

ESTIMANDO MOMENTOS DE COMPRA E VENDA DE BITCOINS USANDO ÁRVORES DE DECISÃO

ESTIMATING BITCOIN BUYING AND SELLING MOMENTS USING DECISION TREES

ESTIMACIÓN DE MOMENTOS DE COMPRA Y VENTA DE BITCOINS UTILIZANDO ÁRBOLES DE DECISIÓN

Itiberê Gonçalves Silva Filho¹
Milton Roberto Heinen²

RESUMO: A popularidade do *bitcoin* vem despertando o interesse de novos investidores no mercado das criptomoedas, porém, muitos acabam endividados pelos erros cometidos. Em vista disso, a proposta deste trabalho é elaborar um modelo capaz de auxiliar investidores novatos em suas decisões de compras e vendas de bitcoins através da técnica de árvore de decisão e de indicadores gráficos. O modelo testou a capacidade de aprendizado e predição da técnica com simulações e validações cruzadas. Os resultados mostraram que foi possível gerar um *feedback* positivo nas simulações ao longo do período histórico da criptomoeda, onde o algoritmo foi capaz de gerar operações que agregassem um lucro significativo em um longo prazo.

Palavras-chave: Bitcoin. Árvore de decisão. Predição.

1272

ABSTRACT: The popularity of bitcoin has aroused the interest of investors in the cryptocurrency market, however, many end up in debt for the mistakes made. In view of this, the purpose of this work is develop a model that can help novice investors in their decisions to buy and sell bitcoins through the decision tree technique and graphic indicators. The model tested the technique's ability to learn and predict with simulations and cross-validation. The results showed that it was possible to generate positive feedback in the simulations over the cryptocurrency's historical period, where the algorithm was able to generate operations that added significant profit in the long term.

Keywords: Bitcoin. Decision Tree. Predict.

RESUMEN: La popularidad del bitcoin ha despertado el interés de nuevos inversionistas en el mercado de las criptomonedas, sin embargo, muchos terminan endeudados por los errores cometidos. Ante esto, el propósito de este trabajo es desarrollar un modelo capaz de ayudar a los inversionistas novatos en sus decisiones de compra y venta de bitcoins a través de la técnica del árbol de decisión e indicadores gráficos. El modelo probó la capacidad de la técnica para aprender y predecir con simulaciones y validación cruzada. Los resultados mostraron que era posible generar un *feedback* positivo en las simulaciones durante el período histórico de la criptomoneda, donde el algoritmo fue capaz de generar operaciones que agregaron una ganancia significativa a largo plazo.

Palabras clave: Bitcoin. Árboles de Decisión. Predicción.

¹ Graduando em Engenharia de Computação – Universidade Federal da pampa (UNIPAMPA). E-mail: itibere.filho@gmail.com.

² Doutorado em Ciência da Computação – Universidade Federal do Rio Grande do Sul (UFRGS).

INTRODUÇÃO

Os criptoativos movimentam uma grande parte do mercado de negócios e continuam ganhando extrema relevância e popularidade entre os meios de investimento nos tempos atuais. Segundo Bashir I (2017), a descentralização gerada pela ausência de um órgão regulador, o anonimato e o funcionamento totalmente virtuais são fatores atraentes que servem como portas de entrada para novos investidores.

O mercado das criptomoedas já é uma realidade na vida cotidiana de muitas pessoas, como nos informa Kovacs L (2021). Por isso, é comum nos dias de hoje um investidor decidir reservar uma parte de seu capital para as criptomoedas por diversos fatores, como um receio motivado pela situação econômica mundial ou por consequência dos impactos sofridos nas economias de diversos países e em suas respectivas moedas.

Para Martins C (2010), uma das principais características é a volatilidade das criptomoedas, característica que acarreta em um comportamento muito mais imprevisível e de altíssimo risco nos curto e médio prazos quando comparado com outros ativos financeiros. Por consequência, alterações maiores podem fazer grandes diferenças na situação geral da moeda, além de fazer muitos compradores perderem dinheiro.

