



Universidade Federal da Paraíba  
Centro de Energias Alternativas e Renováveis  
Departamento de Engenharia Elétrica  
Disciplina de Automação Inteligente

Miguel Marques Ferreira

**Orientador:** Prof. Dr. Juan Moises Mauricio Villanueva

## **Redes Neurais Artificiais**

### **Avaliação 01**



João Pessoa - PB

Fevereiro, 2024

Miguel Marques Ferreira  
**Orientador:** Prof. Dr. Juan Moises Mauricio Villanueva

## **Redes Neurais Artificiais**

### **Avaliação 01**

Relatório técnico apresentado como requisito parcial para aprovação na disciplina de Automação Inteligente, graduação em Engenharia Elétrica.

Universidade Federal da Paraíba – UFPB  
Centro de Energias Alternativas e Renováveis – CEAR  
Departamento de Engenharia Elétrica – DEE  
Disciplina de Automação Inteligente

João Pessoa - PB  
Fevereiro, 2024

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>3</b>
<b>2</b>	<b>DESENVOLVIMENTO</b>	<b>4</b>
<b>2.1</b>	<b>Análise dos dados</b>	<b>4</b>
2.1.1	Cálculo da Energia	4
2.1.2	Cálculo da Taxa de Crescimento de Energia	5
2.1.3	Criação do banco de dados	6
2.1.4	Correlação dos dados	6
<b>2.2</b>	<b>Conjunto de dados de treinamento, validação e teste</b>	<b>7</b>
<b>2.3</b>	<b>Modelos de previsão de demanda semanal de potência</b>	<b>8</b>
2.3.1	Modelo RNA 1	8
2.3.2	Modelo RNA 2	8
2.3.3	Métricas para avaliação de desempenho dos modelos	9
<b>2.4</b>	<b>Hiperparametrização das RNAs</b>	<b>10</b>
<b>2.5</b>	<b>Otimizador Adam</b>	<b>10</b>
<b>3</b>	<b>RESULTADOS E DISCUSSÃO</b>	<b>12</b>
<b>3.1</b>	<b>Desempenho das RNAs para entradas normalizadas</b>	<b>12</b>
3.1.1	Modelo RNA 1	12
3.1.2	Modelo RNA 2	13
<b>3.2</b>	<b>Desempenho das RNAs para entradas não normalizadas</b>	<b>14</b>
3.2.1	Modelo RNA 1	14
3.2.2	Modelo RNA 2	14
<b>3.3</b>	<b>Desempenho das RNAs para diferentes funções de ativação</b>	<b>15</b>
3.3.1	Modelo RNA 1	17
3.3.2	Modelo RNA 2	17
<b>3.4</b>	<b>Melhores Modelos de RNAs</b>	<b>20</b>
3.4.1	Modelo RNA 1	21
3.4.2	Modelo RNA 2	21
<b>4</b>	<b>CONCLUSÃO</b>	<b>25</b>

# 1 Introdução

Uma subestação é uma instalação elétrica responsável por receber, transformar, distribuir e controlar a energia elétrica. Ela atua como um ponto de conexão entre diferentes níveis de tensão em um sistema elétrico, permitindo a transmissão eficiente de eletricidade entre geradores, linhas de transmissão e consumidores. As subestações desempenham um papel crucial na distribuição e controle de energia elétrica em uma rede.

Nessa conjuntura, há um elevado nível de importância para que haja correta medição e previsão de potência elétrica a fim de evitar problemas ou má entrega de energia elétrica. Assim, destacam-se entre os pontos mais importantes da previsão de potência elétrica em uma subestação a necessidade de controlar a rede elétrica, uma vez que é necessário garantir que a demanda de energia seja atendida de forma eficiente e que não ocorram sobrecargas.

Além disso, é necessário que haja eficiência operacional por meio da medição precisa da potência elétrica, a qual permite que os operadores otimizem a operação da subestação, ajustando as configurações conforme necessário para maximizar a eficiência do sistema.

A previsão de demanda, também, é crucial para planejar a capacidade da subestação, detecção de anomalias e balanceamento de cargas. Com base nas tendências históricas e em fatores sazonais, as previsões ajudam a determinar os recursos necessários para atender à demanda futura.

Nesse trabalho, portanto, será apresentando um sistema de previsão de demanda de potências futuras a partir de demandas passadas utilizado Redes Neurais Artificiais (RNA) como atividade de avaliação da unidade 1 da disciplina de Automação Inteligente do curso de Engenharia Elétrica na Universidade Federal da Paraíba (UFPB).

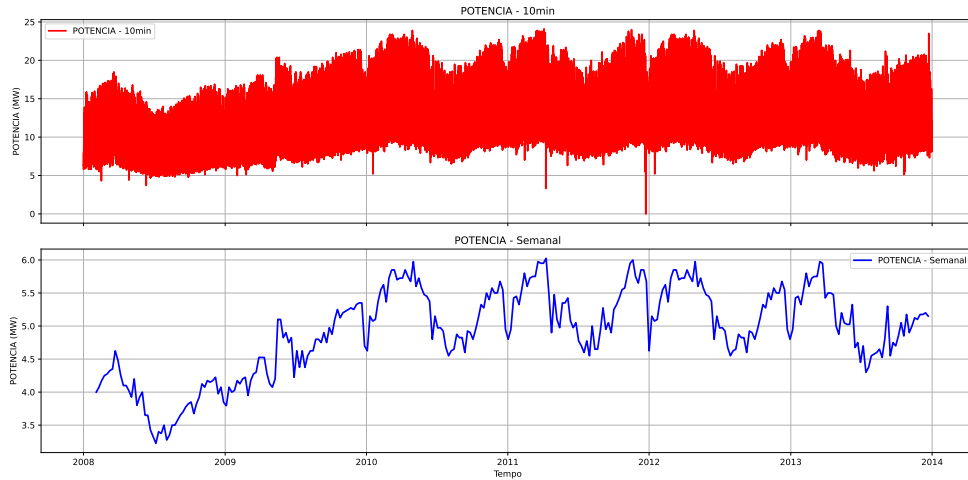
## 2 Desenvolvimento

### 2.1 Análise dos dados

Inicialmente, foi realizada uma análise exploratória dos dados para entender quais grandezas elétricas estavam disponíveis, bem como quais variáveis poderiam ser criadas com base nas grandezas disponíveis para utilizar como fonte de informações para os modelos de previsão baseados em redes neurais artificiais (RNAs). Dessa forma, foi implementado o cálculo da energia e da taxa de crescimento de energia com base na [Equação 2.1](#) e [Equação 2.2](#), respectivamente.

Além disso, os dados disponibilizados eram referentes a potência elétrica com taxa de amostragem de 15 minutos para o intervalo de tempo compreendido de janeiro de 2008 até dezembro de 2013. A [Figura 1](#) ilustra os dados de potência com a taxa de amostragem original, os dados reamostrados semanalmente com o valor máximo no período.

Figura 1 – Dados de potência elétrica.



Fonte: Autoria própria.

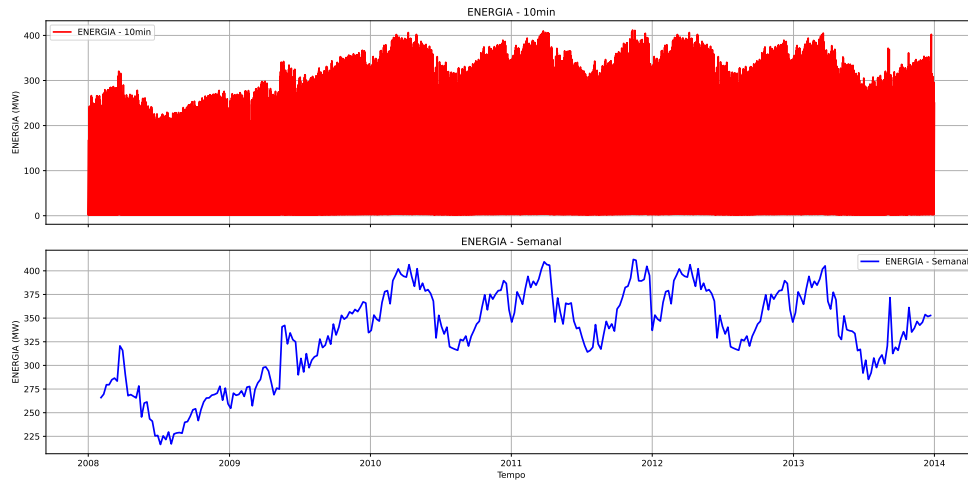
#### 2.1.1 Cálculo da Energia

Para calcular a energia consumida com base na potência medida a cada 15 minutos, você pode usar a seguinte fórmula:

$$E(k) = \sum_{n=1}^{96} \frac{P(i)}{4} \quad (2.1)$$

A Figura 2 ilustra os dados de energia com a taxa de amostragem original, os dados reamostrados semanalmente com o valor máximo no período.

Figura 2 – Dados de energia elétrica.



Fonte: Autoria própria.

