Previsão de Ativos Financeiros com Redes Neurais Multicamadas Otimizadas por Algoritmos Genéticos e Otimização por Enxame de Partículas

Bruno Nicolau M de Souza Conte
Universidade Federal do Pará - UFPA
Belém, PA, Brasil
brunonicolau.conte@gmail.com

Roberto Célio Limão de Oliveira
Universidade Federal do Pará - UFPA
Belém, PA, Brasil
limao@ufpa.br

***Resumo*—Este artigo apresenta uma análise comparativa do desempenho de modelos de Redes Neurais Artificiais (RNAs) para previsão de preços de ativos financeiros da Petrobras, com foco na otimização da arquitetura por meio de um Algoritmo Genético (AG) e Otimização por Enxame de Partículas (PSO). Um modelo de rede neural multicamadas (MLP) é treinado com dados do ativo Petrobras e comparado com arquiteturas otimizadas por AG e PSO. Os resultados indicam que a otimização do modelo neural com AG promove uma melhoria significativa no desempenho preditivo, resultando em um Erro Quadrático Médio (MSE) inferior em comparação com abordagens tradicionais e com o modelo neural otimizado pelo PSO.**

***Palavras-Chave*—Redes Neurais Artificiais, Algoritmos Genéticos, Petrobras, Bolsa de valores, Otimização por Enxame de Partículas, Otimização de Arquitetura, Previsão Financeira.**

I. Introdução

A extrema volatilidade do preço do ativo Petrobras na bolsa de valores impõe um desafio significativo para investidores e analistas, dificultando a tomada de decisões e elevando o risco de perdas. Essa complexidade é agravada pela natureza não linear e pelos múltiplos fatores econômicos e geopolíticos que influenciam o mercado de energia.[1]

Para mitigar essa incerteza, Redes Neurais Artificiais (RNAs) emergem como uma solução promissora, capazes de identificar padrões ocultos e realizar previsões em séries temporais financeiras. Contudo, o desempenho preditivo das RNAs é altamente dependente de uma arquitetura e hiperparâmetros ótimos, cuja definição manual é ineficiente [2].

Este artigo propõe a aplicação de Algoritmos Genéticos (AGs) e Otimização por Enxame de Partículas (PSO) para otimizar automaticamente a configuração das RNAs, buscando aprimorar sua acurácia de previsão. Oferecemos uma análise comparativa dos resultados, demonstrando o potencial dessas meta-heurísticas em gerar modelos preditivos mais robustos e precisos para o mercado de capitais.

II. Mercado de Ações

O mercado de ações, um componente vital da economia global, apresenta um dos ambientes mais complexos e desafiadores para a análise e tomada de decisão. A previsão dos preços de ativos financeiros, como o da Petrobras na bolsa de valores, é uma tarefa de extrema dificuldade. Investidores e analistas frequentemente se deparam com a imprevisibilidade dos movimentos de preços, o que compromete a eficácia das estratégias de investimento e eleva o risco de perdas financeiras. A instabilidade observada, especialmente em ativos de grande liquidez e relevância setorial, como o da Petrobras, cria um cenário onde a precisão preditiva se torna um diferencial competitivo crucial para a sustentabilidade de operações no mercado.

A principal causa dessa dificuldade reside na volatilidade inerente aos mercados financeiros. Preços são influenciados por uma miríade de fatores interligados e muitas vezes imprevisíveis, incluindo: eventos macroeconômicos (inflação, taxas de juros), anúncios de resultados corporativos, notícias políticas (nacionais e internacionais), sentimentos do mercado e, crucialmente, avanços tecnológicos. A Petrobras, por exemplo, é particularmente suscetível a flutuações em preços globais de petróleo e a decisões políticas domésticas, ampliando sua volatilidade. Além disso, a crescente presença de robôs de investimento intensificou a velocidade das transações e a imprevisibilidade de movimentos de curto prazo. Esses sistemas automatizados reagem a eventos em milissegundos, gerando picos e vales abruptos que tornam as abordagens de análise humana tradicional menos eficazes e exigem modelos preditivos cada vez mais sofisticados para acompanhar o ritmo do mercado [3].

Diante da complexidade e da alta volatilidade do mercado de capitais, métodos de previsão tradicionais frequentemente se mostram inadequados. A solução reside na aplicação de técnicas avançadas de Inteligência Artificial, como as Redes Neurais Artificiais (RNAs). As RNAs são capazes de modelar relações não lineares complexas e identificar padrões sutis em grandes volumes de dados históricos, superando as limitações de modelos estatísticos lineares. Ao serem alimentadas com séries temporais de preços e indicadores relevantes, as RNAs podem aprender a dinâmica subjacente do mercado, oferecendo uma ferramenta robusta para a previsão de preços futuros [2].

Apesar do potencial das Redes Neurais, seu desempenho é intrinsecamente ligado à sua configuração. A determinação da arquitetura ideal (número de camadas e neurônios) e dos hiperparâmetros de treinamento é uma tarefa não trivial, que geralmente envolve tentativas e erros manuais. Para otimizar essa etapa crítica, este trabalho oferece a aplicação de meta-heurísticas de otimização, como Algoritmos Genéticos (AGs) e Otimização por Enxame de Partículas (PSO). Essas abordagens bio-inspiradas buscam automaticamente as configurações de RNA que minimizam o erro de previsão, maximizando a acurácia do modelo. Ao automatizar a otimização de hiperparâmetros, podemos gerar modelos preditivos mais robustos e precisos, capacitando investidores e analistas a tomar decisões mais informadas em um ambiente de mercado cada vez mais dinâmico e volátil. Esta otimização visa aprimorar a capacidade da Rede Neural em

prever com maior assertividade os movimentos futuros de ativos como o da Petrobras [4][5].