1273

Seja por erros de inexperiência, desatenção ou até mesmo alguns erros emocionais, Graham B (2003) diz que muitos investidores acabam por não ganhar lucro em suas operações no mercado, seja de criptomoedas como em uma bolsa de ações mais tradicional. Dessa forma, o objetivo aqui proposto é desenvolver um modelo capaz de auxiliar investidores iniciantes a obterem um resultado positivo e descobrirem bons momentos de compra ou venda de *bitcoins* utilizando uma das mais consolidadas técnicas de aprendizado de máquina, a árvore de decisão.

O modelo implementado visa análises puramente matemáticas e técnicas desenvolvidas na linguagem de programação *python*, explorando indicadores gráficos e seus respectivos cálculos, tendências dos preços e a análise numérica de todos os dados através do algoritmo de aprendizado de máquina.

MÉTODOS

Abordando os indicadores gráficos selecionados para a elaboração do código e incrementação do conjunto de dados, foram selecionados para os cálculos cinco indicadores já

consolidados na literatura e muito utilizados pelos investidores, são eles o RSI, MACD, Histograma MACD, estocástico e as Bandas de Bollinger (LEMOS FACA, 2018).

Os indicadores servem de parâmetros para o aprendizado e elaboração da lógica da árvore de decisão. Como nos fala Elder A (2002), podemos repartir os indicadores técnicos abordados entre acompanhadores de tendências e osciladores, onde os osciladores auxiliam nas detecções de sobrecompra ou sobrevenda dos preços (como o RSI e o estocástico), e os acompanhadores de tendência auxiliam concentram na análise da própria tendência (como o MACD, por exemplo).

Segundo RUSSEL S e NORVIG P (2013), o aprendizado de máquina serve como um parâmetro para implementar um agente inteligente capaz de melhorar sua análise e velocidade de resposta partindo dos dados dispostos para ele. Como um agente, a árvore de decisão é capaz de atuar em seu aprendizado como um método para aproximar funções-alvo de valor discreto onde a função aprendida é representada por uma árvore de decisão, assim, alcançando sua resposta final (decisão) através da execução de uma sequência de testes (MITCHELL, 1997).

O algoritmo da árvore de decisão foi escolhido para o desenvolvimento por ser um algoritmo simples e eficiente nas classificações desejadas, além de gerar resultados positivos na base de dados. Outra vantagem da utilização do modelo é sua demonstração não abstrata dos resultados, não sendo um algoritmo caixa-preta e, por consequência, auxiliando visualmente em suas tomadas de decisões e aprendizados.

1274

Destacando trabalhos correlatos, Velankar S, et al. (2018) procuram obter o valor do *bitcoin*, seja através de tentativas de detecção de tendências diárias ou através da otimização de parâmetros capazes de afetar seu valor. Os algoritmos adotados no trabalho foram a regressão bayesiana e as florestas aleatórias.

No trabalho de Rathan K, et al. (2019), objetivo é derivar a previsão do *bitcoin* utilizando diferentes algoritmos de aprendizado de máquina para realizar uma comparação de precisão, onde os resultados da experiência realizada comparados são a árvore de decisão e a regressão linear.

O trabalho de McNally S, et al. (2018) tem como foco verificar com qual precisão a direção do preço do *bitcoin* em dólares americanos consegue ser prevista. A implementação é realizada com diferentes graus de sucesso entre uma rede neural bayesiana recorrente otimizada (RNN) e um modelo LSTM (*long short term memory*). Também é utilizado o modelo ARIMA (*autoregressive integrated moving average*) para uma comparação entre os algoritmos de *deep learning* RNN e LSTM.

O processo de configuração da base de dados teve seu início com a etapa de carregamento dos dados históricos da criptomoeda *bitcoin* diretamente do servidor do *Yahoo Finance*. Utilizando a linguagem *python*, são obtidos os dados do preço de abertura, fechamento, alta, baixa e volume da criptomoeda *bitcoin*, gerando assim a estrutura inicial do conjunto de dados estruturado na forma de um *dataframe* (matriz). As linhas da matriz indicam cada um dos dias obtidos do período histórico, e as colunas representam os preços de abertura, fechamento, alta, baixa e o volume de cada um dos dias.

Após a obtenção dos dados históricos, são calculados, para cada um dos dias de todo o período histórico, os indicadores gráficos RSI, MACD, HMA, Bandas de Bollinger e Estocástico. Os indicadores em questão foram escolhidos por serem já consolidados na literatura do mercado de investimentos e ações. Com o objetivo de se realizar uma lógica de classificação e saída com valores e parâmetros não utópicos, como fechamentos ou altas, por exemplo, foi calculada também a métrica *High, Low, Close divided by 3* (HLC₃) de todos os dias presentes no período.