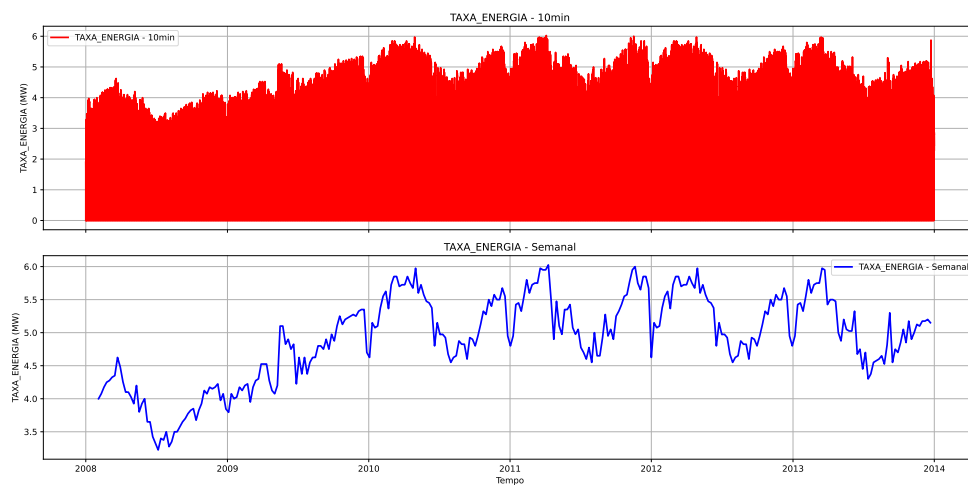
### 2.1.2 Cálculo da Taxa de Crescimento de Energia

A taxa de crescimento de energia pode ser calculada usando a seguinte fórmula:

$$\Delta E(k) = E(k) - E(k - 1) \quad (2.2)$$

A Figura 3 ilustra os dados de energia com a taxa de amostragem original, os dados reamostrados semanalmente com o valor máximo no período.

Figura 3 – Dados de taxa de crescimento de energia elétrica.



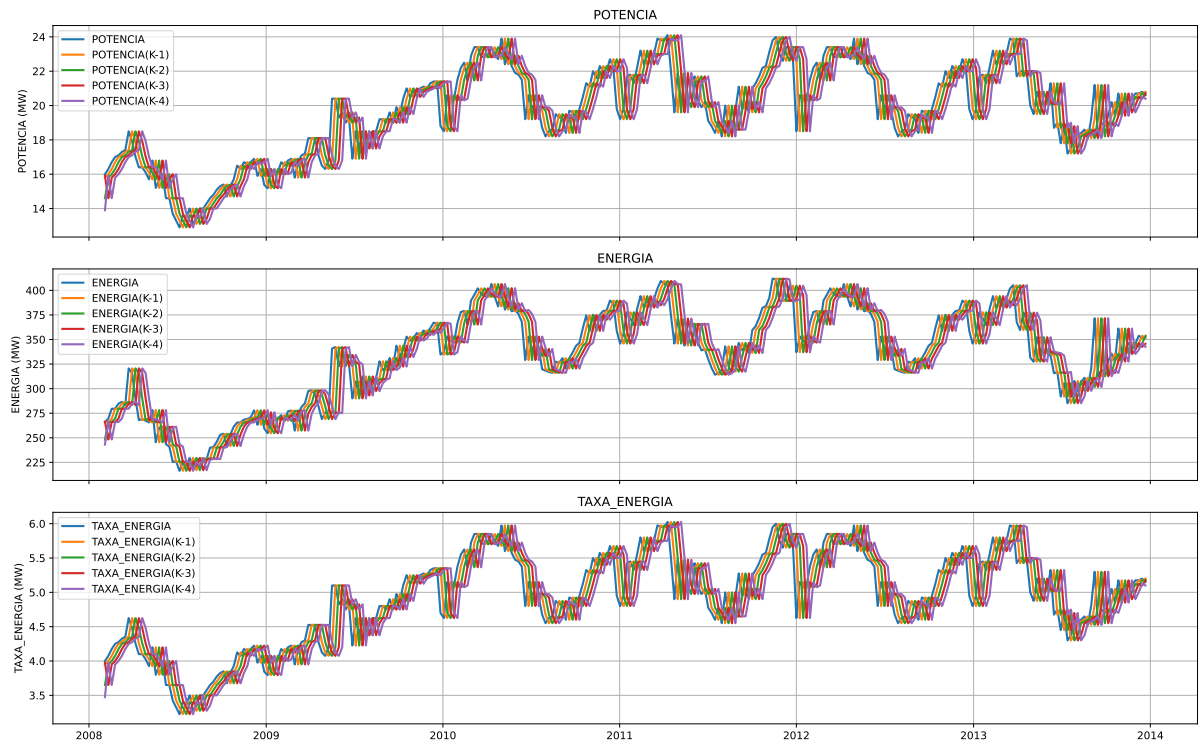
Fonte: Autoria própria.

### 2.1.3 Criação do banco de dados

Diante disso, fez-se necessário gerar o conjunto de dados para efetuar a previsão de demanda. Para tal, foram adicionadas  $N$  colunas com as  $N$  potências em instantes anteriores para alimentar as entradas do modelo (do inglês, *features*), e  $K$  colunas com potências futuras, essas utilizadas como *target* para mostrar ao modelo quais as saídas esperadas.

Na Figura 4 é mostrado um exemplo em que temos 4 dados anteriores e iremos prever 2 dados posteriores. Os valores de  $K$  até  $K - 4$  representam os dados de entrada. Na Figura 5 valores de  $K + 1$  e  $K + 2$  representam os dados de saída.

Figura 4 – Dados de potência em instantes anteriores de  $K$  até  $K - 4$ .

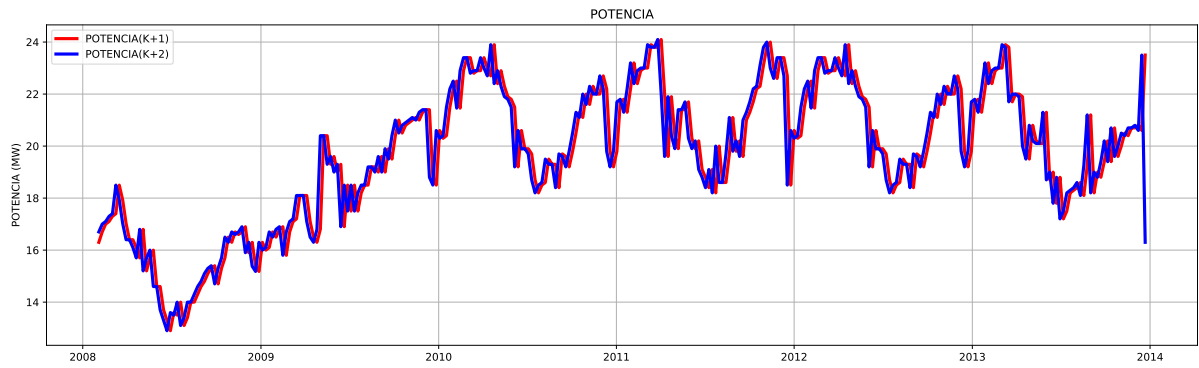


Fonte: Autoria própria.

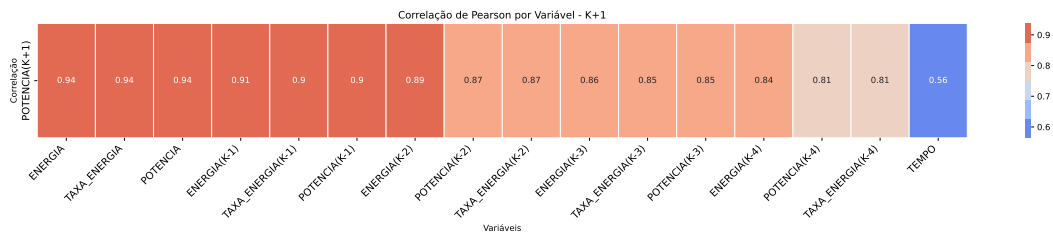
### 2.1.4 Correlação dos dados

A correlação de Pearson foi utilizada com o objetivo de quantificar a força e a direção de uma relação linear entre as variáveis de entrada com as variáveis de saída por meio de uma medida estatística.

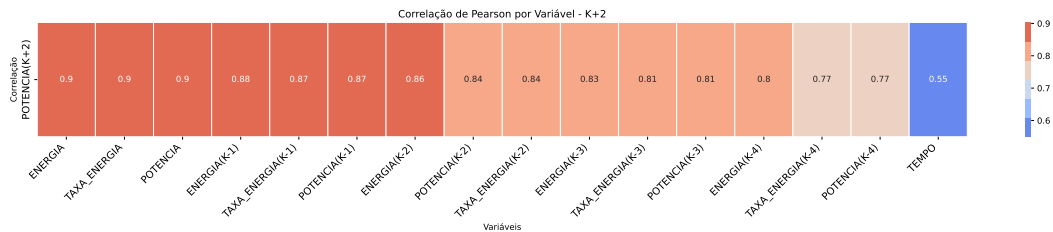
Essa correlação é frequentemente usada para avaliar como as mudanças em uma variável estão associadas às mudanças em outra variável.

Figura 5 – Dados de potência em instantes futuros de  $K$  até  $K + 2$ .

Fonte: Autoria própria.

Figura 6 – Correlação das variáveis do banco de dados com a potência  $K + 1$ .

Fonte: Autoria própria.