III. Trabalhos Relacionados

A crescente complexidade e volatilidade do mercado de ações têm impulsionado a busca por metodologias de previsão mais robustas e precisas. Nesse contexto, a integração de técnicas de inteligência artificial tem se destacado como uma abordagem promissora[6].

Conte e Oliveira [7] apresentam uma abordagem inovadora que combina o método Analytic Hierarchy Process (AHP) com Redes Neurais Artificiais (RNAs) para prever o desempenho de ações na Bolsa de Valores. Neste estudo, o AHP é empregado para atribuir pesos às variáveis relevantes, considerando sua importância relativa na tomada de decisão, enquanto as redes neurais são utilizadas para modelar e prever o comportamento das ações. A combinação dessas duas técnicas visa aprimorar a precisão das previsões, superando as limitações de métodos tradicionais, como a análise fundamentalista e a análise técnica. O desempenho da abordagem é avaliado utilizando dados históricos de empresas listadas na Bolsa de Valores, com métricas como o erro médio absoluto (MAE), erro quadrático médio (MSE) e a Raiz do erro quadrático médio (RMSE) sendo aplicadas para quantificar a acurácia das previsões.

Mghazli [8] foca na análise da eficácia de modelos de previsão a partir do uso de técnicas de pré-processamento em séries temporais financeiras. O principal objetivo do trabalho é demonstrar como um pré-processamento adequado, combinado com o uso de redes neurais morfológicas profundas, pode aumentar significativamente a precisão da previsão em séries temporais financeiras.

Li et al. [9] constroem um dataset com informações nacionais e internacionais sobre o mercado chinês e utilizam a técnica de Grid Search para encontrar os hiperparâmetros de quatro modelos deep learning (Long Short-Term Memory - LSTM, Gated Recurrent Unit - GRU, MLP, and Transformers). Os resultados indicam que a GRU tem um melhor desempenho no MSE entre os modelos utilizados.

Yu et al. [10] desenvolvem um modelo híbrido de predição do 'Dow Jones Industrial Médio diário' combinando uma decomposição modal, uma clusterização e o uso do Algoritmo Genético para sintonizar os hiperparâmetros de uma GRU. Os resultados mostram um RMSE de 224.45 e Medida de Dispersão (R2) igual a 0.9814 para o modelo híbrido.

Yu et al. [11] mostram a utilização de meta-heurísticas para sintonizar os hiperparâmetros de modelos neurais (MLP ou Deep Learning) no processo de predição de séries temporais do setor financeiro.

No geral, os trabalhos apresentados neste capítulo ressaltam a importância crescente da inteligência artificial e de abordagens híbridas na previsão do mercado de ações, buscando superar os desafios impostos pela sua complexidade e volatilidade. Embora cada estudo explore diferentes técnicas e focos, desde a combinação de AHP com RNAs até o pré-processamento de séries temporais e a otimização de hiperparâmetros com meta-heurísticas, todos convergem na busca por maior precisão e robustez nas previsões financeiras, assim, a otimização automática da arquitetura de RNAs utilizando AG e PSO, especialmente para o ativo PETR4, ainda carece de investigação aprofundada, lacuna esta que o presente estudo busca preencher.

IV. Fundamentos

A. Rede Neural Multicamadas (MLP)

Uma Rede Neural Multicamadas (MLP) é uma classe de redes neurais feedforward, composta por uma camada de entrada, uma ou mais camadas ocultas (intermediárias) e uma camada de saída. Cada neurônio em uma camada está conectado a todos os neurônios da camada subsequente. A saída de um neurônio é calculada aplicando-se uma função de ativação à soma ponderada das suas entradas, conforme a Equação 1:

$$y_k = f\left(\sum_{j=1}^{n} w_{kj}\, x_j + \, b_k\right) \tag{1}$$

onde x_1, x_2, ..., n são os sinais de entrada, w_{k1}, w_{k2}, ... , w_{kn} são os respectivos pesos de conexão do neurônio k, b_{k} é o bias, e finalmente, $f(\cdot)$ é a função de ativação.

A função de ativação ReLU (Rectified Linear Unit), comumente utilizada em camadas intermediárias, é definida por:

$$ReLU(x) = \max(0, x) \tag{2}$$

A função de Ativação utilizada na camada de saida é a Linear.

$$f(x) = x \tag{3}$$

O treinamento da rede neural ocorre através da minimização de uma função de perda, como o Erro Quadrático Médio (MSE), que mede a diferença entre as saídas previstas pela rede e os valores reais. O MSE é calculado como:

$$MSE = \frac{1}{n}\sum_{k=1}^{n}(y_k - \hat{y}_k)^2 \tag{4}$$

Onde n é o número de amostras. y_{k} é o valor real da amostra k. $\hat{y}_k$ é o valor previsto pela rede para a amostra k.

B. Algoritmo Genético para Otimização de Hiperparâmetros

Nesta abordagem, um algoritmo genético foi empregado para otimizar o número de camadas e o número de neurônios em cada camada da rede neural. Cada indivíduo (cromossomo) no AG representa uma configuração da arquitetura da rede neural. A aptidão de cada cromossomo é determinada pelo desempenho da rede neural com aquela arquitetura específica, avaliada pelo MSE em um conjunto de validação ou teste.

As principais etapas do AG são:

- Inicialização da População: Geração aleatória de cromossomos que representam diferentes arquiteturas de RNA.
- Avaliação da Aptidão: Para cada cromossomo, uma rede neural é construída e treinada. O MSE obtido é usado como métrica de aptidão (quanto menor o MSE, maior a aptidão).
- Seleção: Indivíduos mais aptos são selecionados para a próxima geração.
- Cruzamento (Crossover): Combinação de partes de cromossomos de pais selecionados para gerar novos descendentes, explorando novas arquiteturas.
- Mutação: Pequenas alterações aleatórias em cromossomos, introduzindo diversidade e evitando mínimos locais [4].