Utilizando a métrica HLC₃, foi elaborada a lógica de classificação ideal para o aprendizado da árvore de decisão. A lógica é um laço de repetição responsável por checar, pelo período determinado pelo usuário, a movimentação das tendências de altas ou baixas no preço da criptomoeda *bitcoin* utilizando como linha de operação os valores gerados pela métrica. Uma sequência de tendências de alta no período indica um momento bom para o algoritmo realizar uma venda, já em uma sequência de tendências de baixa, temos boas indicações de realizar novas compras. Assim, temos uma lógica de classificação baseada em indicadores para apontarem ações de aguardos, compras e vendas da criptomoeda *bitcoin*.

Depois da implementação da lógica de classificação obtida com a métrica HLC₃, são inseridos os indicadores gráficos e a métrica como novas colunas do *dataframe*, gerando assim a estrutura dos dados finais a serem enviados para o aprendizado da árvore de decisão, contendo os valores previsores e sua respectiva classificação.

A Figura 1 representa o gráfico obtido pela lógica do período na métrica HLC₃ seguindo uma tendência de 5 dias. Conseguimos observar que a lógica consegue acompanhar de maneira eficiente as tendências, gerando momentos positivos de compras e vendas e, por consequência, obtendo boas classificações para servirem de aprendizado para a árvore de decisão.

Figura 1 - Gráfico demonstrativo de lógica ideal com o período do HLC3 em 5 dias.



Fonte: SILVA FILHO IG, HEINEN MR, 2022.

Durante os testes realizados, foi possível notar que o modelo possuía um desempenho muito bom em seus *scores* de generalização através da validação cruzada (cerca de 95\% de acerto) em sua configuração padrão/default, ou seja, sem otimização de parâmetros. Todavia, uma capacidade de generalização não indica um bom sucesso nas previsões do modelo, e isso foi observável nos testes preliminares realizados durante a execução do código, onde os algoritmos não conseguiram realizar boas previsões de compras e vendas.

1276

A solução encontrada para melhorar a previsão da árvore foi utilizar uma busca massiva por melhores parâmetros para o algoritmo com um processo de comparação de *scores*. A busca foi realizada através da função *GridSearchCV*, obtida na biblioteca *Sci-kit learn*, que consiste em um processo de pôr busca exaustiva automatizada para checagem da melhor combinação de parâmetros obtida nos parâmetros pré-selecionados.

Os parâmetros da árvore de decisão selecionados para serem otimizados foram o *criterium* (variando entre *entropy* e *gini*), o parâmetro *splitter* (variando entre *best* e *random*), *MaxDepth* (variando entre 5, 10, 15, 20, 25 ou 30) e, por fim, o parâmetro *RandomState* (variando entre 10, 20, 30, 40 e 50). A técnica de busca massiva aumenta o tempo de execução do código, mas melhora significativamente o resultado das previsões em abordagens e simulações reais.

Para fins de padronização entre execuções gerais e os processos de simulação, foram executados diversos testes preliminares durante a elaboração do modelo. Os testes consistiram

principalmente em encontrar uma abordagem satisfatória de visualização e divisão da base de dados para os parâmetros de treino e teste.

Foram testadas inicialmente as abordagens em divisões de treino e teste (*train-test splits*) com algumas divisões de porcentagem em 60\%-40\%, 70\%-30\%, 80\%-20\% para os respectivos treinos e testes. Como as divisões acabavam selecionando apenas alguns períodos da base de dados, a estratégia de divisão entre treino e teste passou a utilizar a função *CrossValPredict*, obtida da biblioteca *Sci-kit learn*. A função é responsável por realizar divisões e previsões do conjunto automatizadas nos dados, respeitando o mesmo comportamento da validação cruzada com *k-folds*.

Com a divisão automática, foi possível gerar e simular momentos de compra e venda durante todo o período histórico do *bitcoin* sem conflitos ou repetições de dados, diminuindo o *overfitting* e incrementando a visualização da capacidade de cada um dos modelos de aprendizado de máquina para todo o período de atuação. Por conta dos novos conjuntos (*folds*) da divisão, a função acaba incrementando a latência de execução do *script*, porém realiza previsões equivalentes a uma função comum de *predict* para cada uma das colunas (*rows*) do conjunto de dados.

