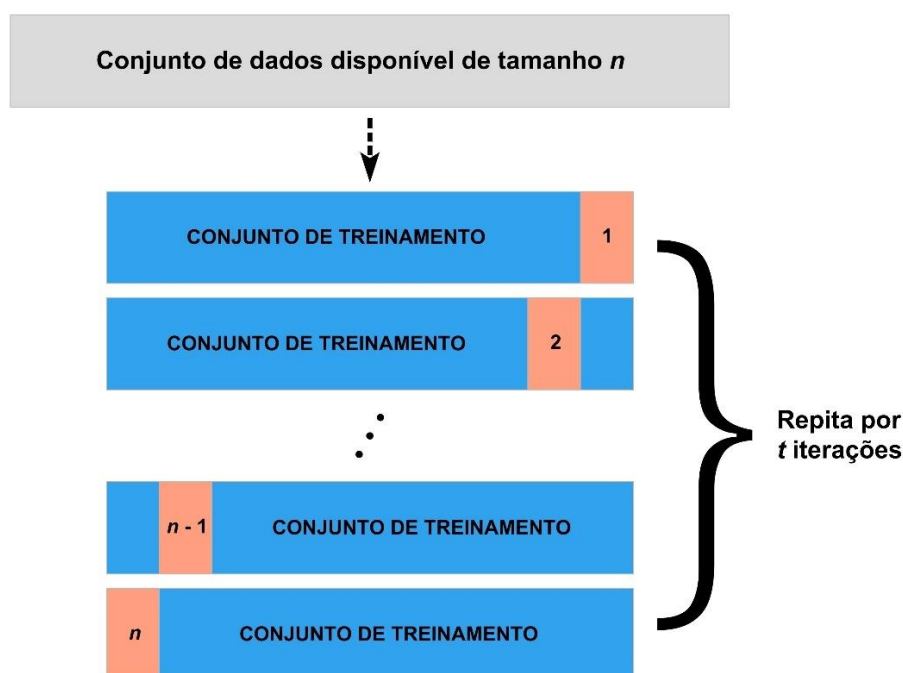


Figura 4. Representação esquemática do método de reamostragem *Repeated Leave-One-Out Cross-Validation*. Um conjunto de dados disponível, de tamanho n é repetidamente dividido em um conjunto de treinamento (mostrado em azul) contendo todas as observações, exceto uma, que corresponde ao conjunto de testes que contém apenas essa observação (mostrado em salmão).

Um resumo da configuração da rede neural utilizada nesse estudo pode ser verificado na Tabela 2 que segue:

Tabela 2: Configuração da rede MLP de arquitetura *feedforward* utilizada no estudo para um conjunto de dados de tamanho $n = 3.224$.

p	t	Nº de Preditores	Conj. de Validação	Função de Treinamento	Função de Ativação	Função de Normal.	Nº de Camadas Ocultas	Nº de Neurônios	Máx. Épocas
1	50	161.200	1.000	<i>trainlm</i>	<i>satlins</i>	<i>mapminmax</i>	3	18	1.000

Os resultados obtidos das predições repetidas são sintetizados pela construção da função de distribuição empírica para cada variável envolvida, conforme delineado na Equação (8). Além disso, o RMSE, MAE e s podem ser calculados para cada ponto de teste, conforme indicado nas Equações (9) a (12).

Análise Estatística e Avaliação de Modelos de RNAs com RLOOCV

A estimativa do valor mais provável, conhecido como valor esperado, não é derivada exclusivamente de uma única predição, mas é baseada em um intervalo, o que permite uma análise mais abrangente. Para fazer isso, organizamos os N valores estimados de um ponto específico em uma ordem estritamente crescente. Isso gera uma função de distribuição empírica para cada ponto de saída, denotado como G . Posteriormente, para uma determinada probabilidade de cobertura estipulada,

representada pelo símbolo α , podemos calcular os percentis desejados, denotados como p . O cálculo de p é feito utilizando a fórmula:

$$p = G_{[(1-\alpha) \times N]} \quad (8)$$

onde os colchetes ($[.]$) indicam arredondamento para o número inteiro mais próximo. Esse procedimento é flexível e adaptável para diferentes valores de α . Esse método se assemelha ao processo empregado na determinação de valores críticos para testes estatísticos que identificam pontos de dados discrepantes (Lehmann, 2012; Rofatto et al., 2020).

O método de reamostragem RLOOCV possibilita a medição do desempenho de predição para todos os 3.224 pontos levantados em campo, tendo em vista que todos esses pontos participam do conjunto de teste. Para a realização dessa avaliação, foram utilizadas as métricas de desempenho Raiz do Erro Quadrático Médio - RMSE (*Root Mean Squared Error*), Erro Médio Absoluto - MAE (*Mean Absolute Error*), e do Desvio Padrão Amostral (s). Essas métricas foram usadas para avaliar o desempenho da predição para cada ponto i da seguinte forma:

$$i = 1, \dots, n \quad (9)$$

$$RMSE_{(i)} = \sqrt{\frac{1}{N} \sum_{k=1}^N (T_k - \hat{O}_k)^2} \quad (10)$$

$$MAE_{(i)} = \frac{1}{N} \sum_{k=1}^N |T_k - \hat{O}_k| \quad (11)$$

$$s_{(i)} = \sqrt{s_{(i)}^2} = \sqrt{\frac{1}{N-1} \sum_{k=1}^N (X_k - \bar{X}_k)^2} \quad (12)$$

onde $\hat{O}_k$ são os N valores estimados para cada ponto de amostra i . T_k são os N valores reais. $\bar{X}_k$ é a média dos N valores estimados para cada ponto de amostra i , enquanto X_k são os N valores estimados. Note que essas métricas de desempenho são calculadas para cada um dos n pontos de amostra disponíveis.

A utilização de RMSE, MAE e s é comum na avaliação de RNAs devido às suas propriedades e relevância em diferentes aspectos do processo de modelagem e predição. RMSE e MAE são métricas que descrevem a acurácia e tendência de um modelo, especialmente em relação às suas predições, e no contexto da avaliação de modelos de RNAs, essas métricas são frequentemente usadas para medir o quão bem um modelo está se ajustando aos dados de treinamento e teste. O Desvio Padrão Amostral (s) é uma medida que quantifica a dispersão dos erros em torno da média, oferecendo informações sobre a variabilidade do desempenho das RNAs.

Resultados

Nesta seção apresentamos os resultados do desempenho do método de reamostragem com base no conjunto total de dados amostrados em campo. Como mencionado em sessões anteriores, o conjunto de dados é referente aos 3.224 pontos com informações espaciais (coordenadas E , N e h), onde um ponto de teste é excluído por vez em um procedimento iterativo, e o restante dos pontos são usados para treinamento da RNA.

