



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

Análise da predição da tendência do preço do Bitcoin utilizando Long Short-Term Memory para as tarefas de Regressão e Classificação

Bruno Couto Mariño

Monografia apresentada como requisito parcial
para conclusão do Curso de Engenharia da Computação

Orientadora

Prof.a Dr.a Roberta Barbosa Oliveira

Brasília
2023

Dedicatória

Dedico este trabalho a Deus, aos meus pais e a todos aqueles que se interessam por inteligência artificial. Também dedico aos estudiosos e amantes do Bitcoin e do mercado financeiro.

Agradecimentos

Agradeço primeiramente a Deus, por ter me proporcionado a possibilidade de estudar estes assuntos fascinantes de inteligência artificial, mercado financeiro e Bitcoin. Agradeço aos meus pais, Juliana Helena e Fernando Cesar, que sempre incentivaram meus estudos e meu desenvolvimento como ser humano. Agradeço à minha namorada, Roberta Souto, e aos meus amigos que sempre me ajudaram em todas as dificuldades enfrentadas. Em especial, um agradecimento à Profa Dra Roberta Barbosa Oliveira pela ótima orientação, explicações e paciência. Por fim, agradeço à Empresa Júnior de Computação CJR, que me proporcionou amizades valiosas, experiências e aprendizados únicos que levarei por toda a minha vida.

Resumo

O trabalho de predição do preço de um ativo no mercado financeiro é uma tarefa clássica. Com o aumento da importância dos criptoativos na economia mundial, os modelos e técnicas de aprendizado de máquina começaram a ser adaptadas para a predição desses novos ativos. Este trabalho propõe uma análise da predição da tendência do preço do Bitcoin para as técnicas de regressão e de classificação utilizando o modelo *Long Short-Term Memory* (LSTM). Também apresenta uma forma de comparar as técnicas, ao transformar o valor encontrado pelo modelo de regressão em classes e analisar os resultados a partir das métricas de desempenho de classificação. Para esse objetivo, são utilizados os preços diários de abertura e de fechamento, máximo e mínimo diários e o volume de negociações em dólares. A classificação utiliza três classes, que são definidas a partir de um limite superior e inferior de variação percentual do preço; já a regressão utiliza o *log* base 10 do preço de fechamento como alvo. Então, os resultados da regressão são transformados em variação percentual e classificados de acordo com os limites de 3%, 5% e 7%. Assim é possível analisar os resultados das métricas de classificação para o modelo de regressão. O melhor cenário para o modelo de classificação apresentou janela de tempo de 15 dias, um limite de 7% e previu a classe da tendência do preço para 1 dia no futuro. Ele obteve um desempenho de 0,95789 para acurácia, precisão, revocação e *f1-score*; e um *Area Under the ROC Curve* médio de 0,96842. Em relação ao modelo de regressão, o melhor cenário utilizou janela de tempo de 30 dias, previu o *log* do preço para 1 dia no futuro. O modelo alcançou um desempenho de 0,01274 para o *Mean Absolute Error*, 0,00027 para *Mean Squared Error*, 0,01667 para *Root Mean Squared Error*, 0,28828% para *Mean Absolute Percentage Error* e 0,98042 para o R^2 . Após transformar o resultado da regressão em classes de acordo com a variação percentual predita, ele obteve um desempenho para um limite de 7% de 0,94553 para acurácia, precisão, revocação e *f1-score*; e um *Area Under the ROC Curve* médio de 0,95915. Ao comparar ambas as técnicas, a classificação se mostrou mais assertiva.

Palavras-chave: Aprendizado de máquina, classificação, regressão, LSTM, criptomoeda, Bitcoin

Abstract

Predicting the price of an asset in the financial market is a classic challenge. With the increasing importance of cryptoassets in the world economy, machine learning models and techniques began to be adapted to predict these new assets. This paper proposes an analysis of Bitcoin price trend prediction using regression and classification techniques employing the Long Short-Term Memory (LSTM) model. It also introduces a way to compare these techniques by transforming the value found by the regression model into classes and analyzing the results using classification performance metrics. For this purpose, daily opening and closing prices, daily highs and lows, and trading volume in American dollars are used. Classification involves three classes defined by upper and lower limits of percentage variation in price, while regression uses the base 10 logarithm of the closing price as the target. The regression results are transformed into percentage variation and classified according to 3%, 5%, and 7% limits. This allows analyzing the classification metrics results for the regression model. The optimal scenario for the classification model presented a 15-day time window, a 7% limit, and predicted the price trend class for 1 day ahead. It achieved a performance of 0.95789 for accuracy, precision, recall, and f1-score, and an average Area Under the ROC Curve of 0.96842. Regarding the regression model, the best scenario used a 30-day time window and predicted the log of the price for 1 day ahead. It achieved a performance of 0.01274 for Mean Absolute Error, 0.00027 for Mean Squared Error, 0.01667 for Root Mean Squared Error, 0.28828% for Mean Absolute Percentage Error, and 0.98042 for R^2 . After transforming the regression result into classes based on the predicted percentage variation, it obtained a performance for a 7% limit of 0.94553 for accuracy, precision, recall, and f1-score, and an average Area Under the ROC Curve of 0.95915. When comparing both techniques, classification proved to be more accurate.

Keywords: Machine learning, classification, regression, LSTM, Bitcoin

Sumário

1	Introdução	1
1.1	Contextualização	1
1.2	Objetivo do Trabalho	3
1.3	Estrutura do Documento	3
2	Fundamentação Teórica	4
2.1	Predição de Ativos no Mercado Financeiro	4
2.1.1	Gráfico de Vela	4
2.1.2	Séries Temporais	5
2.2	Bitcoin	5
2.3	Aprendizado de Máquina	6
2.4	Rede Neural	7
2.5	Métodos de Avaliação de Desempenho	10
2.5.1	Métricas Comuns à Regressão	10
2.5.2	Métricas Comuns à Classificação	12
3	Revisão de Literatura	14
3.1	Mercado Financeiro Tradicional	14
3.2	Análise do Bitcoin	15
3.2.1	Modelos de Classificação	15
3.2.2	Modelos de Regressão	16
3.2.3	Comparação entre Classificação e Regressão	16
3.3	Análise do Ethereum	17
3.4	Considerações Finais	18
4	Metodologia	20
4.1	Base de Dados	21
4.2	Preparação dos Dados	22
4.3	Classificação	25
4.4	Regressão	27

