

REDES NEURAIS ARTIFICIAIS APLICADAS À PREVISÃO E OPERAÇÃO NO MERCADO FINANCEIRO: UM ESTUDO PARA FINS DE OTIMIZAÇÃO DE CARTEIRAS DE INVESTIMENTOS

Renan Aurélio Euzébio¹

RESUMO

O mercado de ações apresenta desafios analíticos e de volatilidade para investidores. Visando apoiar a tomada de decisão, este trabalho explora a aplicação de redes neurais profundas para identificar padrões em dados financeiros e prever o comportamento de ativos. Para tal, desenvolve-se um *pipeline* que compara três arquiteturas de redes neurais — MLP, CNN e LSTM — na previsão de ativos do Índice Bovespa, utilizando dados históricos enriquecidos com indicadores técnicos. A avaliação dos modelos emprega uma metodologia de validação cruzada *Walk-Forward Analysis*, com a Acurácia Direcional como métrica primária e um sistema de *backtesting* para aferir a eficácia financeira de uma estratégia de otimização de carteira. O objetivo é, portanto, investigar a viabilidade de tal abordagem e consolidar a pesquisa em uma plataforma de aplicação prática e educacional para a análise de investimentos com Inteligência Artificial.

Palavras-chave: Redes Neurais; Séries Temporais; Otimização de Carteira.

ABSTRACT

The stock market poses analytical and volatility-related challenges for investors. Aiming to support decision-making, this study explores the application of deep neural networks to identify patterns in financial data and forecast asset behavior. To this end, a pipeline is developed that compares three neural network architectures—MLP, CNN, and LSTM—in predicting assets from Bovespa, using historical data enriched with technical indicators. Model evaluation employs the Walk-Forward Analysis validation methodology, with Directional Accuracy as the primary metric and a backtesting system to assess the financial effectiveness of a portfolio optimization

¹ Discente do Curso de Ciência da Computação da Universidade La Salle - Unilasalle, matriculado na disciplina Trabalho de Conclusão II. E-mail: renan.202210291@unilasalle.edu.br, sob a orientação de Prof. Mozart Lemos de Siqueira. E-mail: mozart.siqueira@unilasalle.edu.br. Data de entrega: 23 junho de 2025.

strategy. The objective is to investigate the feasibility of this approach and to consolidate the research into a practical and educational platform for investment analysis with Artificial Intelligence.

Keywords: Neural Networks; Time Series; Portfolio Optimization.

1 INTRODUÇÃO

A aplicação de Inteligência Artificial (IA) no mercado financeiro tem redefinido as abordagens de análise de investimentos e gestão de ativos. O uso de aprendizado de máquina, em especial as redes neurais profundas, permite a modelagem de dinâmicas não-lineares complexas presentes em séries temporais financeiras, oferecendo a capacidade de identificar padrões históricos para prever comportamentos futuros e, conseqüentemente, informar estratégias de alocação de capital.

Apesar do potencial, a aplicação de tais modelos frequentemente se restringe à previsão de ativos isolados, deixando uma lacuna na transição entre o sinal preditivo e a gestão otimizada de uma carteira diversificada. A simples acurácia em prever a direção de um único ativo não garante um desempenho ótimo quando se gerencia múltiplos ativos simultaneamente, considerando a alocação de risco e capital. Diante disso, o problema de pesquisa que norteia este trabalho é: Como diferentes arquiteturas de redes neurais podem ser empregadas para gerar sinais de negociação e como esses sinais podem ser utilizados em uma estratégia de alocação de capital para otimizar o desempenho de uma carteira de investimentos composta por blue chips² do mercado brasileiro?

O objetivo geral deste trabalho é, portanto, desenvolver e avaliar um *pipeline* completo de aprendizado de máquina que utilize os sinais preditivos de modelos de redes neurais para guiar uma estratégia de otimização de carteira. Para alcançar este fim, foram definidos os seguintes objetivos específicos: (a) implementar e comparar três arquiteturas de redes neurais — *Multi-Layer Perceptron* (MLP), *Convolutional Neural Network* (CNN) e *Long Short-Term Memory* (LSTM) — enriquecidas com *features* de análise técnica; (b) empregar uma metodologia de validação cruzada *Walk-Forward Analysis* (WFA) para o treinamento e avaliação, garantindo a integridade temporal dos resultados; (c) otimizar a função de perda

² Termo utilizado para se referir a ações de empresas de grande porte e solidez financeira.

