

Relatório de Atividades de Doutorado – 2º semestre de 2023

Investigação da radiação natural da praia de Meáípe (ES) utilizando rede neural

Aluna: Miriam Mathias Gigi

Orientadora: Profª Dra. Jacyra Soares (IAG-USP)

Laboratório de Interação Ar - Mar (LIAM) Departamento
de Ciências Atmosféricas – IAG- USP

Outubro 2023

Resumo

A praia de Meaipe, localizada no Espírito Santo, é considerada uma região com altos níveis de radiação natural de baixa dose. A radioatividade neste local é oriunda de areias monazíticas ricas em tório (^{232}Th). Mesmo a praia sendo conhecida como um resort terapêutico, estudos sobre essa região demonstram discordância sobre os efeitos do radônio (^{220}Rn) para a saúde. Cuidado deve ser tomado pois a intensidade da radiação gama apresenta variabilidade espaço-temporal sobre a praia de Meaipe, no qual valores pontuais não podem ser generalizados para a praia toda. Com isso, o objetivo principal deste estudo é entender e mapear a variação temporal e vertical da intensidade de radiação gama sobre a praia de Meaipe com auxílio de rede neural artificial. Pretende-se também estabelecer uma possível relação estatística entre a intensidade da radiação gama e as variáveis meteorológicas medidas em campo.

Palavras chaves:

Redes neurais, Meaipe, areia monazítica, torônio.

1. Objetivos

O objetivo geral deste trabalho é discutir a variação espaço-temporal da intensidade de radiação gama sobre a praia de Meaipe, com a utilização de medidas de campo, pretendendo-se estabelecer a relação da sua variação com variáveis meteorológicas e aperfeiçoar sua estimativa empregando rede neural

Os objetivos específicos são:

- Caracterizar os campos meteorológicos médios de vento, temperatura do ar, umidade do ar, pressão, balanço de radiação, temperatura da superfície e fluxo de calor no solo;
- Caracterizar a intensidade da radiação gama na região costeira da praia de Meaipe;
- Com as redes neurais formular um modelo estatístico que representa a variabilidade espaço-temporal da intensidade da radiação gama em função das variáveis meteorológicas.
- Estimar valores da intensidade da radiação gama.
- Comparar os dados observacionais com os resultados numéricos estimados pela rede neural artificial .

2. Metodologia

Para alcançar os objetivos pré-estabelecidos este trabalho será dividido em duas etapas:

- A. Utilizar os dados medidos pontualmente sobre a área de estudo para:
- Alimentar modelos estatísticos.

Utilizar as redes neurais artificiais:

- Definir arquiteturas e topologias das redes neurais artificiais a serem treinadas para a região.
 - Formular um modelo estatístico que represente a variabilidade temporal da intensidade da radiação gama em função das variáveis meteorológicas.
 - Estimar valores da intensidade da radiação gama.
 - Extrapolar as medidas pontuais da intensidade da radiação gama para uma malha que represente a praia.
- B. Comparar os dados observacionais com os resultados numéricos estimados pela rede neural artificial

2.2. Redes Neurais

As redes neurais artificiais (RNA) são modelos computacionais inspirados no funcionamento do cérebro e foram introduzidas para modelar funções não lineares complexas (Hassan, M. A. et al., 2017).

Dentro das redes neurais há estruturas processadoras denominadas neurônios, os quais permitem a aquisição e aprendizagem do conhecimento pela rede (Silva et al., 2010) a partir de soluções de funções matemáticas (Braga et al., 1998), conforme Figura 1.

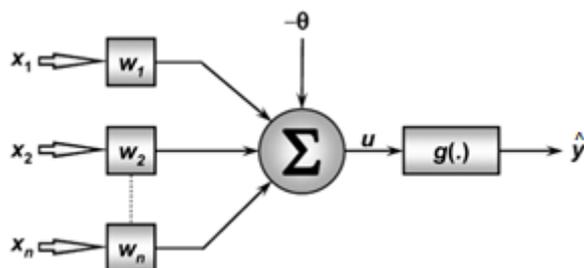


Figura 1– Neurônio artificial (Adaptado de Silva et al., 2010).

Onde, $\vec{X} = \{x_1, x_2, x_3, \dots, x_n\}$ são os dados de entrada em cada passo de tempo, e $\vec{W} = \{w_1, w_2, w_3, \dots, w_n\}$ são os pesos que ponderam esses dados de entrada e quantificam sua relevância, sendo n o número de variáveis de entrada.

O potencial de ativação do neurônio é $u = \sum_{i=1}^n w_j x_i - \theta$, com θ sendo uma constante que determina se o valor do potencial de ativação vai em direção a saída do neurônio (limiar de ativação).

A função de ativação $g(\cdot)$ recebe os valores do potencial de ativação e limita as saídas (\hat{y}) do neurônio dentro de um intervalo razoável. Dentre essas funções a mais utilizada é a função logística:

$$g(u) = \frac{1}{1+e^{-\beta u}} \quad (1)$$

Com β sendo uma constante associada ao nível de inclinação da função à frente do seu ponto de inflexão.