4.5	Comparação entre a Classificação e a Regressão	29
4.6	Análise dos Resultados	29
5	Resultados Experimentais	30
5.1	Classificação da Tendência do Bitcoin	31
5.2	Regressão do <i>Log</i> do Preço do Bitcoin	35
5.3	Transformação em Classes	36
5.4	Análise das Métricas de Desempenho	36
5.4.1	Análise das Métricas de Classificação	36
5.4.2	Análise das Métricas de Regressão	42
5.4.3	Análise das Métricas de Classificação para os Resultados do Modelo de Regressão	44
5.4.4	Comparação entre as Tarefas Regressão e Classificação	46
5.4.5	Reflexões sobre os Resultados Obtidos	46
6	Trabalhos Futuros	49
	Referências	51

Lista de Figuras

2.1	Exemplo de um gráfico de velas <i>candlestick</i> , elaboração própria.	5
2.2	Exemplo da arquitetura de uma rede neural (fonte: elaboração própria). . .	7
2.3	Arquitetura de uma célula LSTM [1].	8
4.1	Esquema representativo das etapas abordadas no trabalho.	21
4.2	Primeiros 5 elementos da base de dados.	22
4.3	Últimos 5 elementos da base de dados.	22
4.4	Dados gerados a partir da variação entre os dias da base de dados.	24
4.5	Gráfico com o preço de fechamento real do conjunto de treino, validação e teste pelo tempo.	25
4.6	Figura representativa dos dados de treino, validação e teste.	26
5.1	Gráficos de perda por época de cada cenário de classificação (1 a 9).	33
5.2	Gráficos de perda por época de cada cenário de classificação (10 a 18). . .	34
5.3	Gráficos de perda por época de cada cenários de regressão.	37

Lista de Tabelas

3.1	Trabalhos abordados neste capítulo.	19
4.1	Cenários testados para a Classificação.	26
4.2	Configurações de Rede Neural para Classificação.	27
4.3	Cenários testados para a Regressão.	28
4.4	Configurações de Rede Neural para Regressão.	28
5.1	Número de amostras de cada classe para cada cenário de Classificação . . .	31
5.2	Resultado da escolha dos hiperparâmetros com <i>Random Search</i>	32
5.3	Resultado da escolha dos hiperparâmetros com <i>Random Search</i> para a re- gressão.	35
5.4	Número de amostras de cada classe para cada cenário do modelo de Re- gressão (1 a 6).	38
5.5	Número de amostras de cada classe para cada cenário do modelo de Re- gressão (7 a 12).	38
5.6	Resultado das métricas de Classificação agrupadas variando o limite. . . .	39
5.7	Resultado das métricas de Classificação agrupadas variando o dia predito. .	40
5.8	Resultado das métricas de Classificação agrupadas variando a janela de tempo.	41
5.9	Resultado das métricas de Regressão agrupadas variando o dia predito. . .	43
5.10	Resultado das métricas de Regressão agrupadas variando a janela de tempo.	43
5.11	Resultado das métricas de classificação para os resultados do modelo de Regressão (1 a 6).	44
5.12	Resultado das métricas de classificação para os resultados do modelo de Regressão (7 a 12).	45
5.13	Comparação dos resultados das métricas de Classificação para os modelos criados, com um limite de 7%.	46

Lista de Abreviaturas e Siglas

Adaboost *Adaptive Boosting.*

ANN *Artificial Neural Network.*

AR *Autoregressive model.*

ARIMA *Auto-Regressive Integrated Moving Average.*

AUC *Area Under Curve.*

BNN *Bayesian Neural Networks.*

BPTT *Backpropagation Through Time.*

CNB *Complement Naive Bayes.*

CNN *Convolutional Neural Network.*

DNN *Deep Neural Network.*

DRN *Deep Residual Network.*

DTC *Decision Tree Classification.*

GRU *Gated Recurrent Unit.*

KNN *K-Nearest Neighbors.*

LR *Logistic Regression.*

LSTM *Long Short-Term Memory.*

MAE *Mean Absolute Error.*

MAPE *Mean Absolute Percentage Error.*

MSE *Mean Squared Error.*

RF *Random Forest.*

RFC *Random Forest Classifier.*

RFR *Random Forest Regression.*

RMSE *Root Mean Squared Error.*

RNN *Recurrent Neural Network.*

ROC *Receiver Operating Characteristic.*

ROI Retorno sobre o investimento, do inglês *Return On Investment.*

SVC *Support-Vector Classifier.*

SVM *Support-Vector Machine.*

SVR *Support-Vector Regression.*

XGBoost *Extreme Gradient Boosting.*

Capítulo 1

Introdução

Este capítulo apresenta a contextualização do tema apresentado, o objetivo do trabalho e a estrutura deste documento.

1.1 Contextualização

O Bitcoin é conhecido como uma criptomoeda que opera por meio de uma tecnologia chamada *blockchain*, ou seja, uma rede descentralizada que cria registros imutáveis de transações, interligando blocos e garantindo segurança por meio de criptografia. O valor de mercado do ativo já chegou a atingir mais de 1 trilhão de dólares e, atualmente, o Bitcoin já está presente no portfólio de algumas empresas, como MicroStrategy¹ e Tesla². Alguns países como El Salvador e República Centro-Africana já até o adotaram como uma das moedas oficiais do país³.

Assim, a presença do Bitcoin no mercado financeiro global não apenas desafia as noções tradicionais de moeda e investimento, mas também destaca o contraste fundamental entre seu modelo descentralizado e os sistemas financeiros convencionais. Enquanto o mercado financeiro tradicional é sustentado por instituições centralizadas, como bancos e bolsas de valores, o Bitcoin opera sem a necessidade de intermediários, baseando-se em um sistema distribuído e descentralizado sem uma autoridade central. Esse contraste de paradigmas ressalta não apenas a complexidade do Bitcoin como ativo, mas também seu potencial para remodelar e influenciar a dinâmica do mercado financeiro global.

O mercado financeiro tradicional, por sua vez, possibilita o financiamento de empresas por meio da venda de participações acionárias. Os compradores, denominados acionistas, adquirem uma ação que representa uma fração da empresa. O preço é determinado pelo

¹valor.globo.com

²br.cointelegraph.com

³valorinveste.globo.com

mercado, especificamente pela interação entre oferta e demanda. Se a demanda por uma ação supera a oferta disponível, seu preço tende a subir. Por outro lado, se a oferta é maior do que a demanda, o preço da ação pode cair. Essa dinâmica é influenciada por uma variedade de fatores, como desempenho financeiro da empresa, perspectivas de crescimento, mudanças em seu setor, condições econômicas globais, notícias, entre outros.