Os resultados da aplicação de 161.200 preditores baseados em redes neurais estão resumidos nas Tabelas 3 e 4. As Figuras 5 a 9 ilustram os resultados obtidos com os preditores baseados em redes neurais criados a partir do método *Repeated Leave-One-Out Cross-Validation* (RLOOCV).

Tabela 3. Resultados do método de reamostragem RLOOCV.

p	t	n	Tempo Decorrido (h min s)	Número de Preditores	Média Global RMSE (m)	Máx. RMSE (m)	Mín. RMSE (m)
1	50	3.224	187:41:00	161.200	0,081 ± 0,002	0,520	0,027

Tabela 4. Médias globais das estatísticas fornecidas pela RNA para predição das altitudes ($\hat{h}$).

Média Global (m)					
$\bar{x}_h$	$\bar{x}_{\hat{h}}$	s	MAE	RMSE	M_d
654,697	654,697	0,061	0,067	0,081	654,696

$\bar{x}_h$ = média das altitudes de referência; $\bar{x}_{\hat{h}}$ = média das altitudes estimadas; s = desvio padrão amostral; MAE = erro médio absoluto; RMSE = raiz do erro quadrático médio; M_d = mediana das altitudes estimadas.

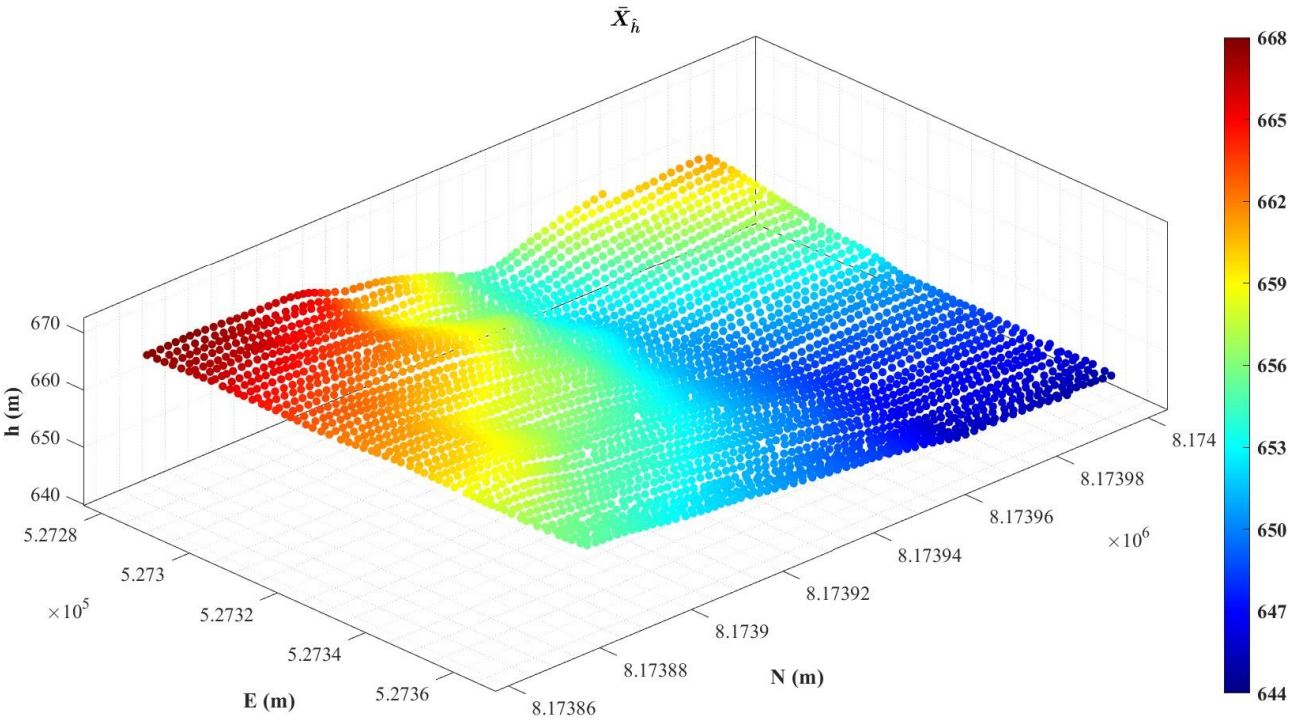
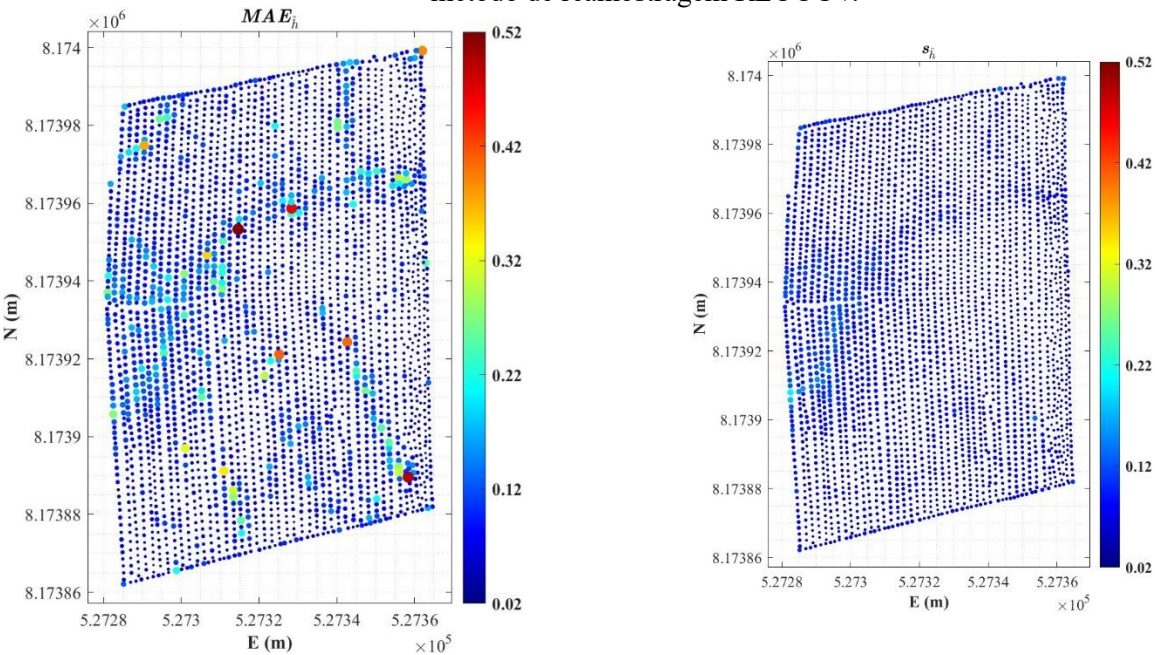


Figura 5. Distribuição espacial 3D da média das altitudes estimadas ($\bar{x}_{\hat{h}}$) pela rede neural com o uso do método de reamostragem RLOOCV.