C. Otimização por Enxame de Partículas (PSO)

A Otimização por Enxame de Partículas (PSO) é uma meta-heurística de otimização bio-inspirada no comportamento social de bandos de pássaros ou cardumes de peixes. Cada "partícula" no enxame representa uma solução candidata no espaço de busca (neste artigo, uma configuração de Arquitetura da RNA). As partículas se movem no espaço de busca, ajustando sua trajetória com base em sua própria melhor posição encontrada (pbest) e a melhor posição encontrada por qualquer partícula no enxame (gbest).

A posição e a velocidade de cada partícula são atualizadas iterativamente. Para uma partícula i em uma dimensão d, as equações de atualização são:

$$v_{i,d}^{t+1} = wv_{i,d}^{t} + c_1 r_1 \left(pbest_{i,d} - x_{i,d}^{t}\right) + c_2 r_2 \left(gbest_d - x_{i,d}^{t}\right) \quad (5)$$

$$x_{i,d}^{t+1} = x_{i,d}^{t} + v_{i,d}^{t+1} \quad (6)$$

Onde $v_{i,d}^{t+1}$ é a nova velocidade da partícula i na dimensão d.$x_{i,d}^{t+1}$é a nova posição da partícula i na dimensão d. w é o fator de inércia.c_1 e c_2 são os coeficientes de aceleração (cognitivo e social, respectivamente).r_1 e r_2 são números aleatórios no intervalo [0,1].$pbest_{i,d}$ é a melhor posição global (gbest) da partícula i até o momento.$gbest_d$ é a melhor posição global (gbest) encontrada por qualquer partícula no enxame até o momento [5].

Na aplicação para otimização de RNAs, cada partícula representa uma arquitetura de rede neural (e.g., número de camadas e neurônios), e a função de aptidão é novamente o MSE da RNA correspondente enquanto que no PSO cada partícula se move no espaço de busca para encontrar novas configurações que possam ter um desempenho melhor.

V. METODOLOGIA

Os dados utilizados neste estudo foram extraídos da bolsa de valores, focando no ativo da Petrobras. A escolha da Petrobras se justifica pela sua relevância no mercado brasileiro e pela alta volatilidade de seus preços, que impõem um desafio significativo para a previsão. O conjunto de dados foi pré-processado para garantir a qualidade e a adequação para o treinamento das redes neurais, como ilustrado na Figura 1.

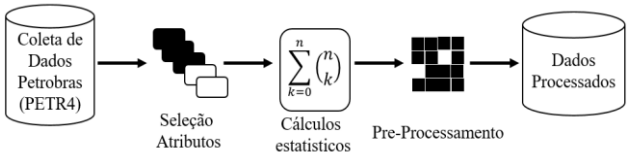

Fig. 1. Visualização do fluxo de dados

Conforme ilustrado na Figura 1, o ciclo de formação dos dados processados inicia-se com a Coleta de Dados da Petrobras (PETR4) diretamente da bolsa de valores. Subsequentemente, é realizada a Seleção de Atributos, onde o preço de fechamento é escolhido como a variável principal para a análise preditiva. Este passo é seguido pela execução de Cálculos Estatísticos sobre os dados selecionados, aplicando a Média Móvel Simples (MMS) de 20 e 50 períodos, Média Móvel Exponencial (EMA) de 20 períodos, as Bandas de Bollinger Superior (SBB) e Inferior (IBB) de 20 períodos. Por fim, a etapa de Pré-Processamento prepara os dados para se utilizado no treinamento da rede neural, resultando em um conjunto de Dados Processados que está otimizado para o treinamento e avaliação dos modelos. Essas etapas garantem a consistência e a adequação dos dados para as análises subsequentes.

As simulações, o treinamento e a otimização das Redes Neurais foram conduzidos em uma workstation equipada com um processador Intel Xeon E5-2690 v4 (14nm), uma placa-mãe E5 MR9A Pro, 64 GBytes de memória RAM DDR4 e uma placa de vídeo AMD Radeon RX 580 com 8GB de VRAM, fornecendo o poder computacional necessário para as operações intensivas de otimização e processamento de dados inerentes ao estudo.

O processo de modelagem e otimização das redes neurais é uma etapa central na metodologia deste estudo. O objetivo principal é não apenas treinar um modelo, mas também encontrar a arquitetura mais performática por meio de técnicas de otimização meta-heurísticas, visando alcançar a melhor capacidade preditiva.

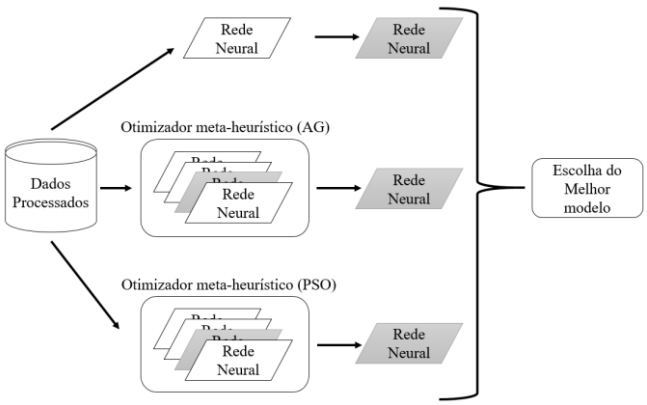

Fig. 2. Seleção do Modelo Reural

Conforme detalhado na Figura 2, os dados processados são a base para três caminhos distintos de construção e otimização de modelos de redes neurais. O primeiro caminho avalia uma Rede Neural Multicamadas (MLP) base. Em paralelo, são exploradas otimizações utilizando algoritmos

meta-heurísticos: uma Rede Neural é otimizada por um Otimizador meta-heurístico Algoritmo Genético (AG), enquanto outra é otimizada por um Otimizador meta-heurístico Otimização por Enxame de Partículas (PSO). Cada uma dessas abordagens visa refinar a arquitetura da rede para maximizar seu desempenho. Ao final desse processo paralelo de experimentação e otimização, é realizada a escolha do melhor modelo, baseada em métricas de desempenho avaliadas nos conjuntos de teste.