Deste contexto, emergiu o desafio de antecipar as flutuações nos preços das ações e índices de mercado. Essa tarefa é complexa devido à multiplicidade de elementos que impactam a oferta e a procura de um ativo. Além disso, a criação de um modelo preditivo perfeito poderia gerar uma mudança na própria dinâmica de demanda pelo ativo, comprometendo sua própria eficácia. Apesar disso, o estudo desse tema é de extrema importância, porque gera informações úteis para a tomada de decisão de investidores, promove a redução da volatilidade e auxilia a entender as movimentações do mercado e detecção de fraudes, por meio da identificação de padrões anômalos.

A partir desse desafio, surgiram diversos modelos que buscaram prever a movimentação do preço das ações utilizando métodos estatísticos como regressão linear e séries temporais [2]. Posteriormente, modelos mais avançados de aprendizado de máquina, como redes neurais, *Support-Vector Machine* (SVM) e árvores de decisão, começaram a ser aplicados também [3] [4]. Mais recentemente modelos de aprendizado profundo (*deep learning*) vêm se mostrando promissores, como o *Long Short-Term Memory* (LSTM) [5]. Além disso, o avanço da análise preditiva no mercado financeiro tradicional tem se estendido para além das fronteiras convencionais, adentrando também o mundo dos criptoativos. Modelos estatísticos e algoritmos de aprendizado de máquina, originalmente desenvolvidos para prever movimentos de preços em ações e ativos tradicionais, têm sido adaptados e aplicados ao mercado de criptomoedas, como o Bitcoin e o Ethereum [6]. Por exemplo, estudos recentes, melhor abordados no Capítulo 3, têm demonstrado a eficácia de modelos preditivos baseados em análise estatística e aprendizado de máquina na previsão dos seus movimentos de preços [4]. Essas abordagens exploram não apenas os aspectos que são comuns ao mercado financeiro, como análise dos preços ou da macroeconomia, mas também as características únicas da tecnologia *blockchain* que influenciam a dinâmica de preços.

Os estudos sobre a previsão de preços de ativos geralmente se dividem em duas abordagens: Classificação e Regressão. O primeiro método procura analisar a tendência do ativo, classificando-a como alta ou baixa se for binária [3], ou usando limites superiores e inferiores se for ternária [7]. Isso permite definir classes que representam áreas acima do “limite superior”, abaixo do “limite inferior” e entre ambos. Já a Regressão busca analisar o preço do ativo ou até mesmo o *Log* dele [8]. Essas abordagens se baseiam no uso de um conjunto de amostras sequenciais, denominada “janela de tempo”, como entrada para

prever o resultado de um valor futuro da sequência. Neste trabalho, serão analisados os dias de negociação do Bitcoin para prever o valor do dia específico no futuro, denominado “dia predito”.

1.2 Objetivo do Trabalho

Diante do cenário da análise preditiva e do aumento da importância do Bitcoin para a sociedade, este trabalho propõe uma análise sobre a tarefa de predição da tendência do preço do ativo utilizando as técnicas classificação e regressão, a rede neural LSTM e os dados históricos e do volume de negociações. Para atingir esse objetivo principal, são definidos três objetivos específicos:

- Analisar os melhores hiperparâmetros da rede neural para as tarefas de Classificação e de Regressão;
- Analisar o impacto dos parâmetros limite, dia predito e janela de tempo no desempenho dos modelos de Classificação e Regressão;
- Propor uma abordagem de comparação entre o modelo de Classificação com o de Regressão a fim de definir qual possui o melhor desempenho.

1.3 Estrutura do Documento

O documento apresenta, no Capítulo 2, uma explicação sobre os fundamentos teóricos necessários e utilizados no projeto. O Capítulo 3, por sua vez, traz um resumo da literatura sobre o tema de aprendizado de máquina para predição de ativos, incluindo o bitcoin. Já o Capítulo 4 demonstra a metodologia aplicada aos experimentos realizados, que possuem seus resultados no Capítulo 5. Por fim, a conclusão e trabalhos futuros são apresentadas no Capítulo 6.

Capítulo 2

Fundamentação Teórica

Este capítulo apresenta uma base de conceitos necessária para a realização deste trabalho, que tem por objetivo estudar técnicas de predição da tendência do preço da moeda digital Bitcoin. São abordados tópicos sobre o mercado financeiro, mercado de criptoativos, métricas de desempenho e redes neurais, que foram necessários para a construção dos modelos de predição criado.

A predição do preço de um ativo, como uma ação na bolsa de valores, é uma tarefa clássica e ao mesmo tempo complexa que já vem sendo estudada há algumas décadas, com o uso de técnicas de regressão linear e aprendizado de máquina (*machine learning*) [9]. Com a rápida difusão das técnicas de aprendizado profundo (*deep learning*), modelos como o LSTM começaram a também serem utilizados para esse objetivo. Além disso, o problema de se prever o preço de um ativo tradicional do mercado foi expandido para o ramo das moedas digitais, como o Bitcoin. Ele, por sua vez, possui características próprias que o diferenciam dos ativos de bolsa de valores tradicionais. Por isso, essas diferenças serão mais abordadas neste capítulo.

2.1 Predição de Ativos no Mercado Financeiro

A tarefa de predição de ativos no mercado financeiro tradicional pode utilizar diversos dados de entrada como dados históricos, da macroeconomia, da análise técnica e fundamentalista [5]. Esta seção aborda conceitos necessários sobre os dados de entrada dos modelos criados.