Figuras 6 e 7. Distribuição espacial do Erro Médio Absoluto (MAE) e Desvio Padrão Amostral (s) das altitudes estimadas ($\hat{h}$).

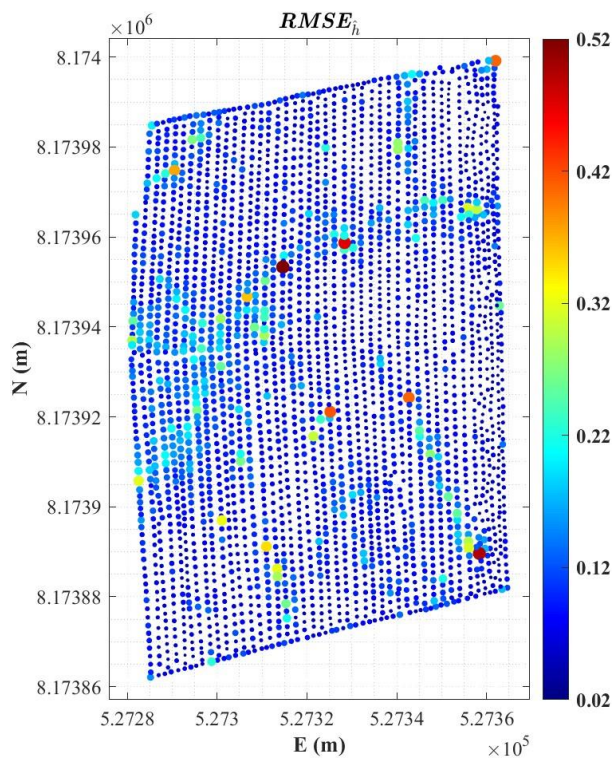


Figura 8. Distribuição espacial do RMSE das altitudes estimadas ($\hat{h}$).

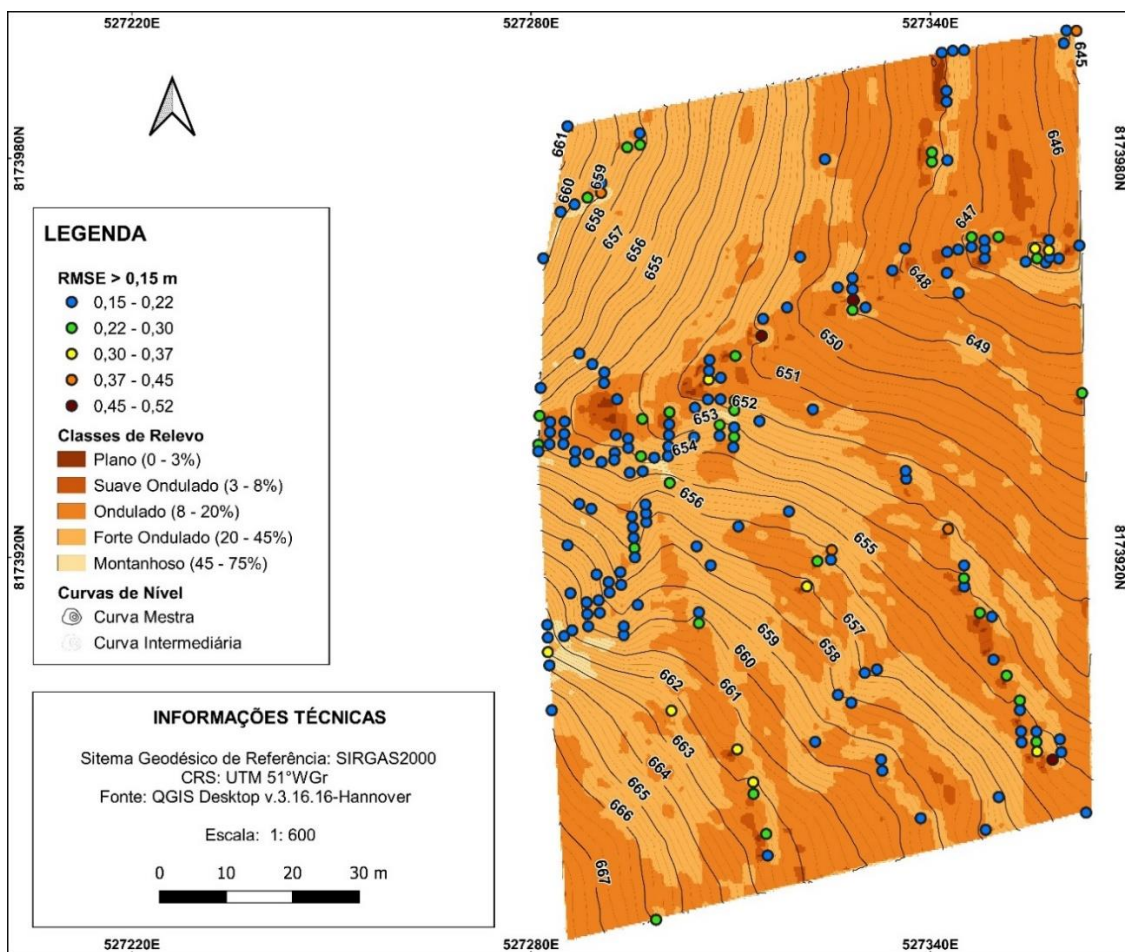


Figura 9. Distribuição espacial do RMSE com valores superiores a 0,15 m, conforme as diferentes classes de relevo.

Discussão

As Tabela 3 e 4 apresentam as médias globais das métricas de desempenho, referente as predições da RNA. Observamos que o RMSE, RMSE máximo e RMSE mínimo foram respectivamente 0,081 m ($\pm 0,002$), 0,520 m e 0,027 m, indicando um bom desempenho do modelo na estimativa das altitudes. Entretanto, é relevante observar que ao empregar um método de reamostragem, existe a possibilidade de que uma ou mais estimativas se afastem consideravelmente de seu valor real, como evidenciado pelo RMSE máximo.