As redes neurais são caracterizadas pela sua arquitetura e topologia. A arquitetura define a quantidade de camadas da RNA e a forma de comunicação entre os neurônios de cada camada. A quantidade dos neurônios e suas funções de ativação de cada camada são chamadas de topologia da RNA (Silva et al., 2016).

Algumas RNA são compostas por três camadas (Figura 2). A camada de entrada recebe os valores dos dados normalizados. A camada oculta ou escondida é responsável pela verificação relativa da importância (peso) das variáveis. A maior parte do processamento da rede é feito nessa camada. A camada de saída processa os resultados gerados na camada anterior.

Como exemplo, a seguir, será descrita uma RNA do tipo *perceptron* multicamadas (*Multilayer Perceptron* –MLP), com um fluxo de informação entre os neurônios unidirecional, do tipo “feedforward” (da camada de entrada em direção a camada de saída), conforme Figura 4.

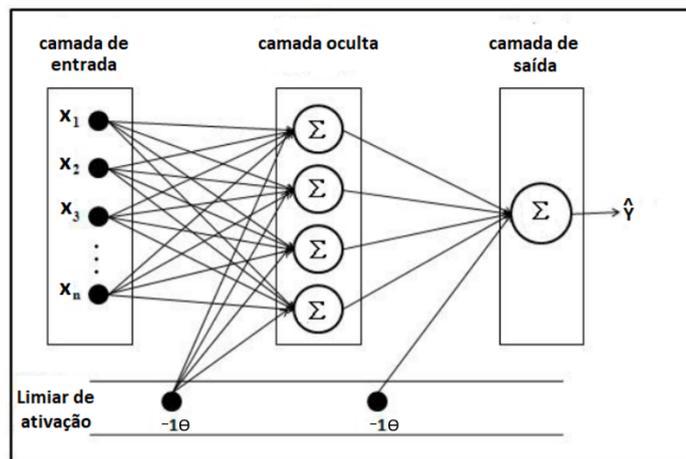


Fig.2 - Rede neural Multilayer Perceptron com uma camada oculta. (Adaptado de Nessrine et al., 2014)

O algoritmo de aprendizagem de uma rede neural é o processo de construção de modelos de predição a partir do treinamento dos seus neurônios com o propósito de ajustar os pesos e os limiares de ativação (Coutinho, 2018) Para a MPL esse treinamento é composto de três etapas.

Na primeira etapa os dados de entrada são propagados para todos os neurônios da camada oculta ponderados pelo peso (W_h^-) de cada neurônio (h). O campo induzido resultante (u) para

cada neurônio da camada oculta é dado por $u_h = \sum_{i=1}^n w_{hi} x_i + \theta_h$, onde i é o número do neurônio da camada de entrada.

O campo u_h é dependente da função de ativação $g_h(u_h)$. O resultado da função g_h é propagado como valor de entrada para o neurônio da camada de saída (o), sendo o campo induzido $u_o = \sum_{h=1}^p w_{oh} g_h + \theta_o$ e o valor estimado $\hat{y} = g_o(u_o)$.

A primeira dificuldade da utilização de redes neurais está na escolha das variáveis de entrada que representem bem o problema e minimizem o erro, $e_o(t) = y(t) - \hat{y}(t)$, entre os valores desejados, ($y(t)$), e os valores estimados, ($\hat{y}(t)$). O algoritmo de treinamento da RNA por retropropagação de erro tem como objetivo minimizar o erro quadrado total (E) produzido

$$\text{pela rede, } E = \frac{1}{2} \sum_{o=1}^p e_o^2(t).$$

Na segunda etapa, uma vez estimado o valor (\hat{y}) pelo algoritmo, os pesos de cada camada são ponderados pela regra geral $w(t + 1) = w(t) + \Delta w$.

Na camada de saída do neurônio (o), a variação do peso pode ser escrita como:

$$\Delta w_{o,h} = \eta g_h(u_h) \delta_o$$

Onde, h é o número do neurônio que transmite o dado da camada oculta, o cálculo de informação de erro é $\delta_o = e_o \frac{\partial g(u_o)}{\partial w}$ e (η) é a taxa de aprendizagem que pondera o ajuste dos pesos

A informação de erro para a camada oculta pode ser escrita como

$$\delta_h = \sum_{o=1}^p (\delta_o w_{oh}) \frac{\partial g_h(u_h)}{\partial w} \quad (2)$$

A alteração dos pesos dos neurônios é dada por:

$$\Delta w_{h,i} = \eta x_i \delta_h \quad (3)$$

Onde i é o número do neurônio da camada de entrada que transmite o dado para o neurônio da camada oculta (h).

Na terceira etapa, os ajustes dos pesos são efetuados, a partir das variações calculadas na segunda etapa.

Resultados

1. Implementação e construção do modelo

Neste trabalho, é proposto um método de ajuste de modelo de redes neurais dentro do software estatístico e gratuito R-Cran (<https://cran.r-project.org/>).