Figura 7 – Correlação das variáveis do banco de dados com a potência  $K + 2$ .

Fonte: Autoria própria.

## 2.2 Conjunto de dados de treinamento, validação e teste

Após a consolidação do banco de valores para os o treinamento dos modelos de RNA, realizou-se a divisão desses em três subconjuntos, sendo esses para treinamento, validação e teste. Para isso, seguiu-se a proporção de 70%, 15% e 15% da quantidade de pontos total e, sobretudo, de forma aleatória, para formar o conjunto de dados utilizados para o treinamento, validação e teste, respectivamente.

Logo após, os dados de potência, energia e taxa de energia foram normalizados com base no valor máximo ( $Z_{max}$ ) e mínimo ( $Z_{min}$ ) a partir da equação matemática apresentada em [Equação 2.3](#), uma vez que os dados de entrada possuíam escalas distintas.



$$Z_t = \frac{Z_{i,t} - Z_{min}}{Z_{max} - Z_{min}} \quad (2.3)$$

## 2.3 Modelos de previsão de demanda semanal de potência

Os modelos baseados em RNA adotados para o desenvolvimento deste trabalho são do tipo *Multilayer Preceptron* (MLP) e constituíram um sistema de previsão de demanda semanal de potências futuras com base em valores de potências passadas, como apresentado na [subseção 2.1.3](#).

A arquitetura das RNAs consistem em camadas de entrada com  $n$  neurônios e uma camada de saída com 2 neurônios, essa destinada a previsão de um valor de potência semanal  $(K + 1)$  e  $(K + 2)$ . Além disso, o otimizador padrão utilizado foi do tipo Adam, e as funções de ativação das camadas ocultas foram alteras dentre as possibilidades *relu* e *sigmoid*, enquanto a camada de saída utilizou a função *sigmoid*, a fim de evitar valores negativos como resultados da previsão de demanda semanal de potência.

### 2.3.1 Modelo RNA 1

O modelo de RNA 1 é responsável por prever a demanda semanal de potência considerando potência (P) e energia (E) de instantes anteriores como valores de entrada. A [Figura 8](#) ilustra o fluxo de informações nas entradas e saídas da RNA 1.

Figura 8 – Fluxo de informações nas entradas e saídas da RNA 1.

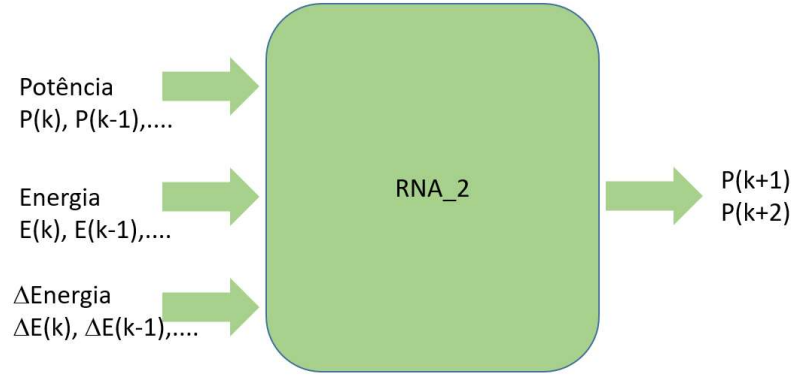


Fonte: Autoria própria.

### 2.3.2 Modelo RNA 2

O modelo de RNA 2 é responsável por prever a demanda semanal de potência considerando potência (P), energia (E) e taxa de crescimento da energia ( $\Delta E$ ) de instantes anteriores como valores de entrada. A [Figura 9](#) ilustra o fluxo de informações nas entradas e saídas da RNA 2.

Figura 9 – Fluxo de informações nas entradas e saídas da RNA 2.



Fonte: Autoria própria.

### 2.3.3 Métricas para avaliação de desempenho dos modelos

Uma vez definido os modelos de RNA para a previsão demanda semanal de potência, é necessário medir o seu desempenho a partir da análise dos erros existentes entre valores reais e previstos para, assim, ajustar os modelos para que esses se adéquem a atividade de prever valores com o menor erro de previsão possível.

Para isso, a avaliação dos modelos durante o processo de treinamento é indispensável. Dessa forma, foram utilizadas as seguintes métricas: *Mean Absolute Error* (MAE), *Mean Squared Error* (MSE) e *Root Mean Squared Error* (RMSE), bem como *Mean Absolute Percentage Error* (MAPE) e *Max Absolute Percentage Error* ( $APE_{max}$ ).

$$MAE = \frac{1}{n} \sum_{i=1}^n |Y_i - \hat{Y}_i| \quad (2.4)$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (2.5)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2} \quad (2.6)$$

$$MAPE = APE_{mean} = \frac{1}{n} \sum_{i=0}^{n-1} \frac{|Y_i - \hat{Y}_i|}{\max(\epsilon, |Y_i|)} \quad (2.7)$$

$$APE_{max} = \max\left(\frac{|Y_i - \hat{Y}_i|}{\max(\epsilon, |Y_i|)}\right) \quad (2.8)$$

## 2.4 Hiperparametrização das RNAs

Com o que foi apresentado em mente, a escolha adequada dos hiperparâmetros impacta significativamente na capacidade do modelo no aprendizado de padrões complexos e generalização para quando houver novos dados.

O método *Random Search*, portanto, surge como uma abordagem eficiente para a hiperparametrização de RNAs. Ele consiste em amostrar aleatoriamente diferentes combinações de hiperparâmetros em um espaço pré-definido e avaliar o desempenho do modelo para cada combinação.

Em contrapartida, a otimização bayesiana é outra alternativa de solução para hiperparametrização de Redes Neurais Artificiais (RNAs), uma vez que esse processo demanda um método de otimização global de funções complexas e caras computacionalmente. Em vez de tratar a função objetivo de um modelo como uma "caixa preta" e tentar encontrar o mínimo local iterativamente, a otimização bayesiana modela a função objetivo como uma distribuição de probabilidade.

A otimização bayesiana é especialmente útil quando a função objetivo é custosa de avaliar, como em casos de otimização de hiperparâmetros de modelos de ML. Juntamente com outras estratégias de otimização de hiperparâmetros, como os métodos *grid search*, *random search* e estratégias evolutivas, a otimização bayesiana tem suas vantagens fundamentadas em um menor custo computacional, a realização de uma busca guiada e, sobretudo, uma rápida convergência para os melhores hiperparâmetros globais.

Entretanto, foram realizados testes entre a otimização bayesiana e a estratégia *random search*, as quais apresentaram performance muito próximas. Dessa forma, optou-se por utilizar o *random search* para procurar a melhor configuração da rede para previsão de demanda semanal de potência.

## 2.5 Otimizador Adam

O otimizador Adam (do inglês, *Adaptive Moment Estimation*) foi o algoritmo de otimização usado no treinamento das RNAs neste trabalho. Ele combina elementos de dois outros otimizadores populares, o método do momento estocástico (SGD com momento) e o RMSprop (*Root Mean Square Propagation*), para melhorar o desempenho durante o treinamento.

Dentre os pontos que levaram a escolha do otimizador Adam, está o fato de que ele utiliza o conceito de momento para acelerar o processo de convergência. Isso ajuda a superar regiões planas ou estreitas na função de custo. Por outro lado, o otimizador Adam incorpora o RMSprop ao adaptar as taxas de aprendizado individualmente para cada parâmetro da rede neural. Tal característica ajuda a atenuar problemas relacionados

com taxas de aprendizado diferentes para diferentes parâmetros.

A fórmula geral para a atualização dos parâmetros  $w$  usando o Adam é:

$$w_{t+1} = w_t - \frac{\text{learning\_rate} \times \hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}$$

Onde:

1.  $w_{t+1}$  são os parâmetros atualizados;
2. `learning_rate` é a taxa de aprendizado;
3.  $\hat{m}_t$  é a média móvel exponencial dos gradientes (momento);
4.  $\hat{v}_t$  é a média móvel exponencial dos gradientes ao quadrado (RMSprop);
5.  $\epsilon$  é uma pequena constante para evitar divisão por zero.

## 3 Resultados e Discussão

### 3.1 Desempenho das RNAs para entradas normalizadas

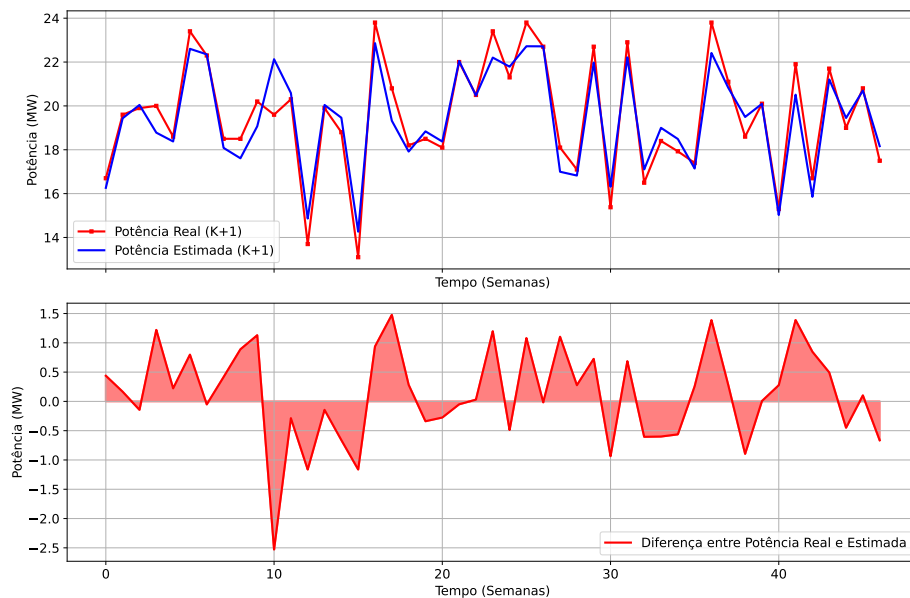
Para cada modelo de previsão de demanda semanal de potência, foi utilizada uma arquitetura semelhante para analisar a necessidade da utilização de entradas normalizadas.

