

Sistema de tomada de decisão no mercado de ações utilizando aprendizado de máquina

Thiago R. S. Santos¹, Oswaldo L. V. Costa¹

¹Departamento de Engenharia de Telecomunicações e Controle
Universidade de São Paulo (USP) 05508-010, Cidade Universitária, São Paulo, SP, Brasil

thiago.ssantos@usp.br, oswaldo@lac.usp.br

Abstract. *This work focuses on studying the application of the Random Forest model in classifying the best moments of buying and selling an asset in the Brazilian stock market, thus working as a trading system. Classification of buying and selling moments is learned using market indicators, calculated from the historical series of prices, to enable the application of non-stationary data in the model, during the pre-processing step, the fractional differentiation technique is applied. In addition, this work includes risk management in the trading strategy.*

Resumo. *Este trabalho se concentra no estudo da aplicação do modelo de Random Forest na classificação dos melhores momentos de compra e venda de um ativo no mercado de ações brasileiro, funcionando assim como um sistema de trading. O aprendizado da classificação dos momentos de compra e venda é realizado com uso de indicadores de mercado, calculados a partir da série histórica de preços. Para possibilitar a aplicação de dados não estacionários no modelo, durante a etapa de pré-processamento, a técnica de diferenciação fracionária é aplicada. Além disso, este trabalho inclui o gerenciamento de risco na estratégia de trading.*

1. Introdução

Prever o movimento de um ativo no mercado de ações, algo de interesse comum entre os investidores, representa um grande desafio na área de finanças. De acordo com [Patel et al. 2015] podemos separar a análise de um ativo no mercado de ações em dois tipos. Um primeiro tipo de análise, conhecida como fundamentalista, explora a situação financeira, econômica e até mesmo o setor na qual a empresa se encontra. Um segundo tipo de análise, conhecida como análise técnica, examina dados do passado, como a variação do preço do ativo ao longo do tempo e o volume de transações. Na análise técnica busca-se encontrar padrões nos movimentos que ocorreram no passado, visando prever futuros movimentos. Para isso, é comum aplicar métodos matemáticos na tentativa de detectar esses padrões e em seguida avaliar se esses padrões persistem no futuro. Com o surgimento de alguns algoritmos de aprendizado de máquina e a capacidade destes algoritmos em detectar padrões em problemas com alto grau de complexidade, a aplicação de técnicas de aprendizado de máquina no mercado financeiro ocasionou o surgimento de diversos estudos na área.

Este trabalho tem como objetivo criar um modelo capaz de identificar o melhor momento de compra e venda de um determinado ativo no mercado de ações utilizando o algoritmo *Random Forest*. Para avaliação do modelo, é realizado um *backtest* com a estratégia criada para o período de 2017 até 2022.

O modelo proposto não apenas ajuda a identificar as oportunidades de investimento mais promissoras, mas também fornece uma estratégia de gerenciamento de risco para mini-

mizar possíveis perdas. A técnica de *Stop Loss* é utilizada para limitar as perdas em caso de movimentos desfavoráveis do mercado. Essa abordagem mais completa e integrada pode ser especialmente valiosa em mercados voláteis e incertos, como o período que compreende a pandemia.

2. Revisão Bibliográfica

Em um dos primeiros trabalhos a apresentar a aplicação de uma técnica de aprendizado de máquina na predição do movimento de um ativo no mercado de ações, [White 1988] estudou o uso de uma rede neural na extração de padrões de uma série temporal com os valores de retorno diário. Posteriormente, outros trabalhos apresentaram uma abordagem com uso de alguns indicadores técnicos, além do uso da série temporal com os valores do ativo. O trabalho de [Jang et al. 1991] utilizou, ainda, 16 indicadores técnicos e também aplicou redes neurais para predição do movimento do mercado de ações da bolsa de valores de Taiwan. Nos anos posteriores, novas técnicas de aprendizado de máquina foram criadas ou aperfeiçoadas. Com isso, surgiram novos trabalhos como [Huang et al. 2005], aplicando *Support Vector Machine* na predição do movimento do mercado de ações da bolsa de valores de Tóquio, [Kumar and Thenmozhi 2006] comparando os resultados entre *Support Vector Machine* e *Random Forest* na predição do movimento do mercado de ações da Índia.