A biblioteca utilizada é a H2O, amplamente documentada e especializada em algoritmos de aprendizagem de máquina e análise preditiva, com estrutura eficiente para lidar com grandes conjuntos de dados (<http://docs.h2o.ai/h2o/latest-stable/h2o-docs/starting-h2o.html>).

O H2O é uma plataforma *open source* desenvolvida pela empresa H2O.ai.. Ela tem interface web intuitiva utilizando localhost e número de IP. Ela é caracterizada por trabalhar com algoritmos estatísticos ou manipulação de dados com memória de forma distribuída gerenciando a troca com disco. Assim, o H2O lida com o volume mínimo de dados em memória e persiste no disco o restante, criando um nível de interoperabilidade entre memória e disco de forma simples, fazendo uma sincronização, quando não espaço em memória, com o disco retornando apenas os dados interessantes para uso.

O H2O é construído em Java e interpreta o código em linguagem R. Sendo o R apenas uma interface cliente do H2O, todas as funções dentro do R estão na verdade chamando o interpretador do H2O, que gerencia os objetos em memória e/ou disco e retorna o resultado via uma REST API.

Utiliza-se então, a sintaxe em R e o pacote H2O faz a parte operacional. O H2O funciona com processamento em paralelo em memória cluster, onde cada computador é um nó, inclusive a máquina do operador, interligados pelo uso das memórias de forma eficiente. Os dados ficam em discos armazenados e particionados através do mecanismo “chave-valor”. A eficiência computacional se dá pelo gerenciamento dessas chaves pelo H2O. Com o IP e o número da chave, mapeia se os dados em qualquer disco.

O modelo de previsão com as redes neurais foi construído com as variáveis meteorológicas e de solo, medidas em campo e calculadas, para prever a intensidade de radiação gama (γ). Os dados medidos de radioatividade da região costeira de Meaípe, em diferentes alturas, serviram como parâmetro para treinamento da rede. O algoritmo da rede neural foi treinado para cada passo de hora com dados do período entre 18/10/2019 e 25/08/2022 com 41234 dados.

O objetivo do treinamento de uma rede neural é encontrar o modelo que gere as melhores previsões. Para que seja uma aplicação bem sucedida, é necessário que se defina a melhor configuração de parâmetros e hiperparâmetros. Para uma rede neural, os parâmetros são as variáveis de entrada e o hiperparâmetros são, por exemplo, os números de camadas e neurônios, ambos com grande poder controle sobre o desempenho do modelo, a função de ativação dos neurônios, a taxa de aprendizagem e o número de épocas, entre outros.

Dentre vários problemas a serem solucionados para implementação de uma rede neural, um deles é a seleção de variáveis - *FeatureSelect* (Boznar, 1993). Um problema já muito conhecido, no qual determina quais padrões devem ser inseridos no conjunto de treinamento, que neste caso, deve incluir o máximo de episódios meteorológicos significativos diferentes (MlakarandBonar), já que variáveis em excesso dificultam a convergência do modelo. Segundo Mlakar (1997) a quantidade de dados tem menos importância do que os padrões de conjuntos de dados.

Os critérios utilizados para selecionar as variáveis importantes para o conjunto de treinamento foram o conhecimento das condições meteorológicas e do solo do local e técnicas de correlações não lineares das variáveis medidas em campo (Alves, 2021) e a importância e fator de contribuição de um parâmetro quando comparada a outro, que correspondem a cálculos realizados com os pesos associados a cada variável distribuídos ao longo da rede durante o treinamento (Mlakar, 1997).

Não existe também, uma metodologia simples para otimizar o número de neurônios e camadas ocultas, isso ocorre através de forma demorada e baseada no treinamento de várias arquiteturas. Segundo (Kroseand Van Der Smagt, 1996), uma única camada oculta pode resolver

problemas de regressão com funções de ativação não lineares. O número de neurônios das camadas ocultas são determinados experimentalmente, e a rede não aprende se o número for muito pequeno, e aprende com muitos detalhes os padrões de treinamento e baixa capacidade de generalização se houver um número muito grande de neurônios (Boznar, 1993; MlakarandBoznar, 1997; Mlakar, 1997).

Uma abordagem, altamente recomendada, para auxiliar na otimização dos hiperparâmetros para um dado conjunto de variáveis já definidas, é a aplicação de pesquisa de grade (*grid search*) sobre um conjunto de hiperparâmetros aleatórios. Ao rodar o algoritmo de *grid search*, ele procura, com paralelismo no H2O, as melhores combinações de hiperparâmetros que variam de acordo com novos conjuntos de variáveis de entrada. Além disso, é importante adicionar estratégias de ponto de parada dentro do algoritmo que determinam quando o aprendizado chegou a um valor desejado, para que o modelo não tenha *overfitting*. Contudo, ainda é recomendado que aos modelos melhor posicionados seja feita, exaustivamente, a verificação das propriedades do modelo (*ModelChecking*), pois o desempenho dos modelos podem variar.