Com relação a Tabela 4, o Erro Médio Absoluto (MAE) foi 0,067 m, menor que o valor do RMSE, justificado pelo procedimento de cálculo que possui uma pontuação linear, onde os resíduos têm mesmo peso. Nesse contexto, o MAE tende a diminuir se comparado ao RMSE, em que os erros maiores recebem mais peso. A média ($\bar{x}_h$) e mediana (M_d) das altitudes estimadas se igualaram ao valor da média de referência ($\bar{x}_h$), indicando uma excelente capacidade de predição.

Conforme é apresentado nas Figuras 6, 7 e 8, observou-se na distribuição espacial das métricas de desempenho MAE, s e RMSE que as taxas apresentaram significativa regularidade em sua grande maioria, tendo em vista a condição de inclinação contínua do relevo em boa parte da área de testes (Figuras 5 e 9).

As métricas de desempenho que apresentaram os piores resultados ficaram agrupadas em porções distintas da área de testes (Figuras 6, 7 e 8), que correspondem aos trechos de talvegue e onde existem mudanças abruptas da inclinação do terreno (Figura 9).

A Figura 9 demonstra com maior clareza essa condição anteriormente mencionada, de modo que os pontos estimados com RMSE superior a 0,15 m estão localizados basicamente nas regiões vertentes, de talvegue ou em trechos do terreno em que há uma mudança significativa de sua inclinação. Esse comportamento dos dados estatísticos demonstrou uma dificuldade do modelo da rede neural em generalizar para regiões de drenagem e de transição da inclinação do terreno.

O trabalho conduzido por Sen e Gumus (2023) destacou, de maneira semelhante, que houve significativo aumento nos erros de elevação no MDE (Modelo Digital de Elevação) interpolado por uma RNA. Esses aumentos foram evidenciados ao longo das linhas de cumeada e drenagem, bem como em pontos de depressão e picos.

Embora as variações abruptas do relevo não sejam capturadas pela RNA - e, portanto, representem um desafio na modelagem de superfícies usando RNAs - em aplicações com medidas anômalas (*outliers*), a RNA pode ser uma vantagem em relação ao método de interpolação espacial Krigagem, já que este último é afetado por tais anomalias. Por exemplo, em um estudo sobre a predição de propriedades químicas do solo utilizando RNA conduzido por Rodrigues et al. (2022), as predições não foram afetadas por essas anomalias, o que poderia ocorrer com o método de Krigagem, que é sensível a *outliers*. Portanto, dependendo da aplicação, o método proposto pode apresentar vantagens, especialmente em um ambiente contaminados por *outliers* pontuais.

Na modelagem de uma superfície de terreno existe a possibilidade da existência de pontos discrepantes (com altitudes incompatíveis) em certas regiões do conjunto total de dados, podendo prejudicar a modelagem realizada e consequentemente suas métricas de desempenho. Conforme apontado por S. Kim et al. (2019), em seu trabalho utilizando imagens estéreo, a existência de pontos discrepantes (*outliers*) podem gerar picos ou depressões irreais em um Modelo Digital de Superfície - MDS.

Neste trabalho não foi abordada uma pré-filtragem de possíveis *outliers* nas altitudes dos pontos levantados por posicionamento pelo GNSS. Portanto, há de se considerar que as piores precisões das altitudes de referência, que tiveram uma variação do desvio padrão (σ) de 0,015 m a 0,086 m (Figuras 10 e 11), não poderiam ser consideradas discrepantes a ponto de prejudicarem a modelagem do terreno, tendo em vista a pequena magnitude dessas medidas de dispersão. Os pontos com piores resultados ficaram agrupados em porções da área de estudo que correspondem a regiões com cobertura arbórea, como árvores e arbustos (Figura 11). As precisões desses pontos que ficaram mais degradadas são, em sua grande maioria, resultado da perda na contagem do número inteiro de ciclos medidos pelo receptor, erro relacionado a propagação do sinal GNSS denominado perdas de ciclo (*cycles slips*). Trata-se do caso em que um ambiente não possui amplo campo de visão, com bloqueio do sinal de um ou mais satélites por construções, árvores, pontes, montanhas etc., impedindo que este chegue até a antena do receptor (Monico, 2008).

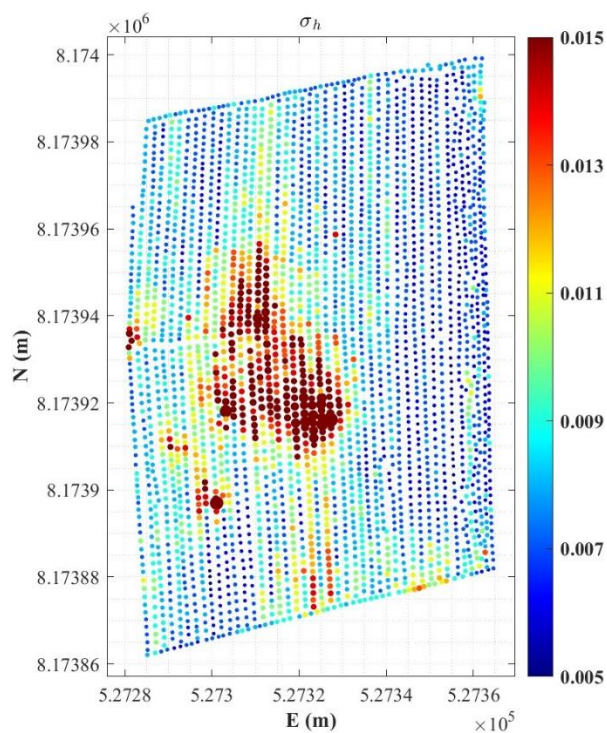


Figura 10. Distribuição espacial do Desvio Padrão (σ) das altitudes de referência (h). A paleta de cores foi limitada a 0,015 m para evidenciar os pontos com maior dispersão (0,015 m a 0,086 m).

