XXXI Congresso de Iniciação Científica Unicamp 2023

Utilização de Séries Temporais para Predição do Preço de Fechamento de Criptomoedas

Palavras-Chave: criptomoedas, aprendizado de máquina, redes neurais artificiais, séries temporais

Autores:

Renan de Oliveira Ferretti, FT - UNICAMP

Prof. Dr. Guilherme Palermo Coelho, FT - UNICAMP

Prof. Dr. Arthur Emanuel de Oliveira Carosia, IFSP - *campus* São João da Boa Vista

1 Introdução

O termo criptomoeda se refere a uma moeda virtual que utiliza criptografia para realizar as suas transações. Tais moedas são organizadas por uma rede ponto-a-ponto chamada *blockchain*, o que garante segurança quando são usadas. Criptomoedas vêm sendo amplamente reconhecidas como um novo método alternativo de troca de capital, o que tem implicações consideráveis para economias emergentes e, em geral, para a economia global [1].

As criptomoedas conquistaram popularidade devido à sua natureza descentralizada, livre da ação de órgãos regulamentadores e baixos custos de transação. Tal popularidade levou a um aumento no volume de negociações e volatilidade do preço deste tipo de ativo [2]. O mercado de criptomoedas possui algumas diferenças em relação aos mercados tradicionais, como maior volatilidade e menor volume [2]. Ressalta-se que um alto nível de volatilidade pode ser usado a favor do investidor para planejar estratégias de negociação a fim de otimizar seus ganhos [3].

Tanto como um mercado emergente quanto como uma linha de pesquisa, as criptomoedas e o comércio de criptomoedas têm tido um aumento notável de interesse [4]. Atualmente, alguns pesquisadores se dedicam a analisar a eficiência do mercado e a volatilidade dos preços de criptomoedas [5].

A diversificação de investimentos é recomendada na gestão de portfólio, o que tem levado investidores a recorrer tanto a ativos mais tradicionais, como *commodities* e imóveis, quanto a alternativas mais recentes, como *Non-fungible tokens* (NFTs) e criptomoedas. Normalmente, os investimentos alternativos têm uma correlação histórica mais baixa com os ativos convencionais, como ações e títulos, o que proporciona uma boa diversificação da carteira. Assim, a criptomoeda pode ser uma boa forma de investimento alternativo [6], chamando atenção da literatura para entender a sua dinâmica, o que é de interesse de investidores [7].

O aprendizado de máquina é uma abordagem eficiente para desenvolver estratégias de compra e venda de criptomoeda [8], porque ele permite a inferência de relacionamentos de dados que muitas vezes não são diretamente observáveis por humanos. Para criar um modelo de aprendizado de máquina, precisamos passar pelos seus estágios de construção, sendo eles a coleta de dados, o pré-processamento dos dados, a análise exploratória dos dados, o ajuste de hiper-parâmetros, a classificação e a avaliação. Em cada etapa podemos utilizar técnicas de aprendizado de máquina, que são caracterizadas por investigar como as máquinas podem adquirir conhecimento através da extração de padrões a partir de um conjunto de dados, buscando o desenvolvimento de algoritmos que permitam que computadores possam se tornar capazes de tomar decisões com certa autonomia [9].

O presente trabalho de iniciação científica buscou realizar a predição da tendência de preço de criptomoedas no próximo dia a partir de dados históricos de suas cotações, i.e. dizer ao usuário se o valor irá subir ou descer no dia seguinte ao momento atual, o que caracteriza um problema de *classificação de dados*. Foram considerados preditores baseados em aprendizado de máquina e na estatística clássica para tal efeito.

Inicialmente, foram estudadas algumas das técnicas mais utilizadas em cada estágio da construção de modelos de predição baseados em aprendizado de máquina: no pré-processamento, a normalização dos dados e o janelamento dos dados; na classificação, as máquinas de vetores-suporte (SVMs, do inglês *support-vector machines*), as redes neurais do tipo perceptron multicamadas (MLP, do inglês *multi-layer perceptron*) e memória de longo-curto prazo (LSTM, do inglês *long short-term memory*), e o modelo autoregressivo integrado de médias móveis (ARIMA, do inglês *autoregressive integrated moving average*); no ajuste de hiperparâmetros, a janela deslizante (do inglês *increasing window cross validation*) e a busca em grade (do inglês *grid search*).

