

Aplicação de Algoritmos de *Machine Learning* na previsão de cotação do *Bitcoin*

Rodrigo de Souza Oliveira¹, Álvaro Viebrantz¹

¹ MBA em Big Data – Fatec – Serviço Nacional de Aprendizagem Industrial (SENAI)
Av. XV de Novembro, 303, Porto - CEP: 78020-300 - Cuiabá-MT

rddsouzaoliveira@gmail.com, alvarowolfx@gmail.com

Abstract. *With the evolution of technology and the growth of the Data Science field, a subject that has become very relevant is the application of advanced Machine Learning techniques to understand assets behavior and support the decision-making process on the financial market. The objective of this work is to apply Machine Learning models, to analyze and understand bitcoin's behavior to forecast its future value and use it as a tool to assist in investment strategies. To achieve that goal, a dataset of Bitcoin quotation data was created by collecting historical data using a crypto exchange API, containing data starting in December 2017. The models developed in this article were able to understand and forecast Bitcoin price fluctuations, with a highlight to the Multilayer Perceptron model. This work paved the way for a range of possibilities for future work to better understand model architectures available that can generalize the behavior of Bitcoin over time and to the development of other tools that can automate decision-making operation on the Bitcoin market.*

Resumo. *Com a evolução da tecnologia e o crescimento da área de Ciência de Dados, um assunto que se tornou muito relevante é a aplicação de técnicas avançadas de Aprendizado de Máquina para entender o comportamento dos ativos e apoiar o processo de tomada de decisão no mercado financeiro. O objetivo deste trabalho é aplicar modelos de Aprendizado de Máquina, analisar e compreender o comportamento do Bitcoin para prever seu valor futuro e usá-lo como uma ferramenta para auxiliar nas estratégias de investimento. Para atingir esse objetivo, um conjunto de dados de cotação de Bitcoin foi criado por meio da coleta de dados históricos usando uma API de uma corretora, contendo dados a partir de dezembro de 2017. Os modelos desenvolvidos neste artigo foram capazes de entender e prever as oscilações dos preços do Bitcoin, com destaque para o modelo Multilayer Perceptron. Este trabalho abre caminho para uma gama de possibilidades de trabalhos futuros para melhor compreender as arquiteturas de modelos disponíveis que podem generalizar o comportamento do Bitcoin ao longo do tempo e para o desenvolvimento de outras ferramentas que podem automatizar a operação de tomada de decisão no mercado de Bitcoin.*

1. Introdução

Investimento é a aplicação de algum recurso, com a expectativa de algum ganho futuro. A incerteza do ganho, ou um possível prejuízo, caracteriza o risco do investimento. No

contexto de aplicação financeira, pode-se caracterizar o investimento como a aplicação de dinheiro que não gere custos, tenha expectativa de lucro futuro e que não exija esforços relevantes¹.

Diversos estudos, já consolidados, trazem que a diversificação do portfólio de investimentos se demonstra como uma eficiente forma para redução dos riscos, conforme trazido em [Oda et al. 1998]. A diversificação de investimentos é uma técnica que visa a diluição dos riscos e a maximização dos ganhos [Pactual 2017].

Aliado à sua popularização nos últimos anos, o Bitcoin se tornou uma alternativa para os investidores comporem suas carteiras de investimentos e despertou-se a necessidade de melhor entender seu comportamento. Porém, a oscilação e a incerteza do futuro do Bitcoin, gera muita insegurança, principalmente entre os investidores mais conservadores [Oliveira 2020].

Como a tecnologia e as novas ferramentas de análise de dados, podem contribuir para transformar o bitcoin como uma alternativa de investimento, minimizando-se os riscos das operações?

A evolução da tecnologia proporcionou o surgimento de novas ferramentas de análise de dados. Estas ferramentas podem contribuir para transformar o bitcoin como uma alternativa de investimento, pois, através do entendimento de seu comportamento, possibilitam a minimização dos riscos das operações com a moeda.

O Objetivo deste trabalho é a implementação e teste de modelos capazes de analisar dados históricos das cotações do Bitcoin, para identificar o comportamento do preço do Bitcoin e projetar as cotações futuras.

Os modelos desenvolvidos tiveram como ideia, possibilitar que o investidor tenha uma ferramenta técnica para auxiliar no processo de tomada de decisões sobre as estratégias que investimentos que serão adotadas.

Justifica-se o desenvolvimento deste trabalho, pela importância que o Bitcoin conquistou no mercado financeiro nos últimos anos, pelo desafio na aplicação de técnicas extremamente avançadas para análise de dados com a finalidade de prever o valor futuro do ativo, que é um dos grandes desafios dos cientistas de dados.