Após serem realizadas as predições de saída do modelo, as mesmas sofrem a aplicação de alguns filtros de checagem para que as saídas geradas correspondam com simulações de *trades* e sejam executadas corretamente. As regras a serem aplicadas pelos filtros de classificação são as seguintes:

- O primeiro momento de interação deve ser uma compra, caso não seja, o algoritmo deverá começar a margem disponível comprando para determinar as operações de *trading* subsequentes.
- O último momento de interação deve ser uma venda, caso não seja, o algoritmo ignora as **X** compras realizadas finalmente até atingir a venda, demarcando o término de todas as operações.
- O algoritmo deve sempre alterar entre uma compra seguida de uma venda, sempre nesta ordem. Caso a ordem não esteja habilitada, são selecionados os primeiros momentos de compra e venda de uma possível sequência das tendências, de modo a deixar o algoritmo mais parecido com uma escolha humanizada, pois em situações reais dificilmente conseguiremos realizar uma operação em um momento ideal.

- Para funcionar corretamente, o algoritmo deve possuir a mesma quantidade de compras e vendas realizadas, onde $C[] - V[] = 0$. Caso $C[] > V[]$ as últimas X compras serão desconsideradas para realizar uma equivalência. Caso $V[] > C[]$ as últimas X vendas serão desconsideradas.

Com a função de predição pronta e os filtros devidamente aplicados nas saídas, o modelo já é capaz de realizar automaticamente as configurações da base de dados, encontrar os melhores hiperparâmetros para o algoritmo de *machine learning*, gerar as médias de acurácia (validação cruzada) do mesmo e realizar previsões e simulações de *trades* com os dados.

A última parte do processo de aprendizado e execução foi responsável por realizar os cálculos e configurações de exibição dos dados obtidos para as simulações de operações (*trades*). O código gera como saída uma tabela contendo a quantidade de *trades* realizadas, o valor de cada uma das compras e sua equivalência na criptomoeda *bitcoin*, o valor de venda de cada operação, o valor de lucro e/ou prejuízo e o processo de investimento, assim como o montante inicial investido e o resultado obtido após todas as *trades* serem realizadas (montante final). A Figura 2 demonstra um exemplo de tabela gerada com os valores do investimento e suas respectivas operações de lucro ou prejuízo.

Figura 2 - Demonstração da tabela de saída das simulações de *trades* realizadas

INDICE	COMPRAS	QUANTIDADE	VENDAS	LUCRO OU PREJUIZO	INVESTIMENTO (US\$)
1	459.31	1.08858939	753.7	320.47	820.47
2	2931.75	0.27985638	1825.95	-309.46	511.0
3	2374.65	0.21519119	2600.49	48.6	559.6
4	8170.75	0.06848858	18006.74	673.65	1233.26
5	50062.41	0.02463438	56077.1	148.17	1381.42
TOTAL	63998.87	-	79263.99	881.42	1381.42