Atualmente existem diversos trabalhos relacionados com a aplicação de aprendizado de máquina no mercado financeiro. Alguns trabalhos aplicam aprendizado de máquina na solução de um problema de regressão, onde um modelo é treinado utilizando uma série temporal histórica com valores de preço do ativo e em seguida o modelo treinado é utilizado na previsão de valores de preço futuros. Uma segunda abordagem, se baseia na solução de um problema de classificação, técnica abordada neste trabalho. A literatura sugere que esta abordagem apresenta melhores resultados [Leung et al. 2000]. O sistema proposto neste trabalho foi inspirado no trabalho apresentado por [Nascimento et al. 2020], onde foi aplicado o uso de redes neurais na criação de um modelo capaz de classificar o melhor momento de compra e venda de um ativo, utilizando alguns indicadores técnicos e a série de fechamento estacionária. Porém, neste trabalho, é apresentada a aplicação da técnica de aprendizado de máquina conhecida como *Random Forest*, desenvolvida com base no modelo de árvores de decisão.

Para inclusão da série com valores de fechamento como variável preditora do modelo de *Random Forest*, é também aplicada a técnica de diferenciação fracionária para tratar o problema da não estacionariedade.

Essa abordagem foi inspirada em alguns trabalhos que visam melhorar o desempenho de modelos de aprendizado de máquina com o tratamento prévio de variáveis não estacionárias, como a diferenciação da série de fechamento [Nascimento et al. 2020], a aplicação da técnica de *Empirical Mode Decomposition* [Chowdhury et al. 2019] e a utilização da diferenciação fracionária [De Prado 2018]. Essas abordagens visam transformar a série, reduzindo a dependência temporal e capturando características relevantes para melhorar a capacidade preditiva dos modelos.

3. Metodologia

A metodologia aplicada neste trabalho se baseia em cinco etapas principais: extração de dados, pré-processamento dos dados, treinamento e validação do modelo, validação da estratégia de trading e análise dos resultados.

3.1. Extração e pré-processamento

Primeiramente foram extraídos os dados históricos de cada ativo, contendo valores de abertura O_i , fechamento C_i , máximo H_i , mínimo L_i e volume V_i para cada instante de tempo i . Os dados coletados representam valores diários, sendo assim, a indicação i representa um dia. Após a extração dos dados, foi realizado o cálculo e análise dos indicadores de mercado apresentados na Tabela 1. Além destes indicadores, o valor da própria série de fechamento C_i foi transformada em uma nova série \tilde{C}_i estacionária por meio da diferenciação fracionária. Tanto a série \tilde{C}_i quanto os indicadores de mercado calculados foram utilizados para criação das variáveis preditoras do modelo de aprendizado de máquina.

Tabela 1. Indicadores de Mercado

Indicador	Expressão
RSI	$100 - \frac{100}{1 + \frac{\sum_{l=0}^{13} UP_{i-l}/14}{\sum_{l=0}^{13} DW_{i-l}/14}}$
Williams %R	$-100 \frac{\max_{l=0}^{13} (H_{i-l}) - C_i}{\max_{l=0}^{13} (H_{i-l}) - \min_{l=0}^{13} (L_{i-l})}$
Bollinger Band %	$\frac{(C_i - (MA_i - 2\sigma_i))}{((MA_i + 2\sigma_i) - (MA_i - 2\sigma_i))}$
Keltner Channel %	$\frac{(C_i - (EMA_i - 2ATR_i))}{((EMA_i + 2ATR_i) - (EMA_i - 2ATR_i))}$
CCI	$\frac{(M_i - A_i)}{(0.015D_i)}$
MACD	$EMA_{12i} - EMA_{26i}$
Force Index (FI)	$(C_i - C_{i-1})V_i$
Volume-Price Trend (VPT)	$VPT_{i-1} + (V_i \frac{C_i - C_{i-1}}{C_{i-1}})$
Daily Log Return (DLR)	$\ln(\frac{C_i}{C_{i-1}})$

Notas: UP representa movimentos de subida no preço de fechamento; DW representa movimentos de queda no preço de fechamento; EMA representa a média móvel exponencial do preço de fechamento; MA representa a média móvel simples do preço de fechamento; ATR representa a média móvel do True Range, definido como $\max(H_i - L_i, |H_i - C_{i-1}|, |L_i - C_{i-1}|)$; M_i é a média entre H_i, L_i e C_i ; A_i é a média móvel simples de M_i ; D_i é o desvio médio entre M_i e A_i ;

3.1.1. Indicadores de mercado

1. Relative Strength Index (RSI): este indicador, primeiramente apresentado por [Wilder 1978], é usado para medir a força e a velocidade de um movimento de preço;
2. Williams (%R): reflete o preço de fechamento em relação ao valor mais alto de um determinado período;
3. Bollinger Band %: quantifica a relação entre o valor de fechamento e as bandas de Bollinger e ajudam a identificar pontos de virada em uma tendência.