Huber de forma *data-driven*, ajustando seu parâmetro *delta* à volatilidade de cada ativo; (d) desenvolver e avaliar uma estratégia de alocação de capital que utilize os sinais preditivos para gerenciar dinamicamente uma carteira multi-ativo; e (e) validar a performance financeira da carteira otimizada contra um benchmark de mercado (*Buy and Hold*).

2 DESENVOLVIMENTO

2.1 Trabalhos Relacionados

A aplicação de Redes Neurais Artificiais (RNAs) para a predição de séries temporais no mercado financeiro brasileiro tem sido objeto de estudo recorrente na academia. A pesquisa na área abrange desde a modelagem de estratégias de investimento até a exploração de arquiteturas de rede específicas para a negociação de ativos.

Nesse contexto, Correia (2017) investigou a modelagem de uma estratégia financeira baseada na combinação de diferentes RNAs, com foco em criar um sistema de apoio à tomada de decisão. A abordagem demonstra o potencial de ir além da simples previsão de preços, buscando validar a utilidade prática dos modelos em um contexto de negociação. De forma complementar, o trabalho de Dametto (2018) aprofundou o estudo da aplicação de RNAs para a predição de séries temporais financeiras, reforçando o potencial dessas técnicas para analisar o comportamento de ativos da bolsa brasileira.

Mais recentemente, a pesquisa tem se concentrado em arquiteturas especializadas. Nascimento (2020), por exemplo, aplicou Redes Neurais Convolucionais (CNNs) diretamente à negociação de ativos, explorando a capacidade deste tipo de rede em identificar padrões locais e motivos recorrentes nos dados de mercado.

O presente trabalho se insere nesta linha de pesquisa, buscando expandir as análises anteriores ao realizar uma avaliação comparativa de múltiplas arquiteturas (MLP, CNN e LSTM) e ao introduzir uma metodologia de validação e otimização estruturada, com o objetivo de não apenas prever, mas também validar a eficácia financeira das previsões geradas através de um sistema de *backtesting*.

2.2 Metodologia e Pipeline do Projeto

A metodologia adotada neste trabalho estrutura-se em um pipeline computacional automatizado, projetado para sistematicamente avaliar a capacidade de modelos de aprendizado de máquina em prever séries temporais do mercado de ações brasileiro. O processo abrange desde a aquisição e tratamento dos dados até a avaliação final do desempenho preditivo e sua aplicação em uma simulação de estratégia de investimento.

O fluxo de trabalho compreende três fases principais. A primeira fase consiste no pré-processamento dos dados, que inclui a coleta de dados históricos de ativos selecionados, a aplicação de engenharia de atributos para agregar indicadores técnicos relevantes e a transformação dos dados em um formato de janela deslizando, adequado para o treinamento de modelos supervisionados.

A segunda fase é dedicada à modelagem preditiva. Nesta etapa, são desenvolvidos e avaliados modelos baseados em diferentes arquiteturas de redes neurais, a fim de identificar a abordagem mais eficaz para o problema. O treinamento é conduzido sob uma metodologia de validação que preserva a integridade temporal dos dados, sendo o processo complementado pela otimização dos parâmetros de cada modelo.

Finalmente, a terceira fase concentra-se na aplicação prática, onde a eficácia preditiva dos modelos é utilizada para guiar uma estratégia de otimização de carteira. Nesta etapa, os sinais de previsão gerados para cada ativo individual são agregados e servem como entrada para uma lógica de alocação de capital. O desempenho desta estratégia de portfólio é então avaliado através de um sistema de *backtesting*, que afere o retorno financeiro da carteira gerenciada dinamicamente em comparação a um benchmark de mercado.

2.3 Coleta e Pré-processamento dos Dados

A fase de coleta e pré-processamento de dados é fundamental para o sucesso de qualquer modelo de aprendizado de máquina, especialmente em domínios de séries temporais financeiras. Esta seção detalha o processo adotado, desde a justificativa da escolha dos ativos até a estruturação final dos dados para o treinamento das redes neurais.