O desempenho de predição do modelo está associado com a forma que foi apresentado os parâmetros para o algoritmo durante a aprendizagem. Quando o modelo é avaliado com os mesmos dados incluídos na etapa de treinamento, pode ocorrer o *overfitting*. Nesse processo de sobreajuste, o modelo memoriza os padrões dos dados mas não consegue generalizá-los. O modelo tem desempenho aceitável durante os testes, mas não consegue prever dados não vistos por ele.

É possível minimizar a escolha de modelos sobre ajustados, como o que tem melhor desempenho, separando os dados em subconjuntos de treinamento, teste e validação. Após o treinamento, verifica-se o desempenho do modelo com os dados de teste e uma segunda avaliação com os dados de validação. Se o modelo obter bom desempenho nos dados de validação, ele não tem *overfitting*.

Contudo, pode ser um problema separar os dados em subconjuntos, pois é importante disponibilizar o máximo de dados possíveis para o treinamento. Se o conjunto de dados for pequeno isso pode ser um problema ainda maior. Como solução, é possível executar o algoritmo com uma técnica chamada validação cruzada (*cross-validation*).

A validação cruzada é um método para avaliar e comparar modelos de aprendizagem de máquina. A técnica estatística separa os dados, em subconjuntos para treinamento e teste chamados de *k-folds*.

O treinamento do modelo com o *cross-validation* ocorre de forma intercalada. Os *k-folds*

não são fixos, o subconjunto que é utilizado num primeiro momento como dados de treinamento, são utilizados para o teste no processo de aprendizagem seguinte, de forma que nunca os dados para o treinamento e teste são os mesmos em uma rodada. O número de *k-folds* determina quantas vezes o algoritmo passa pelo processo de aprendizagem sobre dados, possibilitando maior poder de generalização e minimizando o *overfitting*.

Durante o processamento da rede neural com o pacote H2O, ocorre de forma simultânea, a busca pelo melhor conjunto de hiperparâmetros (*gridsearch*), os treinamentos com a separação e intercalação dos dados para treinamento e teste e (*cross-validation*) e a validação.

Antes de entrar na rede as variáveis devem ser normalizadas para que tenham a mesma influência sobre a rede neural, e a escolha foi assumir intervalos entre -1 e 1 que tem sido amplamente usada na área de meio ambiente. Além disso, os dados devem ser “randomizados” uma vez que se usa o algoritmo de otimização da descida gradiente estocástica - *StochasticGradientDescent* (SGD) para atualizar os pesos a cada interação (Bottou, 1998, 2010, 2012).

Para o treinamento, as variáveis de entrada definidas para o aprendizado das RNA foram altura (A), Umidade específica (q), Temperatura do solo (TS), Temperatura do ar (TA), Radiação de onda longa emitida pela superfície (LW up), Radiação de onda longa emitida pela atmosfera (Lwdn), Onda curta incidente (OC dn), Onda curta refletida (OC up), hora (h), dia (d). Os valores foram submetidos a médias horárias selecionadas daqueles que tinham valores de intensidade de radiação gama associados. Após o treinamento, elas foram ordenadas de acordo com a sua importância e fator de contribuição segundo o método de Gedeon (Gedeon, 1997).

Além disso, as linhas de dados foram misturadas, uma vez que o algoritmo de otimização *StochasticGradientDescent* (SGD) para a função de perda é usado, por ser mais rápido, para atualizar os pesos a cada iteração, (Bottou, 1998, 2010, 2012; CandelandParmar; 2015). Uma coluna de ID foi adicionada aos dados para auxiliar no pós-processamento quando os dados retornarem ao original.

Os dados utilizados para esse trabalho foram separados pela técnica *holdout*, sendo 80% para base de treinamento e 20% para base de validação. Na base de treinamento foi aplicada o *crossvalidation*, com *5-folds*. Cada *fold* tem 20% dos dados da base de treinamento. Em cada rodada 4 *folds* foram utilizados para treinamento e 1 *fold* para teste.

Com *grid search*, obteve-se inúmeros modelos ajustados aos dados com diferentes métricas de desempenho, no entanto, o critério de escolha para o estudo foi aquele que apresentou

a melhor curva de aprendizagem, que é aquela que descreve o percurso da curva do erro do modelo à medida que o número de épocas decorre.

A RNA foi constituída com duas camadas ocultas composta por 10 neurônios cada. A função de ativação para os neurônios das camadas ocultas foi uma função tangente hiperbólica, amplamente usada em estudos ambientais. A função é uma função logística redimensionada e deslocada a sua simetria em torno de 0, o que faz a convergência do modelo ser mais rápida (Candeland Parmar; 2015). Para o nó de saída, usou uma função linear ou identidade, uma vez que se deseja um valor real e sua transmissão direta para o *output*, ela copia o seu valor de entrada para o seu valor de saída (Zaim et al, 2023).