Assim, para o contexto de comparação, foi aplicada ao modelo RNA 1 e RNA 2 uma arquitetura consistente em uma camada de entrada com  $n$  neurônios ( $n$  como a quantidade de variáveis), duas camadas ocultas com função de ativação *relu* e com 25 e 16 neurônios, respectivamente, bem como uma camada de saída com 2 neurônios. Além disso, o otimizador utilizado foi do tipo Adam, e a função de ativação da camada de saída foi a função *sigmoid*, a fim de evitar valores negativos como resultados da previsão.

#### 3.1.1 Modelo RNA 1

A [Figura 10](#) apresenta a previsão de demanda semanal do modelo de RNA 1 para uma semana à frente, isto é, potência ( $K + 1$ ).

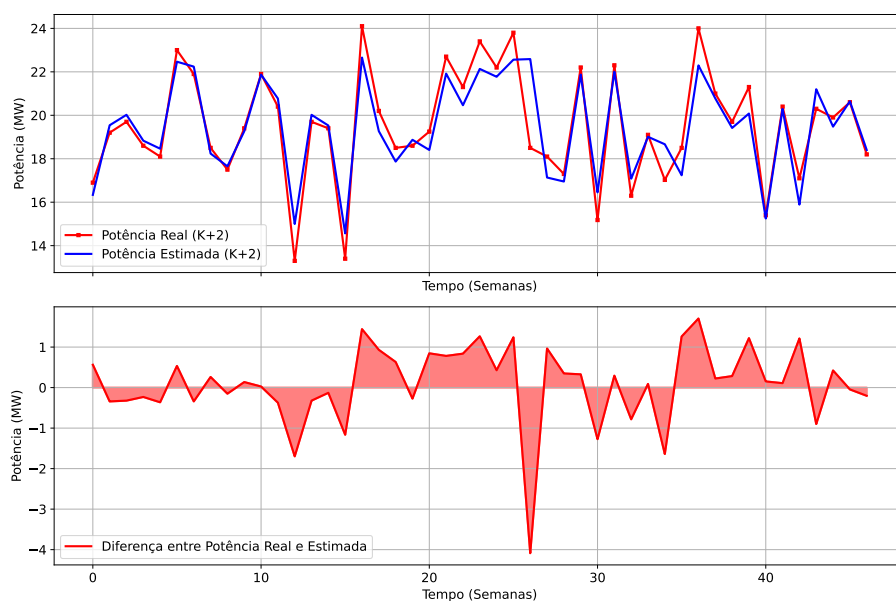
Figura 10 – RNA 1 - Entradas normalizadas e saídas na faixa real de valores.



Fonte: Autoria própria.

De forma semelhante, a [Figura 11](#) apresenta a previsão de demanda semanal do modelo de RNA 1 para duas semanas à frente, isto é, potência ( $K + 2$ ).

Figura 11 – RNA 1 - Entradas normalizadas e saídas na faixa real de valores.

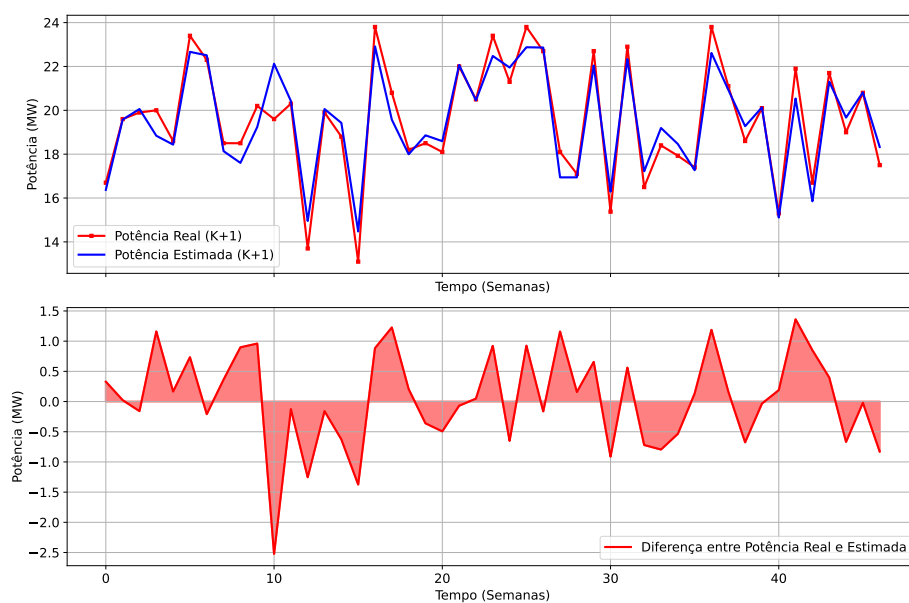


Fonte: Autoria própria.

### 3.1.2 Modelo RNA 2

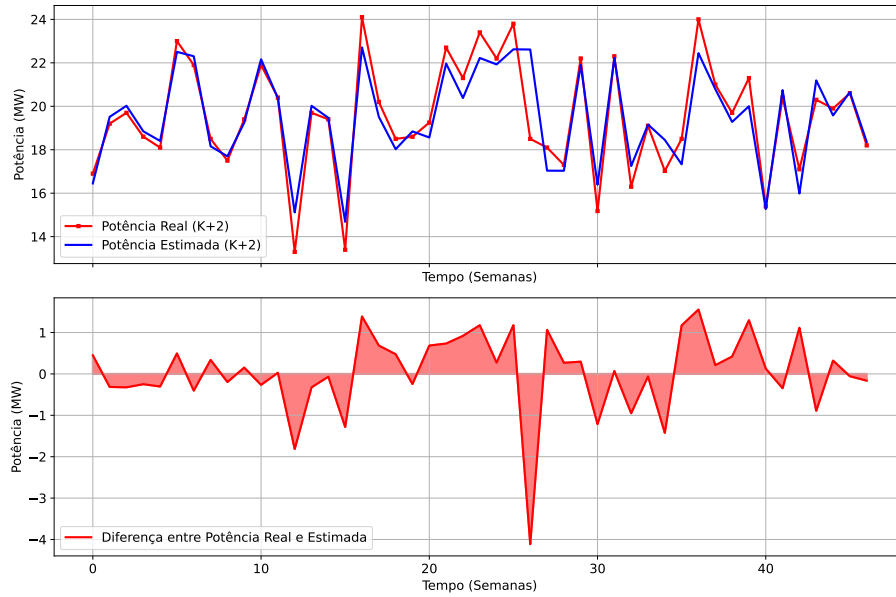
A [Figura 12](#) apresenta a previsão de demanda semanal do modelo de RNA 2 para uma semana à frente, isto é, potência ( $K + 1$ ), enquanto a [Figura 13](#) apresenta a previsão de demanda semanal do modelo de RNA 2 para duas semanas à frente.

Figura 12 – RNA 2 - Entradas normalizadas e saídas na faixa real de valores.



Fonte: Autoria própria.

Figura 13 – RNA 2 - Entradas normalizadas e saídas na faixa real de valores.



Fonte: Autoria própria.

## 3.2 Desempenho das RNAs para entradas não normalizadas

Dessa vez, submetendo os modelos de RNA 1 e 2 um conjunto de dados de entrada sem a aplicação da normalização, foi visto que ambas as redes contaram com suas saídas saturadas. Para tal, não houve convergência no processo de otimização durante o treinamento e as redes não conseguiram aprender corretamente.