2.3.1 O Mercado de Ações Brasileiro e a Escolha dos Ativos

O Índice Bovespa (Ibovespa) é o principal indicador de desempenho das ações negociadas na B3, a bolsa de valores do Brasil. Ele é composto por uma carteira teórica formada pelas ações com maior negociabilidade e representatividade no mercado. O cálculo do índice leva em consideração a média ponderada do valor de mercado ajustado das empresas, considerando o volume negociado nos últimos 12 meses.

As chamadas blue chips referem-se às ações de empresas de grande porte, com comprovada solidez financeira, elevado valor de mercado e liquidez. Esses ativos desempenham papel central na composição do Índice, dada sua representatividade e volume de negociação. Entre os exemplos mais relevantes destacam-se PETR4 (Petrobras), VALE3 (Vale) e ITUB4 (Itaú Unibanco), que foram os ativos selecionados para este estudo. Essas empresas, por possuírem forte influência sobre o comportamento do índice, são alvos recorrentes de análises quantitativas e preditivas.

2.3.2 Aquisição dos Dados Brutos

Para a aquisição dos dados históricos, utilizou-se a biblioteca de código aberto *yfinance* em Python, que provê uma interface de acesso aos dados da plataforma Yahoo Finance³. Foram requisitados os dados diários de cotação — contendo os valores de abertura, máxima, mínima, fechamento e volume (OHLCV) — para cada um dos ativos blue chip selecionados. A extração contemplou um período de cinco anos, garantindo um volume de dados substancial para o treinamento e validação dos modelos. Na chamada da função de download, foi explicitamente configurado o parâmetro *multi_level_index* como falso, a fim de assegurar que o *DataFrame* resultante possuísse uma estrutura de colunas simples e padronizada.

2.3.3 Engenharia de Atributos com Análise Técnica

Considerando que o preço de fechamento isoladamente oferece um contexto limitado sobre a dinâmica do mercado, procedeu-se com a etapa de engenharia de atributos para enriquecer o conjunto de dados. Para tal, utilizou-se a biblioteca *finta*⁴ para calcular e anexar indicadores de análise técnica amplamente utilizados. Foram

³ Disponível em: <https://finance.yahoo.com>. Acesso em: 15 mar. 2025.

⁴ Disponível em: <https://pypi.org/project/finta/>. Acesso em: 15 mar. 2025.

incorporadas Médias Móveis Simples (SMA) de 10 e 50 períodos, com o objetivo de capturar tendências de curto e longo prazo, respectivamente. Adicionalmente, foi incluído o Índice de Força Relativa (RSI) de 14 períodos, um oscilador de momento que indica potenciais condições de sobrecompra ou sobrevenda do ativo. O volume de negociação diário também foi mantido como uma feature relevante.

2.3.4 Preparação das Sequências para Modelagem

A etapa final do pré-processamento consiste em transformar a série temporal multivariada em um formato adequado para o aprendizado supervisionado, utilizando a técnica de janela deslizante (*sliding window*). Para cada ponto no tempo, uma janela de 20 dias consecutivos de todos os atributos é utilizada como a entrada do modelo (X). O alvo da previsão (y) é definido como o preço de fechamento do dia imediatamente subsequente à janela. Antes da criação das sequências, todo o conjunto de dados é normalizado através do *MinMaxScaler* da biblioteca *scikit-learn*⁵, que ajusta todos os atributos para a escala entre 0 e 1. Esta normalização é um passo crucial para garantir a convergência estável e eficiente dos algoritmos de redes neurais. O mesmo objeto *scaler* ajustado nos dados de treino de cada *fold* da validação é então utilizado para transformar os dados de teste correspondentes.

2.4 Arquitetura dos Modelos Preditivos

A fim de avaliar comparativamente a eficácia de diferentes abordagens de aprendizado profundo para a previsão de séries temporais financeiras, foram selecionadas e implementadas três arquiteturas de redes neurais distintas. A escolha contemplou um modelo de *baseline* tradicional (MLP), um modelo capaz de extrair *features* espaciais locais (CNN 1D) e um modelo recorrente, especializado em dependências temporais (LSTM). As subseções seguintes detalham a estrutura de cada um.

2.4.1 Modelo de Baseline: Multi-Layer Perceptron (MLP)

Como ponto de referência (*baseline*), implementou-se um *Multi-Layer Perceptron* (MLP), uma arquitetura de rede neural do tipo *feed-forward*. Para este

⁵ Disponível em: <https://scikit-learn.org/0.21/documentation.html>. Acesso em: 15 mar. 2025

modelo, a janela de entrada de 20 dias com 5 atributos cada é "achatada" (*flattened*), transformando a entrada sequencial em um único vetor de características. A estrutura é composta por múltiplas camadas densas (*Dense*) com a função de ativação ReLU (*Rectified Linear Unit*) para introduzir não-linearidade, sendo intercaladas com camadas de *Dropout* para fins de regularização.