A MLP é composta por uma camada de entrada, uma ou mais camadas ocultas (intermediárias) e uma camada de saída. Cada neurônio em uma camada é conectado a todos os neurônios da camada subsequente. A saída de um neurônio é calculada aplicando-se uma função de ativação à soma ponderada das suas entradas, com a função ReLU (Rectified Linear Unit) empregada nas camadas intermediárias e a função Linear na camada de saída. O treinamento da MLP visa minimizar o Erro Quadrático Médio (MSE), uma métrica que quantifica a diferença entre os valores previstos pela rede e os valores reais.

No AG cada indivíduo, ou cromossomo representa uma configuração específica da arquitetura da rede neural, incluindo o número de camadas ocultas e a quantidade de neurônios em cada uma delas. A aptidão de cada cromossomo é determinada pelo desempenho da rede neural correspondente, avaliada pelo MSE em um conjunto de validação. O processo do AG envolve etapas de inicialização da população (geração aleatória de arquiteturas), avaliação da aptidão, seleção dos indivíduos mais aptos, cruzamento (crossover) para gerar novas arquiteturas e mutação para introduzir diversidade e evitar mínimos locais.

Adicionalmente, a Otimização por Enxame de Partículas (PSO) foi aplicada para otimizar a arquitetura das RNAs. Inspirada no comportamento social de bandos, cada "partícula" no enxame representa uma solução candidata (configuração de arquitetura da RNA). As partículas ajustam suas trajetórias no espaço de busca com base em sua melhor posição individual (pbest) e a melhor posição encontrada por qualquer partícula no enxame (gbest). A função de aptidão para o PSO também é o MSE da RNA correspondente à arquitetura de cada partícula. O PSO busca iterativamente as configurações que minimizam o erro de previsão da rede, explorando o espaço de busca de forma eficiente [5].

VI. Resultados

Para a análise e modelagem do ativo da Petrobras (PETR4), os dados históricos de preços foram obtidos diretamente da plataforma MetaTrader 5 (MT5), utilizando a biblioteca MetaTrader 5 no Python. Este processo incluiu o estabelecimento de uma conexão segura entre Corretora XP Investimentos e a plataforma contratada MetraTrader 5. Em seguida, foram coletadas as últimas 5.000 barras (candles) do ativo PETR4 no intervalo de tempo de 1 minuto (mt5.TIMEFRAME_M1) utilizando a biblioteca MetaTrader5. Os dados brutos foram então convertidos para um DataFrame do pandas, com a coluna de tempo devidamente formatada [12][13].

O conjunto total de 5000 amostras foi cuidadosamente dividido para treinamento, validação e teste, garantindo a robustez e a generalização do modelo. A distribuição dos dados é a seguinte: 60% para treinamento, 20% para validação e 20% para teste. Essa divisão permite que o modelo aprenda com um conjunto de dados, ajuste seus hiperparâmetros com outro e, finalmente, seja avaliado em dados não vistos. A Figura 3 ilustra a visualização temporal desses dados.

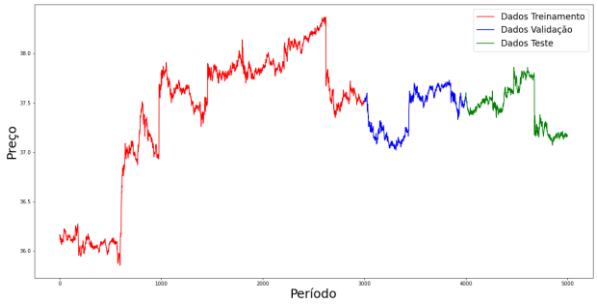

Fig. 3. Visualização dos dados de Treinamento, Validação e Teste

Os dados de entrada para a rede neural consistiram nos três últimos preços de fechamento do ativo e nos cinco principais indicadores técnicos, a Média Móvel Simples (MMS) de 20 e 50 períodos, Média Móvel Exponencial (EMA) de 20 períodos, as Bandas de Bollinger Superior (SBB) e Inferior (IBB) de 20 períodos e desvio padrão de 2.

A Média Móvel Simples (MMS) de 20 e 50 períodos são utilizadas para identificar tendências e padrões de movimento dos preços, ao longo do tempo, conforme a Equação7.

$$MMS = \frac{\sum_{i=1}^{n} Preço_i}{n} \tag{7}$$

onde $Preço_i$ é preço de fechamento do período atual e n o número de períodos para o cálculo.

A Média Móvel Exponencial (MME) de 20 períodos, atribui maior peso aos preços mais recentes, sendo calculada com base em uma constante chamada de alpha, conforme a Equação8.

$$MME_{Hoje} = Preço_{Hoje} * K + MME_{Ontem} * (1 - K)$$

$$Onde\ k = \frac{2}{n+1} \tag{8}$$

onde $Preço_{Hoje}$ é preço de fechamento do período atual e n o número de períodos para o cálculo.

As Bandas de Bollinger Superior (SBB) e Inferior (IBB) de 20 períodos e desvio padrão de 2, sendo utilizadas para identificar a volatilidade dos preços de um ativo e possíveis pontos de reversão ou continuação de tendências, conforme a Equação 9, 10 e 11:

$$MMS = \frac{\sum_{i=1}^{n} Preço_i}{n} \tag{9}$$

$$SBB = MMS + SD\sqrt{\frac{\sum_{i=1}^{n}(Preço_i - MMS)^2}{n}} \tag{10}$$

$$IBB = MMS - SD\sqrt{\frac{\sum_{i=1}^{n}(Preço_i - MMS)^2}{n}} \tag{11}$$

onde SD é desvio padrão, Preço é preço de fechamento do período atual, MMS representa o valor médio para os n períodos (calculado pela Equação11) e n o número de períodos para o cálculo.