Dessa forma, foi possível fazer comparações entre os quatro algoritmos de classificação implementados e chegar a observações importantes sobre as possíveis vantagens e desvantagens de cada um no contexto de previsão da tendência de preço de criptomoedas. O presente resumo está organizado da seguinte maneira: a Seção 2 apresenta a metodologia empregada para a realização da pesquisa. Já os resultados obtidos e suas discussões foram descritos na Seção 3. Por fim, a Seção 4

traz as conclusões extraídas ao longo de todo o desenvolvimento do trabalho.

2 Metodologia

Para a realização das atividades foi utilizado o banco de dados público do *Yahoo Finance*[1], que contém dados diários de todas as criptomoedas do mercado, desde o seu lançamento até os dias de hoje. A base de dados em questão possui dados do preço de abertura, preço de fechamento, maior preço alcançado no dia, menor preço alcançado no dia e volume de transações da criptomoeda no dia.

No presente trabalho, foram selecionadas cinco criptomoedas para realizar os experimentos, sendo que o critério de decisão utilizado para realizar esta escolha foi o valor de mercado. Valor de mercado é o termo usado para se referir ao preço que o mercado está pagando por uma empresa, no caso a detentora da criptomoeda. Ele é calculado multiplicando-se o número de criptomoedas em circulação pelo preço atual de cada uma. O próprio site do Yahoo Finance fez essa ordenação e as criptomoedas escolhidas foram Bitcoin (BTC), Ethereum (ETH), Tether (USDT), BNB e USD Coin (USDC). Após a obtenção da base de dados a ser utilizada no projeto, foi necessário definir uma metodologia para os experimentos, conforme a Figura 1. Todo esse processo é descrito mais detalhadamente nesta seção.

2.1 Pré-processamento de dados

Nesta etapa aplicamos um conjunto de técnicas para converter os dados brutos em dados preparados, ou seja, dados em formatos úteis para a nossa aplicação. Inicialmente, foi constatado que a BNB, BTC e ETH não sofreram muitas variações em seu preço até determinado período. Portanto, consideramos as amostras coletadas a partir desse período de mudança, visto que, antes disso, a alteração no preço não era significativa e a sua rápida ascensão acabou comprometendo o funcionamento do modelo. A Figura 2 demonstra a separação da base de dados da BTC em dois períodos, em azul temos o intervalo que foi considerado para os experimentos. Nessa etapa, separamos a BNB e a ETH entre antes e depois de 2020, a BTC entre antes e depois de 2018 e o resto das criptomoedas não foram separadas em dois períodos pois não apresentaram tais variações.

Após isso, foi criada a variável de saída (a ser predita) com base na comparação do preço de fechamento da criptomoeda no dia atual e no dia anterior. Para isso, definimos a variável de saída como sendo um (1), representando uma subida no preço da criptomoeda, e zero (0), representando uma descida ou uma não-variação no preço, sendo o último extremamente raro.

Posteriormente, aplicamos um janelamento de quatro dias pela nossa base, ou seja, agrupamos todos os dados de três dias atrás com os dados do dia atual a fim de fornecer mais dados de entrada para o modelo. Dessa maneira, espera-se que ele seja mais eficaz ao prever a

Figura 1: Diagrama da metodologia empregada.