A camada de saída consiste em um único neurônio, uma vez que o objetivo do modelo é a predição de um único valor, correspondente ao preço de fechamento do ativo (GÉRON, 2020). Conforme recomendado para tarefas de regressão onde o valor de saída não possui limites predefinidos, não foi utilizada nenhuma função de ativação nesta camada de saída, permitindo que o modelo produza valores em qualquer escala. Embora o erro quadrático médio seja uma função de perda comum para regressão, a presença de *outliers* nos dados financeiros motivou a escolha da Huber Loss, que, conforme aponta Géron (2020), combina a precisão do erro quadrático para pequenas variações com a menor sensibilidade a grandes erros, característica do erro absoluto.

2.4.2 Modelo Convolutacional (CNN 1D)

A segunda arquitetura avaliada é uma Rede Neural Convolutacional unidimensional (CNN 1D). Esta abordagem utiliza camadas convolucionais (*Conv1D*) que operam deslizando múltiplos filtros (ou *kernels*) através da sequência de entrada. Cada *kernel* é projetado para aprender a detectar um padrão sequencial muito curto, de comprimento não superior ao seu próprio tamanho, como uma pequena formação de preços ou um padrão em um indicador técnico (GÉRON, 2020).

O resultado da aplicação destes filtros é um conjunto de mapas de atributos unidimensionais, onde cada mapa destaca a presença do padrão que seu respectivo *kernel* aprendeu a reconhecer nos dados. Embora camadas convolucionais possam ser combinadas com camadas recorrentes, Géron (2020) aponta que elas são suficientemente poderosas para serem usadas de forma independente na modelagem de sequências, o que justifica a implementação de uma arquitetura puramente convolutacional neste trabalho. No modelo implementado, são utilizadas camadas *Conv1D* com a função de ativação ReLU (*Rectified Linear Unit*), seguidas por camadas *MaxPooling1D* para reduzir a dimensionalidade dos mapas de atributos e *Dropout* para regularização. Após a extração de padrões, uma camada *Flatten*

prepara os dados para as camadas densas finais, que conduzem à predição de um único valor de saída.

2.4.3 Modelo Recorrente (LSTM)

A terceira e mais complexa arquitetura implementada é a *Long Short-Term Memory* (LSTM), um tipo de Rede Neural Recorrente (RNN). Esta arquitetura foi escolhida para superar uma limitação fundamental das redes recorrentes simples: o problema da memória de curto prazo. Devido às transformações sequenciais nos dados, informações dos primeiros passos de uma longa série temporal podem ser perdidas, impedindo que o modelo aprenda padrões de longo prazo (GÉRON, 2020).

Para solucionar esta questão, foram introduzidas células com memória de longo prazo, como a LSTM. Sua arquitetura interna, que inclui múltiplos "portões" (*gates*) e um estado de célula que atua como memória de longo prazo, permite que o modelo retenha informações por períodos mais extensos. Segundo Géron (2020), isso resulta não apenas na capacidade aprimorada de detectar dependências de longo prazo nos dados, mas também em uma convergência de treinamento mais rápida em comparação com células recorrentes básicas. A estrutura implementada neste trabalho é do tipo "empilhada" (*stacked*), utilizando duas camadas *keras.layers.LSTM*⁶ sequenciais.

2.5 Processo de Treinamento e Otimização

A eficácia de um modelo de aprendizado de máquina não depende apenas de sua arquitetura, mas também da solidez de seu processo de treinamento e da otimização de seus parâmetros. Esta seção detalha a metodologia de validação empregada para garantir a integridade temporal dos resultados, bem como os processos de otimização aplicados tanto à função de perda quanto aos hiperparâmetros estruturais das redes neurais.

2.5.1 Metodologia de validação cruzada Walk-Forward Analysis (WFA)

Para a avaliação dos modelos, foi empregada a metodologia de validação cruzada WFA, uma abordagem consistente para séries temporais que preserva a ordem cronológica dos dados, previne o viés de dados futuros (*lookahead bias*).