Montante inicial: 500

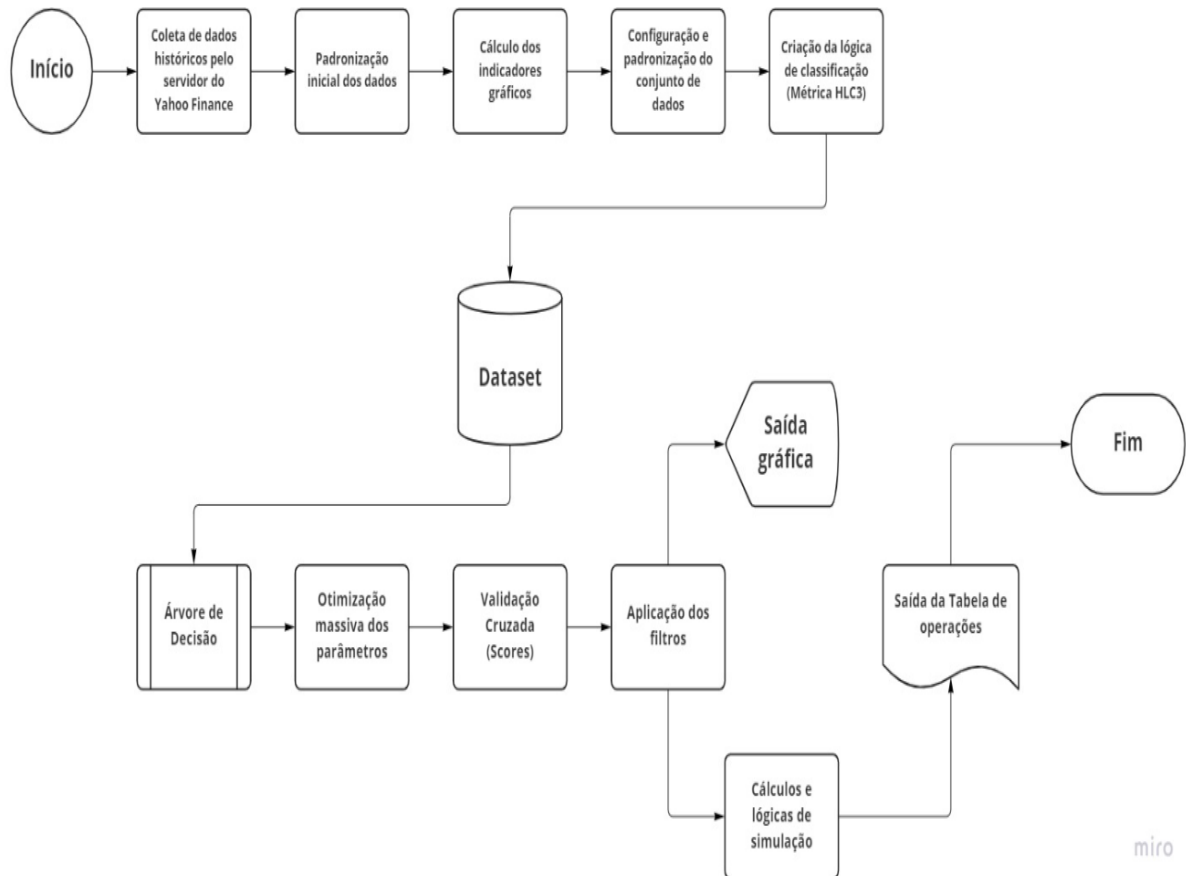
Montante final: 1381.42

Fonte: SILVA FILHO IG, HEINEN MR, 2022

Por fim, temos na Figura 3 um fluxograma ilustrativo de todo o processo de execução do código. O modelo contido no fluxograma retrata as etapas de obtenção dos dados históricos pelo servidor do *Yahoo Finance*, o cálculo e adição dos indicadores gráficos e da métrica HLC_3 no conjunto de dados, o processo de execução do algoritmo de árvore de decisão, as otimizações de

parâmetros e validações cruzadas, os filtros de saídas obtidas do modelo, cálculos lógicos e as informações de saída disponibilizadas para o usuário.

Figura 3 - Fluxograma do processo de execução completo do código



Fonte: SILVA FILHO IG, HEINEN MR, 2022.

RESULTADOS E DISCUSSÕES

Para comprovar a eficiência do código e do algoritmo de árvore de decisão atuando nos dados históricos da criptomoeda bitcoin, foram realizados dez testes de todo o fluxo de execução do script em *python*, como demonstrado na Figura 4. Em cada um dos dez testes realizados, estão informados o período utilizado na métrica HLC₃ (coluna *P.HLC₃*), a porcentagem de acerto da validação cruzada (coluna *VC*), a quantidade de operações realizadas (coluna *Trades*), o montante inicial dos investimentos (coluna *MI*), o montante final obtido (coluna *MF*) e, por fim, o valor de lucro ou prejuízo obtido ao final do investimento (coluna *LouP*).

Figura 4 - Testes executados do código

Teste	P. HLC3 (dias)	VC (%)	Trades	MI (US\$)	MF (US\$)	LouP (US\$)
1	5	97.21	2	500	2676.14	2176.14
2	4	92.25	3	500	17014.18	16514.18
3	3	88.59	5	500	4553.09	4053.09
4	6	95.54	2	500	7872.78	7372.78
5	7	96.34	2	500	13161.38	12661.38
6	8	96.78	1	500	65418.89	64918.89
7	9	99.31	1	500	9769.11	9269.11
8	2	75.18	7	500	23095.77	22595.77
9	1	56.7	24	500	4941.17	4441.17
10	10	98.59	3	500	31236.78	30736.78

Fonte: SILVA FILHO IG, HEINEN MR, 2022.