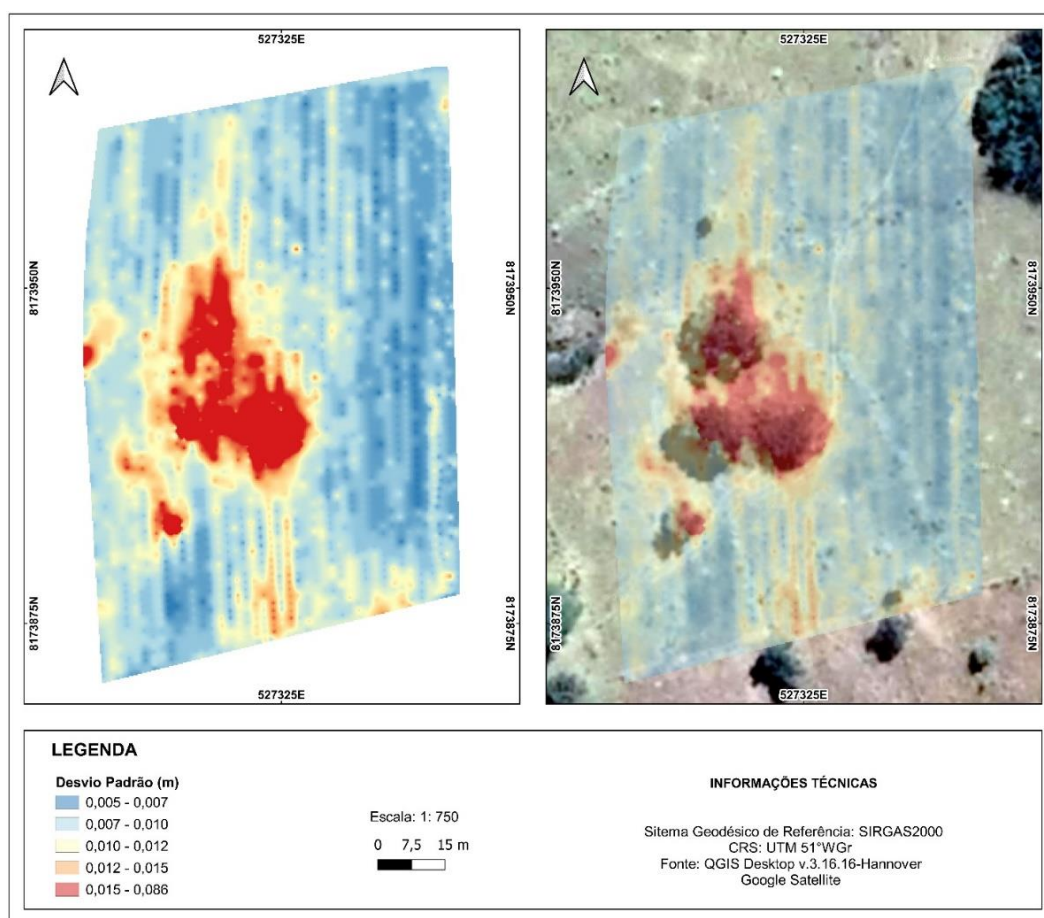


Figura 11. Distribuição espacial do Desvio Padrão (σ) das altitudes de referência (h) interpoladas pelo método Krigagem Ordinária.

Investigações sobre possíveis *outliers* em dados altimétricos ficarão a cargo de trabalhos futuros com enfoque na detecção de erros em modelos de rede neural como este aqui desenvolvido.

Com relação às altitudes estimadas pela RNA, as métricas de desempenho com piores taxas (Figuras 6, 7 e 8) apresentaram comportamento distinto das altitudes de referência, estando correlacionadas as características topográficas da área em questão.

A rede neural demonstrou ser sensível ao modelar a superfície do terreno quando altitudes de pontos específicos apresentam discrepâncias quando comparadas ao padrão das altitudes dos pontos vizinhos. Essa piora nas taxas do MAE e RMSE, que por uma equivocada interpretação poderia ser atribuída a presença de *outliers* altimétricos, na verdade estão relacionadas a presença de irregularidades do terreno oriundas de processos naturais ou por ação antrópica. A existência de áreas vertentes, com linhas de drenagem, naturalmente são áreas onde o modelo oferece uma piora nas métricas de desempenho devido a inclinação do terreno aumentar as diferenças das altitudes entre os pontos. Porém, a existência de uma boa distribuição de pontos nessas áreas possibilitou uma transição, digamos, mais suavizada entre a altitudes, que consequentemente ofereceu taxas de erro menores quando comparadas a pequenas porções que apresentaram as piores taxas. Essas são estreitas faixas do terreno que exibem mudanças repentinas no relevo, ocasionadas pela construção de antigos terraceamentos (ou “curvas de nível”) que ainda perduram na área de pastagem (Figuras 9 e 11). É possível que a resposta da rede neural a essas áreas em específico fosse mais bem representada com a densificação de pontos, aprimorando o aprendizado do modelo, de forma a não considerar um número maior de altitudes discrepantes como anomalias.

A RNA desenvolvida neste contexto, por meio do método de reamostragem *Repeated Leave-One-Out Cross-Validation*, apresenta vantagens em relação a configurações mais simples de RNAs. A aplicação de técnicas de reamostragem contribui para aprimorar a capacidade de generalização da rede neural, permitindo que ela identifique padrões mais robustos nos dados. Esses métodos oferecem uma abordagem mais sólida para estimar a incerteza associada às previsões da rede. Além disso, ao utilizar tentativas, é possível obter uma estimativa da variabilidade das previsões, fornecendo uma medida de confiança associada a cada uma delas. Essas tentativas permitem

descrever a distribuição de probabilidade do preditor, em contraste com a prática comum de fornecer apenas uma previsão pontual, que é frequentemente utilizada em redes neurais.

Em tese, a metodologia aqui proposta pode ser aplicada a outros tipos de modelagem de superfície, como batimetria, modelos geoidais/quase-geoidais, partindo do pressuposto que os dados de entrada da rede neural sejam pontos espacializados, ou seja, com coordenadas tridimensionais (x , y e z) do conjunto de dados.

Porém a dúvida com relação ao modelo aqui desenvolvido recai sobre sua aplicabilidade diante de grandes conjuntos de dados. Tendo em vista o formato como os parâmetros da rede neural foram configurados, utilizando o método RLOOCV, que é computacionalmente intensivo quando aplicado a amplos conjuntos de dados, somado a proposta de diversas realizações (tentativas), possivelmente não seria praticável nessas condições.

Perante tal fato, seria interessante a utilização do método de reamostragem *Houd-out Trials*, abordado por Rofatto et al. (2023), que pode ser mais apropriado por utilizar um conjunto de testes maior, onde o usuário pode escolher a porcentagem de divisão da amostra para cada conjunto (treinamento, validação e teste) assim como o número de tentativas.

A configuração de rede neural com melhor desempenho alcançado custou 187 horas e 41 minutos (Tabela 3). Diante do grande empenho em termos de capacidade computacional, assim como tempo decorrido, é de se questionar a viabilidade do uso do método de reamostragem adotado.