A qualidade dos dados é crucial em qualquer projeto de aprendizado de máquina, especialmente para a previsão de preços no mercado financeiro, onde a presença de imperfeições pode comprometer a confiabilidade das previsões. Para garantir a integridade dos dados, foi realizado a limpeza e o tratamento de valores ausentes. Os valores ausentes, que podem surgir por diversas razões, foram preenchidos com a mediana conforme a Equação12 e 13.

Se n é ímpar:

$$Mediana: \frac{n+1}{2} \tag{12}$$

Se n é par:

$$t_1 = \frac{n}{2} \;\; e \;\; t_2 = \frac{n}{2} + 1$$

$$Mediana = \frac{t_1 + t_2}{2} \tag{13}$$

onde n é o número de elementos;

O modelo MLP baseline foi configurado com 6 entradas, 7 camadas intermediárias com 24, 20, 22, 11, 3, 9, 11 neurônios respectivamente, utilizando a função de ativação ReLU, e uma camada de saída com 1 neurônio utilizando função Linear. O treinamento da rede neural baseia-se na minimização do Erro Quadrático Médio (MSE), que mede a diferença entre as saídas previstas e os valores reais. A Figura 4 demonstra a convergência do MSE durante o treinamento do modelo, tanto para o conjunto de treinamento quanto para o de validação.

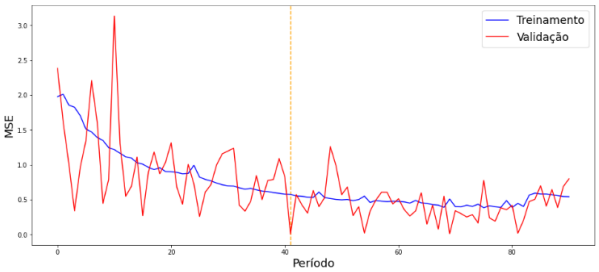

Fig. 4. Convergência da Rede Neural MLP

Conforme ilustrado na Figura 4, observa-se a trajetória do MSE para os conjuntos de treinamento (linha azul) e validação (linha vermelha) ao longo do período de treinamento. Inicialmente, ambos os erros tendem a diminuir, indicando que a rede está aprendendo os padrões dos dados.

A linha tracejada vertical (em amarelo) marca o ponto de seleção do melhor modelo, onde o conjunto de pesos é selecionado, pois o desempenho no conjunto de validação atinge seu ponto ótimo antes de iniciar uma possível degradação. Essa linha representa uma parada antecipada (early stopping), uma técnica para evitar o overfitting, que ocorre quando o modelo começa a aprender o ruído dos dados de treinamento e perde a capacidade de generalizar para dados novos. A convergência dos erros antes do sobreajuste e a estabilidade após esse ponto demonstram a eficácia do processo de treinamento e a capacidade da rede de generalizar para dados não vistos.

A avaliação final do modelo preditivo é realizada pela comparação direta de suas estimativas com os valores reais ao longo do tempo. Esta visualização é crucial para entender a capacidade do modelo em capturar a dinâmica da série temporal do ativo, revelando tanto a sua acurácia quanto os momentos de maior desvio entre a previsão e a realidade do mercado.

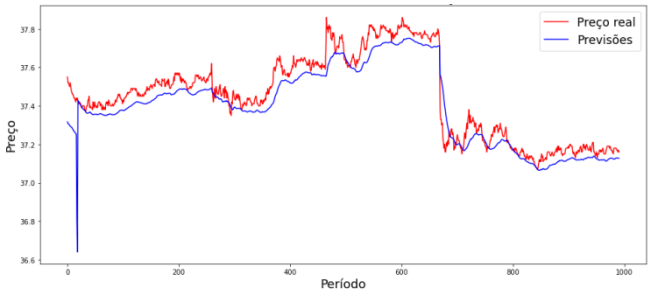

Fig. 5. Previsão da Rede Neural MLP

Conforme ilustrado na Figura 5, a linha vermelha representa o Preço real do ativo ao longo do período analisado, enquanto a linha azul corresponde às Previsões geradas pelo modelo.

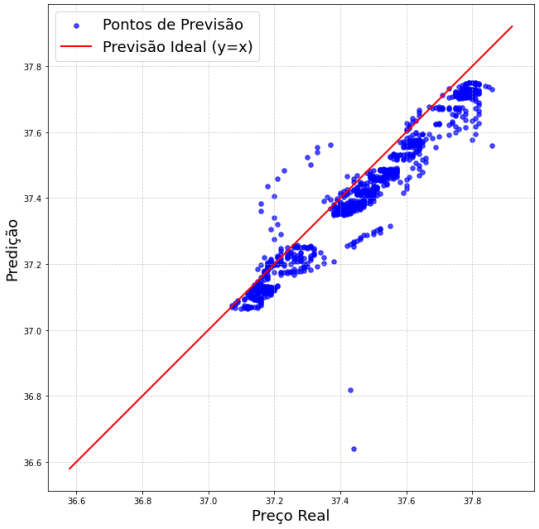

Fig. 6. Preço Real x Predição Modelo MLP

Conforme ilustrado na Figura 6, o gráfico de dispersão apresenta os "Pontos de Previsão" (em azul) do modelo MLP, plotados em relação aos "Preços Reais" no eixo horizontal e as "Previsões" no eixo vertical. A linha vermelha diagonal representa a "Previsão Ideal (y=x)", indicando onde os valores previstos seriam exatamente iguais aos valores reais. A alta concentração dos pontos azuis próxima a essa linha vermelha sinaliza uma forte correlação e uma notável precisão do modelo MLP na previsão do preço do ativo, o que demonstra sua capacidade de aprender e

replicar os padrões subjacentes da série temporal financeira. Complementarmente, o coeficiente de determinação (R^2) de 0.8617 corrobora essa observação, indicando que aproximadamente 86,17% da variância nos preços reais do ativo são explicados pelo modelo. O R² é a seguinte:

$$SS_{res} = \sum_{i=1}^{n} (y_i - \hat{y})^2 \tag{14}$$

$$SS_{tot} = \sum_{i=1}^{n} (y_i - \bar{y})^2 \tag{15}$$

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \tag{16}$$

Onde SS_{res} é a soma dos quadrados da diferencas entre os valores reais observado y_i e os valores previstos pelo modelo $\hat{y}$. SS_{tot} é a soma dos quadrados das diferenças entre o valores reais observados y_i e a média dos valores reais $\bar{y}$ e o n é o número de observações.