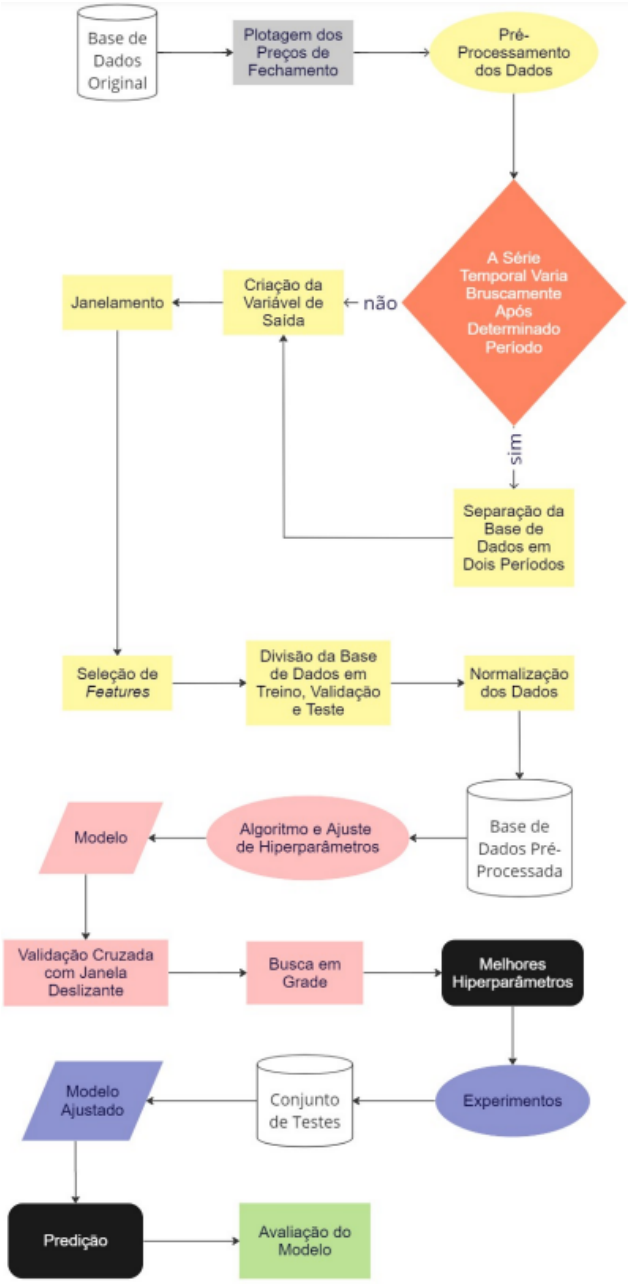
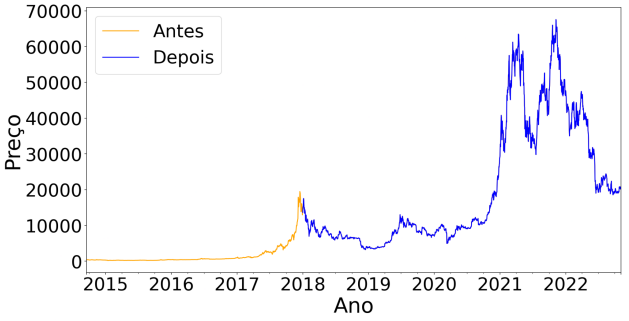


Figura 2: Divisão da base de dados em dois períodos.



tendência do preço de cada dia, visto que ele terá mais dados para tal efeito. É importante mencionar, que os dados, devidamente janelados, foram divididos em dois conjuntos: 80% para treinamento e validação, e 20% para

[1] https://finance.yahoo.com/crypto/

teste. Esta divisão foi feita uma única vez e todos os modelos consideraram os mesmos conjuntos de treinamento/validação e teste. Por fim, utilizamos os valores máximos e mínimos do conjunto de treinamento e validação para realizar a normalização min-max em cada atributo da base de dados, com o objetivo de evitar que o algoritmo fique enviesado para as variáveis com maior ordem de grandeza. A Tabela 1 mostra todas as *features* que foram empregadas como entrada dos modelos baseados em aprendizado de máquina.

Tabela 1: *features* que serviram de entrada para a SVM, MLP e LSTM

Feature
Preço de abertura
Alta
Baixa
Preço de fechamento

2.2 Algoritmos e ajuste de hiperparâmetros

Foram implementadas para a etapa de experimentos deste projeto o algoritmo de aprendizado de máquina *support-vector machine* (SVM), a rede neural artificial do tipo *multi-layer perceptron* (MLP), a rede neural artificial do tipo *long short-term memory* (LSTM) e o modelo autoregressivo integrado de médias móveis (ARIMA). Eles foram escolhidos devido ao seu amplo uso na literatura para a predição de séries temporais [10][11][12]. Em especial, o ARIMA foi utilizado pois aborda o problema com outra perspectiva, sem utilizar aprendizado de máquina. Ele é popular na predição de séries temporais pelo fato de observar diferentes estruturas temporais nos dados, como a tendência e a sazonalidade. Para realizar os experimentos foram desenvolvidos quatro *scripts* que utilizam módulos com uma implementação pronta de cada algoritmo, utilizando a linguagem de programação Python. Portanto, os scripts desenvolvidos tiveram como objetivo criar o modelo a partir dos parâmetros que serão passados a ele.