No entanto, o questionamento aqui não se trata em verificar a viabilidade de todo o processo (aquisição de dados pelo GNSS em três dias de campo, pós-processamento desses dados e o processamento ininterrupto de 187 horas e 41 minutos da RNA), quando comparado a consolidadas técnicas de aquisição de dados para determinação da elevação do terreno, como é o caso de tecnologias de Varredura a Laser (*Laser Scanning*), que permite a medição rápida de milhões de pontos, ou mesmo a técnica Lidar (*Light Detection and Ranging*), que oferece um nível de detalhe em sua análise que pode chegar a milhões de pontos por quilômetro quadrado (Lemmens, 2011). Para ambas as técnicas, a determinação da modelagem da superfície do terreno envolve a seleção de um método de interpolação espacial, como Rede Irregular Triangular (TIN), Vizinho mais Próximo, Ponderação pelo Inverso da Distância (IDW),

Krigagem etc. (Bagheri et al., 2014; Baltensweiler et al., 2017; Lemmens, 2011; J. Li e Heap, 2008).

O enfoque envolvido aqui neste estudo está em aprimorar o uso das RNAs como uma alternativa eficiente aos modelos convencionais de interpolação espacial amplamente utilizados para modelagem de superfícies em diversos campos das Geociências, como a modelagem da superfície do terreno, batimetria, modelagem da superfície geoidal/quase-geoidal etc.

A ideia central desse estudo é propor um modelo de superfície matemática por meio de uma Rede Neural Artificial, utilizando um método de reamostragem para a estimativa da distribuição de probabilidade aproximada para o preditor, em vez de simplesmente fornecer uma predição pontual.

Conclusões

Redes Neurais Artificiais aplicadas a predição de elevações é um método não determinístico que nos últimos anos vem requerendo diversas investigações voltadas a um procedimento alternativo de interpolação espacial.

Este estudo apresenta uma nova abordagem, onde o método de reamostragem incorporado às redes neurais possibilita que identifiquemos a distribuição experimental de probabilidade para as altitudes a serem estimadas, em vez de apenas apresentar uma única estimativa. Como resultado, podemos analisar o quão bem a predição funciona para cada ponto específico de amostragem.

Apresentamos o método de reamostragem denominado *Repeated Leave-One-Out Cross-Validation* (RLOOCV) em um modelo de rede neural. O método RLOOCV é uma ampliação do conhecido método *Leave-One-Out Cross-Validation* (LOOCV), e consiste em selecionar uma observação do conjunto total de dados, treinando as $n - 1$ observações restantes, em um processo iterativo onde cada uma das observações são selecionadas individualmente.

O método de reamostragem RLOOCV demonstrou um desempenho global sólido na estimativa de altitudes, como indicado pelos valores de métricas de desempenho. A análise revelou que o modelo é consistente em diferentes partes do conjunto de dados, evidenciado pela pequena diferença entre os valores globais médios das altitudes mínima e máxima estimadas.

No entanto, a análise espacial revelou que a capacidade da rede neural em generalizar é afetada em áreas com variabilidade topográfica, ou seja, áreas com variações significativas de

inclinação do terreno, como nas regiões vertentes, de talvegue e áreas com mudanças abruptas na inclinação. Estes resultados fornecem informações valiosas para futuras melhorias e otimizações no modelo, bem como para aplicações práticas em regiões com características topográficas semelhantes.

Trabalhos futuros podem voltar seus esforços em como aprimorar o gradiente do relevo nos procedimentos de interpolação espacial em redes neurais. Nesse processo de investigação, seria interessante a exploração de arquiteturas de redes mais complexas ou adaptações específicas no treinamento da rede para lidar melhor com regiões de variabilidade topográfica.

A configuração da estrutura da rede neural, somada à grande dimensão do conjunto de dados, exigiu um alto desempenho computacional, tanto do ponto de vista de desempenho como em tempo decorrido. Este aspecto deve ser levado em consideração em aplicações práticas que requerem eficiência computacional.

Devido à natureza aleatória da rede neural, existe a possibilidade de que uma ou mais predições obtidas por meio do processo de reamostragem apresentem desvios substanciais em relação aos seus valores reais. Tendo em vista que são geradas várias estimativas para cada variável de saída da rede neural, seria viável considerar a exclusão dessas estimativas discrepantes (*outliers*). Dando continuidade ao contexto da aplicação da RNA desenvolvida neste estudo, seria tanto interessante quanto necessário a elaboração de uma segunda abordagem envolvendo a ideia de ajustar o modelo de superfície aqui proposto e modelar os resíduos (*outliers*) deste ajuste por meio de uma nova RNA configurada para essa finalidade.

Desse modo, o método aqui proposto pode ser aplicado em comparação a outros métodos de reamostragem, como *Hold-out Trials*, *Leave-P-Out Cross-Validation* e *Repeated Cross-Validation* (J. Kim, 2009; Rofatto et al., 2023; Zhong et al., 2020), a saber.

Agradecimentos

O trabalho teve apoio do Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq (Processo nº 421278/2023-4).

Referências

Abbas, A. I., Alhamadani, O. Y. M., & Mohammed, M. U. (2022). The application of an artificial neural network for 2d coordinate transformation. *Journal of Intelligent Systems*,