A estrutura deste trabalho está organizada da seguinte forma: A Metodologia (Capítulo 2), onde são abordados os principais passos para coleta de dados, análise exploratória e o desenvolvimento do modelo, A Revisão de Literatura (Capítulo 3), onde são apresentados os conceitos acerca do Bitcoin, Machine Learning e sobre as Redes Neurais Artificiais, A Apresentação da Pesquisa (Capítulo 4) e Discussão de Resultados (Capítulo 5), onde é abordado os resultados obtidos em cada etapa do desenvolvimento do modelo e o resultado da previsão realizada e por último, as Considerações Finais (Capítulo 6), onde se conclui sobre as aplicações do modelo e recomendações de futuros estudos.

2. Metodologia

A finalidade dessa pesquisa tem natureza aplicada e abordagem qualitativa e quantitativa para realizar estudo de caráter descritivo.

¹Detalhes em Blog Rico. O Que é Investimento e Por Que Poupança é Ruim. Recuperado em 3 de outubro de 2020, de <https://blog.rico.com.br/o-que-e-investimento>

Foi realizado um levantamento bibliográfico sobre os temas abordados neste artigo, através da coleta de informações em livros, revistas, sites e outros artigos.

Para a coleta e análise dos dados utilizou-se um Macbook Pro 2018 com memória RAM de 16GB, processador Intel i7 de 2.2Ghz, vídeo Radeon Pro 555x de 4Gb e um Flash Storage de 251Gb, além de sistema operacional MacOS Catalina, com acesso à internet.

Utilizou-se como linguagem de programação o R [R Core Team 2020] para todas as etapas de análises apresentadas neste artigo e o Latex [Goossens et al. 1993] para escrita. Como banco de dados, utilizou-se o Microsoft SQL Server, hospedado na nuvem Azure.

2.1. Coleta dos dados

A coleta dos dados se iniciou em dezembro de 2017, através da API REST da corretora de criptomoedas Bitfinex. Estruturou-se uma rotina em linguagem R, dentro de uma máquina virtual no ambiente Google Cloud Platform (GCP), que acessava a API e armazenava os dados em um banco de dados SQL Server, num período de 10 em 10 segundos.

A rotina, consultava dois grupos de informações, fornecidos pela API: *"Ticker"* e *"Orderbook"*. O primeiro grupo, trazia informações sobre a cotação no momento da consulta e trazia os seguintes campos:

- Valor atual de venda
- Valor atual de compra
- Último valor negociado
- Menor valor negociado das últimas 24h
- Maior valor negociado nas últimas 24h
- Volume total negociado nas últimas 24h

O segundo grupo, trazia o *"livro de ordens"*, que era a listagem de todas as ordens de compra e venda na corretora, vigentes no momento da consulta. Essa consulta retornava milhares de linhas e para reduzir o espaço de armazenamento, optou-se por resumir as informações antes de salvar no banco de dados. Os dados resumidos traziam as seguintes informações tanto das ordens de compra, quanto das ordens de venda:

- Mínimo
- 1o Quartil
- Mediana
- 3o Quartil
- Máximo

Ao final, o conjunto de dados ficou com as variáveis apresentadas na Tabela 1. A variável *Last* será a variável resposta dos modelos, ou seja, a variável que iremos tentar estimar através dos modelos a serem desenvolvidos.

2.2. Análise exploratória

Com os dados já estruturados e armazenados num banco de dados, o próximo passo foi realizar uma análise exploratória no conjunto de dados.

Como o tempo de resposta da API pode variar, analisou-se os intervalos entre cada consulta efetivamente realizada. Após essa análise, foram realizadas algumas tratativas nos dados para se chegar ao *dataset* final.

Tabela 1. Variáveis coletadas através da API

Variável	Descrição
AIL	Límite inferior dos preços de venda do livro de ofertas
AQ1	1º Quartil dos preços de venda do livro de ofertas
AMD	Mediana Quartil dos preços de venda do livro de ofertas
AQ3	3º Quartil dos preços de venda do livro de ofertas
ASL	Límite superior dos preços de venda do livro de ofertas
BIL	Límite inferior dos preços de compra do livro de ofertas
BQ1	1º Quartil dos preços de compra do livro de ofertas
BMD	Mediana Quartil dos preços de compra do livro de ofertas
BQ3	3º Quartil dos preços de compra do livro de ofertas
BSL	Límite superior dos preços de compra do livro de ofertas
BAMOUNT	Volume total das ofertas de compra
AAMOUNT	Volume total das ofertas de venda
Bid	Valor da última oferta de compra
Ask	Valor da última oferta de venda
Last	Valor da última negociação realizada
Low	Menor negociação realizada no timeframe
High	Maior negociação realizada no timeframe
Volume	Volume total das negociações realizadas
Timestamp	Momento da última negociação
datetime	Data e hora da última negociação

Fonte: Dados do estudo

2.3. Modelos

Neste trabalho foram utilizados os modelos de Regressão Linear, Random Forest e *Multi Layer Perceptron* (MLP), considerando como base de treinamento os dados compreendidos entre novembro de 2017 até dezembro de 2019 e para validação os dados entre janeiro e março de 2020.