⁶ Disponível em: <https://www.tensorflow.org/guide/keras?hl=pt-br>. Acesso em: 15 mar. 2025

Esta técnica é considerada uma solução eficaz para o problema do sobreajuste (*overfitting*), um dos maiores desafios no desenvolvimento de estratégias de negociação. Diferentemente da validação cruzada tradicional, esta técnica simula um cenário realista de implantação, onde o modelo é periodicamente re-treinado com novos dados, sendo a aproximação mais fiel possível da maneira como um sistema otimizável é utilizado em tempo real

A configuração do teste *walk-forward*, conforme descrito por Pardo (2008), envolve a divisão sequencial dos dados em múltiplos períodos de otimização (*in-sample*) e teste (*out-of-sample*). O processo implementado neste trabalho inicia-se com o treinamento do modelo em um conjunto de dados históricos de três anos (período *in-sample*). A avaliação é então realizada em uma janela de tempo subsequente de três meses (período *out-of-sample*). Em seguida, a janela de validação "desliza" para frente no tempo em um mês, e os dados do período de teste anterior são incorporados ao conjunto de treinamento, que se expande. Um novo modelo é então treinado do zero com este conjunto de dados atualizado e avaliado na nova janela de teste. Este ciclo se repete até que todo o período de dados disponível seja percorrido.

2.5.2 Otimização da Função de Perda (Huber Loss)

A seleção da função de perda é um passo crítico no treinamento de modelos de regressão. Embora a função baseada no erro quadrático (MSE) seja amplamente utilizada, sua formulação tende a ser influenciada por observações com grandes erros residuais, os chamados *outliers*. O desempenho desta função de perda degrada em dados com distribuições de erro de cauda longa, como é comum em séries financeiras.

Para contornar essa sensibilidade, optou-se pela função de perda de Huber, um critério de M-regressão robusto que combina as melhores propriedades do Erro Quadrático Médio (MSE) e do Erro Absoluto Médio (MAE) (HUBER; RONCHETTI, 2009). Sua característica híbrida opera de forma quadrática para erros pequenos, menores que um limiar δ (*delta*), e de forma linear para erros maiores. Essa estrutura confere maior resistência a valores atípicos, característica do MAE, sem sacrificar a sensibilidade a erros menores, típica do MSE.

A eficácia da *Huber Loss*, contudo, depende da correta definição de seu hiperparâmetro *delta*. Um valor fixo para todos os ativos seria subótimo, visto que

cada um possui um perfil de volatilidade distinto. Portanto, para cada ativo foi analisada a distribuição estatística das variações diárias absolutas de preço e valor de *delta* foi então definido programaticamente como a média aritmética entre o desvio padrão e o terceiro quartil desta distribuição, resultando em uma função de perda adaptada às características de cada série temporal.

2.5.3 Otimização de Hiperparâmetros da Arquitetura

Além da função de perda, os hiperparâmetros da arquitetura de cada rede neural, como o número de neurônios por camada e a taxa de *dropout*, foram otimizados de forma automatizada. Para este fim, utilizou-se a biblioteca *KerasTuner*⁷, configurada com o algoritmo de busca *RandomSearch*. Este algoritmo explora o espaço pré-definido de hiperparâmetros, selecionando combinações aleatórias e treinando modelos para cada uma delas por um número limitado de épocas. A combinação que resultou no menor valor de erro no conjunto de validação (*validation loss*) foi então selecionada como a arquitetura ótima para o treinamento final daquele *fold* específico, garantindo que cada modelo fosse ajustado às nuances do seu respectivo conjunto de dados de treinamento.

2.6 Sistema de Backtesting e Avaliação de Estratégia

A validação final de um sistema de previsão para o mercado financeiro reside em sua capacidade de gerar resultados financeiros superiores a uma estratégia passiva. Portanto, a acurácia preditiva dos modelos desenvolvidos foi utilizada como insumo para um sistema de *backtesting* projetado para simular a performance de uma estratégia de alocação de capital em uma carteira multi-ativo. Esta seção detalha a lógica da estratégia de investimento e a metodologia de simulação empregada para aferir seu desempenho.