Para a obtenção dos hiperparâmetros dos diferentes modelos de classificação baseados em aprendizado de máquina, realizamos uma busca em grade (*grid search*) baseada em validação cruzada com janela deslizante de 5 pastas (5-*folds*). Essa espécie de validação cruzada consiste em deixar um tamanho dinâmico para o treinamento e um fixo para a validação, desse modo a janela de treinamento cresce a cada iteração, conforme a Figura 3. A busca em grade levou em conta, para as redes neurais, a quantidade de camadas ocultas, a quantidade de neurônios em cada camada e a taxa de aprendizado do algoritmo de treinamento, conforme a Tabela 2. Já para o SVM foi considerado o tipo de kernel a ser usado e seu coeficiente, e um limite superior na fração de erros de margem e um limite inferior na fração de vetores de suporte (nu), ambos representados por um mesmo valor, conforme a Tabela 3. Ao final, a combinação de valores dos hiperparâmetros que levou cada modelo ao melhor desempenho em termos da média da métrica F1-*score* no conjunto de validação foi a escolhida.

Figura 3: Validação cruzada com janela deslizante de 3 pastas [13]

Para o modelo ARIMA, não foi utilizada a busca em grade no seu ajuste de hiperparâmetros, ao invés disso seguimos as seguintes etapas. Primeiramente, realizamos o teste de *Dickey-Fuller* a fim de verificar a estacionaridade da série temporal dos preços de fechamento de cada criptomoeda. As séries da BNB, BTC e ETH foram identificadas como não estacionárias, portanto foi aplicada uma diferenciação nos dados delas para adaptação ao modelo, o que não foi o caso da USDC e USDT. Após isso, em sua construção, definimos os parâmetros como ARIMA (3,0,1) ou ARIMA (3,1,1), a depender da criptomoeda. Sendo o primeiro parâmetro indicador de quantos dias anteriores ao atual ele deve considerar, o segundo a quantidade de diferenciações realizadas na série temporal e o terceiro a ordem da média móvel.

Tabela 2: valores de hiperparâmetros da MLP e LSTM testados na busca em grade

hiperparâmetros	**valores**			
quantidade de camadas ocultas	1	2	3	4
neurônios em cada camada	10	30	50	70
taxa de aprendizado	0,001	0,005	0,01	

3 Resultados e Discussões

Nessa seção serão apresentados os resultados obtidos através da implementação da metodologia apresentada anteriormente. Um repositório com o código desenvolvido, bem como os dados pré-processados utilizados, está disponível no Github[2].

Na Tabela 4 podemos encontrar os resultados da etapa de ajuste de hiperparâmetros para as redes neurais MLP e LSTM, com o número de camadas ocultas, número de neurônios por camada e taxa de aprendizado

[2] https://github.com/Renanferretti

selecionados. Já na Tabela 5, temos a seleção de hiper-parâmetros da SVM, com o tipo de kernel e seu coeficiente, e o valor do limite superior na fração de erros de margem e do limite inferior na fração de vetores de suporte (nu).

Tabela 3: valores de hiperparâmetros da SVM testados na busca em grade

hiperparâmetros	**valores**			
kernel	linear	poly	rbf	sigmoid
coeficiente do kernel	scale		auto	
nu	0,2	0,4	0,6	0,8

A Tabela 6 apresenta os desempenhos obtidos por cada modelo de aprendizado de máquina para o conjunto de testes em termos de F1-score e acurácia, para cada uma das criptomoedas selecionadas para este projeto.

Os melhores hiperparâmetros encontrados para as redes neurais se alteram para cada criptomoeda, exceto a quantidade de camadas ocultas do modelo. Uma RNA com duas camadas ocultas é a arquitetura mais vezes encontrada dentre os hiperparâmetros com melhor desempenho no conjunto de validação, sendo vista em 70% dos casos. Já o valor do limite superior na fração de erros de margem e do limite inferior na fração de vetores de suporte (nu), após se ajustar os hiperparâmetros da SVM, está presente em 100% dos casos.