O modelo MLP formado treinado utilizando o pacote Keras [Chollet 2015], que cria uma camada para execução do ambiente Tensorflow [Abadi et al. 2015].

Os modelos de Regressão Linear e Random Forest, foram treinados através do framework *Tidymodels* [Kuhn and Wickham 2020b], utilizando como motor de treinamento os pacotes *glmnet* [Friedman et al. 2010] e *randomForest* [Liaw and Wiener 2002]

3. Revisão de Literatura

Neste tópico serão apresentados os principais conceitos abordados no estudo, contextualizando o Bitcoin, os tipos de aprendizados de máquina e os principais pontos que envolvem as redes neurais artificiais.

3.1. Bitcoin

O Bitcoin consiste em uma rede estruturada ponto-a-ponto, constituída por diversos computadores que assumem o mesmo papel de validadores das transações que são realizadas entre eles [Nakamoto 2019].

Todas as transações que são realizadas pela rede do Bitcoin, são criptografadas, dificultando sua rastreabilidade.

De acordo com [Shawn 2017], o Bitcoin foi a primeira moeda digital a resolver o problema do gasto duplo, utilizando uma solução descentralizada para armazenamento dos registros das transações.

Outra característica da moeda é que não existe nenhum governo ou instituição que regulamenta suas transações e seu preço. Esses fatores ajudaram a contribuir para a popularização do Bitcoin nos últimos anos e consequentemente a sua valorização, saindo de centavos de dólares em 2009 para cerca de 7 mil em 2018, conforme imagem abaixo.



Figura 1. Preço Histórico do Bitcoin

Fonte: <https://www.coindesk.com/price/bitcoin>

Atualmente, estima-se mais de 375 milhões de dólares sejam comercializados em bitcoin em apenas um dia.

3.2. Machine Learning

Segundo [Kelleher et al. 2020], Machine Learning pode ser definido como a automação de processos que extraem padrões dos dados. Compreende um conjunto de técnicas computacionais que têm a característica de "aprender" com os dados.

De acordo com [Tanaka 2018], existem três tipos de algoritmos de *Machine Learning*:

- **Supervisionado:** Quando se diz ao algoritmo o que é cada entrada (rótulo) e ele aprende quais são as características que influenciam a entrada a ser o que ela é.

- **Não supervisionado:** Quando não se diz ao algoritmo o que é cada entrada, ou seja, os dados não são rotulados. O algoritmo classifica as entradas conforme suas características semelhantes.
- **Por reforço:** Define-se um sistema de recompensas e punições aos possíveis resultados para que o algoritmo possa ponderar as escolhas a serem feitas.

3.3. Redes Neurais Artificiais

Redes Neurais Artificiais (RNAs). São modelos matemáticos, compostos por unidades de processamentos simples, que calculam determinadas funções matemáticas. Uma rede neural artificial é um modelo de computação inspirado na forma como a estrutura do cérebro dos mamíferos processa informação [De Souto et al. 2003].

As unidades de processamento, também chamadas de nós ou neurônios, são dispostas em uma ou várias camadas e interligadas por conexões. As conexões estão ligadas a pesos, que ponderam os valores recebidos por cada neurônio.

O conhecimento sobre o problema em consideração está guardado dentro dos exemplos que têm que estar obrigatoriamente disponíveis. O algoritmo de aprendizagem generaliza esses dados e memoriza o conhecimento dentro dos parâmetros adaptáveis da rede, que são os pesos [Rauber 2005].

A camada que recebe os dados é chamada de camada de entrada, a camada de saída é responsável por traduzir o resultado da RNA e quaisquer outras são denominadas de camadas ocultas, ou intermediárias, vide Figura 2 abaixo. Em cada uma das camadas é aplicada uma função de ativação. Durante a fase de treinamento, a RNA ”aprende” ajustando-se os pesos [Bishop 1996].