2.6.1 Estratégia de Alocação com Filtro de Sinais

A construção de uma estratégia de negociação, conforme Pardo (2008), é composta por três componentes principais: as regras de entrada e saída, o gerenciamento de risco e o dimensionamento da posição (*position sizing*). A estratégia desenvolvida neste trabalho aborda cada um destes elementos. O núcleo

⁷ Disponível em: https://www.tensorflow.org/tutorials/keras/keras_tuner?hl=pt-br. Acesso em: 15 mar. 2025.

da estratégia é um filtro de média móvel de sinais, projetado para gerar decisões de negociação mais estáveis, que são então submetidas a um conjunto de regras de execução para gerenciar o portfólio:

2.6.1.1 Regra de entrada (Compra)

Uma ordem de compra é executada para um ativo somente se todas as seguintes condições forem atendidas: (a) seu sinal suavizado ultrapassar um limiar de confiança positivo; (b) o ativo não estiver atualmente em carteira; e (c) o preço atual do ativo estiver acima de sua média móvel de 50 dias. Esta última condição atua como um filtro de entrada (*entry filter*), cuja finalidade, segundo Pardo (2008), é adicionar uma camada de confirmação ao sinal primário para melhorar a qualidade da operação;

2.6.1.2 Regra de saída (Venda)

Uma posição existente em um ativo é liquidada se qualquer uma das seguintes condições ocorrer: (a) o preço do ativo atingir seu nível de *stop-loss*. A implementação de um *stop-loss* é um mecanismo fundamental de gerenciamento do risco da operação (*trade risk*), que busca limitar a perda de capital em negociações individuais (PARDO, 2008); ou (b) o sinal suavizado do ativo cair abaixo de um limiar de confiança negativo;

2.6.1.3 Manutenção e rebalanceamento

Se uma posição existente não atender a nenhuma das regras de saída, ela é mantida no portfólio. O portfólio é rebalanceado diariamente, e a alocação de capital, um aspecto do dimensionamento da posição (PARDO, 2008), é feita de forma equponderada entre os ativos que satisfazem as condições de compra ou manutenção.

Esta abordagem mantém a característica *Long-Only*, focando em reduzir operações desnecessárias e em proteger o capital. As premissas simplificadoras, como a desconsideração de custos de transação e do *slippage* — definido por Pardo (2008) como a diferença entre o preço esperado de uma ordem e seu preço de execução real —, foram mantidas nesta fase do backtest.

2.6.2 Metodologia de Simulação e Benchmark de Comparação

O processo de simulação foi conduzido pelo script de *backtesting*, que itera cronologicamente sobre os períodos de teste definidos pela validação cruzada WFA. Para garantir a ausência de viés de dados futuros, a simulação carrega, para cada *fold*, o conjunto de modelos específicos que foram treinados apenas com informações disponíveis até o início daquele período.

O desempenho da estratégia é medido pela evolução do valor total da carteira ao longo do tempo. Para contextualizar os resultados, a performance da estratégia ativa é comparada com a de um benchmark de portfólio passivo 1/N. Esta estratégia de referência consiste em alocar o capital inicial em partes iguais entre todos os ativos do universo (1/3 para cada ticker) no primeiro dia da simulação e manter essas posições fixas (*Buy and Hold*) até o final do período. A métrica final de sucesso é o Retorno Total da carteira gerenciada pelos modelos em comparação com o retorno do portfólio passivo de referência.

2.7 Resultados

Esta seção apresenta os resultados quantitativos obtidos através do pipeline de experimentação. Os resultados são divididos em duas análises complementares: primeiramente, o desempenho preditivo de cada arquitetura de rede neural, medido pela acurácia direcional; em seguida, o desempenho financeiro da estratégia de otimização de portfólio, avaliado através do sistema de backtesting.

2.7.1 Desempenho preditivo das arquiteturas

A métrica primária para avaliar a capacidade preditiva dos modelos foi a Acurácia Direcional Média, calculada ao longo dos múltiplos *folds* da validação cruzada WFA. Esta métrica afere a frequência com que cada modelo previu corretamente a direção (alta ou baixa) do preço do ativo no dia seguinte. A Tabela 1 sumariza o desempenho médio de cada arquitetura para os ativos estudados.

Tabela 1 – Acurácia Direcional Média por Ativo e Arquitetura

Ativos (<i>tickers</i>)	Acurácia direcional média (%)		
	CNN	LSTM	MLP
ITUB4.SA	49,13	52,33	47,26
PETR4.SA	48,65	52,80	50,90
VALE3.SA	48,00	52,32	51,38

Fonte: Elaborado pelo autor (2025).