Tabela 4: Melhores hiper-parâmetros da MLP e LSTM.

Criptomoeda	**Modelo**	**Camadas ocultas**	**Neurônios por camada**	**Taxa de aprendizado**
BNB	MLP	2	70	0,001
	LSTM	2	30	0,001
BTC	MLP	4	70	0,01
	LSTM	1	50	0,005
ETH	MLP	2	70	0,005
	LSTM	3	70	0,001
USDC	MLP	2	50	0,001
	LSTM	2	10	0,001
USDT	MLP	2	70	0,005
	LSTM	2	10	0,01

Notamos que a USDC e a USDT levaram aos mesmos valores de hiperparâmetros. Uma hipótese para isso ter ocorrido é que as séries temporais dessas criptomoedas são muito similares, portanto os hiperparâmetros também são parecidos. Outro ponto a se ressaltar é que os valores definidos para o kernel não são muito diversos entre si, dado que foram fornecidos quatro tipos diferentes e apenas dois conseguiram as melhores métricas no conjunto de validação.

Tabela 5: Melhores hiper-parâmetros da SVM.

Criptomoeda	**Kernel**	**Coeficiente do kernel**	**nu**
BNB	linear	auto	0,2
BTC	poly	scale	0,2
ETH	linear	auto	0,2
USDC	poly	scale	0,2
USDT	poly	scale	0,2

Na Tabela 6 foram destacadas as melhores medidas obtidas para cada criptomoeda. De maneira geral, nos experimentos realizados tivemos comportamentos diferentes por parte de cada algoritmo. A SVM e a LSTM apresentaram resultados satisfatórios para todas as cinco criptomoedas, com destaque para a BNB, BTC e ETH. A arquitetura usada pela SVM para essas três criptomoedas foi kernel linear, coeficiente do kernel automático e 0,2 de nu (BNB); kernel polinomial, coeficiente do kernel escalar e 0,2 de nu (BTC); e kernel linear, coeficiente do kernel automático e 0,2 de nu (ETH). Já a arquitetura empregada pela LSTM nelas foi de 2 camadas ocultas, 30 neurônios por camada e 0,001 como taxa de aprendizado (BNB); 1 camada oculta, 50 neurônios por camada e 0,005 como taxa de aprendizado (BTC); e 3 camadas ocultas, 70 neurônios por camada e 0,001 como taxa de aprendizado (ETH). Em relação à rede neural do tipo MLP, ela teve um ótimo desempenho na predição do preço de fechamento da BNB e ETH, porém não conseguiu ter o mesmo rendimento para as demais criptomoedas. Já o ARIMA atingiu resultados inferiores aos modelos baseados em aprendizado de máquina para todas criptomoedas. A SVM obteve o melhor desempenho entre todas as predições ao realizar a previsão para a BNB, com F1-score de 98,94% e acurácia de 98,89%.

Dessa forma, considerando a complexidade do modelo e seu desempenho, verificamos que os modelos mais simples são competitivos com opções inspiradas em redes neurais artificiais, podendo até superar determinadas abordagens.

4 Conclusões

Este trabalho de iniciação científica visou utilizar séries temporais de criptomoedas a fim de realizar um

estudo comparativo entre diferentes algoritmos de aprendizado de máquina e da estatística clássica. O projeto mostrou, através de diferentes modelos de predição, que o emprego de técnicas baseadas em aprendizado de máquina menos complexas podem levar a desempenhos tão bons, se não melhores, que modelos baseados em redes neurais artificiais.

Dentre as opções testadas, a SVM foi a que atingiu o melhor desempenho médio entre as cinco criptomoedas utilizadas; a LSTM também alcançou resultados bastante satisfatórios, mas um pouco inferiores aos da SVM. Em relação às criptomoedas, a BNB foi a que obteve os melhores resultados no conjunto de testes ao se considerar todos os classificadores.

Tabela 6: Desempenho dos modelos para cada criptomoeda.