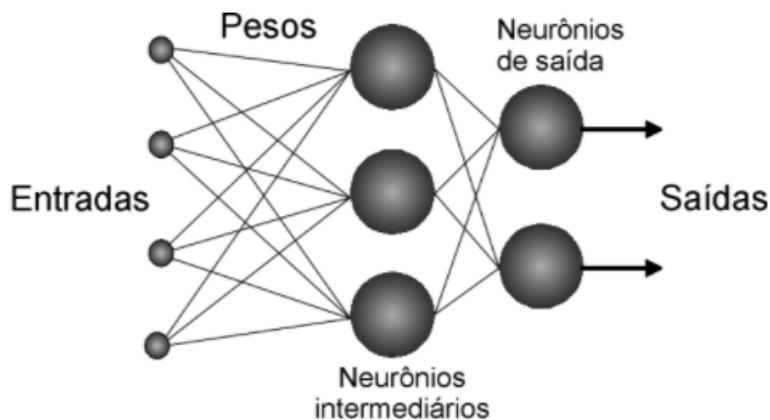


Figura 2. Exemplo de uma Rede Neural Artificial de 2 camadas com 4 entradas e 2 saídas

Fonte: <https://cerebramente.org.br/n05/tecnologia/rna.htm>, consulta em 02 de outubro de 2020

3.3.1. Multilayer Perceptron

A *Multilayer Perceptron* (MLP), ou Perceptron de Múltiplas Camadas em tradução direta, é um tipo de Rede Neural artificial que possui mais de uma camada de neurônios, inter-

ligados em alimentação direta. Uma introdução às Redes Neurais e Perceptrons pode ser vista em [Bailer-Jones et al. 2001], mais detalhes sobre a MLP em [Sarle 1994] e sobre a alimentação direta em [Bishop et al. 1995].

A Figura 3 esquematiza uma MLP com duas camadas ocultas:

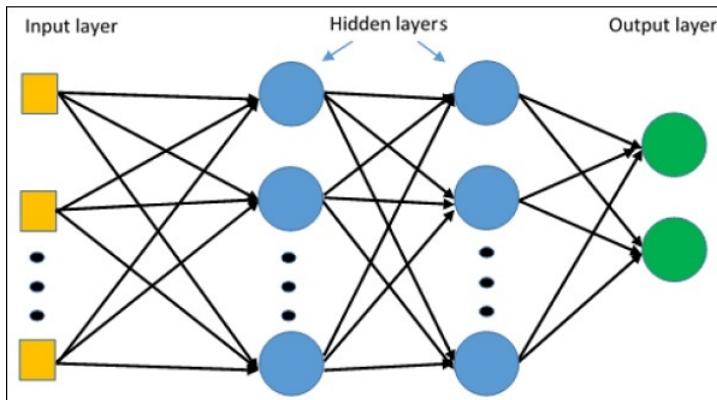


Figura 3. Exemplo de uma MLP com duas camadas ocultas

Fonte: https://www.tutorialspoint.com/tensorflow/tensorflow_multi_layer_perceptron_learning.htm, consulta em 02 de outubro de 2020

3.4. Regressão Linear Múltipla

Modelos de regressão linear, assumem que existem uma relação linear entre uma variável y (também chamada de variável dependente) e p variáveis independentes (ou preditoras) [Rodrigues 2012]. O modelo de regressão linear múltipla é definido conforme abaixo:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \varepsilon_i, i = 1, \dots, n \quad (1)$$

em que

- y_i é o valor da variável dependente na observação i , $i = 1, \dots, n$;
- x_{ip} é o valor da i -ésima observação da p -ésima variável independente;
- β_p são os coeficientes da regressão;
- ε_i é o erro aleatório na i -ésima observação.

Os coeficientes β_j representam a média esperada na variável resposta Y , quando a variável X_j sofre um acréscimo unitário, enquanto todas as outras X_k , $k \neq j$ são mantidas constantes.

O coeficiente β_0 corresponde ao intercepto do plano de regressão. Se o modelo incluir $X_j = 0$, então β_0 será a média de Y nesse ponto. Caso contrário, não existe interpretação prática para β_0 .

3.5. Random Forests

Os Random Forests (ou floresta aleatória em tradução direta) é um algoritmo de aprendizado de máquina, proposto por [Breiman 2001], que consiste na geração de diversas árvores de decisão, baseadas em seleções aleatórias e variáveis. Após, é realizada uma "votação" entre os modelos e a decisão mais votada é a resposta do algoritmo.

É dado pelo modelo:

$$h(\mathbf{x}, \Theta_k), k = 1, \dots, p \quad (2)$$

em que Θ_k são vetores aleatórios *iid* e cada árvore, vota para a classe mais popular da entrada \mathbf{x} .