O R^2 varia de 0 a 1. Um R^2 próximo de 1 indica que o modelo explica uma grande proporção da variância da variável dependente, ou seja, as previsões do modelo se ajustam bem aos dados reais. Um R^2 próximo de 0 indica que o modelo explica muito pouco ou nenhuma da variância da variável dependente, sugerindo um ajuste fraco.

A Tabela I compara as métricas de desempenho do modelo MLP baseline com as do artigo previamente publicado [7]. Os resultados do MLP baseline foram: MAE de 0.06122, MSE de 0.0063346 e RMSE de 0.0795904. Em comparação, o artigo publicado alcançou MAE de 0.041, MSE de 0.003 e RMSE de 0.058 [7].

TABELA I. COMPARAÇÃO DESEMPENHOS

Métrica	Artigo Publicado [7]	MLP
MAE	0.041	0.06122
MSE	0.003	0.0063346
RMSE	0.058	0.0795904

Para otimizar a arquitetura e os hiperparâmetros da Rede Neural, um Algoritmo Genético foi implementado. O AG buscou automaticamente a configuração ideal das camadas e o número de neurônios por camada. A Tabela II detalha os parâmetros do AG utilizado, incluindo 50 gerações totais, 5 indivíduos de elitismo, máximos de 30 camadas e 30 neurônios por camada, e probabilidades de cruzamento (75%), mutação (5%) e "ring" (17%).

TABELA II. CONFIGURAÇÃO AG

Configuração Do Algoritmo Genetico	
Total de geração	50
Quantidade elitismo	5
Quantidade maxima de Camada	30
Quantidade maxima de Neuronio por camada	30
Probabilidade Cruzamento	75%
Probabilidade Mutação	5%
Probabilidade Ring	17%

Cada cromossomo na população representa uma configuração da rede neural. A função fitness de cada cromossomo é avaliada pelo desempenho da rede neural correspondente, medido pelo MSE no conjunto de teste. Para calcular o fitness, a rede neural MLP é construída com a arquitetura definida pelo cromossomo, utilizando função de ativação ReLU nas camadas ocultas e Linear na camada de saída. O modelo é então compilado com o otimizador 'rmsprop' e função de perda 'mean_squared_error', e treinado por até 100 épocas com um batch_size de 1. O treinamento incorpora EarlyStopping (com patience=12) e ModelCheckpoint para salvar o modelo que apresentou o menor val_loss. Após o treinamento, o modelo salvo é carregado, e suas previsões no conjunto de teste são utilizadas para calcular o MSE final, que serve como o valor de fitness para cromossomo.

Cada Rede Neural foi submetida a um conjunto de testes, para avaliar a capacidade do modelo em prever as ações da bolsa de valores em dados ainda não conhecidos. Para a avaliação do modelo sobre o conjunto de dados de teste, foram utilizados três métricas de desempenho como: erro médio absoluto (MAE), erro quadrático médio (MSE) e a raiz do erro quadrático médio (RMSE), A MAE é calculado da seguinte maneira:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |y_i - d_i| \tag{17}$$

onde y_i é o valor de saída e d_ié o valor desejado. MSE e RMSE são apresentados a seguir, respectivamente.

$$MSE = \frac{1}{n}\sum_{i=1}^{n} (y_i - d_i)^2 \tag{18}$$

onde y_i é o valor de saída e d_ié o valor desejado.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (y_i - d_i)^2} \tag{19}$$

onde y_i é o valor de saída e d_ié o valor desejado.

A Figura 7 exibe a evolução do erro (MSE) do melhor cromossomo encontrado por geração, demonstrando a capacidade do AG em convergir para soluções de menor erro.

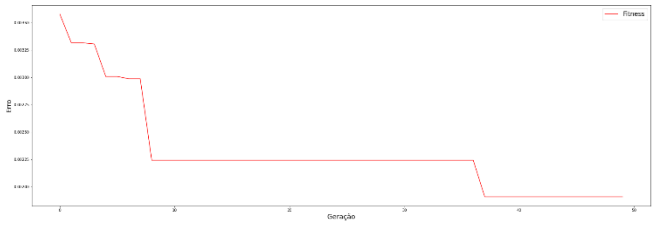

Fig. 7. Melhor Fitness por Geração

A Figura 8 ilustra a média do erro da população por geração.

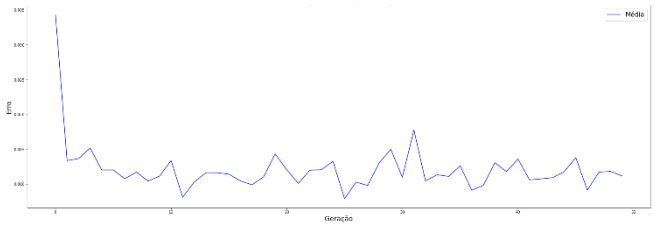

Fig. 8. Médias por Geração

A melhor solução encontrada pelo AG, após 50 gerações, resultou em uma rede neural com 4 camadas intermediárias, contendo 11, 7, 23 e 26 neurônios, respectivamente, com função de ativação ReLU nas camadas intermediárias e Linear na camada de saída. Este modelo otimizado obteve um MSE de 0.00190881 nos dados de teste. O tempo de execução para essa otimização foi de aproximadamente 2 dias e 23 horas e 3 minutos.