2.1.1 Gráfico de Vela

É comum os artigos que estudam sobre predição do valor de mercado de um ativo utilizarem dados históricos de negociação, que compreendem o maior preço do dia, o menor, o da

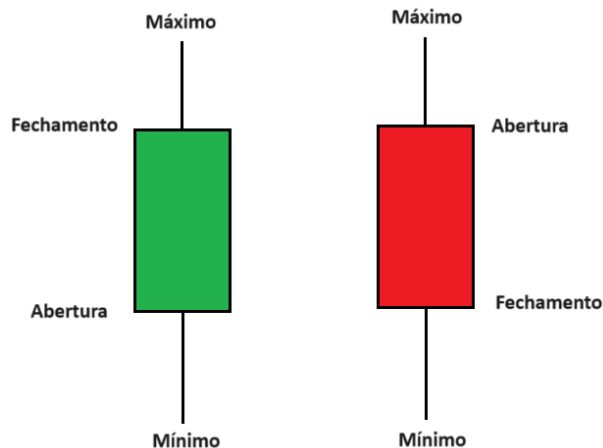


Figura 2.1: Exemplo de um gráfico de velas *candlestick*, elaboração própria.

primeira negociação (abertura) e o da última negociação (fechamento) [10]. Esses dados podem ser representados através do gráfico de uma vela, em inglês, *candlestick*. A Figura 2.1 representa um dia no gráfico de vela. A cor verde indica que o preço de fechamento foi superior ao de abertura, ou seja, o ativo se valorizou naquele período analisado, já a cor vermelha indica o oposto.

2.1.2 Séries Temporais

Séries temporais são uma coleção de observações coletadas de forma sequencial ao longo de um tempo. De acordo com Zhongyang *et al.* [11], em geral, são usadas em problemas de predição para adivinhar um valor futuro no tempo $t+h$, utilizando os dados disponíveis até o tempo t , sendo t um dado tempo e h um número natural maior que 1 para que $t+h$ seja um tempo superior a t . Na Equação 2.1 há uma exemplificação de uma série temporal $x_t, x_{t-1}, \dots, x_{t-N+1}$ utilizada para prever um valor futuro x_{t+h} utilizando N valores de entrada. O N pode ser denominado também *window size* ou *time step*.

$$x_{t+h} = f(x_t, x_{t-1}, \dots, x_{t-N+1}). \quad (2.1)$$

2.2 Bitcoin

É definido como um sistema de dinheiro eletrônico ponto a ponto pelo seu criador Satoshi Nakamoto [12], um pseudônimo criado por um autor anônimo. De acordo com Sheth e Dattani [13], também pode ser descrito como um banco de dados de registros descentralizado e distribuído que promove transações rápidas e confiáveis sem um gerenciamento centralizado. Uma rede de nós P2P (*peer to peer*) permite o armazenamento, verificação

e auditoria das transações. Uma vez que a transação é confirmada na *blockchain*, é impossível excluí-la, alterá-la ou adulterá-la. A confirmação é realizada por mecanismos de consenso da rede, sem a necessidade de intervenientes de instituições financeiras. Além disso, o Bitcoin possui o código aberto e pode ser acessado pelo Github¹ e o download realizado pelo site oficial Bitcoin.org²

Devido à sua expressiva valorização desde sua criação, começou a ser mais conhecido e estudado. A moeda chegou a ser negociada a 0,00099 dólar em 2009, já em 2021 atingiu a marca dos 60000 dólares³. Devido à essa alta, muitos programadores e economistas começaram a estudar formas de prever a movimentação desse ativo. Essa tarefa já era estudada para o mercado financeiro tradicional, como é mostrado no Capítulo 3, porém apresenta algumas diferenças, como a própria arquitetura *blockchain*, sua característica descentralizada e pelo fato do ativo ser inerentemente fora do sistema tradicional.

Por ter sido criado com a intenção de ser descentralizado, algumas informações não são acessíveis diretamente através da rede. Como não há um local centralizado para a compra dele, algumas informações utilizadas para entender a rede, como preços de negociação e volume em dólares negociado, podem ser obtidas através de corretoras, que são empresas que trabalham facilitando o comércio da moeda. Por outro lado, há dados que podem ser acessíveis de forma transparente à todos conectados à rede, como força computacional, dificuldade de mineração, tamanho médio do bloco, taxa de transação média, inflação, número de nós conectados, taxa de uso da rede, entre outros. Ao longo dos anos surgiram diversas abordagens que agrupam alguns desses dados, sejam de corretoras ou da *blockchain*, para aplicar em algoritmos de aprendizado de máquina a fim de prever futuras movimentações do ativo [8].

2.3 Aprendizado de Máquina

A área da computação conhecida como aprendizado de máquina se dedica ao estudo de algoritmos capazes de imitar o processo de aprendizagem humano. Seu objetivo é treinar modelos para identificar e compreender padrões nos dados, possibilitando a replicação desse aprendizado em conjuntos de dados desconhecidos. O programa criado pode aprender padrões a partir de rótulos pré-definidos, que seria aprendizado supervisionado, ou aprender ao longo do seu treinamento quais são os rótulos e conjuntos similares para a criar os agrupamentos sem a interferência direta, que seria aprendizado não supervisionado [14]. O problema estudado neste trabalho é o primeiro tipo, o aprendizado supervisionado, ou seja, o algoritmo é treinado por um conjunto de dados previamente rotulados.

¹github.com

²bitcoin.org

³buybitcoinworldwide.com

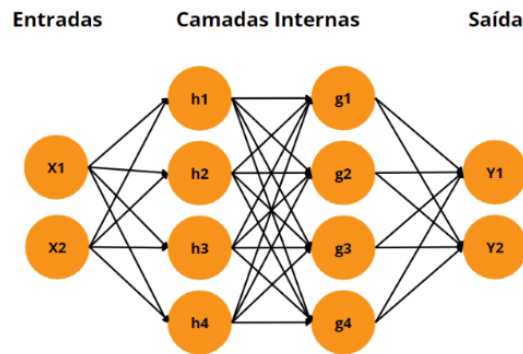


Figura 2.2: Exemplo da arquitetura de uma rede neural (fonte: elaboração própria).

Também é interessante diferenciar problemas de regressão dos de classificação [15]. Para o primeiro, a variável de saída é contínua e numérica. Isso significa que o objetivo é prever um valor numérico dentro de um intervalo ou espectro, como previsão do preço nominal de um ativo em determinado dia. Já para a classificação, a variável de saída é categórica e discreta. Essa técnica busca agrupar os dados em categorias, como classificar se o preço de um ativo vai subir ou cair em determinado dia, prever a tendência.

2.4 Rede Neural

A Rede Neural, ou Rede Neural Artificial, do inglês *Artificial Neural Network* (ANN), é uma arquitetura de aprendizado de máquina que busca simular o funcionamento do cérebro humano [16]. Nela são implementadas camadas compostas por nós (células), que funcionam como neurônios artificiais, que se conectam com a definição de pesos e limites. Já uma rede neural profunda consiste em uma rede neural artificial que possui três ou mais camadas com entradas, saídas e camadas internas. A Figura 2.2 exemplifica a arquitetura de uma rede neural com duas camadas internas de quatro nós. As setas indicam os pesos; as células X_i representam as entradas; h_i os neurônios da primeira camada interna; g_i os neurônios da segunda camada interna e Y_i as saídas.