4. Keltner Channel %: indicador técnico que utiliza bandas de volatilidade para identificar possíveis pontos de entrada e saída no mercado financeiro;
5. CCI: mede a variação de preço do ativo e sua variação de preço médio;
6. Moving Average Convergence Divergence (MACD): mostra a relação entre duas médias móveis de preços, sendo uma de curto prazo EMA_s e uma de longo prazo EMA_l ;
7. Force Index (FI): é um indicador de momento ponderado por volume, calculado como a diferença entre o fechamento atual e o fechamento anterior, multiplicado pelo volume;
8. Volume-Price Trend (VPT): é baseado em um volume cumulativo contínuo que adiciona ou subtrai um múltiplo da alteração percentual na tendência do preço da ação e no volume atual, dependendo dos movimentos de máximo ou mínimo do investimento;
9. Daily Log Return (DLR): é uma medida da variação percentual, no logaritmo natural, do preço de uma ação de um dia para o outro;

3.1.2. Diferenciação Fracionária

Para a utilização da série de fechamento junto com os indicadores de mercado na criação das variáveis preditoras, foi realizado o processo de diferenciação fracionária para que o modelo de aprendizado seja capaz de capturar padrões invariantes no tempo durante o aprendizado. Uma solução comumente adotada em séries temporais financeiras é a transformação da série temporal de preços de um ativo em uma série temporal de retornos. Por outro lado, cada ponto na série de preços possui uma relação de dependência com pontos anteriores, enquanto que a série de retornos apresenta uma perda significativa nesta relação de dependência. Este dilema é abordado por [De Prado 2018], que apresenta uma solução alternativa com uso da diferenciação fracionária. A noção de diferenciação fracionária na aplicação de predição de séries temporais foi apresentada por [Hosking], com o uso da técnica ARIMA. Para entender o processo de diferenciação fracionária na predição de séries temporais, podemos definir o operador B como um atraso. Aplicando o operador B , por exemplo, no preço de fechamento de uma ação $C_i \in \mathbb{R}$ em um dado instante i , obtemos um valor de C_{i-1} que representa o preço de fechamento da ação em um instante $i - 1$, ou seja,

$$BC_i = C_{i-1} \quad (1)$$

Podemos utilizar o operador B para obter uma nova série a partir da diferenciação de C_i como:

$$C'_i = C_i - C_{i-1} = (1 - B)C_i \quad (2)$$

Considerando $i = 1, \dots, n$, a equação (2) representa a transformação da série com valores de preço, normalmente não estacionária, na série de retorno. De forma geral, a diferenciação na ordem d da série pode ser obtida por:

$$(1 - B)^d C_i \quad (3)$$

que pode então ser resolvida com o auxílio da série binomial, com $d \in \mathbb{R}$, da seguinte forma [De Prado 2018]:

$$(1 - B)^d = \sum_{z=0}^{\infty} \binom{d}{z} (-B)^z = 1 - dB + \frac{d(d-1)}{2!} B^2 - \dots \quad (4)$$

Definindo ω como:

$$\omega = \{1, -d, \frac{(d-1)d}{2!}, \dots, (-1)^z \prod_{m=0}^{z-1} \frac{(d-m)}{z!}, \dots\} \quad (5)$$

e sendo ω_z o z -ésimo termo de ω , a série diferenciada na ordem d pode então ser obtida como:

$$\tilde{C}_i = \sum_{z=0}^{\infty} \omega_z C_{i-z} \quad (6)$$

O somatório pode ser aproximado considerando os primeiros termos de ω . Considerando que devemos encontrar o valor de d que torne a nova série obtida uma série estacionária, e que o valor d está vinculado a quantidade de memória que precisa ser removida para obter a estacionariedade, precisamos então encontrar o menor valor d que satisfaça a condição de estacionariedade. Este processo pode ser realizado de modo iterativo, aumentando o valor de d gradativamente e avaliando a não estacionariedade da série utilizando, por exemplo, o teste de Dickey-Fuller Aumentado (ADF) [Dickey and Fuller 1979]. O teste ADF é um teste estatístico usado para verificar se uma série temporal possui raiz unitária, o que significa que a série é não estacionária.