4. Discussão dos resultados

A proposta deste artigo é a criação de modelos que consiga entender o comportamento do preço do Bitcoin e fazer projeções para auxiliar a tomada de decisões de compra e venda, com o intuito de otimizar os resultados dos *trades*.

O primeiro passo para o desenvolvimento deste modelo foi a coleta de dados, através da API da corretora *Bitfinex*, e que armazenava o preço do Bitcoin, entre outras informações, a cada dez segundos.

Na tabela 2 é possível verificar algumas estatísticas acerca do conjunto de dados.

Tabela 2. Sumarização dos dados

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
AIL	3278	6384	7797	7917	9442	19891
AQ1	3561	6890	8217	8456	9877	20542
AMD	3706	7838	8889	9357	11202	22125
AQ3	3874	8442	10762	11031	12926	25195
ASL	4219	9371	13501	14550	19359	36100
BIL	0	1783	4111	4511	7300	10475
BQ1	1001	4350	6108	6110	7863	13026
BMD	2822	5448	6979	6917	8528	16156
BQ3	3067	5977	7412	7437	9097	18124
BSL	3278	6384	7797	7916	9441	19890
BAMOUNT	789.5	3785.3	9342.7	141543.1	129697.3	835193.2
AAMOUNT	923.5	4961.4	7571.0	7961.8	10788.1	20640.8
Bid	3278	6384	7797	7916	9441	19890
Ask	3278	6384	7797	7918	9442	800000
Last	3278	6384	7797	7916	9442	19891
Low	0	6252	7502	7644	9183	18734
High	0	6519	8049	8169	9725	19891
Volume	0	7102	13983	23613	31641	213851
datetime	2017-11-22 22	2018-06-25 11	2019-02-11 10	2019-01-31 06	2019-09-06 16	2020-09-29 02

Fonte: Dados do estudo

Apresenta-se na Figura 4 abaixo, a variação do preço do Bitcoin, desde o início da coleta dos dados até o momento da elaboração deste estudo. Pode-se perceber que no final do ano de 2018, o preço do Bitcoin atingiu seu pico máximo, chegando a quase 20 mil dólares. Após um longo período de queda, no primeiro semestre do ano de 2019, o preço aumentou consideravelmente, chegando a cerca de 13 mil dólares.

Outro ponto relevante que se observa, é um grande gap entre os meses de abril e outubro de 2020. Esse fato se deu, por problemas na aplicação que consome os dados. Por isso, serão descartados os registros superiores ao mês de abril/2020.

O próximo passo foi fazer uma análise das informações que foram coletadas. Durante o processo de estruturação do programa que iria fazer a coleta dos dados, observou-se que as vezes a API apresentava oscilações de performance, ou seja, demorando muito para retornar o resultado da solicitação.

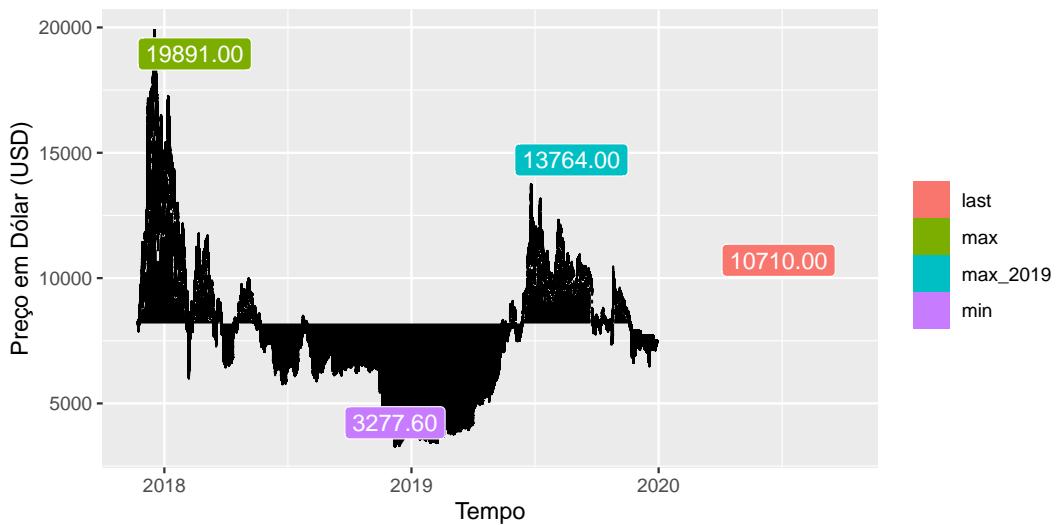


Figura 4. Evolução do preço do Bitcoin, extraído da API da Bitfinex

Fonte: Dados do estudo

Verificou-se então a diferença entre cada período coletado. Verifica-se, Tabela 3 que existem sete registros com diferença superior a um dia.