Com os dados e resultados obtidos, foi possível perceber que o algoritmo de árvores de decisão atuou de maneira muito eficiente no conjunto de dados da *bitcoin* e seus respectivos indicadores gráficos adicionados. O pior valor obtido descreve um lucro de US \$2174.14, com a realização de apenas duas *trades* com o montante inicial de US \$500. Abordando o melhor lucro atingido, conseguimos notar um período da métrica HLC₃ seguindo uma tendência de 8 dias, onde o algoritmo conseguiu, partindo do mesmo montante inicial de 500 dólares, realizar um lucro de US \$64918.89. 1280

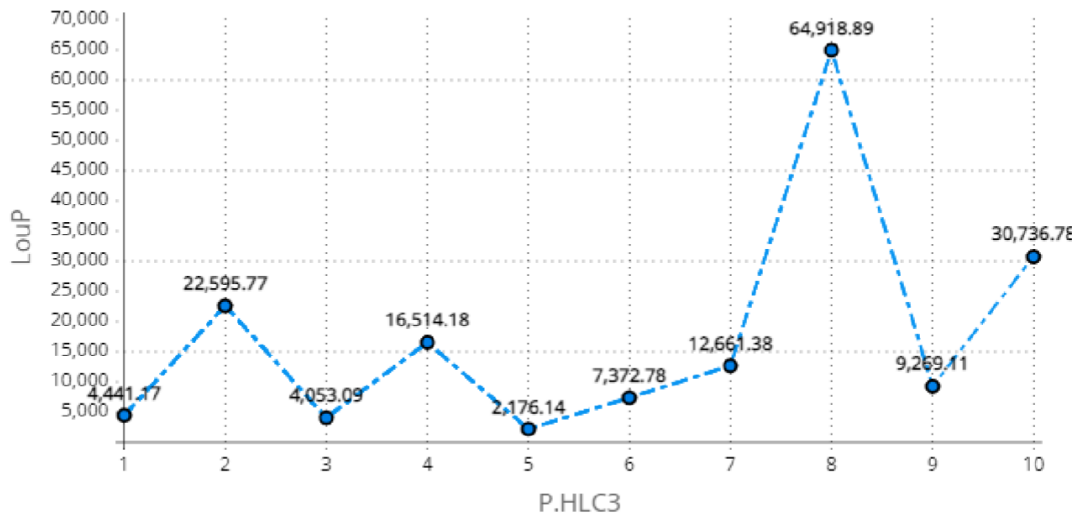
Através da disponibilidade dos dados finais, conseguimos perceber duas regras que podem ser generalizadas e adotadas como esperadas para o modelo, são elas:

- Períodos da métrica HLC₃ mais elevados tendem a realizar menos *trades*, já os períodos menores, realizam mais operações;
- Períodos menores na métrica HLC₃ influenciam na taxa de acerto (*scores*) da validação cruzada, pois, acaba sendo mais difícil para o modelo generalizar uma classificação com valores mais aleatórios e variados. Para casos com menores períodos, a taxa de acertos acaba sendo mais elevada, mantendo uma proporcionalidade inversa já esperada.

A lógica pensada para o modelo seria atuar em opções de períodos maiores, garantindo mais segurança para investidores novatos, porém, foi perceptível a capacidade do modelo de conseguir obter lucro em todos os testes realizados, independente do período selecionado. Os períodos

acabaram influenciando no aprendizado da árvore, porém, não foi encontrada nenhuma lógica crescente ou decrescente visível através dos dados, logo, o resultado é fruto do raciocínio do algoritmo combinado com a aplicação dos filtros. A **Figura 5** demonstra os períodos da métrica HLC₃ comparados com suas taxas de lucro.