3.1.3. Definição das variáveis preditoras

Com os indicadores de mercado apresentados na Tabela 1 e a nova série de fechamento estacionária \tilde{C}_i , obtida pelo método da diferenciação fracionária, foram definidas as variáveis preditoras que são utilizadas no modelo de aprendizado de máquina. Para facilitar a captura de padrões e tendências no tempo, o vetor $\mathbf{x}_i \in \mathbb{R}^l$, que representa as variáveis preditoras, é composto pelos indicadores e a série de fechamento estacionária calculados entre os instantes de tempo i e $i - 5$. Com a utilização de 9 indicadores de mercado, a série de fechamento estacionária e a utilização de 6 instantes de tempo, temos $l = 60$, ou seja,

$$\mathbf{x}_i = \begin{bmatrix} RSI_i \\ \vdots \\ RSI_{i-5} \\ \vdots \\ \tilde{C}_i \\ \vdots \\ \tilde{C}_{i-5} \end{bmatrix} \quad (7)$$

Definido o vetor \mathbf{x}_i para o instante de tempo i , é preciso relacioná-lo a variável $y_i \in \{-1, 0, 1\}$ capaz de representar uma tendência de subida, descida ou estabilidade no preço do ativo, a partir do ponto i analisado.

3.1.4. Rotulagem dos dados

De modo geral, podemos definir a variável y_i , com 1 representando o melhor momento de compra e -1 representando o melhor momento de venda do ativo, enquanto 0 representa um momento de espera. A técnica de rotulagem dos dados utilizada neste trabalho é semelhante à apresentada por [Nascimento et al. 2020], definindo os momentos de compra, venda e espera com base nos valores de máximo e mínimo do preço do ativo em uma janela móvel. Foi considerada uma janela de 7 dias descartando os valores de máximo ou mínimo que são obtidos nos limites do intervalo avaliado. Com isso, o treinamento de um modelo de aprendizado de máquina pode ser realizado através do conjunto $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$.

3.2. Modelo de aprendizado de máquina

O aprendizado de máquina pode ser definido como “o campo de estudo que dá aos computadores a habilidade de aprender sem serem explicitamente programados” [Simon 2013]. Os algoritmos de aprendizado de máquina são capazes de capturar padrões com uso de dados e possuem aplicações em diversas áreas. Este trabalho se concentra na utilização do modelo conhecido como *Random Forest*, que se baseia na utilização de conjuntos de árvores de decisão.

3.2.1. Árvores de decisão

Árvores de decisão é um tipo de algoritmo de aprendizado de máquina bastante utilizado em problemas de classificação, onde os dados são classificados por meio de uma sequência de testes lógicos definidos como nós. O treinamento de uma árvore de decisão envolve encontrar as regras de decisão que melhor se ajustam aos dados de treinamento. Usando o vetor \mathbf{x}_i como exemplo e representando uma determinada variável preditora através do índice j a construção da regra de divisão é realizada através da condição se $x_{ij} \leq \epsilon$ ou não, onde ϵ é algum limiar e x_{ij} representa o elemento da posição j do vetor \mathbf{x}_i . A construção de uma árvore de decisão é feita pela escolha da variável e da regra associada a variável, que melhor divide um conjunto de itens em cada passo [Breiman et al. 2017]. Por simplificação, representamos os parâmetros j e ϵ do nó m como $\theta_m = (j_m, \epsilon_m)$, onde θ_m define a regra de divisão no nó m . Dada uma amostra de treinamento, a escolha de θ_m é avaliada com base na sua capacidade de separação dos dados no nó m , avaliada através de uma métrica definida como impureza. Representando todo o conjunto de treinamento como $\mathcal{L}_n = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$, com x_{ij} o elemento j do vetor \mathbf{x}_i , e lembrando que θ_1 representa os parâmetros j e ϵ do nó 1, podemos então realizar a primeira divisão dos dados em dois grupos com tamanhos n_e e n_d :

$$\mathcal{L}_{n_d}^{direita}(\theta_1) = \{(\mathbf{x}_i, y_i) | x_{ij_1} \leq \epsilon_1\} \quad (8)$$

$$\mathcal{L}_{n_e}^{esquerda}(\theta_1) = \{(\mathbf{x}_i, y_i) | x_{ij_1} > \epsilon_1\} \quad (9)$$

onde $\mathcal{L}_{n_d}^{direita}$ e $\mathcal{L}_{n_e}^{esquerda}$ representam subconjuntos da amostra de treinamento total \mathcal{L}_n , e:

$$n = n_e + n_d \quad (10)$$

A impureza associada a divisão pode então ser calculada como:

$$G(\mathcal{L}_n, \theta_1) = \frac{n_d}{n} H(\mathcal{L}_{n_d}^{direita}(\theta_1)) + \frac{n_e}{n} H(\mathcal{L}_{n_e}^{esquerda}(\theta_1)) \quad (11)$$

onde H representa uma função de perda. Uma das principais funções de perda utilizada para avaliar o erro da divisão em cada nó é conhecida como *Gini Index*. Considerando que após a divisão da amostra de treinamento \mathcal{L}_n de tamanho n , são geradas novas amostras $\mathcal{L}_{n_d}^{direita}$ e $\mathcal{L}_{n_e}^{esquerda}$ com novos tamanhos n_d e n_e , para generalizar a explicação do cálculo da função *Gini Index* podemos representar de forma geral uma nova amostra gerada como \mathcal{L}_{n_f} de tamanho n_f .

$$H(\mathcal{L}_{n_f}) = \sum_k p_k(1 - p_k) \quad (12)$$

em que p_k representa a proporção de dados com classe igual a k após a divisão no nó analisado. Sendo $I_i(k) \in \mathbb{I}_{n_f}$ o resultado da função indicadora aplicada a y_i presente no conjunto de dados \mathcal{L}_{n_f} , podemos definir p_k conforme a equação abaixo:

$$I_i(k) = \begin{cases} 1 & \text{se } y_i = k \\ 0 & \text{se } y_i \neq k \end{cases} \quad (13)$$

$$p_k = \frac{1}{n_f} \sum_{I \in \mathbb{I}_{n_f}} I_i(k) \quad (14)$$

O parâmetro θ_1 é obtido de modo a minimizar a impureza dada pela equação (11). Em seguida, a árvore é construída recursivamente por meio de uma série de divisões que minimizam a impureza de cada novo nó criado.

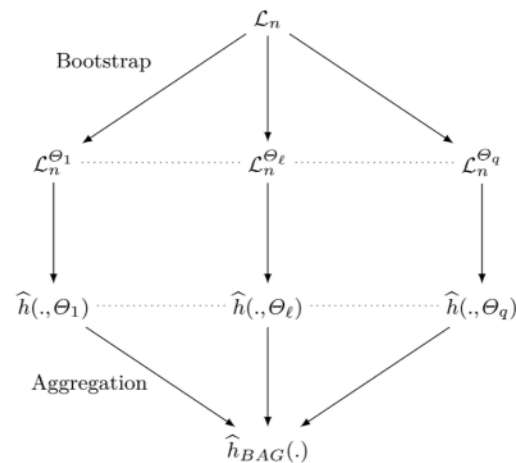
3.2.2. Random Forest

Como uma forma de melhorar o poder preditivo de uma árvore de decisão, [Ho 1995] apresentou pela primeira vez o conceito de *Random Decision Forest*, criando um conjunto de árvores de decisão no mesmo conjunto de dados e utilizando todas as árvores para realização da predição. Em seguida, [Breiman 2001] apresenta uma técnica conhecida como *bagging*, capaz de melhorar a estabilidade e precisão do algoritmo, criando um número adicional de conjuntos de treinamento por amostragem uniforme e substituindo o conjunto de treinamento original. Com isso, surge então a técnica de aprendizado de máquina, conhecida hoje como *Random Forest*. Com os dados de treinamento, são realizadas amostragens com reposição para criação de um novo conjunto de dados. A Figura 1 ilustra o processo de *bootstrap*, onde a partir da amostra de treinamento \mathcal{L}_n novas amostras são geradas. Cada árvore $\hat{h}(\cdot)$ é construída a partir de um novo conjunto de dados gerado. Após a criação de q árvores de decisão, cada árvore tem um peso unitário na classificação final [Breiman 2001].

Atualmente, algumas bibliotecas desenvolvidas na linguagem Python, como a biblioteca *scikit-learn* utilizada neste trabalho, realizam a classificação final do modelo como uma média da probabilidade de predição de cada árvore.

4. Desenvolvimento e resultados

O processo de treinamento e validação foi realizado com o uso de uma técnica de janela deslizante, onde os dados de 2006 até 2022 foram divididos em um conjunto de validação, um conjunto de treinamento e um conjunto de teste. Para cada ano no conjunto de validação foram utilizados os 10 anos anteriores para treinamento e teste do modelo de aprendizado de máquina. Os resultados apresentados nas tabelas 3 e 5 representam os resultados



Fonte: [Genuer and Poggi 2020].