Rede Neural Recorrente

As redes neurais recorrentes, do inglês *Recurrent Neural Network* (RNN) são modelos de redes neurais projetadas especificamente para lidar com dados sequenciais, como séries temporais, texto, fala, entre outros. A principal diferença fundamental entre elas e uma ANN convencional está na capacidade da RNN de manter uma forma de memória de curto prazo que permite processar seqüências de dados, por meio de um algoritmo de propagação

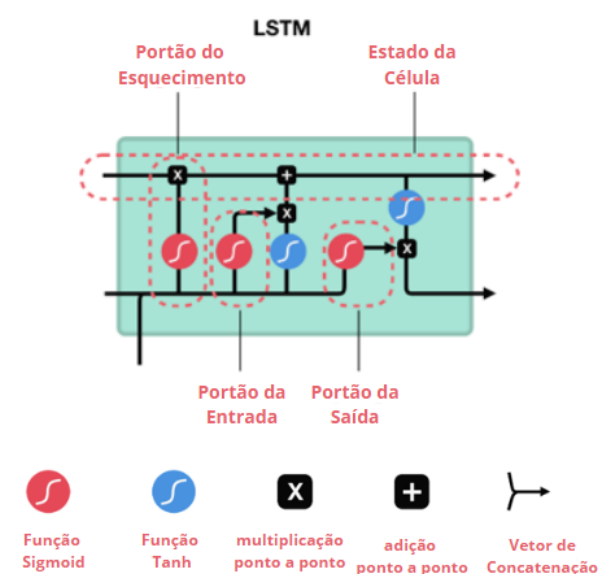


Figura 2.3: Arquitetura de uma célula LSTM [1].

retroativa através do tempo ou, em inglês, *Backpropagation Through Time* (BPTT). Uma RNN, quando é treinada, processa dados ao longo de várias etapas temporais. Durante esse processo, a rede calcula os gradientes, que representam as taxas de mudança da função de perda em relação aos parâmetros da rede, que são usados para atualizar os pesos de forma a minimizar o erro entre a saída prevista e a saída real, permitindo que o modelo aprenda e melhore sua capacidade de realizar previsões.

Um problema enfrentado por elas é o dissipação do gradiente [17], que pode dificultar a retenção de informações de longo prazo em sequências muito longas. Nesse caso, os primeiros valores passam a ser esquecidos, há uma dificuldade de transportar informações para as etapas posteriores. O problema ocorre quando esse valor diminui à medida que se propaga no tempo, ocasionando o esquecimento com os sequências grandes.

Long Short-Term Memory

A fim de resolver os problemas de dissipação de gradiente, foram criados alguns modelos como LSTM [18], que busca transitar informações entre as camadas. Ele implementa um mecanismo de portões que regulam o fluxo de informações para o nó. Eles decidem quais informações em uma sequência são importantes e quais podem ser descartadas. Assim, o *LSTM* é insensitivo à grandes intervalos, porque consegue transmitir informações relevantes ao longo de uma longa cadeia de sequências para fazer previsões. A Figura 2.3 representa uma célula *LSTM* com seus mecanismos de portões. No caso do *LSTM*, temos os portões esquecimento, entrada e saída:

- Esquecimento (*forget gate*) - Decide qual informação deve ser esquecida ou lembrada recebendo o estado da célula anterior e a entrada atual. Pode ser encontrado por meio da Equação 2.2, em que f_t representa a saída do portão no instante de tempo t , σ a função sigmoide, que comprime os valores para o intervalo entre 0 e 1, H_{t-1} a saída anterior, X_t a entrada atual, W_f a matriz dos pesos das conexões e b_f o viés da rede neural.

$$f_t = \sigma(W_f \cdot [H_{t-1}, X_t] + b_f) \quad (2.2)$$

- Entrada (*input gate*) - Utiliza o mesmo mecanismo que o portão do esquecimento, mas decide quais informações devem ser adicionadas no estado atual, para serem transmitidas futuramente. Para decidir quais valores de entrada serão atualizados, neste exemplo é utilizada uma função sigmoide, representada na Equação 2.3. Já para gerar valores candidatos para serem adicionados ao estado da célula, é usada como exemplo uma função tanh, representada na Equação 2.4, em que i_t a saída da porta de entrada, tanh a tangente hiperbólica, que comprime os valores de entrada para um intervalo entre -1 e 1 e \tilde{C}_t representa os valores candidatos gerados pela função tanh.

$$i_t = \sigma(W_i \cdot [H_{t-1}, X_t] + b_i) \quad (2.3)$$

$$\tilde{C}_t = \tanh(W_C \cdot [H_{t-1}, X_t] + b_C) \quad (2.4)$$

- Estado da célula - a Equação 2.5 demonstra como é feita a atualização do estado da célula, combinando o estado anterior com os novos valores. Nela, C_t representa o estado atual da célula e C_{t-1} o estado da célula anterior.

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \quad (2.5)$$

- Saída (*output gate*) - Define quais informações do estado atual serão passadas para a saída, levando em consideração o estado anterior e o atual. A saída é representada pela Equação 2.6 e o estado da célula atual é passado por uma função tanh (Equação 2.7), em que o_t representa a saída da porta de saída e H_t a saída final da célula LSTM.

$$o_t = \sigma(W_o \cdot [H_{t-1}, X_t] + b_o) \quad (2.6)$$

$$H_t = o_t \cdot \tanh(C_t) \quad (2.7)$$

2.5 Métodos de Avaliação de Desempenho

A fim de avaliar o desempenho do modelo preditivo, é necessário fazer uso de métricas de desempenho específicas para o problema em questão. Neste trabalho, o problema é analisado tanto através de uma ótica de regressão quanto de classificação, portanto, é necessário apresentar as métricas mais comuns utilizadas nesses problemas. Um trabalho de resumo proposto por Verma *et al.* [5] analisou 50 artigos relacionados à predição do preço de ativos no mercado financeiro e entendeu que para problemas de regressão (previsão do preço das ações) são mais utilizados o erro médio absoluto, erro quadrático médio, a raiz do erro quadrático médio e erro absoluto percentual médio; enquanto para problemas de classificação (predição da tendência das ações), acurácia, precisão, revocação e *F-score* são considerados. Já Wang e Yan [7] utilizaram a área sobre a curva característica para medir o desempenho do modelo de classificação da tendência do preço do Bitcoin.