A Figura 9 demonstra a convergência do MSE durante o treinamento do modelo, tanto para o conjunto de treinamento quanto para o de validação escolhido pelo AG.

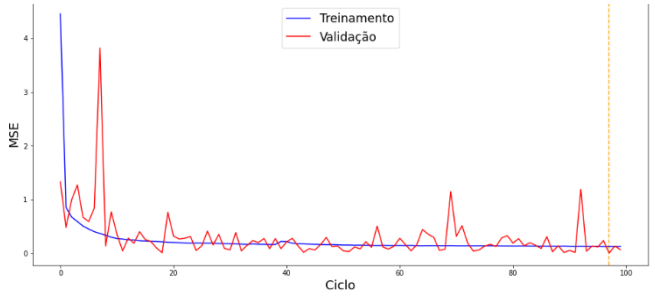

Fig. 9. Convergência da MLP escolhida pelo AG

A Otimização por Enxame de Partículas (PSO) foi empregada como uma meta-heurística para otimizar a arquitetura e os hiperparâmetros da Rede Neural Artificial (RNA), buscando a configuração que minimizasse o Erro Quadrático Médio (MSE). O experimento foi configurado com os seguintes parâmetros para o algoritmo PSO:

TABELA III. CONFIGURAÇÃO PSO

Configuração do PSO	
Número de Partículas	10
Número Máximo de Iterações	50
Fator de Inércia (w)	0.5
Coeficiente de Aceleração Cognitiva (c1)	1
Coeficiente de Aceleracão Social (c2)	2
Domínio de Busca para o Número de Neurônios	1 a 30

Cada partícula no enxame representa uma configuração da rede neural, especificamente o número de neurônios para cada uma das sete camadas ocultas. A aptidão (fitness) de cada partícula é avaliada pelo desempenho da rede neural correspondente, medido pelo MSE no conjunto de teste. Para calcular o fitness, a rede neural MLP é construída com a arquitetura definida pela partícula, utilizando função de ativação ReLU nas camadas ocultas e Linear na camada de saída. O modelo é então compilado com o otimizador 'rmsprop' e função de perda 'mean_squared_error', e treinado por até 100 épocas com um batch_size de 1. O treinamento incorpora EarlyStopping (com patience=12) e ModelCheckpoint para salvar o modelo que apresentou o menor val_loss. Após o treinamento, o modelo salvo é carregado, e suas previsões no conjunto de teste são utilizadas para calcular o MSE final, que serve como o valor de fitness para a partícula.

Ao longo das 50 iterações, as partículas ajustam suas posições e velocidades, convergindo em direção à melhor solução encontrada globalmente (melhor_posicao_global) e individualmente (melhores_posicoes_pessoais), buscando minimizar o erro da rede neural. Ao final do processo, o PSO identificou uma configuração de rede neural que resultou em um MSE de 0.001923 nos dados de teste. A melhor arquitetura encontrada pelo PSO consistiu em 7 camadas intermediárias com a seguinte quantidade de neurônios: [19, 14, 25, 23, 22, 17, 10]. O tempo total de execução para o algoritmo PSO foi de aproximadamente 3 dias e 20 horas, refletindo o custo computacional associado à otimização de hiperparâmetros de redes neurais através de meta-heurísticas.

A avaliação comparativa dos modelos é essencial para determinar a eficácia das abordagens de otimização de hiperparâmetros. Analisamos o desempenho do modelo MLP baseline, que representa uma rede neural com arquitetura definida de forma convencional, e o comparamos com o modelo otimizado por Algoritmo Genético (AG) e modelo gerado pela Otimização por Enxame de Partículas (PSO). A Tabela IV consolida as métricas de desempenho (MSE) para cada abordagem, complementando com as informações do artigo de referência.

TABELA IV. CONFIGURAÇÃO PSO

Métrica	**Artigo Publicado**	**MLP Baseline**	**AG + MLP**	**PSO + MLP**
MSE	0.003	0.0063346	0.00190881	0.001923

Conforme a Tabela IV, o modelo MLP baseline, sem otimização de hiperparâmetros por meta-heurísticas, apresentou um MSE de 0.0063346. Em comparação, o artigo previamente publicado alcançou um MSE de 0.003, indicando que a configuração utilizada em nosso MLP baseline foi menos eficaz que a do estudo de referência.

No entanto, a aplicação das meta-heurísticas de otimização demonstrou uma melhoria substancial. O modelo de Rede Neural otimizado pelo Algoritmo Genético (AG) alcançou o menor MSE de 0.00190881. Este resultado representa uma redução de aproximadamente 68.2% no MSE em relação ao MLP baseline e 36.3% em relação ao modelo

do artigo publicado, evidenciando a capacidade do AG em explorar o espaço de arquiteturas e hiperparâmetros de forma eficiente para encontrar soluções superiores. A arquitetura encontrada pelo AG foi de 4 camadas intermediárias com [11, 7, 23, 26] neurônios, respectivamente.

O algoritmo Otimização por Enxame de Partículas (PSO) também demonstrou um desempenho notável, resultando em um MSE de 0.001923. Embora marginalmente superior ao AG neste, o PSO também conseguiu superar significativamente tanto o MLP baseline quanto o modelo do artigo publicado. A arquitetura otimizada pelo PSO, composta por 7 camadas intermediárias com [19, 14, 25, 23, 22, 17, 10] neurônios, destaca a capacidade dessas meta-heurísticas de descobrir configurações de rede complexas que seriam difíceis de obter por meio de tentativas manuais ou buscas sistemáticas.