Tabela 3. Lag entre os registros

	Data Final	Data Inicial	Diferença
1	2020-09-29 02:24:54	2020-04-11 02:26:54	4103:58:00.27338
2	2018-12-29 16:01:59	2018-12-14 10:53:04	365:08:55.01703
3	2020-04-08 20:43:17	2020-04-04 04:09:28	112:33:48.94099
4	2019-01-01 21:07:53	2018-12-29 17:28:01	75:39:51.56921
5	2020-04-01 02:25:02	2020-03-29 21:13:41	53:11:21.09238
6	2020-04-11 02:25:10	2020-04-09 02:46:24	47:38:46.28917
7	2020-03-29 20:34:06	2020-03-28 14:00:28	30:33:37.74005
8	2020-04-03 02:25:00	2020-04-02 03:52:29	22:32:30.92597
9	2020-04-04 02:24:55	2020-04-03 05:42:33	20:42:22.44385
10	2019-01-07 14:15:12	2019-01-07 07:00:10	07:15:01.36821

Fonte: Dados do estudo

Essa primeira análise, subsidiou a decisão de que, apesar dos dados serem coletados a cada dez segundos, o modelo confeccionado analisará o período de um dia, ou seja, o modelo irá projetar, com base nas informações atuais, o preço do bitcoin no próximo dia.

Com isso, o terceiro passo adotado foi filtrar os dados, obedecendo o período de um dia. Com isso, o gráfico do preço histórico ficou Figura 5 abaixo.

Após, prosseguiu-se com a geração das bases de treinamento e teste. Utilizou-se como base de treinamento o período compreendido entre 01/11/2017 e 31/12/2019 e como base de testes o período entre 01/01/2020 a 31/03/2020. Normalizou-se as variáveis para todas ficarem compreendidas no intervalo entre menos um e um.

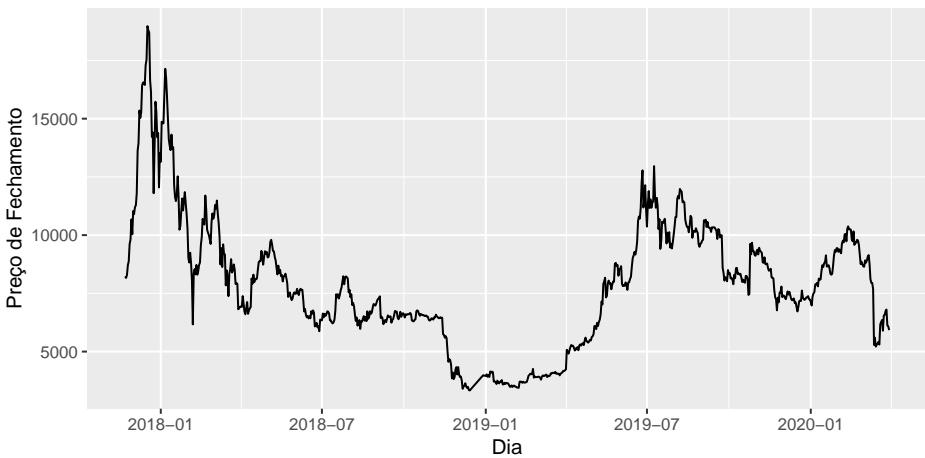


Figura 5. Evolução do preço do Bitcoin, após filtros

Fonte: Dados do estudo

Executou-se os modelos de Regressão (*lr*) e *Random Forests* (*rf*), removendo as variáveis autocorrelacionadas. Neste passo, utilizou-se o método *step_corr* do pacote *recipes* [Kuhn and Wickham 2020a], que tem como finalidade remover variáveis que possuem correlações com outras variáveis, maiores que 0,9.

Para o MLP (*nn*), utilizou-se todas as variáveis na camada de entrada, considerando 16 camadas ocultas com 32 neurônios cada e 100 épocas para o treinamento.

Nos três modelos, a variável resposta (ou variável-alvo) foi o *Last*, ou seja, o último valor obtido no dia.

Na Figura 6 abaixo, é possível observar o comparativo entre o preço previsto (no eixo x) e o preço real (no eixo y), dos modelos aplicados na base de testes. Para facilitar a interpretação, foi desenhada uma linha tracejada vermelha, onde se lê que quanto mais próximos os pontos estiverem desta linha, mais preciso será o modelo.