2.5.1 Métricas Comuns à Regressão

Esta etapa apresenta as funções para o cálculo das métricas de desempenho para a regressão.

Erro Médio Absoluto

O erro médio absoluto, do inglês *Mean Absolute Error* (MAE), calcula a média das diferenças absolutas entre as previsões do modelo e os valores reais. É uma métrica que não penaliza grandes erros de forma significativa. Na Equação 2.8 há o cálculo da média da diferença entre o valor predito \hat{y} e o real y , com o n representando o número total de observações ou amostras no conjunto de dados. Quanto menor o valor de MAE, significa que melhor são os resultados preditos pelo modelo.

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (2.8)$$

Erro Quadrático Médio

O erro quadrático médio, do inglês *Mean Squared Error* (MSE), calcula a média das diferenças ao quadrado entre as previsões do modelo \hat{y} e os valores reais y . Essa métrica penaliza erros maiores mais do que o MAE, tornando-o sensível a *outliers*, que são valores muito discrepantes dos outros. A representação do cálculo desse erro é encontrado na Equação 2.9.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2.9)$$

$$TVN = \frac{VN}{VN + FP} \quad (2.10)$$

$$TPR = \frac{VP}{VP + FN} \quad (2.11)$$

Raiz do Erro Quadrático Médio

A raiz do erro quadrático médio, do inglês *Root Mean Squared Error* (RMSE), é a raiz quadrada do MSE. Ela é útil para entender o erro médio em termos da unidade dos valores de destino e é representada na Equação 2.12.

$$RMSE = \sqrt{MSE} \quad (2.12)$$

Erro Percentual Absoluto Médio

O Erro Percentual Absoluto Médio, do inglês *Mean Absolute Percentage Error* (MAPE), mostra a porcentagem de erro em relação aos valores reais, em outras palavras, encontra a média da diferença entre o valor previsto e o real equivale, tanto para mais quanto para menos. Essa métrica está representada pela Equação 2.13.

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100\% \quad (2.13)$$

Coefficiente de Determinação

O coeficiente de determinação, também conhecido como R^2 avalia quão próximos os dados estão da linha de regressão ajustada. Nele, o valor de 1 equivale à um perfeito ajuste e 0 indica que o modelo não faz um ajuste melhor do que uma linha horizontal que passa pela média dos valores observados. É possível que os valores também sejam negativos, o que sugere que o modelo de regressão está se ajustando pior aos dados do que uma linha horizontal que passa pela média dos valores observados. Essa métrica é calculada utilizando a soma dos quadrados das diferenças entre os valores observados (y) com os valores previstos pelo modelo (\hat{y}) e a soma dos quadrados das diferenças entre os valores observados (y) e a média dos valores observados (\bar{y}) como descrito na Equação 2.14.

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (2.14)$$

2.5.2 Métricas Comuns à Classificação

Esta etapa apresenta as funções para o cálculo das métricas de desempenho para classificação. As médias são calculadas considerando todas as instâncias (amostras de dados) individualmente, independentemente da classe a que pertencem. Todas as previsões e rótulos verdadeiros são tratados como se fossem de uma única classe. Isso significa que cada instância tem o mesmo peso na métrica, independentemente de qual classe ela pertença. Essa abordagem é útil quando há classes muito desbalanceadas, porque avalia o desempenho global do modelo em todas as instâncias de forma igual. Já, por meio do *None*, é possível entender o comportamento específico para cada classe.

Acurácia

A acurácia representa a proporção de previsões corretas feitas pelo modelo em relação ao total de observações. Enquanto em análises binárias, uma classe é designada como verdadeira e a outra como falsa, em problemas ternários, todas as classes são consideradas simultaneamente, não apenas em relação a uma verdadeira e uma falsa. A acurácia avalia o quão precisas foram as previsões para todas as três classes, determinando a proporção total de previsões corretas em relação ao número total de observações. É possível calcular a acurácia de acordo com a Equação 2.15. Nela, os Verdadeiros Positivos (VP) são as observações corretamente previstas como pertencentes a uma classe específica, enquanto os Verdadeiros Negativos (VN) são observações corretamente previstas como não pertencentes a uma classe específica. Para problemas com mais de uma classe, os valores de VP representa a soma dos verdadeiros positivos de cada classe, enquanto VN, a soma dos verdadeiros negativos de cada classe.

$$\text{Acurácia} = \frac{\text{VP} + \text{VN}}{\text{Total}} \quad (2.15)$$

Precisão Média

A precisão mede a proporção de observações positivas classificadas corretamente em relação ao total de observações classificadas como positivas. É útil para avaliar a qualidade das previsões positivas do modelo e minimizar os falsos positivos. Seu cálculo é apresentado na Equação 2.16, em que Falsos Positivos (FP) são os valores que foram erroneamente preditos como a classe alvo. Para problemas com mais de uma classe, FP representa a soma dos falsos positivos específicos de cada classe.

$$\text{Precisão} = \frac{\text{VP}}{\text{VP} + \text{FP}} \quad (2.16)$$

Revocação Média

Revocação, também conhecido como *recall* do inglês, mede a taxa de verdadeiros positivos, ou seja, a proporção de observações positivas classificadas corretamente em relação ao total de observações reais positivas. Ela é útil para avaliar a capacidade do modelo de identificar corretamente todas as observações positivas. Seu cálculo é apresentado na Equação 2.17, em que Falsos Negativos (FN) são os valores que foram erroneamente preditos como a não pertencentes à classe alvo. Para problemas com mais de uma classe, FN representa a soma dos falsos negativos específicos de cada classe.

$$\text{Revocação} = \frac{\text{VP}}{\text{VP} + \text{FN}} \quad (2.17)$$

F1-Score

O F1-Score é a média harmônica da precisão e da revocação. Ele é útil quando o objetivo do problema é equilibrar a precisão e a revocação, especialmente se houver um desequilíbrio entre as classes.

$$\text{F1-Score} = \frac{2 \cdot \text{Precisão} \cdot \text{Revocação}}{\text{Precisão} + \text{Revocação}} \quad (2.18)$$

Área Sob a Curva Característica de Operação do Receptor

A área sob a curva característica de operação do receptor, do inglês *Area Under Curve* (AUC), é criada a partir da área formada sobre a curva *Receiver Operating Characteristic* (ROC). Esta, por sua vez, é criada traçando a taxa de verdadeiros positivos em relação à taxa de falsos positivos para vários valores de limiar. A AUC mede a probabilidade de uma amostra aleatória ser classificado como uma classe positiva [7]. O cálculo do seu valor médio foca no desempenho geral do modelo em todas as classes. É importante considerar não apenas a métrica global, mas também as individuais para cada classe, pois essas métricas podem fornecer informações valiosas sobre como o modelo está se saindo em relação a cada classe específica [19].