Ambas as meta-heurísticas, AG e PSO, provaram ser ferramentas poderosas para a otimização de hiperparâmetros de Redes Neurais em problemas de previsão financeira. Elas são capazes de navegar em espaços de busca não lineares e de alta dimensionalidade, que são características da otimização de arquiteturas de redes neurais, levando a modelos com acurácia preditiva superior. O custo computacional associado a essas otimizações, embora elevado (cerca de 3 dias por algoritmo), é justificado pela melhoria na performance dos modelos, que é crítica para a tomada de decisões no volátil mercado de ativos financeiros como o da Petrobras.

VII. Considerações Finais

Este artigo investigou a aplicação e o impacto da otimização de arquiteturas de Redes Neurais Artificiais (RNAs) por meio de meta-heurísticas bio-inspiradas, especificamente o Algoritmo Genético (AG) e a Otimização por Enxame de Partículas (PSO), para a desafiadora tarefa de previsão de preços do ativo Petrobras na bolsa de valores. A volatilidade intrínseca do mercado financeiro e a complexidade das interações entre fatores econômicos e geopolíticos tornam as abordagens preditivas tradicionais insuficientes, o que justifica o uso de técnicas avançadas como as RNAs.

Os resultados obtidos demonstram de forma inequívoca o potencial dessas meta-heurísticas em aprimorar significativamente o desempenho preditivo dos modelos de RNA. Enquanto um modelo MLP baseline apresentou um Erro Quadrático Médio (MSE) de 0.0063346, as otimizações por AG e PSO reduziram esse erro para 0.00190881 e 0.001923, respectivamente. Essa melhoria substancial valida a premissa de que a configuração ótima de uma RNA não é trivial e que a busca automatizada por essa arquitetura pode levar a resultados muito superiores aos obtidos por métodos heurísticos ou manuais. A capacidade tanto do AG quanto do PSO de navegar em espaços de busca complexos e de alta dimensionalidade permitiu a descoberta de configurações de rede que maximizam a acurácia, mesmo com o custo computacional associado a essas otimizações, que se mostrou elevado, mas justificado pela precisão alcançada.

Em suma, este estudo reforça a viabilidade e a eficácia da combinação de Redes Neurais Artificiais com Algoritmos Genéticos e Otimização por Enxame de Partículas para a previsão de séries temporais financeiras. As ferramentas propostas representam um avanço importante para investidores e analistas, oferecendo modelos mais robustos e precisos que podem auxiliar na tomada de decisões estratégicas em um ambiente de mercado cada vez mais dinâmico e imprevisível.

Referencias

[1] CARDOSO, Regis; KELM, Helena. Análise Fundamentalista - Ações PETR4. 2020. Disponível em: https://conteudos.xpi.com.br/acoes/petr4/. Acesso em: 28 abr. 2025.

[2] GOODFELLOW, Ian. Deep learning. Cambridge, Massachusetts: The Mit Press, 2016.

[3] RESEARCH XP. Guia para iniciantes: o que é Day Trade e como funciona! 2024. Disponível em: https://conteudos.xpi.com.br/aprenda-a-investir/relatorios/day-trade/. Acesso em: 13 maio 2024.

[4] NEUMANN, Eyal. Hands-On Genetic Algorithms with Python: Apply genetic algorithms to solve real-world AI and machine learning problems. Birmingham: Packt Publishing, 2019.

[5] WALKER, Brian (Ed.). Particle Swarm Optimization (PSO): Advances in Research and Applications. Nova York: Nova Science Publishers, 2017.

[6] Vuković, D.B., Dekpo-Adza, S. & Matović, S. AI integration in financial services: a systematic review of trends and regulatory challenges. Humanit Soc Sci Commun 12, 562 (2025). https://doi.org/10.1057/s41599-025-04850-8.

[7] CONTE, Bruno Nicolau; OLIVEIRA, Roberto Célio Limão. Análise Multicritério e Rede Neurais Artificiais para prever ações na bolsa de valores. Anais do XVI Congresso Brasileiro de Inteligência Computacional, [S.L.], p. 1-6, 31 dez. 2023. SBIC. http://dx.doi.org/10.21528/cbic2023-074.

[8] MGHAZLI, Ya-Sin; ARAðJO, Ricardo; SEIXAS, José Manoel. Superando a Hipótese do Mercado Eficiente Fraca com Redes Neurais Morfológicas Profundas. Anais do XVI Congresso Brasileiro de Inteligência Computacional, [S.L.], p. 1-8, 31 dez. 2023. SBIC. http://dx.doi.org/10.21528/cbic2023-119.

[9] Chin Yang Lin, João Alexandre Lobo Marques. Stock market prediction using artificial intelligence: A systematic review of systematic reviews. Social Sciences & Humanities Open, Volume 9, 100864, 2024. https://doi.org/10.1016/j.ssaho.2024.100864

[10] Hanyue Yu, Jiyang Dong, Bin Li. Economic insight through an optimized deep learning model for stock market prediction regarding Dow Jones industrial Average index. Expert Systems with Applications, 128473, 2025. https://doi.org/10.1016/j.eswa.2025.128473.

[11] L. Yu, Y. Chen and Y. Zheng, "A Multi-Market Data-Driven Stock Price Prediction System: Model Optimization and Empirical Study," in IEEE Access, vol. 13, pp. 65884-65899, 2025, doi: 10.1109/ACCESS.2025.3559169.

[12] INVESTIMENTOS, Xp. Como realizar contratação e instalação do MetaTrader 5. Disponível em: https://atendimento.xpi.com.br/artigo/1476-como-realizar-contratacao-e-instalacao-do-metatrader-5. Acesso em: 28 abr. 2025.

[13] METAQUOTES. Módulo MetaTrader para integração com Python. 2025. Disponível em: https://www.mql5.com/pt/docs/python_metatrader5. Acesso em: 28 abr. 2025.