O processo de aprendizagem é dinâmico e alguns hiperparâmetros podem ter mais relevância na aceleração ou desaceleração da convergência no aprendizado do algoritmo, dentre eles estão a taxa de aprendizagem e o *momentum* (MlakarandBonar, 1997). Taxa de aprendizagem muito alta pode garantir uma convergência rápida, mas corre o risco de atingir mínimos locais da função de erro, ou passar a não convergir por não chegar ao erro mínimo global, já taxas muito baixas utiliza muito tempo para convergir e tem o problema do custo computacional, mas pode ter resultados melhores.

Para a busca de uma convergência mais rápida e aprendizagem mais eficiente foi utilizada o algoritmo de aprendizagem adaptativa ADADELTA (Zeiler, 2012), que combina automaticamente os hiperparâmetros taxa de aprendizagem e *momentum*. Na busca de pesquisa não adaptativos desses hiperparâmetros pode se chegar especificar até 7 parâmetros para eles. Neste caso, simplificando a busca, apenas dois hiperparâmetros precisam ser usados ρ , $\rho = 0.9$ que é similar ao *momentum* e está relacionado com a memória dos pesos anteriores para a atualização e o segundo Épsilon, $\epsilon = 1e10^{-8}$ que está relacionado com decaimento gradual da taxa de aprendizagem durante o processo de treinamento (CandelandParmar; 2015).

Os métodos de otimização dos algoritmos são interativos e necessitam de um critério de parada (*stoppingmetric*). Dentro do pacote H2O existem diversas possibilidades, mas os dois escolhidos de forma automática pelo sistema foram com base no erro quadrático médio (RMSE) com o valor de tolerância definido igual a zero para convergir ao menor erro possível. E a segunda métrica chamada *stopping round* permite que o modelo interrompa o treinamento caso o valor de critério de parada não melhore para um dado valor específico das rodadas de treinamentos, a partir de uma média móvel simples.

Abaixo resumo dos hiperparâmetros usados no treinamento da rede neural, incluindo os

valores pré-determinados pelo algoritmo:

Tabela 1- Hiperparâmetros da rede neural

Hiperparâmetro	Valor
Função de ativação	Tanh
n-fold	5
Época	638
ρ	0.9
ϵ	$1e10^{-8}$
<i>Momentum ramp</i>	$1e10^6$
<i>RateAnealling</i>	$1e10^{-6}$
Taxa de aprendizagem	0.005
<i>Stopping metric</i>	MSE
<i>Stopping tolerance</i>	0
<i>Stopping round</i>	5

2. Avaliação do desempenho dos modelos

Utiliza-se a avaliação do modelo para determinar o bom desempenho na previsão de novos dados. A saída de um modelo de regressão é um valor numérico, cujo existem valores de referência que ajudam no cálculo do erro cometido pelo modelo.

As bases de dados de validação e treinamento foram avaliadas por diferentes indicadores estatísticos calculados dentro do software R pela função *modeval* da biblioteca *Sirad*. Dentre eles estão, RMSE, MSE, MAE e R^2 .

3. Resultados preliminares

O modelo gerado com a curva de aprendizagem com comportamento considerável obteve um comportamento descendente do erro (MSE) à medida que ocorria a aprendizagem do modelo na sequência de época. Dois fatores podem influenciar em termos de aprendizagem do modelo, o número de épocas e a taxa de aprendizagem. Observa-se na figura 3 (A) que o modelo consegue

convergir, visto que as curvas estão juntas e apresentaram um comportamento descendente semelhante ao longo das épocas demonstrando estabilidade e boa generalização. Esta indicação de boa generalização, ou ausência de overfitting entra em concordância com os valores da tabela 2, no qual os valores de R2 para as amostras de treinamento e validação são 0,99. O gráfico de dispersão entre os valores horários observados e previstos, com dados das três alturas utilizadas, indica proximidade com a linha 1:1 e com R2 0,99, o que apoia a ideia da altura ser uma variável super importante para radiação gama e exibe potencial preditivo dp perfil vertical do modelo (FIG. 3B).