Capítulo 3

Revisão de Literatura

Este capítulo é referente à revisão de literatura realizada para dar suporte à tarefa de predição do preço de negociação do Bitcoin. Para isso, ele aborda artigos que realizam estudos sobre predição dos preços de ativos do mercado, sendo eles ações, Bitcoin e outros criptoativos. Também são analisados artigos com diferentes abordagens, tanto de regressão, que é a previsão do preço futuro do ativo, quanto de classificação, previsão da tendência do preço futuro do ativo. Para este último, há a possibilidade de classificar em duas ou três categorias.

3.1 Mercado Financeiro Tradicional

A tarefa de predição do preço de um ativo no mercado financeiro já é um assunto bastante estudado na literatura. Antes de analisar o mercado específico do Bitcoin, é importante analisar artigos referentes ao mercado tradicional. Ming-Wei *et al.* [20] utilizam dados de 34 índices financeiros pelo período de seis anos a fim de realizar um estudo sobre uso de modelos de aprendizado de máquina para predição do preço de ativos. Os autores estudam os fatores como maturidade do mercado, metodologia, composição dos atributos e horizonte tempo. Após diversas comparações, eles entendem que a maturidade do mercado e o horizonte de tempo analisado afetam a acurácia do modelo preditivo, porém, o uso de indicadores técnicos não melhoraram o desempenho, se comparado apenas com o uso dos preços históricos.

Por outro lado, o estudo proposto por Nabipour *et al.* [3] utilizou 10 indicadores técnicos criados a partir dos preços de movimentação do ativo para prever sua tendência, subir ou cair. Quatro grupos do mercado de ações foram analisados, sendo eles, financeiros, petróleo, minerais não metálicos e metais básicos da bolsa de valores de Teerã. Os autores utilizaram os modelos Árvore de Decisão (*Decision Tree*), *Random Forest* (RF), *Adaptive Boosting* (Adaboost), *Extreme Gradient Boosting* (XGBoost), *Support-Vector Classifier*

(SVC), *Naïve Bayes*, *K-Nearest Neighbors* (KNN), *Logistic Regression* (LR), *ANN*, *RNN* e *LSTM*. Foram analisados os impactos dos indicadores técnicos em sua forma contínua e uma posterior transformação deles para binários, de acordo com a especificidade de cada indicador. A abordagem binária se mostrou mais eficaz, porém com apenas uma pequena melhora em relação aos modelos de *deep learning*. Estes, sendo *RNN* e *LSTM*, se mostraram superiores aos demais modelos para as métricas de F1-score, Acurácia e *AUC*.

3.2 Análise do Bitcoin

Os problemas de predição da movimentação do preço do Bitcoin são divididos, em geral, em tarefas de regressão e classificação. A primeira busca prever o seu valor e a segunda a sua tendência.

3.2.1 Modelos de Classificação

O trabalho do Helfenstein [21] utiliza o modelo *LSTM* para classificar se o valor do Bitcoin subiu ou caiu, através de uma análise do histórico de preço semanal e do volume de negociações. Usa apenas 418 valores, referentes aos preços semanais. O artigo varia a janela de observação e as épocas. Também analisa a utilização de dados macroeconômicos junto ao modelo, porém gerou uma acurácia menor (51,82%), se comparada à utilização apenas dos dados históricos do bitcoin (52,9%). Também gera uma métrica de retorno sobre o investimento, simulando caso o ativo fosse comprado ou vendido de acordo com o modelo criado, e métricas de acurácia. Comparou os resultados à algumas estratégias de negociação de ativos do mercado financeiro. Apenas com os dados do bitcoin, o trabalho alcançou uma acurácia de 52,9% e um retorno sobre o investimento de 122,82%.

O impacto que as postagens da rede social “Twitter”, atualmente denominada “X”, tem no preço do Bitcoin é um assunto analisado por Kanji *et al.* [4] utilizando as postagens relativas ao Bitcoin, seus preços históricos e dados da rede *blockchain*. As postagens são classificadas de acordo com uma análise de sentimento e comparadas com o preço do ativo, a fim de verificar se o preço vai aumentar ou diminuir no dia seguinte. São utilizados para a análise de sentimentos os modelos de aprendizado de máquina *LR*, *SVC*, *CNB*, *RFC*, *DTC*, junto com a técnica de *10-fold cross validation*. Para prever os preços da criptomoeda, os autores testam os modelos *LR*, *SVM*, *LSTM*, *GRU*, *LSTM-GRU* *SVM-GRU*. Após treino por 100 épocas e com uma janela de observação de 14 dias, o modelo *SVM-GRU* obteve a melhor acurácia e precisão, 60,86% e 60% respectivamente, enquanto o modelo *SVM-LSTM* e o *SVM* obtiveram a melhor revocação (60%).

Já Wang e Yan [7] estudaram uma abordagem diferente, não só prever se o preço do Bitcoin se movimentaria para cima ou para baixo, mas sim prever movimentações dentro

de três categorias: subiu mais que o limite superior, caiu mais que o limite inferior ou ficou entre o limite superior e inferior. Para isso, utiliza dados diários de negociação do ativo do mês de setembro de 2014 até maio de 2022, um total de 2794 dias. Os autores criam uma taxa de mudança, que é relacionada ao valor percentual de mudança no preço dentro de um determinado período. Os valores de entrada no modelo são: preço mínimo, preço de fechamento e taxa de mudança que variou entre 7, 15 e 30 dias anteriores à data analisada. As classes são criadas a partir de um limite que variou entre 3%, 5% e 7%. Além disso, o modelo é utilizado para prever a variação de 1, 3, 5 e 7 dias no futuro. Dos modelos *LSTM* e *GRU*, o primeiro obteve o melhor resultado; 0,701; na métrica *AUC*, com o limite em 7%, a variação de 30 dias anteriores como entrada e a previsão de 3 dias no futuro.