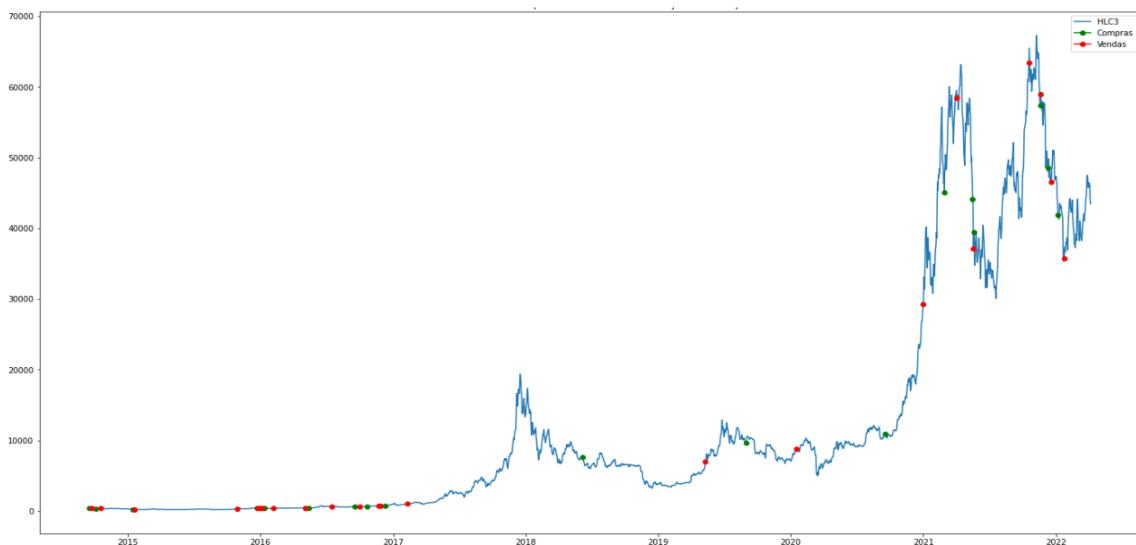
Figura 5 - Lucro ou prejuízo pelo período da métrica HLC₃.



Fonte: SILVA FILHO, IG, HEINEN MR, 2022.

Outra saída do modelo é a exibição gráfica gerada pelas trades realizadas pelo algoritmo da árvore de decisão ao longo do período da métrica utilizado. A **Figura 6** demonstra as operações realizadas e escolhidas pelo algoritmo em uma das execuções completas do código.

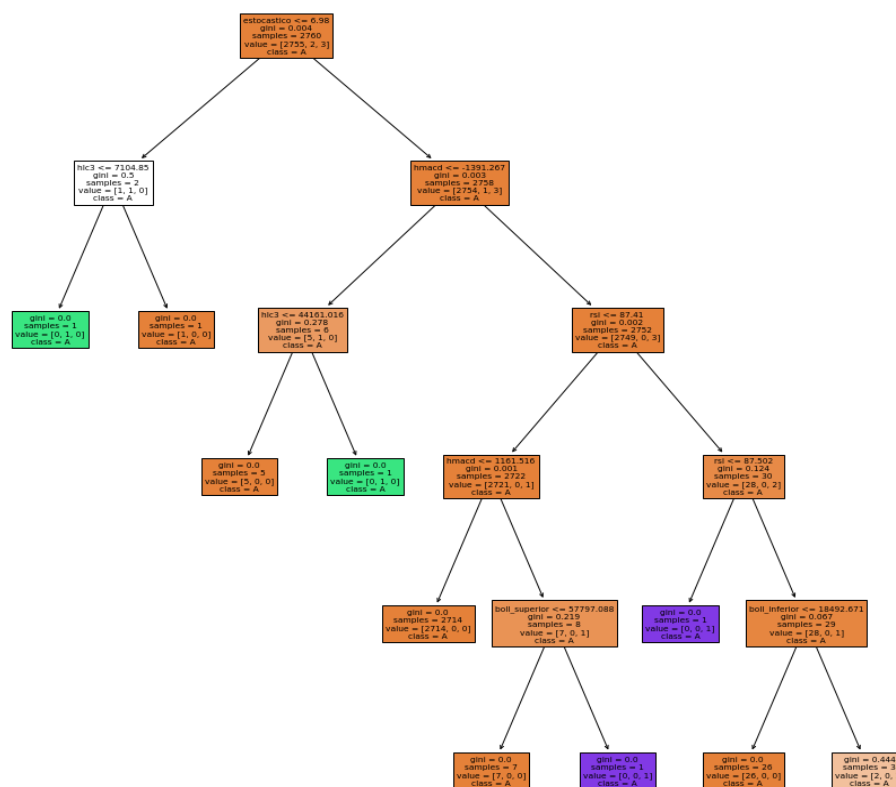
Figura 6 - Momentos de compra e venda realizados pelo algoritmo de árvore de decisão.