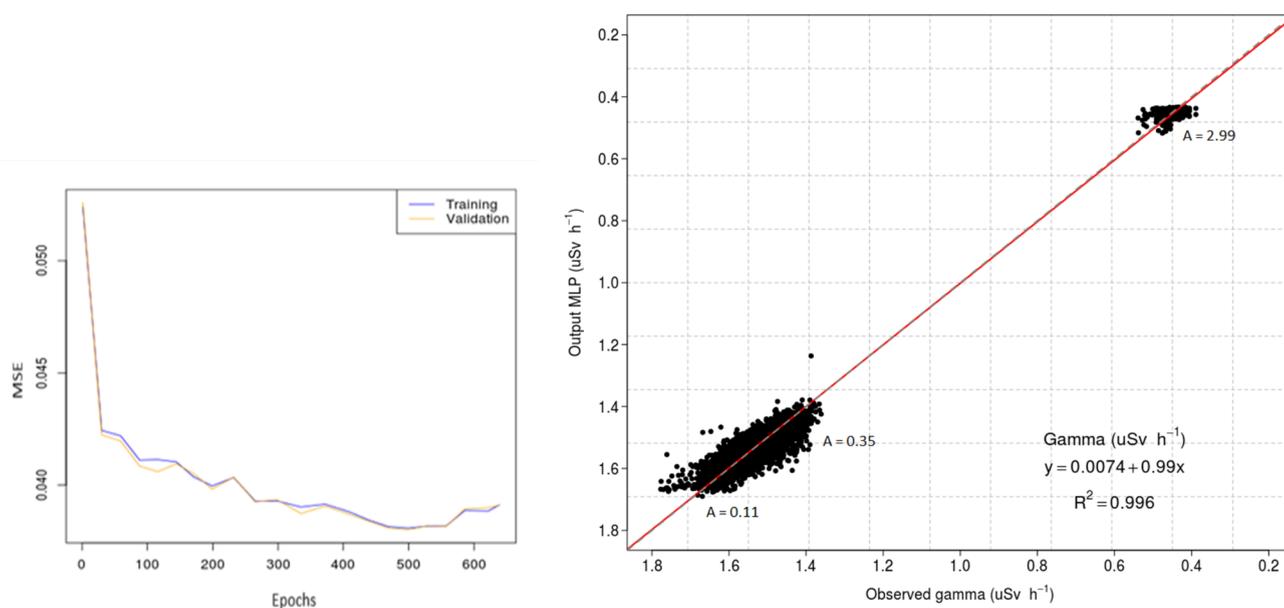


Fig.3- (A) Evolução do MSE como função de desempenho dos dados de treinamento e validação. (B) Gráfico de dispersão entre valores horários observados e previstos de intensidade radiação gama em três alturas (8.215 pontos).

O indicador MAE representa o erro (para mais ou menos) em relação ao valor correto gerado pelo modelo. O RMSE penaliza erros que são muito diferentes do valor exato. Na tabela 2 pode-se verificar que ambas as métricas apresentam pequenas diferenças, mas os valores de MAE são menores que de RMSE, isso apresenta pequena variância entre os dados das amostras. Contudo, o modelo consegue ter um bom desempenho do comportamento médio da amostra, bem como da tendência, sem conseguir atingir todos os pontos extremos, como visto na figura 4.

Tab.2- Avaliação do modelo

Métrica	Treinan	Valid
RMSE	0.0306	0.0302
MSE	$9.3 \cdot 10^{-4}$	$9.1 \cdot 10^{-4}$
MAE	0.0224	0.0223

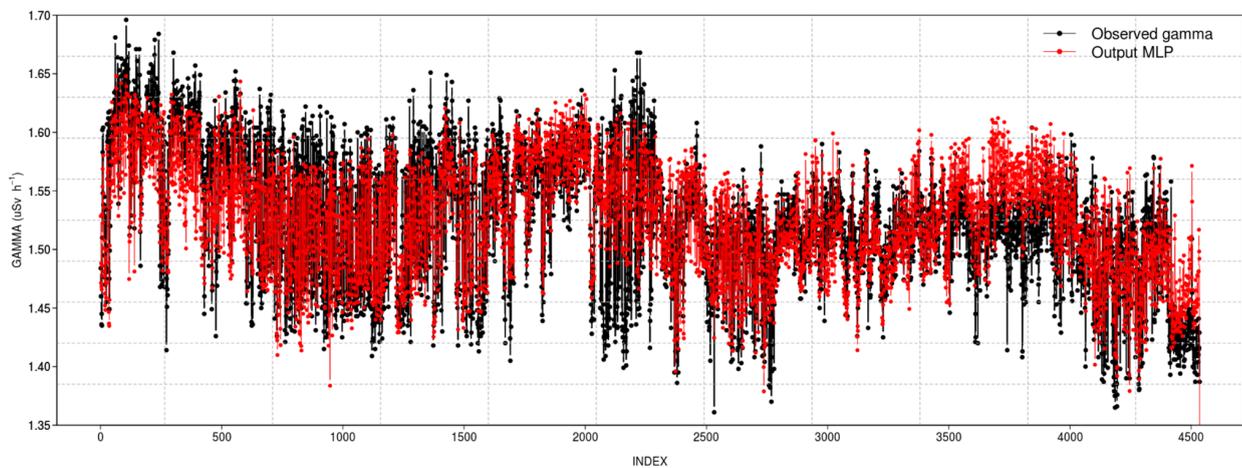


Fig.4- Comparação entre os dados observados e previstos pelo modelo para a altura 0.35 m

O local de medida está em uma região de areia monazítica e todas as medidas da intensa radiação gama são oriundas da série do Tório. Mesmo sendo uma região litorânea, a areia em si que está sendo medida na área próxima ao sensor, não entra em contato com o mar. Por isso, pode-se assumir que não ocorre movimentação, ou ocorre movimentação muito lenta dos grãos que originam o torônio ($Rn\ 220$). Outro fator que restringe a areia, em um "bloco fixo" em uma determinada e pequena região, é a grama presente em cima da areia. Por falta de movimentação da areia, não há a necessidade de pensar, nesse experimento, no acréscimo/decréscimo de material radioativo sólido (grão de areia) no ponto de medida.