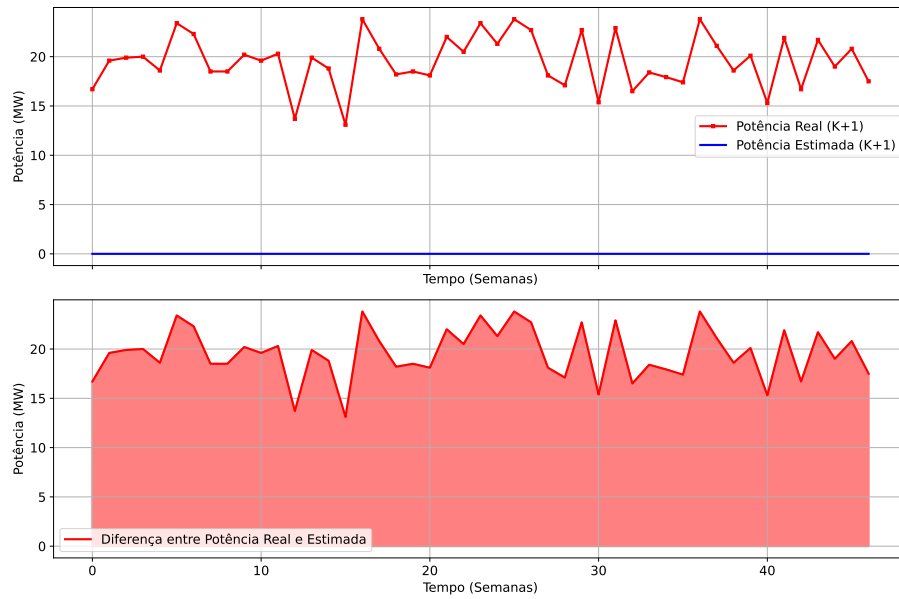
### 3.2.1 Modelo RNA 1

A [Figura 14](#) e [Figura 15](#) mostram que a falta da normalização das entradas para reduzir a escala dos dados, ocasionou o não aprendizado da rede e, possivelmente ocasionou problemas de estabilidade numérica durante o treinamento.

### 3.2.2 Modelo RNA 2

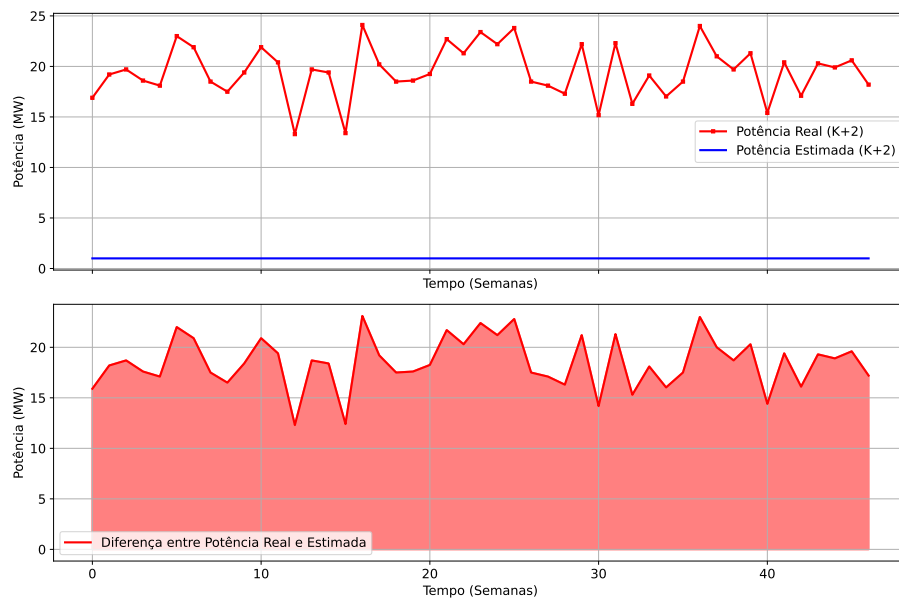
De forma semelhante ao que aconteceu com os modelos de RNA 1, a [Figura 16](#) e [Figura 17](#) indicam também que a não normalização das entradas do modelo ocasionou a saturação das saídas do modelo.

Figura 14 – RNA 1 - Entradas não normalizadas.



Fonte: Autoria própria.

Figura 15 – RNA 1 - Entradas não normalizadas.



Fonte: Autoria própria.

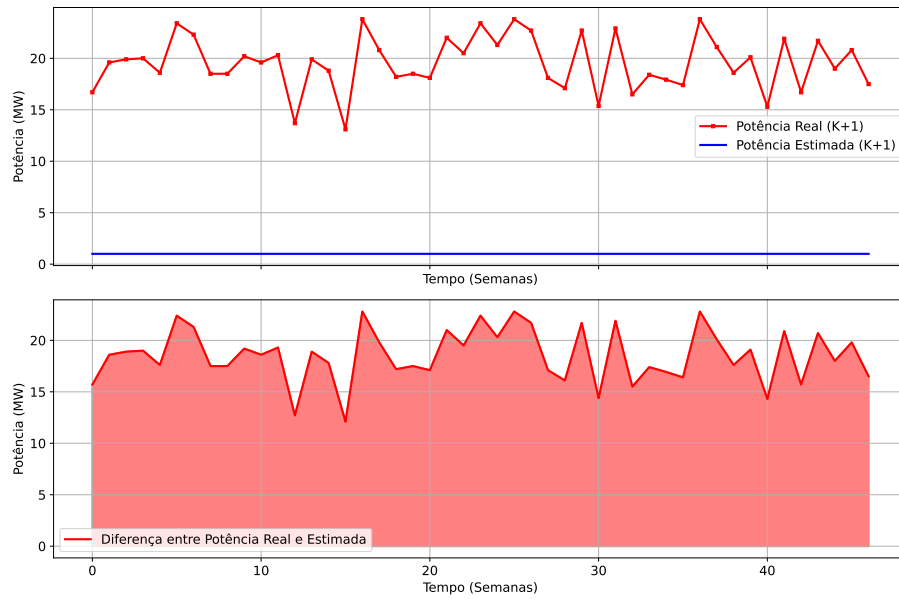
### 3.3 Desempenho das RNAs para diferentes funções de ativação

O desempenho das RNAs pode variar para diferentes funções de ativação devido às características intrínsecas de cada função.

Para comparação, foi aplicada ao modelo RNA 1 e RNA 2 uma arquitetura

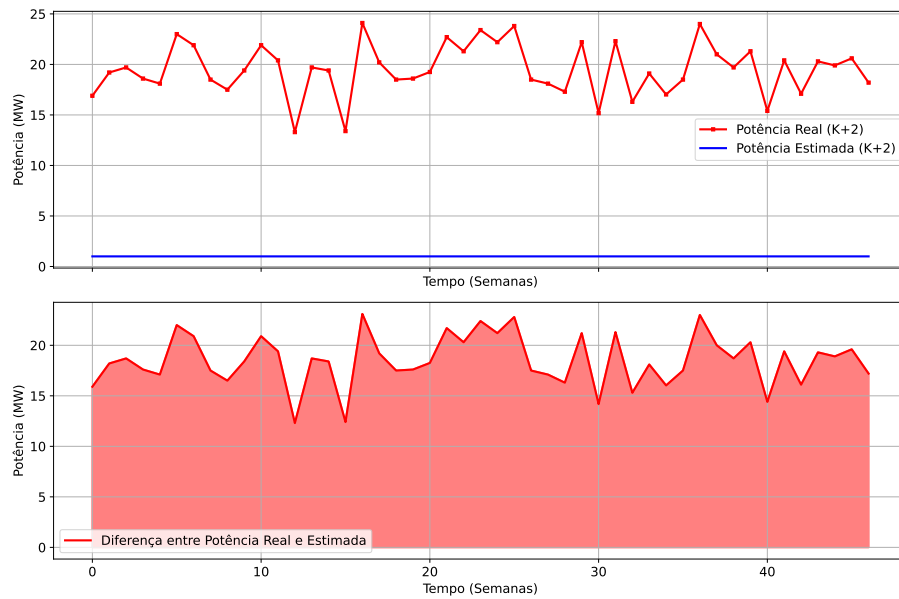


Figura 16 – RNA 2 - Entradas não normalizadas.



Fonte: Autoria própria.

Figura 17 – RNA 2 - Entradas não normalizadas.



Fonte: Autoria própria.

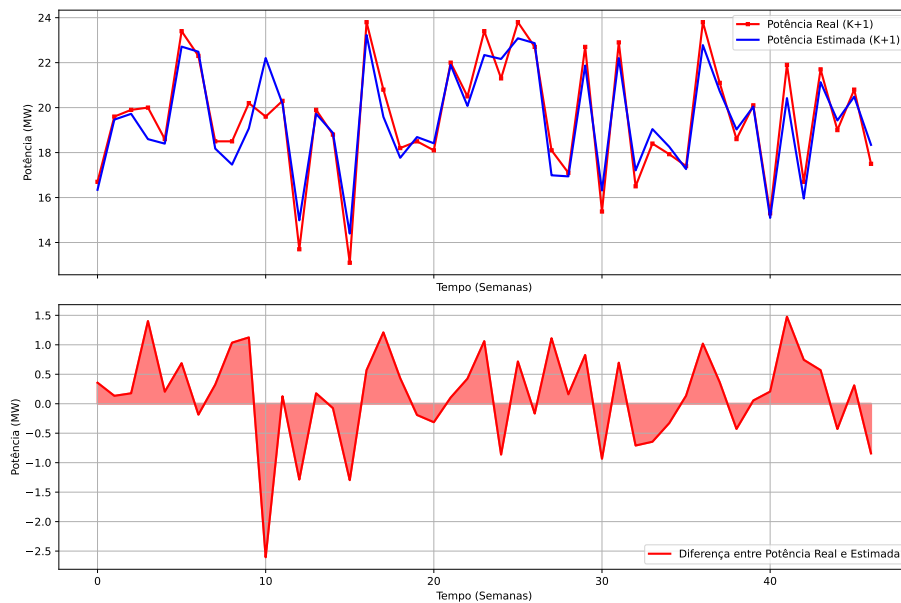
consistente em uma camada de entrada com  $n$  neurônios ( $n$  como a quantidade de variáveis), duas camadas ocultas com função de ativação *relu* e com 25 e 16 neurônios, respectivamente, bem como uma camada de saída com 2 neurônios. Além disso, o otimizador utilizado foi do tipo Adam, e a função de ativação da camada de saída foi a função *sigmoid*, a fim de evitar valores negativos como resultados da previsão.