Fonte: SILVA FILHO IG, HEINEN MR, 2022.

Por fim, a última saída gerada pelo código é o diagrama da árvore de decisão, demonstrando, através de uma ilustração, toda a lógica de raciocínio e aprendizado utilizada pelo algoritmo para compreender o conjunto de dados do período histórico e seus valores obtidos pelo cálculo dos indicadores gráficos para gerarem uma classificação significativa e eficiente como resposta. A Figura 7 demonstra o diagrama com todo o processo de aprendizado da árvore de decisão gerado em uma das execuções do script.

Figura 7 - Diagrama das estratégias para os momentos de compra e venda criadas pela árvore de decisão.



Fonte: SILVA FILHO IG, HEINEN MR, 2022.

CONSIDERAÇÕES FINAIS

Este trabalho buscou detalhar o processo de desenvolvimento de um modelo em código utilizado para executar e simular operações de compras e vendas na criptomoeda *bitcoin* através do algoritmo de árvore de decisão. Com base nos resultados obtidos dos testes realizados, o modelo apresenta seu funcionamento correto dentro do planejado, sendo capaz de gerar uma resposta positiva de aprendizagem para a base de dados em que o processo é executado, além de implementar toda a sequência lógica de cálculos, visualizações gráficas, aplicações de filtros, simulações com lucros muito positivos e a possibilidade de utilização para a previsão de qualquer

ativo buscando uma classificação futura. Foi realizada uma implementação funcional em linguagem *python* capaz de auxiliar investidores novatos com um *feedback* de saída simplificado de três respostas (compra, venda e aguardo) para os dados recebidos através de uma integração que utilizou lógicas e parâmetros obtidos de indicadores gráficos, validações cruzadas para gerar testes de acurácia, buscas automáticas de dados históricos da criptomoeda *bitcoin*, previsões através de um algoritmo de *machine learning*, otimização automatizada de parâmetros e a utilização de uma lógica de compra e venda baseada na métrica HLC₃.

REFERÊNCIAS

BASHIR I. **Mastering Blockchain: Deeper insights into decentralization, cryptography, Bitcoin, and popular Blockchain frameworks**. 1ª ed. Packt Publishing Ltd. 2017; 869 p.

LEMONS FACCA. **Análise técnica dos mercados financeiros: Um Guia Completo e Definitivo dos Métodos de Negociação de Ativos**. 2ª ed. Saraiva Uni. 2018; 528 p.

ELDER A. **Come Into My Trading Room: A Complete Guide to Trading**. 1ª ed. Elsevier. 2002; 302 p.

GRAHAM B. **The Intelligent Investor**. 6ª ed. HarperCollins. 2003; 672 p.

KOVACS L. 2021. **O que é a mineração de criptomoedas? [Investimentos]**. Technoblog. Disponível em: <https://tecnoblog.net/443449/o-que-e-a-mineracao-de-criptomoedas-investimentos/> Acesso em: 23 ago. 2021.

MARTINS C. **Os Supersinais da Análise Técnica: Guia para investimentos lucrativos na bolsa**. 1ª ed. Elsevier. 2010; 168 p.

MCNALLY S, et al. **Predicting the Price of Bitcoin Using Machine Learning**. 2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP), 2018; 26: 339-343.

MITCHELL TM. **Machine Learning**. 1ª ed. McGraw-Hill. 1997; 432 p.

RATHAN K, et al. **Crypto-Currency price prediction using Decision Tree and Regression techniques**. 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), 2019; 3: 190-194.

RUSSEL S, NORVIG P. **Inteligência Artificial**. 3ª ed. GEN LTC. 2013; 1016 p.

VELANKAR S. **Bitcoin price prediction using machine learning**. 2018 20th International Conference on Advanced Communication Technology (ICACT), 2018; 20: 144-147.