- 31(1), 739–752. <https://doi.org/10.1515/jisys-2022-0033>
- Abdi, H., & Williams, L. J. (2010). Jackknife. Em D. M. Salkind (Org.), *Encyclopedia of Research Design* (p. 655–660). Sage. <https://personal.utdallas.edu/~herve/abdi-Jackknife2010-pretty.pdf>
- Afsari, R., Nadizadeh Shorabeh, S., Bakhshi Lomer, A. R., Homaei, M., & Arsanjani, J. J. (2023). Using artificial neural networks to assess earthquake vulnerability in urban blocks of tehran. *Remote Sensing*, 15(5). <https://doi.org/10.3390/rs15051248>
- Aggarwal, C. C. (2018). *Neural networks and deep learning* (1st ed.). Springer Cham. <https://doi.org/10.1007/978-3-319-94463-0>
- Akwensi, P. H., Brantson, E. T., Niipele, J. N., & Ziggah, Y. Y. (2021). Performance evaluation of artificial neural networks for natural terrain classification. *Applied Geomatics*, 13(3), 453–465. <https://doi.org/10.1007/s12518-021-00360-9>
- Bagheri, H., Sadeghian, S., & Sadjadi, S. Y. (2014). The assessment of using an intelligent algorithm for the interpolation of elevation in the DTM generation. *Photogrammetrie, Fernerkundung, Geoinformation*, 2014(3), 197–208. <https://doi.org/10.1127/1432-8364/2014/0220>
- Baltensweiler, A., Walthert, L., Ginzler, C., Sutter, F., Purves, R. S., & Hanewinkel, M. (2017). Terrestrial laser scanning improves digital elevation models and topsoil pH modelling in regions with complex topography and dense vegetation. *Environmental Modelling and Software*, 95, 13–21. <https://doi.org/10.1016/j.envsoft.2017.05.009>
- Cakir, L., & Yilmaz, N. (2014). Polynomials, radial basis functions and multilayer perceptron neural network methods in local geoid determination with GPS/levelling. *Measurement: Journal of the International Measurement Confederation*, 57, 148–153. <https://doi.org/10.1016/j.measurement.2014.08.003>
- Celisse, A. (2008). Model selection via cross-validation in density estimation, regression, and change-points detection. <https://theses.hal.science/tel-00346320>
- Cetó, X., Céspedes, F., & delValle, M. (2013). Assessment of Individual polyphenol content in beer by means of a voltammetric bioelectronic tongue. *Electroanalysis*, 25(1), 68–76. <https://doi.org/10.1002/elan.201200299>
- Cetó, X., Céspedes, F., Pividori, M. I., Gutiérrez, J. M., & Del Valle, M. (2012). Resolution of phenolic antioxidant mixtures employing a voltammetric bio-electronic tongue. *Analyst*, 137(2), 349–356. <https://doi.org/10.1039/c1an15456g>
- Chen, W., Pourghasemi, H. R., Kornejady, A., & Zhang, N. (2017). Landslide spatial modeling: introducing new ensembles of ann, mxent, and svm machine learning techniques. *Geoderma*, 305, 314–327. <https://doi.org/10.1016/j.geoderma.2017.06.020>
- Chernick, M. R. (2012). Resampling methods. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(3), 255–262. <https://doi.org/10.1002/widm.1054>
- Coelho, F. F., Giasson, E., Campos, A. R., de Oliveira e Silva, R. G. P., & Costa, J. J. F. (2021). Geographic object-based image analysis and artificial neural networks for digital soil mapping. *Catena*, 206. <https://doi.org/10.1016/j.catena.2021.105568>
- Ebrahimi, M., Safari Sinangani, A. A., Sarikhani, M. R., & Mohammadi, S. A. (2017). Comparison of artificial neural network and multivariate regression models for prediction of Azotobacteria population in soil under different land uses. *Computers and Electronics in Agriculture*, 140, 409–421. <https://doi.org/10.1016/j.compag.2017.06.019>
- Efron, B. (1982). The jackknife, the bootstrap and other resampling plans. *Society for Industrial and Applied Mathematics*. <https://doi.org/10.1137/1.9781611970319>
- Fanos, A. M., Pradhan, B., Alamri, A., & Lee, C. W. (2020). Machine learning-based and 3d kinematic models for rockfall hazard assessment using lidar data and gis. *Remote Sensing*, 12(11). <https://doi.org/10.3390/rs12111755>
- Farolfi, G., & Del Ventisette, C. (2016). Contemporary crustal velocity field in alpine mediterranean area of italy from new geodetic data. *GPS Solutions*, 20(4), 715–722. <https://doi.org/10.1007/s10291-015-0481-1>
- Fuso, F., Crocetti, L., Ravanelli, M., & Soja, B. (2024). Machine learning-based detection of TEC signatures related to earthquakes and tsunamis: the 2015 Illapel case study. Em *GPS Solutions* (Vol. 28). Springer Science and

- Business Media Deutschland GmbH.
<https://doi.org/10.1007/s10291-024-01649-z>
- Haykin, S. (1998). *Neural networks - a comprehensive foundation* (2nd ed.). Prentice Hall.
- Hilborn, E. D., Catanzaro, D. G., & Jackson, L. E. (2012). Repeated holdout cross-validation of model to estimate risk of Lyme disease by landscape characteristics. *International Journal of Environmental Health Research*, 22(1), 1–11.
<https://doi.org/10.1080/09603123.2011.588320>
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning* (1st ed.). Springer.
<https://doi.org/10.1007/978-1-4614-7138-7>
- Jang, J. S. R., Sun, C. T., & Mizutani, E. (1997). Neuro-fuzzy and soft computing - a computational approach to learning and machine intelligence. *IEEE Transactions on Automatic Control*, 42(10), 1482–1484.
<https://doi.org/10.1109/tac.1997.633847>
- Jian, L., Wang, X., Jiang, W., Hao, H., Xi, R., & Yang, L. (2024). Improved tide level prediction model combined GA-BP neural networks and GNSS SNR data. *Advances in Space Research*, 74(4), 1595–1608.
<https://doi.org/10.1016/j.asr.2024.05.030>
- Jiang, H. (2021). *Machine learning fundamentals*. Em *Machine Learning Fundamentals*. Cambridge University Press.
<https://doi.org/10.1017/9781108938051>
- Kaloop, M. R., & Hu, J. W. (2015). Optimizing the de-noise neural network model for gps time-series monitoring of structures. *Sensors (Switzerland)*, 15(9), 24428–24444.
<https://doi.org/10.3390/s150924428>
- Kaloop, M. R., & Kim, D. (2014). Gps-structural health monitoring of a long span bridge using neural network adaptive filter. *Survey Review*, 46(334), 7–14.
<https://doi.org/10.1179/1752270613Y.0000000053>
- Kavzoglu, T., & Saka, M. H. (2005). Modelling local gps/levelling geoid undulations using artificial neural networks. *Journal of Geodesy*, 78(9), 520–527.
<https://doi.org/10.1007/s00190-004-0420-3>
- Khaledian, Y., & Miller, B. A. (2020). Selecting appropriate machine learning methods for digital soil mapping. *Applied Mathematical Modelling*, 81, 401–418.
<https://doi.org/10.1016/j.apm.2019.12.016>
- Kim, J. H. (2009). Estimating classification error rate: repeated cross-validation, repeated hold-out and bootstrap. *Computational Statistics and Data Analysis*, 53(11), 3735–3745.
<https://doi.org/10.1016/j.csda.2009.04.009>
- Kim, S., Rhee, S., & Kim, T. (2019). Digital surface model interpolation based on 3D mesh models. *Remote Sensing*, 11(1).
<https://doi.org/10.3390/rs11010024>
- Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection.
<https://www.ijcai.org/Proceedings/95-2/Papers/016.pdf>
- Konakoglu, B., Cakir, L., & Gökalp, E. (2016). 2D coordinate transformation using artificial neural networks. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 42(2W1), 183–186.
<https://doi.org/10.5194/isprs-archives-XLII-2-W1-183-2016>
- Lehmann, R. (2012). Improved critical values for extreme normalized and studentized residuals in Gauss-Markov models. *Journal of Geodesy*, 86(12), 1137–1146.
<https://doi.org/10.1007/s00190-012-0569-0>
- Lemmens, M. (2011). *Geo-information* (J. D. Gatrell & R. R. Jensen, Orgs.; 2011th ed, Vol. 5). Springer . <https://doi.org/10.1007/978-94-007-1667-4>
- Li, J., & Heap, A. D. (2008). A review of spatial interpolation methods for environmental scientists (Vol. 23). *Geoscience Australia*.
- Li, Y., Li, L., Chen, C., & Liu, Y. (2023). Correction of global digital elevation models in forested areas using an artificial neural network-based method with the consideration of spatial autocorrelation. *International Journal of Digital Earth*, 16(1), 1568–1588.
<https://doi.org/10.1080/17538947.2023.2203953>
- Lorenz, R., & Petryna, Y. (2024). Monitoring of thermal deformations of a highway bridge: comparison of geodetic measurements, finite element simulations and AI predictions. *Journal of Physics: Conference Series*, 2647(25), 252035.
<https://doi.org/10.1088/1742-6596/2647/25/252035>
- Moghtased-Azar, K., & Zaletnyik, P. (2009). Crustal velocity field modelling with neural network and polynomials. *Observing our Changing Earth*, 133.