Para o modelo 1, foi testadas em suas camadas ocultas as funções de ativação *relu* e *sigmoid* a fim de verificar a convergência no treinamento e, sobretudo, a performance dos modelos calculada com base na previsão da demanda semanal de potência futura. Dessa forma, ambas as funções de ativação nas camadas ocultas fizeram com que o modelo tivesse convergência durante a etapa de treinamento, bem como boas métricas de erros entre seus resultados previstos e reais de potência.

### 3.3.1 Modelo RNA 1

Assim, a Figura 18 e Figura 19 ilustram as saídas do modelo 1 quando suas camadas ocultas possuíam função de ativação *relu*.

Figura 18 – RNA 1 - Camadas ocultas com função de ativação *relu*.



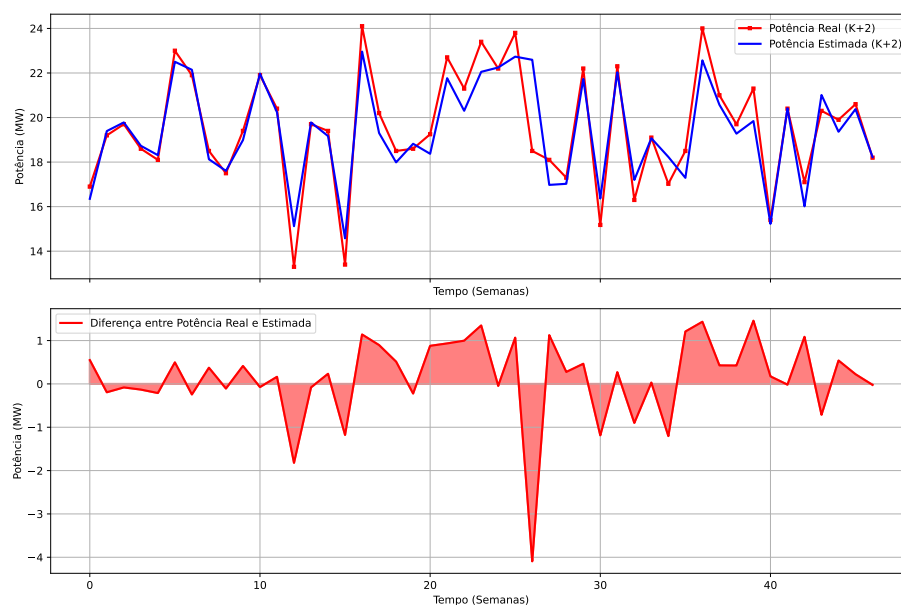
Fonte: Autoria própria.

A Figura 20 e Figura 21 ilustram as saídas do modelo 1 quando suas camadas ocultas possuíam função de ativação *sigmoid*.

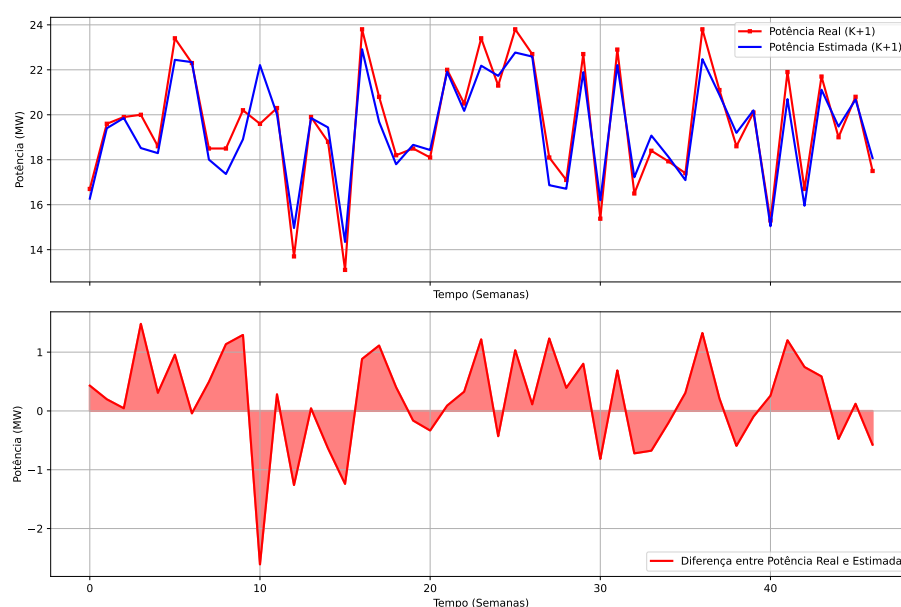
Métrica		MAE	MSE	RMSE	MAXE	MAPE	Max_APE
<i>relu</i>	K + 1	0.6219	0.6342	0.7964	2.6046	3.2708	13.2890
	K + 2	0.6735	0.9350	0.9669	4.0929	3.5963	22.1235
<i>sigmoid</i>	K + 1	0.6508	0.6798	0.8245	2.6100	3.4093	13.3163
	K + 2	0.6830	0.9338	0.9663	3.9903	3.6368	21.5693

### 3.3.2 Modelo RNA 2

Assim, a Figura 22 e Figura 23 ilustram as saídas do modelo 1 quando suas camadas ocultas possuíam função de ativação *relu*.

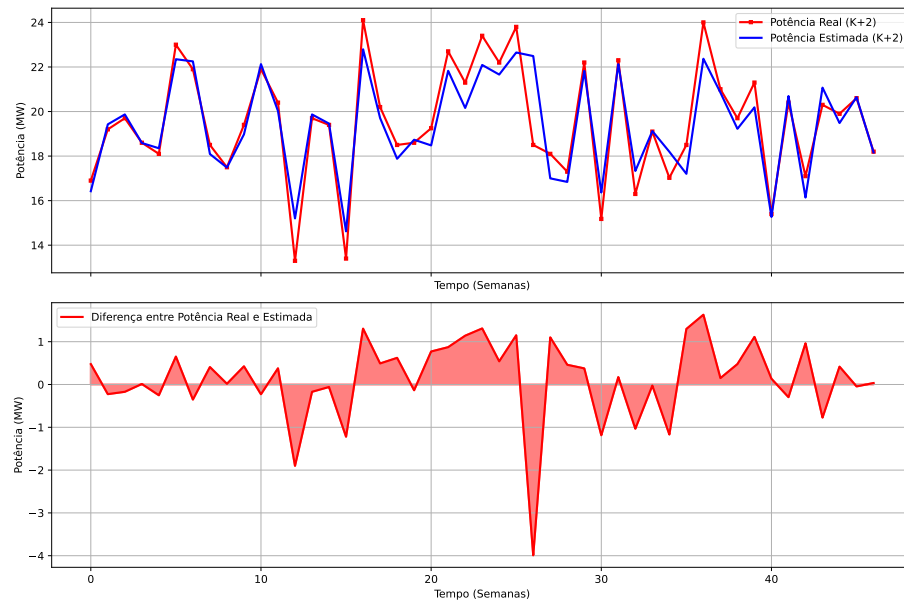
Figura 19 – RNA 1 - Camadas ocultas com função de ativação *relu*.

Fonte: Autoria própria.

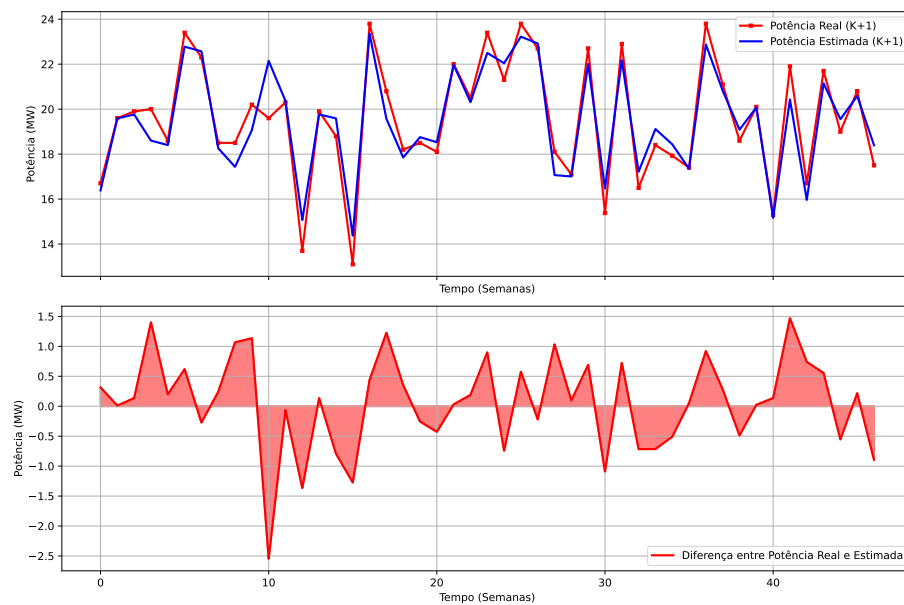
Figura 20 – RNA 1 - Camadas ocultas com função de ativação *sigmoid*.