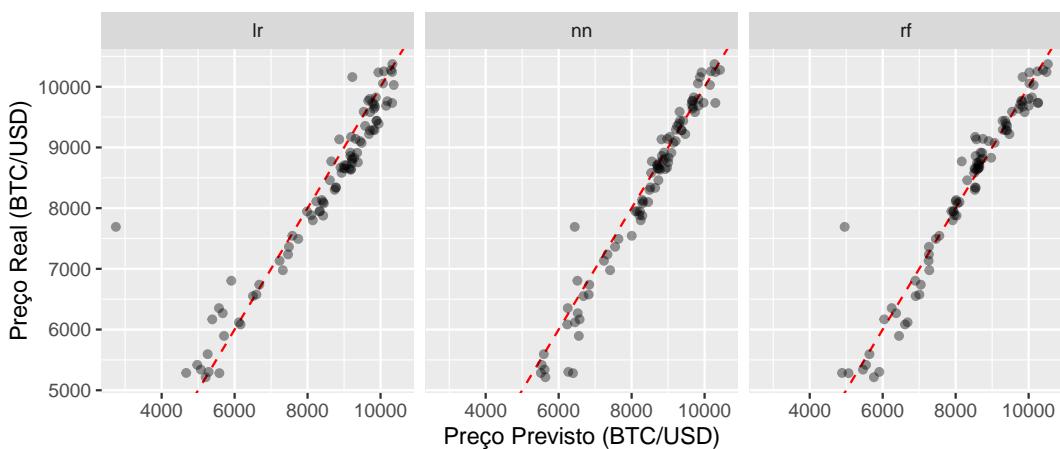


Figura 6. Evolução do preço do Bitcoin, após filtros

Fonte: Dados do estudo

Detaca-se um ponto discrepante (à esquerda da linha tracejada), em todos os modelos. Isso indica, que neste ponto específico, os modelos não conseguiram estimar de

forma precisa o preço.

Na figura 7, pode-se analisar os valores previstos (em vermelho) e reais (em azul) através do tempo. Detaca-se o modelo de Regressão Linear (lr) que não conseguiu prever muito bem os valores em meados do mês de março, enquanto os modelos Random Forest (rf) e MLP (nn) conseguiram ser mais aderentes nas previsões realizadas.

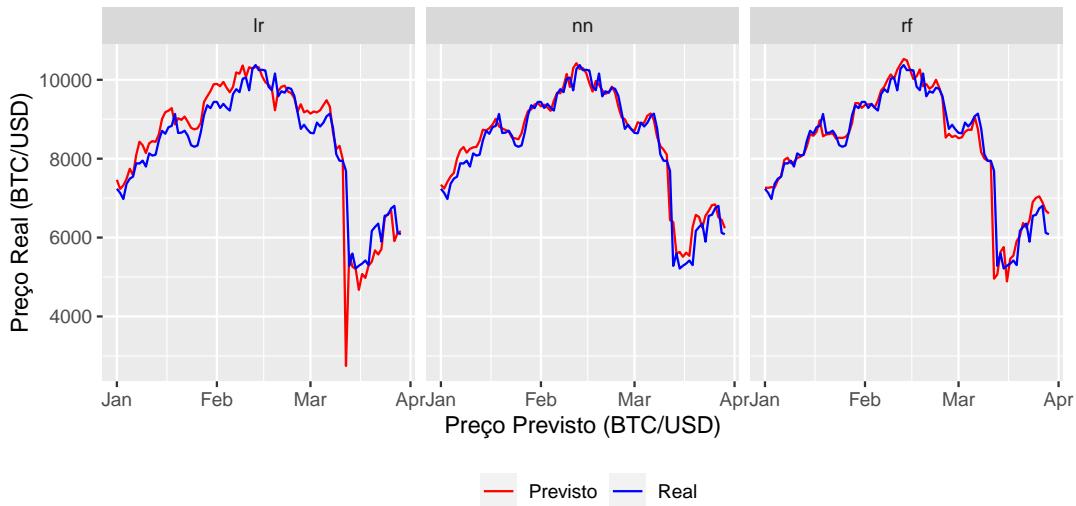


Figura 7. Evolução do preço do Bitcoin, após filtros

Fonte: Dados do estudo

Já na tabela 4, são apresentadas as métricas *Root Mean Squared Error* (rmse), R^2 (rsq) e *Mean Absolute Error* (mae). Para o rmse e mae, quanto menor os valores observados, ou seja, menor os erros observados (diferença entre os valores estimados e reais), melhor o desempenho do modelo. Para o R^2 , que é o coeficiente de determinação do modelo, que traz o quanto da variação total dos dados é efetivamente explicada pelo modelo, quanto maior os valores, melhor o desempenho do modelo.