Estes resultados indicam que o modelo LSTM, com sua capacidade de capturar dependências temporais de longo prazo, demonstrou uma vantagem preditiva consistente e estatisticamente relevante sobre o acaso (50%) para todos os ativos no universo do estudo. A questão subsequente, a ser explorada na próxima seção, é se esta vantagem estatística pode ser traduzida em um resultado financeiro positivo através da estratégia de portfólio.

2.7.2 Desempenho da Estratégia de Portfólio (*Backtesting*)

Para validar a eficácia prática dos sinais gerados, foi executado um *backtesting* da estratégia de alocação de portfólio, conforme descrito na seção 2.6.1. A simulação utilizou os sinais dos modelos de melhor desempenho para cada ativo (os "vencedores" identificados na Tabela 1) para gerenciar dinamicamente uma carteira com capital inicial de R\$ 100.000,00. O desempenho da estratégia foi comparado ao de um benchmark passivo 1/N (*Buy and Hold*). Os resultados consolidados são apresentados na Tabela 2.

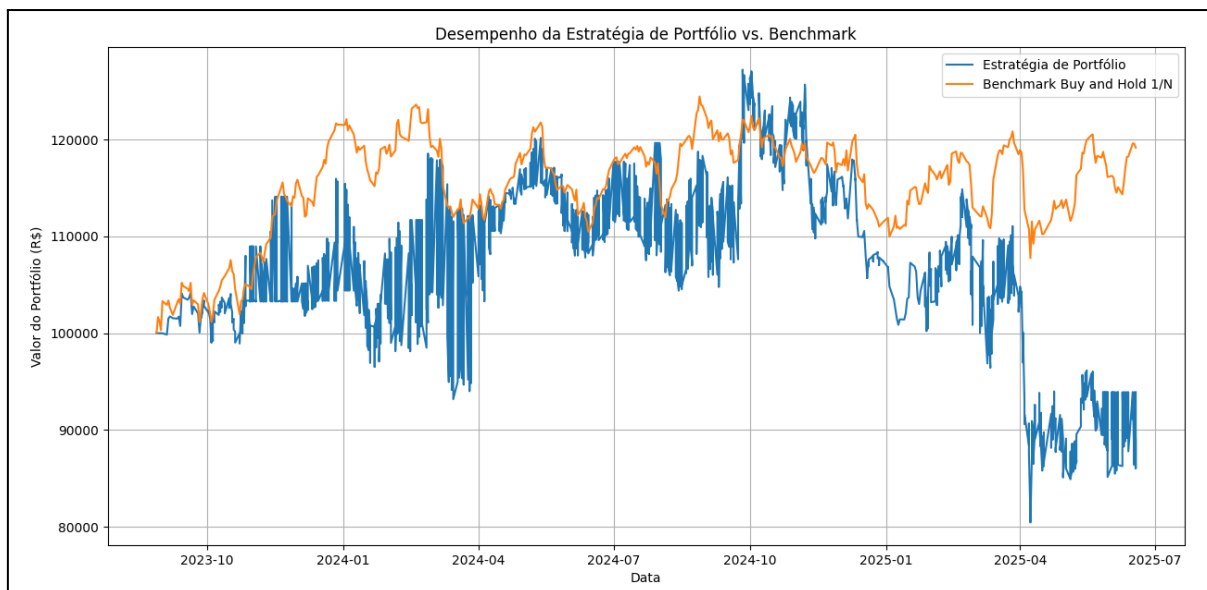
Tabela 2 – Resultado Final do Backtest de Portfólio

Estratégia	Valor final (R\$)	Retorno total (%)
Estratégia do portfólio	93.906,32	-6,09
Benchmark Buy and Hold 1/N	119.156,18	19,16

Fonte: Elaborado pelo autor (2025).

A Figura 1 ilustra visualmente a evolução do capital para ambas as abordagens ao longo do período de teste.

Figura 1 – Desempenho da Estratégia de Portfólio vs. Benchmark



Fonte: Elaborado pelo autor (2025).

Nota 1: Período de simulação de 28/08/2023 a 18/06/2025.