Fonte: Autoria própria.

A [Figura 24](#) e [Figura 25](#) ilustram as saídas do modelo 1 quando suas camadas ocultas possuíam função de ativação *relu*.

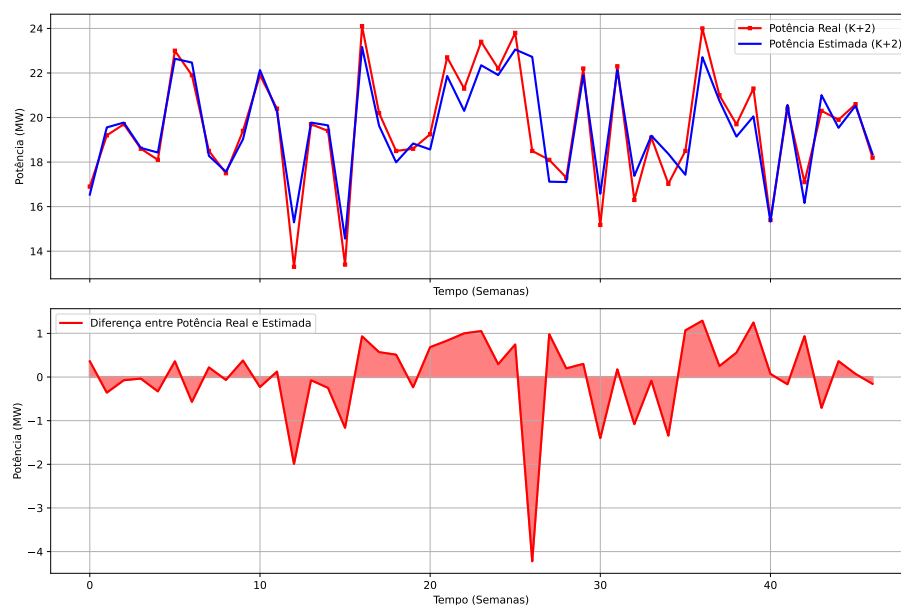
Figura 21 – RNA 1 - Camadas ocultas com função de ativação *sigmoid*.

Fonte: Autoria própria.

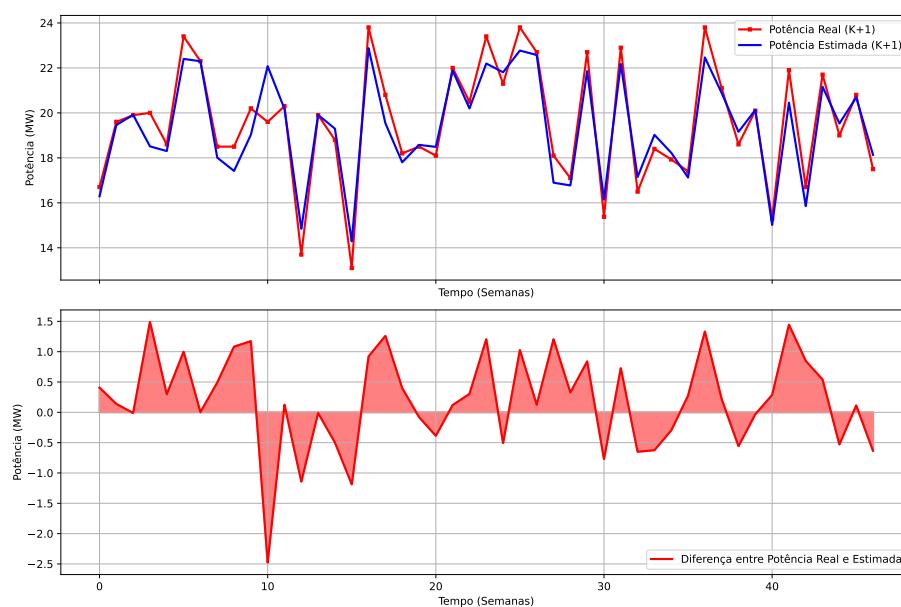
Figura 22 – RNA 2 - Camadas ocultas com função de ativação *relu*.

Fonte: Autoria própria.

Métrica		MAE	MSE	RMSE	MAXE	MAPE	Max_APE
<i>relu</i>	K + 1	0.6134	0.6262	0.7913	2.5474	3.2516	12.9970
	K + 2	0.6404	0.8956	0.9464	4.2223	3.4590	22.8232
<i>sigmoid</i>	K + 1	0.6404	0.6677	0.8171	2.4767	3.3406	12.6363
	K + 2	0.6772	0.9203	0.9593	3.8894	3.5796	21.0239

Figura 23 – RNA 2 - Camadas ocultas com função de ativação *relu*.

Fonte: Autoria própria.

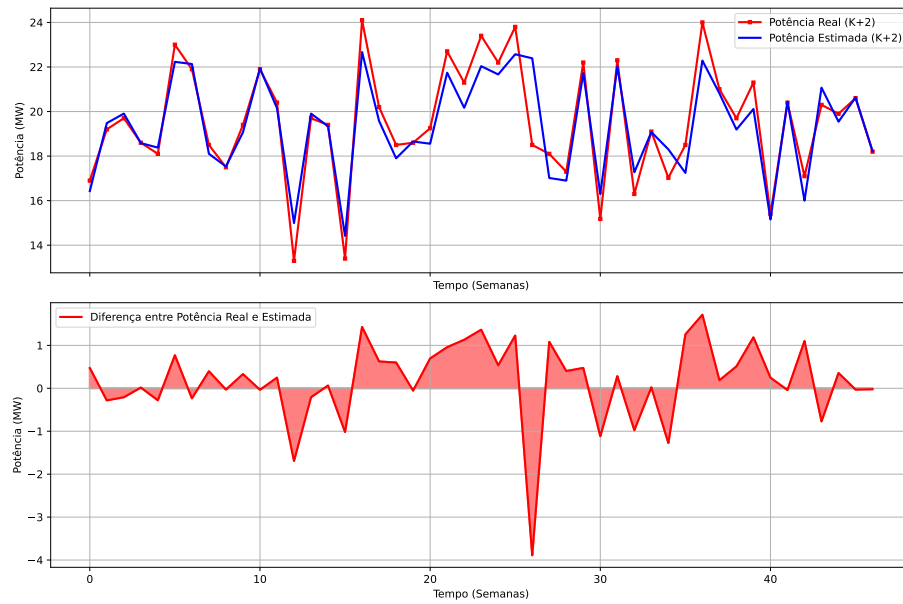
Figura 24 – RNA 2 - Camadas ocultas com função de ativação *sigmoid*.

Fonte: Autoria própria.

### 3.4 Melhores Modelos de RNAs

Na fase de teste, última etapa deste trabalho, buscou-se encontrar a melhor rede neural com as seguintes condições:

1. Máximo Erro Percentual Absoluto < 10%;

Figura 25 – RNA 2 - Camadas ocultas com função de ativação *sigmoid*.

Fonte: Autoria própria.

2.  $\text{MAPE} < 5\%$ ;

### 3.4.1 Modelo RNA 1

A melhor configuração encontrada para o modelo RNA 1 consiste em uma com uma camada de entrada com  $n$  neurônios ( $n$  como a quantidade de variáveis), duas camadas ocultas com função de ativação *relu* e com 68 e 70 neurônios, respectivamente, bem como uma camada de saída com 2 neurônios. Além disso, o otimizador utilizado foi do tipo Adam, e a função de ativação da camada de saída foi a função *sigmoid*, a fim de evitar valores negativos como resultados da previsão.

Na [Tabela 1](#) estão disponibilizadas todas as métricas para avaliação da performance do modelo RNA 1 para a previsão de demanda de potência elétrica.

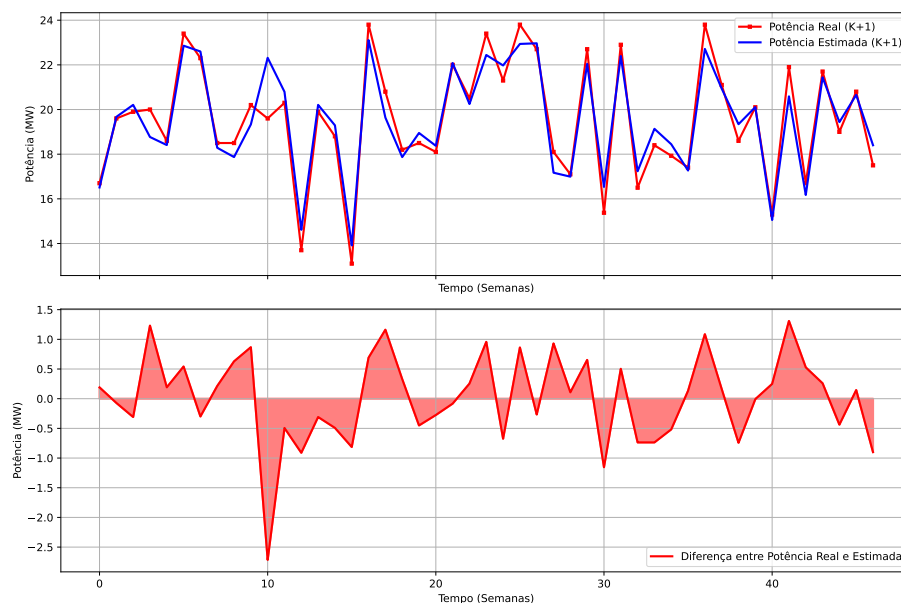
Tabela 1 – RNA 1 - Métricas para avaliação da performance do modelo.