O torônio é um gás que decai em aproximadamente 55 s, sendo um fator limitante da saída do solo e altura atingida. Considera-se dois fatores para descrever o baixo limite alcançado pelo torônio isolado na atmosfera, sem a presença de radônio. O primeiro é a meia vida curta do gás que limita o alcance de longas distâncias verticais, antes de decair. (Alves, 2021) afirma que em Meaípe, não há detecção de variações significativas da intensidade de radiação gama em cerca de 3m. Cigna (2005) e Fontan et al. (1966) encontraram resultados semelhantes. O segundo fator para a baixo alcance vertical do torônio, são as suas propriedades termodinâmicas, já que ele é cerca de oito vezes mais pesado que o ar atmosférico, e tende a se concentrar na atmosfera mais baixa próxima à superfície (SINGH et al., 2020). Por esses motivos, acredita-se que o transporte vertical não exerce influência relevante no alcance do torônio na atmosfera e que a turbulência atmosférica exerce papel importante na variação do torônio apenas em regiões onde

ele coexiste com o radônio que tem uma meia-vida maior (~3,8 dias), permitindo maior interação dos isótopos com os processos da camada limite planetária.

O percurso do gás, do seu surgimento até chegar ao sensor, é dividido em duas etapas: (I) percurso solo (II) percurso atmosfera e então decai.

No (I) percurso solo tem sido visto que temperatura e umidade do solo podem interferir na sua difusão até o limite solo-superfície. No (II) percurso atmosfera, por ser um gás inerte, o radônio não interage quimicamente com nada, sua movimentação pode estar associada como a de um gás ideal e a temperatura, o faz expandir e atingir os sensores fora do solo.

Não sendo possível distinguir a origem quando ocorre movimentação gás, dentro da atmosfera é possível verificar dois percursos (i) horizontal e (ii) vertical, no entanto, devido a complexidade do movimento, dedicou -se a entender apenas a variação com altura. As medidas verticais, pressupõem medidas só oriundas da região do solo logo abaixo do sensor. Tem sido observado que em médias horária da intensidade de radiação gama, a velocidade do vento medida em 4,30 m não apresentou correlação com a variação na (i) movimentação vertical, uma vez que o gás decai em menos de 3,00m. No local, há poucos acumulados de precipitação (Alves, 2021).

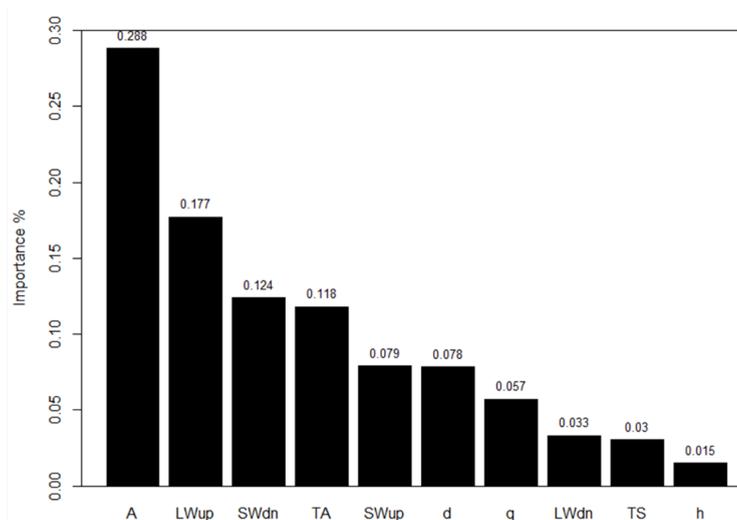


Fig. 4 - Escala de importância das variáveis

Dentre várias configurações de variáveis de entrada, houve uma melhora no desempenho na aprendizagem da rede neural quando adicionada as variáveis dia do ano e hora, nas quais, parecem ajudar o algoritmo a entender as variações sazonais e diárias das variáveis. Considerando que o gás radônio comporta-se como um gás ideal, grandezas termodinâmicas, como a temperatura do ar podem auxiliar na sua movimentação. A radiação LWup é responsável pelo seu

aquecimento, o dia do ano, pode ajudar a mostrar quais são os dias de verão que ocorrem maior aquecimento. Os átomos de torônio que chegam na superfície do solo são transportados para a atmosfera pelo processo de exalação. A temperatura do solo e temperatura do ar é uma grandeza física que dá ao gás a energia térmica para aumentar seu grau de agitação médio e conseguir se expandir. Essa agitação térmica das moléculas em escala microscópica contribui para o gás chegar ao sensor antes de decair. A hora pode ser associada à exposição solar, fator que contribui para aumento de temperatura do solo e ar durante o dia. A umidade específica é um fator que ajuda a verificar a importância da umidade no comportamento da radiação gama na atmosfera, uma vez que não temos suas medidas no solo mas sabemos da sua relevância. Como ela aparece como sendo uma das três mais importantes, pode-se associar como sendo um fator de impedimento de exalação solo-atmosfera, quando ocorre a presença de muita saturação de água na atmosfera.