Figura 1. Representação do modelo de Random Forest.

obtidos com o conjunto de validação para os anos de 2017 até 2022. Primeiramente os dados de alguns dos principais ativos listados na bolsa de valores do Brasil foram extraídos, como PETR4, Petróleo Brasileiro SA Petrobras Preference Shares; VALE3, Vale S.A.; ITUB4, Itaú Unibanco; BBDC4, Banco Bradesco SA Preference Shares; ABEV3, AMBEV; BOVA11, iShares Ibovespa Fundo de Índice. Os dados obtidos possuem valores diários de abertura O_i , fechamento C_i , máximo H_i , mínimo L_i e volume V_i . Com isso, para cada ano avaliado, os valores extraídos são divididos de acordo com treinamento, teste e validação. Os indicadores listados na Tabela 1 são calculados e em seguida é realizada a diferenciação fracionária para diferentes valores de d , truncando o somatório na equação (6) até o décimo termo. A Tabela 2 apresenta o resultado obtido para o ativo PETR4 utilizando o período de treinamento de 2011 até 2020.

Tabela 2. Resultado do teste ADF (Augmented Dickey-Fuller) com constante

d	ADF	5%
0,00	-1,72	-2,86
0,11	-2,26	-2,86
0,22	-3,08	-2,86
0,33	-4,36	-2,86
0,44	-6,35	-2,86
0,56	-9,50	-2,86
0,67	-14,40	-2,86
0,78	-21,10	-2,86
0,89	-27,83	-2,86
1,00	-32,35	-2,86

O resultado do teste é apresentado na coluna *ADF* e para rejeitar a hipótese nula e concluir que a série é estacionária precisamos utilizar o valor de d tal que o valor *ADF* seja menor que 2,86, considerando um nível de significância no teste de 5%. O valor de d é definido como menor valor tal que o teste de estacionariedade seja válido. A Figura 2 representa a

transformação da série C_i em \tilde{C}_i para o ativo PETR4, utilizando $d = 0,22$.

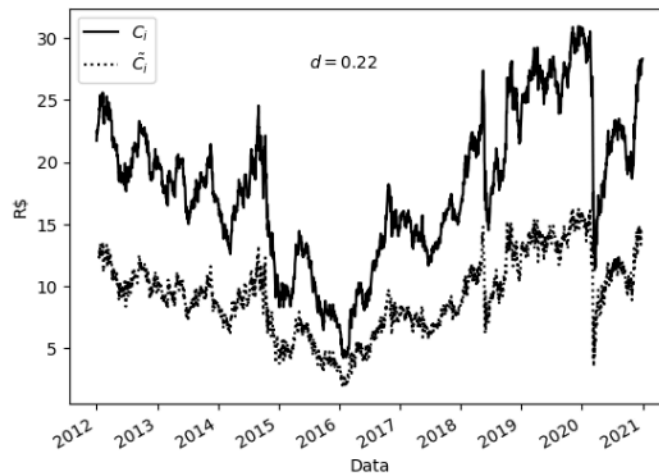


Figura 2. Diferenciação da série de fechamento (PETR4).

Com o valor de d definido a nova série \tilde{C}_i e os indicadores de mercado mencionados anteriormente, o vetor \mathbf{x}_i é obtido conforme equação (7). Após isso, é realizada a rotulagem dos dados, conforme apresentada anteriormente e a variável y_i é obtida. Com o conjunto de treino é realizado o treinamento do modelo de *Random Forest* com o uso de 100 árvores de decisão, um número mínimo de dados por nó de 7 e uma profundidade máxima de 9. Após essa etapa, é realizada a classificação dos dados no conjunto de teste e em seguida o modelo é aplicado no ano de validação, utilizando o mesmo valor de d , obtido com os dados de treinamento, para o cálculo de \tilde{C}_i . O processo ocorre da mesma forma para os outros anos posteriores, obtendo um novo conjunto de treino e retreinando o modelo a cada um ano. Ao final da simulação, os valores obtidos no conjunto de validação foram avaliados.

4.1. Avaliação do modelo de aprendizado

Primeiramente foram avaliados os resultados relacionados à classificação do modelo de aprendizado de máquina. A Tabela 3 apresenta a acurácia obtida em cada ativo, que representa a porcentagem de previsões corretas realizadas pelo modelo em relação ao número total de amostras. Além disso, são apresentadas as métricas de precisão e *F1 Score*. Estas métricas são calculadas individualmente para cada classe e o resultado apresentado representa a média ponderada considerando todas as classes -1 , 0 e 1 .