- https://doi.org/10.1007/978-3-540-85426-5_93
- Monico, J. F. G. (2008). Posicionamento pelo GNSS (2nd ed.). Editora Unesp.
- Özarpacı, S., Kılıç, B., Bayrak, O. C., Taşkıran, M., Doğan, U., & Floyd, M. (2024). Machine learning approach for GNSS geodetic velocity estimation. *GPS Solutions*, 28(2). <https://doi.org/10.1007/s10291-023-01607-1>
- Pacci, S., Dengiz, O., Alaboz, P., & Saygın, F. (2024). Artificial neural networks in soil quality prediction: significance for sustainable tea cultivation. *Science of the Total Environment*, 947. <https://doi.org/10.1016/j.scitotenv.2024.174447>
- Provost, F., & Fawcett, T. (2013). *Data science for business* (1st ed.). O'Reilly Media.
- Quenouille, M. H. (1949). Problems in plane sampling. *The Annals of Mathematical Statistics*, 20(3), 355–375. <http://www.jstor.org/stable/2236533>
- Raschka, S., & Mirjalili, V. (2017). *Python machine learning: machine learning and deep learning with Python, scikit-learn, and TensorFlow* (2nd ed.). Packt Publishing.
- Reitermanová, Z. (2010). *Data splitting*. Matfyzpress.
- Rodrigues, B. P., Rofatto, V. F., Matsuoka, M. T., & Teles Assunção, T. (2022). Resampling in neural networks with application to spatial analysis. *Geo-Spatial Information Science*, 25(3), 413–424. <https://doi.org/10.1080/10095020.2022.2040923>
- Rofatto, V. F., Bonimani, M. L. S., Matsuoka, M. T., Klein, I., & Campos, C. C. de. (2023). Resampling methods in neural networks: from point to interval application to coordinate transformation. *Journal of Surveying Engineering*, 149(1). <https://doi.org/10.1061/jsued2.sueng-1366>
- Rofatto, V. F., Matsuoka, M. T., Klein, I., Veronez, M. R., & da Silveira, L. G. (2020). A monte carlo-based outlier diagnosis method for sensitivity analysis. *Remote Sensing*, 12(5). <https://doi.org/10.3390/rs12050860>
- Santos, H. G. dos, Jacomine, P. K. T., Anjos, L. H. C. dos, Oliveira, V. Á. de, Lumbrreras, J. F., Coelho, M. R., Almeida, J. A. de, Araújo Filho, J. C. de, Oliveira, J. B. de, Cunha, T. J. F., & Embrapa Solos. (2018). *Sistema brasileiro de classificação de solos* (5th ed.). Embrapa. <https://www.agroapi.cnptia.embrapa.br/portal/assets/docs/SiBCS-2018-ISBN-9788570358004.pdf>
- Sen, A., & Gumus, K. (2023). Comparison of different parameters of feedforward backpropagation neural networks in DEM height estimation for different terrain types and point distributions. *Systems*, 11(5). <https://doi.org/10.3390/systems11050261>
- Shao, J., & Tu, D. (1995). *The jackknife and bootstrap* (1st ed.). Springer Science+Business Media. <https://doi.org/10.1007/978-1-4612-0795-5>
- Silva, I. N., Spatti, D. H., & Flauzino, R. A. (2016). *Redes neurais artificiais para engenharia e ciências aplicadas* (M. A. Eckersdorff, Org.; 2nd ed.). Artliber.
- Stopar, B., Ambrožič, T., Kuhar, M., & Turk, G. (2006). GPS-derived geoid using artificial neural network and least squares collocation. *Survey Review*, 38(300), 513–524. <https://doi.org/10.1179/sre.2006.38.300.513>
- Sun, D., Lee, V. C., & Lu, Y. (2016). An intelligent data fusion framework for structural health monitoring. *ICIEA*, 11, 49–54. <https://doi.org/10.1109/ICIEA.2016.7603550>
- Tukey, J. W. (1958). Bias and confidence in not-quite large samples. *The Annals of Mathematical Statistics*, 614–623.
- Wang, L., & Yu, F. (2020). Jackknife resampling parameter estimation method for weighted total least squares. *Communications in Statistics - Theory and Methods*, 49(23), 5810–5828. <https://doi.org/10.1080/03610926.2019.1622725>
- Yadav, S., & Shukla, S. (2016). Analysis of k-fold cross-validation over hold-out validation on colossal datasets for quality classification. *Proceedings - 6th International Advanced Computing Conference, IACC 2016*, 78–83. <https://doi.org/10.1109/IACC.2016.25>
- Yager, R. R., & Zadeh, L. A. (1992). *An introduction to fuzzy logic applications in intelligent systems* (1st ed.). Springer Science+Business Media. <https://doi.org/10.1007/978-1-4615-3640-6>
- Zhang, P. (1993). Model selection via multifold cross validation. *Em The Annals of Statistics* (Vol. 21, Número 1, p. 299–313). <http://www.jstor.org/stable/3035592>
- Zhang, R., Bian, S., & Li, H. (2021). Rspcn: super-resolution of digital elevation model based on recursive sub-pixel convolutional neural networks. *ISPRS International Journal*

- of Geo-Information, 10(8).
<https://doi.org/10.3390/ijgi10080501>
- Zhang, Y., Yu, W., & Zhu, D. (2022). Terrain feature-aware deep learning network for digital elevation model superresolution. ISPRS Journal of Photogrammetry and Remote Sensing, 189, 143–162. <https://doi.org/10.6084/m9.figshare.19597201>
- Zhong, Y., He, J., & Chalise, P. (2020). Nested and repeated cross validation for classification model with high-dimensional data. Revista Colombiana de Estadística, 43(1), 103–125. <https://doi.org/10.15446/rce.v43n1.80000>