4. Atividades realizadas:

A aluna:

- Foi aprovada na avaliação de proficiência em inglês (2°SEM/2020).
- Obteve os créditos necessários para o depósito da tese (1°SEM/2021)
- Realizou o estágio supervisionado (2°SEM/2021).
- Foi aprovada no exame de qualificação com nota 8,4 (1°SEM/2022).
- Participou da Conferência AIA - 2023 (2°SEM/2023)
- Ganhou prêmio - Melhor apresentação pôster AIA 2023 (2°SEM/2023)
- Participou da V Escola de Física Jayme Tiomno - (2°SEM/2023)

5. Próximas etapas

- Análise, interpretação e discussão dos resultados.
- Escrever a Tese e Artigo

6. Cronograma:

As atividades necessárias para o desenvolvimento da tese estão elencadas abaixo e distribuídas dentro do período:

- a) Análise, interpretação e discussão dos resultados.
- b) Elaboração de artigos científicos e da tese,

c) Revisão.

d) Versão final da tese.

Tabela 3 – Cronograma de Atividades.

Atividades	8S 2023	9S 2024	10S 8/2024
a)			
b)			
c)			
d)			

7. Referência bibliográfica

- [1] ALVES, T. P. D. S. (2021) Investigação observacional da atmosfera e da radioatividade natural emitida pelas areias monazíticas de Meaípe/Guarapari/ES – Brasil. (Dissertação de mestrado, Instituto de Astronomia, Geofísica e Ciências Atmosféricas, Universidade de São Paulo)
- [2] BRAGA, A. D. P., CARVALHO, A. P. L. F., LUDERMIR, T. B. (1998). Fundamentos de Redes Neurais Artificiais: 11ª Escola de Computação. DCC/IM, COOPE/Sistemas, NCE/UFRJ, Rio de Janeiro. 246p.
- [3] BOTTOU, L. (1998). “Online Algorithms and Stochastic Approximations”. Online Learning and Neural Networks. Cambridge University Press. ISBN 978-0-521-65263-6
- [4] BOTTOU, L. (2010). “Large-scale machine learning with SGD.” Proceedings of COMPSTAT’2010. Physica-Verlag HD. 177-186.
- [5] BOTTOU, L. (2012) “SGD tricks.” Neural Networks: Tricks of the Trade. Springer Berlin Heidelberg, 421-436.
- [6] CIGNA, A. A. (2005) Radon in caves. International Journal of Speleology, v. 34, n. 1, p. 1.
- [7] COUTINHO, E. R., SILVA, R. M. D., MADEIRA, J. G. F., COUTINHO, P. R. D. O. D., BOLOY, R. A. M., and DELGADO, A. R. S. (2018). Application of Artificial Neural Networks (ANNs) in the Gap Filling of Meteorological Time Series. Revista Brasileira de

Meteorologia, 317-328.

- [8] FONTAN, J. (1966). Measurement of the diffusion of radon, thoron and their radioactive daughter products in the lower layers of the Earth's atmosphere. *Tellus*, v. 18, n. 2-3, p. 623-632.
- [9] HASSAN, M. A., KHALIL, A. KASEB, S. and KASSEM, M. A. (2017). Potential of four different machine-learning algorithms in modeling daily global solar radiation. *Renewable Energy journal*. 52-62.
- [10] KROSE, B. and VAN DER SMAGT, P. *An Introduction to Neural Networks*, eighth ed., University of Amsterdam, 1996.
- [11] M. BOZNAR, M. LESJAK, P. MLAKAR. (1993) .Neural network-based method for short-term predictions of ambient SO₂ concentrations in highly polluted industrial areas of complex terrain, *Atmospheric Environment*. 27B:221 .
- [12] MLAKAR P, BONAR M.(1997)Perceptron neural-network-based model predicts air pollution. In: Adeli H,editor. *Intelligent Information Systems IIS97*, Grand Bahama Island, Bahamas, December 8–10,1997. Proceedings. Los Alamos, CA: IEEE Computer Society;. p. 345.
- [13] MLAKAR P. (1997) Determination of features for air-pollution forecasting models. In: Adeli H, editor. *Intelligent Information Systems IIS97*, Grand Bahama Island, Bahamas, December 8–10, 1997. Proceedings. Los Alamos, CA: IEEE Computer Society;. p. 350–54.
- [14] SILVA, I. D., SPATTI, D. H., and FLAUZINO, R. A. (2010). *Redes neurais artificiais para engenharia e ciências aplicadas*. São Paulo: Artliber, 23(5), 33-111.
- [15] SILVA, L. P., PERES, S. M., BOSCARIOLI, C. (2016). *C. Introdução a mineração de dados: com aplicações em R*. First.
- [16] SINGH, B. (2020) Quantification of radon/thoron exhalation rates of soil samples collected from district Faridabad of Southern Haryana, India. *Journal of Radioanalytical and Nuclear Chemistry*, v. 326, n. 1, p. 831-843.