Tabela 3. Métricas para avaliação do modelo de classificação

Ativo	Acurácia(%)	Precisão(%)	F1(%)
ABEV3	62,37	69,74	64,39
BBDC4	60,92	67,83	62,82
BOVA11	64,30	68,76	65,68
ITUB4	63,54	70,99	65,28
PETR4	63,75	70,92	65,56
VALE3	63,96	69,11	65,62

Apesar da eficácia da estratégia de trading depender da qualidade do modelo de aprendizado de máquina, é necessário avaliar métricas que estão diretamente relacionadas com o

resultado financeiro na aplicação da estratégia, como a taxa de acerto e o retorno sobre o investimento.

4.2. Avaliação financeira

Para avaliação do resultado financeiro do sistema de trading proposto foram considerados inicialmente os primeiros sinais de compra de cada ativo e, a partir de uma posição de compra, o próximo sinal de venda é considerado e assim sucessivamente. Ou seja, não são considerados sinais de compra onde a última posição foi de compra e da mesma forma para venda.

Por simplificação, foi considerado uma unidade genérica para representar o valor de compra do ativo. Dessa forma, o desempenho da estratégia foi medido com base na variação percentual em relação a essa unidade genérica. Além disso, na simulação, considera-se que a entrada envolve a aplicação de todo o capital em uma posição de compra, e a saída implica em zerar completamente essa posição de compra. É importante ressaltar que os resultados apresentados não levam em consideração o custo de cada transação, um fator relevante a ser considerado em uma aplicação prática.

Os resultados obtidos foram comparados com a estratégia de *Buy and Hold* e com uma estratégia de trading popular conhecida como *Moving Average Crossover*, que se baseia no cruzamento de duas curvas, uma que representa a tendência de curto prazo e uma segunda curva representando uma tendência de longo prazo. Visando avaliar a capacidade do modelo em aprender padrões e classificar corretamente os dados de entrada, foram comparados os resultados da taxa de acerto da estratégia proposta com a estratégia de *Moving Average Crossover* (MAC), conforme Tabela 4. A taxa de acerto mede a porcentagem de negociações bem-sucedidas em relação ao número total de negociações executadas. Por possuir apenas uma negociação, a estratégia de *Buy and Hold* não foi avaliada para essa métrica.

Tabela 4. Taxa de acerto e índice de Sharpe

Ativo	Taxa de acerto (%)			Índice de Sharpe		
	Modelo*	Modelo	MCA	Modelo*	Modelo	MCA
ABEV3	59,49	59,49	45,95	0,50	0,50	0,19
BBDC4	59,09	59,09	41,67	0,47	0,33	0,45
BOVA11	72,84	72,84	40,00	0,60	0,37	0,63
ITUB4	63,55	64,49	40,54	0,72	0,48	0,69
PETR4	65,69	66,67	41,03	0,58	0,52	0,27
VALE3	74,70	74,70	55,56	0,87	0,58	0,75

Nota: Modelo* representa o modelo com a inclusão de *Stop Loss* e *Take Profit*.

Quando comparamos apenas a taxa de acerto, podemos notar que em todos os ativos a estratégia proposta apresenta um melhor resultado, indicando um desempenho melhor em termos de identificação de oportunidades de negociação bem-sucedidas. No entanto, há outras métricas e fatores importantes a serem considerados ao avaliar o desempenho de uma estratégia, como o índice de Sharpe que mede a relação entre o retorno da estratégia e o risco assumido. Comparando o índice de Sharpe apresentado na Tabela 4, podemos notar um pior desempenho no modelo, para os ativos BBDC4, BOVA11, ITUB4 e VALE3. Uma possível explicação para uma maior taxa de acerto, mas um menor índice de Sharpe, é que a estratégia de trading pode estar gerando lucros consistentes, mas esses lucros podem

estar sendo obtidos em negociações que apresentam um risco, indicando a necessidade de uma melhor gestão de risco do modelo. Para isso, foi realizada a avaliação de um modelo com a inclusão de *Stop Loss* e *Take Profit* para a proteção do lucro e limitação da perda. Por simplificação, foi considerado o mesmo limite para ambos os ativos, sendo o limite de *Stop Loss* de 10% e o limite *Take Profit* de 20%. É possível notar que o gerenciamento do risco eleva significativamente tanto no índice sharpe quanto no resultado líquido apresentado na Tabela 5. O resultado líquido é apresentado como uma porcentagem do capital investido para cada ativo em diferentes estratégias, incluindo a estratégia de *Buy and Hold*.