Com isso, pode-se observar que, confirmando a tendência observada nas Figuras 6 e 7, o modelo de Regressão Linear Múltipla, apresentou a pior performance os três, quando o modelo *Multilayer Perceptron*, apresentou os melhores resultados.

Tabela 4. Métricas dos modelos

	lr	nn	rf
rmse	642.854	302.027	381.754
rsq	0.864	0.963	0.928
mae	360.761	214.231	213.004

Fonte: Dados do estudo

5. Considerações finais

O Objetivo deste trabalho foi implementar e testar modelos capazes de analisar dados históricos das cotações do Bitcoin, para identificar o comportamento do preço e projetar as cotações futuras.

Construiu-se então três modelos (Regressão Linear Múltipla, *Random Forest* e *Multilayer Perceptron*), para auxiliar na tomada de decisões e estratégias de investimentos em Bitcoin.

Os resultados mostram que, o modelo *Multilayer Perceptron* conseguiu se ajustar às variações do preço da moeda de forma mais eficiente que os demais, o tornando em uma ferramenta viável para auxiliar na definição de estratégia de investimento.

Outro ponto a ser destacado é que, devido à variação do cenário econômico e outros quesitos que influenciam o mercado de cryptomoedas, a constante avaliação e otimização do modelo, com uso de novos dados, é indispensável para se mantenha uma ferramenta útil para tomada de decisões.

É recomendável que, caso o valor do Bitcoin se estabilize nos próximos meses, um novo modelo deva ser construído para que se alcance melhores resultados, inclusive com a aplicação de outras técnicas. Este modelo deve ser abordado em estudos futuros.

Além disso, para que se coloque em prática o conhecimento gerado pelo modelo, é importante o desenvolvimento de uma ferramenta que automatize o processo de tomada de decisões de compra e venda, com base nas projeções geradas pelo modelo. Esta ferramenta será confeccionada em estudos futuros.

Referências

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.
- Bailer-Jones, C. A., Gupta, R., and Singh, H. P. (2001). An introduction to artificial neural networks. *arXiv preprint astro-ph/0102224*.
- Bishop, C. M. (1996). Neural networks: a pattern recognition perspective.
- Bishop, C. M. et al. (1995). *Neural networks for pattern recognition*. Oxford university press.
- Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.
- Chollet, F. (2015). keras. <https://github.com/fchollet/keras>.
- De Souto, M., Lorena, A., Delbem, A., and de Carvalho, A. (2003). Técnicas de aprendizado de máquina para problemas de biologia molecular. *Sociedade Brasileira de Computação*, 1(2).
- Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22.
- Goossens, M., Mittelbach, F., and Samarin, A. (1993). *The LaTeX Companion*. Addison-Wesley, Reading, Massachusetts.
- Kelleher, J. D., Mac Namee, B., and D’arcy, A. (2020). *Fundamentals of machine learning for predictive data analytics: algorithms, worked examples, and case studies*. MIT press.
- Kuhn, M. and Wickham, H. (2020a). *recipes: Preprocessing Tools to Create Design Matrices*. R package version 0.1.13.
- Kuhn, M. and Wickham, H. (2020b). *Tidymodels: a collection of packages for modeling and machine learning using tidyverse principles*.
- Liaw, A. and Wiener, M. (2002). Classification and regression by randomforest. *R News*, 2(3):18–22.
- Nakamoto, S. (2019). Bitcoin: A peer-to-peer electronic cash system. Technical report, Manubot.
- Oda, A. L., Senger, M. C. M., and Chára, A. N. (1998). Um estudo sobre diversificação na bolsa de valores de são paulo. *XXII Encontro da Associação Nacional de Pós-Graduação em Administração (ENANAP)*.
- Oliveira, J. J. (2020). *Bitcoin é investimento de risco, mas dá para ganhar sem cair em golpes*. Acesso em 28 de setembro de 2020.
- Pactual, B. (2017). *Diversificação de investimentos: o que é e quais as vantagens*. Acesso em 28 de setembro de 2020.

- R Core Team (2020). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.
- Rauber, T. W. (2005). Redes neurais artificiais. *Universidade Federal do Espírito Santo*, page 29.
- Rodrigues, S. C. A. (2012). *Modelo de regressão linear e suas aplicações*. PhD thesis, Universidade da Beira Interior.
- Sarle, W. S. (1994). Neural networks and statistical models.
- Shawn, R. (2017). *Blockchain Key Terms, Explained*. Acesso em 28 de setembro de 2020.
- Tanaka, M. (2018). *3 tipos de aprendizado caracterizam o Machine Learning*. Acesso em 28 de setembro de 2020.