A análise dos resultados quantitativos e da curva de capital demonstra que a estratégia de portfólio ativa, apesar de ser guiada por modelos com uma acurácia direcional superior a 50%, não conseguiu superar o desempenho do benchmark passivo. A estratégia apresentou alta volatilidade e rebaixamentos significativos, resultando em um retorno negativo ao final do período.

3 CONSIDERAÇÕES FINAIS

Este trabalho se propôs a investigar a viabilidade da aplicação de redes neurais artificiais para a otimização de uma carteira de investimentos no mercado de ações brasileiro. Para isso, foi desenvolvido um pipeline completo que realizou a coleta de dados, engenharia de atributos, treinamento, otimização e avaliação comparativa de três arquiteturas de redes neurais — MLP, CNN e LSTM — sob uma rigorosa metodologia de validação cruzada WFA.

A análise do desempenho preditivo dos modelos revelou resultados promissores. Conforme apresentado na Tabela 1, a arquitetura LSTM demonstrou uma superioridade consistente, alcançando uma acurácia direcional média superior a 52% para todos os ativos estudados, indicando uma vantagem estatística sobre uma previsão aleatória. Contudo, o *backtesting* da estratégia de alocação de capital revelou que esta vantagem estatística não se traduziu em um desempenho

financeiro superior ao *benchmark*. A Figura 1 ilustra que a estratégia, apesar de momentos de acerto, apresentou alta volatilidade e rebaixamentos significativos.

Respondendo à questão de pesquisa, conclui-se que a metodologia proposta é eficaz na construção e avaliação de modelos com capacidade preditiva, mas a tradução direta de previsões de preço (regressão) em uma estratégia de alocação de capital, na sua forma atual, não foi suficiente para otimizar o desempenho da carteira de forma a superar uma abordagem passiva. A alta reatividade aos sinais diários, mesmo após filtrações, sugere que a natureza do sinal preditivo deve ser aprimorada.

Diante do exposto, e a partir das conclusões e limitações identificadas, sugerem-se algumas linhas de investigação como trabalhos futuros para a evolução desta pesquisa. Recomenda-se a refatoração do problema de regressão para um de classificação binária, onde o modelo passaria a prever a direção do movimento em vez do valor exato do preço, explorando a geração de probabilidades que podem atuar como sinais de confiança mais estáveis. Adicionalmente, propõe-se a incorporação de novas classes de atributos, como dados macroeconômicos (taxa de juros, inflação) e a cotação do dólar, para fornecer aos modelos um contexto mais amplo sobre o ambiente econômico. Recomenda-se também, como uma importante linha de investigação, a aplicação de metodologias de *ensemble*, que consistem em combinar as previsões dos modelos LSTM, CNN e MLP para gerar um sinal final mais estável e diversificado (GÉRON, 2020), em vez de depender de uma única arquitetura "vencedora" para cada ativo. Por fim, o desenvolvimento de uma interface interativa no formato de dashboard traduziria os achados da pesquisa em uma ferramenta de aplicação prática e educacional, permitindo a exploração visual dos resultados e a execução de novas previsões.

REFERÊNCIAS

CORREIA, J. M. A. **Estudo e modelagem de uma estratégia financeira baseada na combinação de redes neurais artificiais para apoiar a tomada de decisão no mercado financeiro**. 22 fev. 2017. Disponível em:

<https://catalogodeteses.capes.gov.br/catalogo-teses/>. Acesso em: 19 abr. 2025.

DAMETTO, R. C. **Estudo da aplicação de redes neurais artificiais para predição de séries temporais financeiras**. 05 ago. 2018. Disponível em:

<https://catalogodeteses.capes.gov.br/catalogo-teses/>. Acesso em: 19 abr. 2025.

GÉRON, Aurélien. **Mãos à Obra: Aprendizado de Máquina com Scikit-Learn, Keras & TensorFlow**. 2. ed. Rio de Janeiro: Alta Books, 2020.

HUBER, Peter J.; RONCHETTI, Elvezio M. **Robust statistics**. 2. ed. Hoboken: John Wiley & Sons, 2009.

NASCIMENTO, D. G. **Redes neurais convolucionais aplicadas à negociação de ativos no mercado financeiro**. 10 dez. 2020. Disponível em:

<https://catalogodeteses.capes.gov.br/catalogo-teses/>. Acesso em: 12 out. 2024.

PARDO, Robert. **The Evaluation and Optimization of Trading Strategies**. 2. ed. Hoboken: John Wiley & Sons, 2008.