Tabela 5. % Percentual de Lucro por ativo e estratégia

Ativo	Modelo*(%)	Modelo(%)	<i>Buy And Hold</i> (%)	<i>MAC</i> (%)
ABEV3	42,94	4,57	-11,52	8,56
BBDC4	43,89	23,44	-4,32	41,31
BOVA11	46,94	26,98	75,21	49,87
ITUB4	78,00	47,22	4,69	76,81
PETR4	69,24	66,59	56,45	12,00
VALE3	117,23	79,98	205,12	113,94

Nota: Modelo* representa o modelo com a inclusão de *Stop Loss* e *Take Profit*.

A gestão de risco é uma parte fundamental de qualquer estratégia de trading, independentemente do momento do mercado. No entanto, em períodos de crise e incerteza, como durante a pandemia, a importância de gerenciar adequadamente o risco se torna ainda mais crucial.

5. Conclusão

Os resultados da avaliação do modelo de aprendizado mostram que ele obteve uma acurácia semelhante aos apresentados em trabalhos anteriores. Por sua vez, a taxa de acerto da estratégia proposta revelou um desempenho satisfatório, em linha com as expectativas dadas a acurácia do modelo de aprendizado utilizado. Já o retorno líquido obtido com a estratégia de trading proposta mostram que, em geral, ela obteve um desempenho superior em relação às estratégias de *Buy and Hold* e *Moving Average Crossover*, especialmente quando considerada a inclusão de *Stop Loss* e *Take Profit*. Vale destacar que alguns fatores externos interferem no mercado de ações fazendo com que os preços subam ou desçam inesperadamente, como o efeito da pandemia. Em futuros trabalhos seria interessante avaliar a inclusão de diferentes valores de *Stop Loss* e *Take Profit* para cada ativo. Além disso, os parâmetros utilizados no cálculo dos indicadores técnicos podem ser estudados. Outros tópicos interessantes seriam a inclusão de novas variáveis preditoras, a utilização de diferentes técnicas de rotulagem dos dados e a avaliação de outros modelos de aprendizado de máquina.

Referências

- Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.
- Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (2017). *Classification and regression trees*. Routledge.
- Chowdhury, U. N., Chakravarty, S. K., Hossain, M. T., and Ahmad, S. (2019). Empirical mode decomposition based ensemble random forest model for financial time series forecasting.
- De Prado, M. L. (2018). *Advances in financial machine learning*. John Wiley & Sons.
- Dickey, D. A. and Fuller, W. A. (1979). Distribution of the estimators for autoregressive time series with a unit root. *Journal of the American statistical association*, 74(366a):427–431.
- Genuer, R. and Poggi, J.-M. (2020). Random forests. In *Random Forests with R*, pages 33–55. Springer.
- Ho, T. K. (1995). Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282. IEEE.
- Hosking, J. M. (1981), “. *Fractional differencing*. *Biometrika*, 68(1):165–76.
- Huang, W., Nakamori, Y., and Wang, S.-Y. (2005). Forecasting stock market movement direction with support vector machine. *Computers & operations research*, 32(10):2513–2522.
- Jang, G.-S., Lai, F., Jiang, B.-W., and Chien, L.-H. (1991). An intelligent trend prediction and reversal recognition system using dual-module neural networks. In *Proceedings First International Conference on Artificial Intelligence Applications on Wall Street*, pages 42–43. IEEE Computer Society.
- Kumar, M. and Thenmozhi, M. (2006). Forecasting stock index movement: A comparison of support vector machines and random forest. In *Indian institute of capital markets 9th capital markets conference paper*.
- Leung, M. T., Daouk, H., and Chen, A.-S. (2000). Forecasting stock indices: a comparison of classification and level estimation models. *International Journal of forecasting*, 16(2):173–190.
- Nascimento, D., Costa, A., and Bianchi, R. (2020). Stock trading classifier with multichannel convolutional neural network. In *Anais do XVII Encontro Nacional de Inteligência Artificial e Computacional*, pages 282–293. SBC.
- Patel, J., Shah, S., Thakkar, P., and Kotecha, K. (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert systems with applications*, 42(1):259–268.
- Simon, P. (2013). *Too big to ignore: the business case for big data*, volume 72. John Wiley & Sons.
- White, H. (1988). Economic prediction using neural networks: The case of ibm daily stock returns. In *ICNN*, volume 2, pages 451–458.
- Wilder, J. W. (1978). *New concepts in technical trading systems*. Trend Research.