Métricas	MAE	MSE	RMSE	MAXE	MAPE	Max_APE
<b>K + 1</b>	0.5866	0.5604	0.7486	2.7159	3.0610	13.8564
<b>K + 2</b>	0.6813	0.9138	0.9559	4.1299	3.6025	22.3236

### 3.4.2 Modelo RNA 2

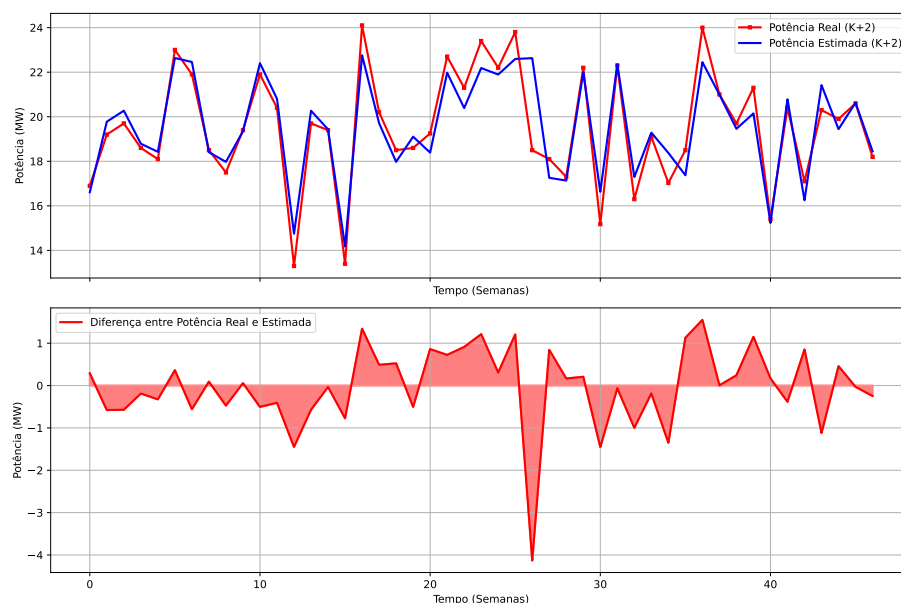
A melhor configuração encontrada para o modelo RNA 2 consiste em uma com uma camada de entrada com  $n$  neurônios ( $n$  como a quantidade de variáveis), três camadas ocultas com função de ativação *relu* e com 84, 78 e 19 neurônios, respectivamente, bem

Figura 26 – RNA 1 - Previsão de demanda semanal de potência (K+1) pelo melhor modelo encontrado.



Fonte: Autoria própria.

Figura 27 – RNA 1 - Previsão de demanda semanal de potência (K+2) pelo melhor modelo encontrado.

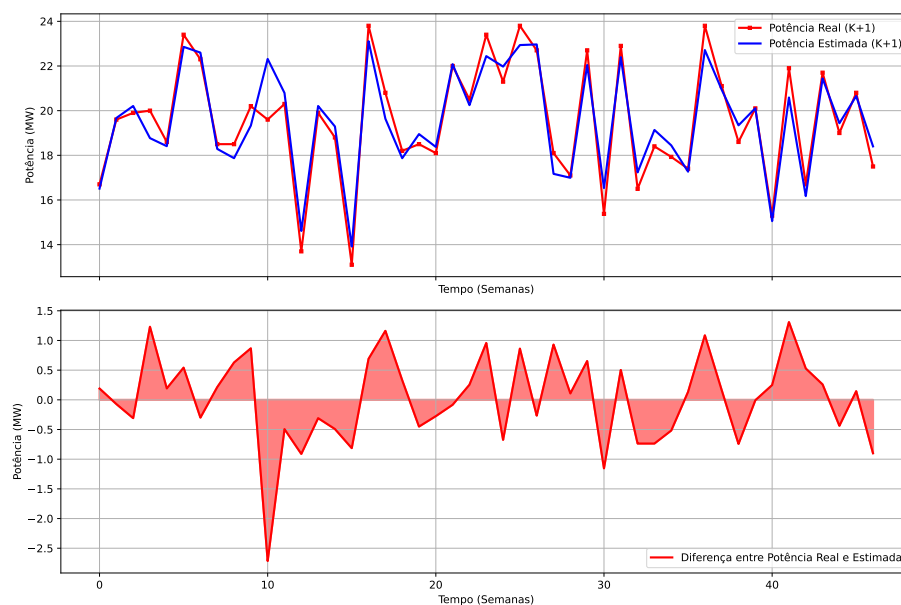


Fonte: Autoria própria.

como uma camada de saída com 2 neurônios. Além disso, o otimizador utilizado foi do tipo Adam, e a função de ativação da camada de saída foi a função *sigmoid*, a fim de evitar valores negativos como resultados da previsão.

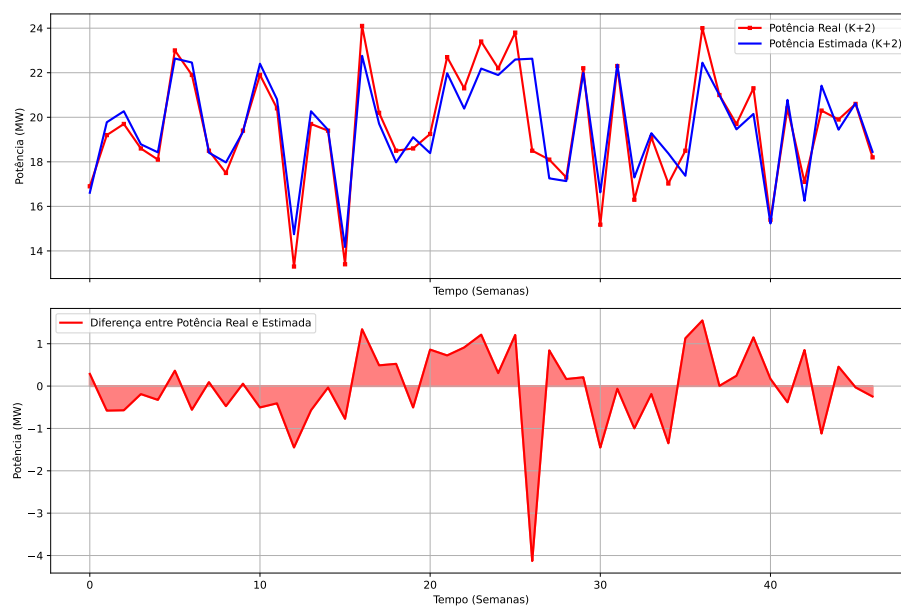
Na [Tabela 2](#) estão disponibilizadas todas as métricas para avaliação da performance do modelo RNA 2 para a previsão de demanda de potência elétrica.

Figura 28 – RNA 1 - Previsão de demanda semanal de potência (K+1) pelo melhor modelo encontrado.



Fonte: Autoria própria.

Figura 29 – RNA 1 - Previsão de demanda semanal de potência (K+2) pelo melhor modelo encontrado.



Fonte: Autoria própria.



Tabela 2 – RNA 2 - Métricas para avaliação de performance do modelo.

<b>Métricas</b>	<b>MAE</b>	<b>MSE</b>	<b>RMSE</b>	<b>MAXE</b>	<b>MAPE</b>	<b>Max_APE</b>
<b>K + 1</b>	0.6084	0.6130	0.7829	2.8954	3.2146	14.7723
<b>K + 2</b>	0.6667	0.8968	0.9470	4.2896	3.5590	23.1870

## 4 Conclusão

O sistema de previsão de demanda de potências futuras baseado em Redes Neurais Artificiais (RNAs) apresenta-se como uma ferramenta robusta e eficaz para antecipar com precisão as necessidades futuras de energia elétrica com base em padrões de demanda passados. Ao longo deste estudo, foi explorado e implementado uma arquiteturas de RNAs, testando diferentes configurações de hiperparâmetros e avaliando o desempenho do modelo em dados históricos.

Os resultados obtidos demonstram que as RNAs são capazes de capturar com sucesso as complexas relações temporais e não lineares presentes nos dados de demanda de potência. A capacidade de generalização dos modelos treinados foi evidenciada através da comparação entre as previsões e os dados reais, mostrando consistência e acurácia em diversos cenários.

No entanto, os requisitos de performance dos modelos não foram atingidos e, assim, para encontrar a melhor arquitetura foi visto que a eficácia das RNAs é sensível à qualidade e representatividade dos dados utilizados para treinamento.

Por fim, este estudo destaca a importância contínua do desenvolvimento e aprimoramento de sistemas de previsão de demanda de energia elétrica, especialmente diante da crescente complexidade dos sistemas de energia. As RNAs oferecem uma promissora abordagem preditiva que pode ser integrada efetivamente nas estratégias de gestão de demanda, planejamento de recursos e operação eficiente de redes elétricas.