Criptomoeda	Modelo	F1-*score*	Acurácia
BNB	SVM	**98,94%**	**98,89%**
	MLP	95,74%	95,58%
	LSTM	95,18%	95,03%
	ARIMA	49,85%	48,46%
BTC	SVM	**97,04%**	**97,16%**
	MLP	64,75%	47,88%
	LSTM	93,06%	92,27%
	ARIMA	45,55%	47,50%
ETH	SVM	**94,91%**	**95,02%**
	MLP	85,32%	82,32%
	LSTM	92,48%	92,82%
	ARIMA	48,18%	48,97%
USDC	SVM	82,91%	**84,34%**
	MLP	72,72%	62,61%
	LSTM	**84,96%**	83,48%
	ARIMA	56,01%	61,19%
USDT	SVM	85,47%	83,45%
	MLP	71,40%	59,85%
	LSTM	**87,50%**	**85,89%**
	ARIMA	58,37%	59,61%

Como trabalhos futuros, recomenda-se buscar uma melhora na performance do modelo ARIMA por meio de uma melhor configuração de seus hiperparâmetros, a união dos quatro modelos em um conjunto de preditores (*ensemble*) e que, a partir disso, seja feita uma simulação de investimento para cada uma das criptomoedas empregadas neste trabalho a fim de verificar se haveria lucro ao final e, se houver, de quanto seria.

Referências Bibliográficas

[1] NASIR, M.A.; HUYNH, T.L.D.; NGUYEN, S.P.; DUONG, D. Forecasting cryptocurrency returns and volume using search engines. **Financ. Innov**. 2019, 5, 2.

[2] ALONSO-MONSALVE, Saúl; SUÁREZ-CETRULO, Andrés L.; CERVANTES, Alejandro; QUINTANA, David. Convolution on neural networks for high-frequency trend prediction of cryptocurrency exchange rates using technical indicators. **Expert Systems With Applications**, v. 149, jul. 2020.

[3] BOURI, Elie; LAU, Chi Keung Marco; LUCEY, Brian; ROUBAUD, David. Trading volume and the predictability of return and volatility in the cryptocurrency market. **Finance Research Letters**, v. 29, p. 340-346, jun. 2019.

[4] FARELL, Ryan, "An Analysis of the Cryptocurrency Industry" (2015). **Wharton Research Scholars**. 130.

[5] BALCILAR, Mehmet; BOURI, Elie; GUPTA, Rangan; ROUBAUD, David. Can volume predict Bitcoin returns and volatility? A quantiles-based approach. **Economic Modelling**, [S.L.], v. 64,, p. 74-81, ago. 2017.

[6] TRIMBORN, Simon; LI, Mingyang; HÄRDLE, Wolfgang Karl. Investing with Cryptocurrencies: a liquidity constrained investment approach. **Journal Of Financial Econometrics**, v. 18, n. 2, p. 280-306, 3 jun. 2019.

[7] NASIR, Muhammad Ali et al. Forecasting cryptocurrency returns and volume using search engines. **Financial Innovation**, v. 5, n. 1, p. 1-13, 2019.

[8] MCNALLY, Sean; ROCHE, Jason; CATON, Simon. Predicting the Price of Bitcoin Using Machine Learning. **2018 26Th Euromicro International Conference On Parallel, Distributed And Network-Based Processing (Pdp)**, mar. 2018.

[9] RUSSEL, Stuart; NORVIG, Peter. **Artificial Intelligence**. 3. ed., Elsevier, 1995. 1132 p.

[10] REBANE, Jonathan et al. Seq2Seq RNNs and ARIMA models for cryptocurrency prediction: A comparative study. In: **SIGKDD Fintech'18, London, UK, August 19-23, 2018**. 2018.

[11] JAY, Patel et al. Stochastic neural networks for cryptocurrency price prediction. **Ieee access**, v. 8, p. 82804-82818, 2020.

[12] POONGODI, M. et al. Prediction of the price of Ethereum blockchain cryptocurrency in an industrial finance system. **Computers & Electrical Engineering**, v. 81, p. 106527, 2020.

[13] DE OLIVEIRA CAROSIA, Arthur Emanuel. **Previsão do mercado de ações brasileiro com o uso de análise de sentimentos, indicadores técnicos e valores de ações**. 2022. Tese de Doutorado. [sn].