3.2.2 Modelos de Regressão

A fim de estudar e prever o preço do Bitcoin e sua volatilidade, Jang e Lee [8] utilizam dados da rede *blockchain*, dados históricos de preços e dados macroeconômicos como taxas de câmbio. Os autores optaram por medir o *log* do preço e o *log* da volatilidade, haja vista as grandes diferenças nominais entre os valores mais antigos e mais recentes dessas variáveis. O modelo *Bayesian Neural Networks* (BNN) com *10-fold cross-validation* foi o que se sobressaiu em relação à regressão linear e ao *Support-Vector Regression* (SVR), que foram utilizados como benchmark, atingindo 0,0138 de *MAPE* e 0,0039 de *RMSE* para o *log* do Preço no grupo de teste. Para o *log* da volatilidade, atingiu 0,5090 de *MAPE* e 0,2546 de *RMSE*. Foram realizados também experimentos de seleção de features e implementação do *framework* Rollover. Porém o valor com menor erro se manteve sendo o modelo *BNN* sem o Rollover.

Malhotra *et al.* [22] focaram em identificar qual modelo de aprendizado de máquina era melhor para prever a flutuação no preço do Bitcoin: regressão linear, regressão Ridge, regressão LASSO, *LSTM*, *GRU* ou *ARIMA*. Foram utilizados quatro meses de análise. *ARIMA* foi o modelo que obteve o melhor desempenho, com 0,02294 para a métrica *MAE*. O *LSTM* obteve uma pontuação de 0,02821, o segundo melhor.

3.2.3 Comparação entre Classificação e Regressão

As métricas utilizadas para as tarefas de regressão e classificação são diferentes, por isso muitos artigos tratam de apenas uma abordagem. Apesar disso, Ji *et al.* [15] comparam o retorno sobre o investimento seguindo cada uma das estratégias. Utilizam os preços históricos do Bitcoin e atributos de sua *blockchain* para realizar uma análise comparativa entre os modelos *Deep Neural Network* (DNN), *LSTM*, *CNN*, *Deep Residual Network*

(DRN), além de testarem uma combinação entre eles. O modelo *LSTM* se mostrou ligeiramente superior para a tarefa de regressão, enquanto o *DNN* para a classificação binária (subir ou descer). Os autores também relacionam o retorno financeiro, caso a análise do algoritmo fosse seguida como investimento. Nela, o modelo de classificação se mostrou mais assertivo. No projeto, consideraram a série histórica de preços do Bitcoin, bem como 29 atributos de sua *blockchain*. Após uma seleção das mais relevantes para o problema, utilizaram 18 para prever o preço do dia seguinte. Também foi aplicada a técnica de *5-fold cross validation*.

3.3 Análise do Ethereum

Como os artigos que buscam prever o preço do Bitcoin utilizam seus dados históricos para a análise, todos são criados se baseando no mesmo banco de dados. A fim de estudar outros cenários, mas ainda no contexto similar, é interessante estudar sobre o Ethereum, que é uma moeda digital similar ao Bitcoin e foi criada por Vitalik Buterin em 2015¹. Estudiosos buscaram também prever o preço futuro da moeda utilizando diferentes abordagens, como Tejaswi *et al.* [6] que utilizaram quatro modelos de aprendizado de máquina para o problema: *RFR*, *SVR*, *KNN*, *XGBoost*; e três de *deep learning*: *LSTM*, *GRU*, *GRU* em conjunto com *LSTM*. Com os dados históricos de preços e volatilidade de 2315 dias, que representam de agosto 2015 a dezembro de 2021, realizou-se o pré-processamento, treino, avaliação dos modelos e posterior comparação entre eles, a fim de encontrar o que é o mais adequado. O *LSTM* foi o modelo de *deep learning* com o melhor resultado para as métricas *RMSE*, *MSE*, *MAE* e *R2 score*, atingindo os valores 253,683; 64355,564; 115,908; 0,965, respectivamente, para o conjunto de teste.

Outra abordagem também utilizada para auxiliar a tarefa de predição do valor de ativos no mercado é a análise de sentimentos. Arezooji [23], busca analisar a *blockchain* do Ethereum, seu histórico de preços e as tendências de pesquisa no Google (*Google Trends*) a fim de encontrar relações que podem indicar manipulação de mercado e criação de bolhas. O trabalho analisa as movimentações de transações e de contratos na rede *blockchain*, revelando uma possível tendência. Então ela é comparada com o volume de pesquisas *online* pela moeda Ethereum e pelo Bitcoin, a fim de gerar uma variável de interesse público. Após uma análise do preço e das tendências geradas, foi identificado possíveis momentos em que há agentes com muito dinheiro investido e com possíveis informações internas que podem estar manipulando o mercado após uma queda de preços. Por fim, é realizado um estudo de correlação cruzada entre os preços da moeda Ethereum e das tendências analisadas, para prever a movimentação futura dos preços. O artigo

¹ethereum.org

conclui que a fim de realizar boas predições, é interessante considerar aspectos financeiros e sociais, além de que a abordagem proposta é positiva para estimar a criação de bolhas e manipulação de mercado.

3.4 Considerações Finais

A fim de facilitar o entendimento, a Tabela 3.1 apresenta um resumo dos trabalhos analisados neste capítulo. A partir do estudo desses trabalhos que incorporam diferentes abordagens para a tarefa de predição de ativos, é possível tomar algumas conclusões. Primeiramente, apesar de não existir um consenso na literatura, o modelo *LSTM* demonstra ser eficiente para o problema em questão. Isso também foi observado pelo trabalho de resumo de Verma *et al.* [5] que, após analisarem 50 artigos entre 2018 a 2021 relacionados à predição do preço de ativos no mercado financeiro, entenderam que os modelos de *deep learning*, como *LSTM* e *SVM*, são os mais populares. Também, é possível entender que uma abordagem interessante é a adição e escolha de quais atributos do ativo são mais relevantes para o modelo, no caso do bitcoin, é positivo analisar dados dos preços históricos, da macroeconomia, de sentimento de mercado e da rede *blockchain*. Por fim, em relação às métricas de desempenho, *MSE*, *RMSE*, *MAE* e *MAPE* são mais utilizados para problemas de regressão, enquanto Acurácia, Precisão, Recall, F-score e *AUC* para problemas de classificação [5] [7].

Então, este trabalho apresenta uma nova abordagem de comparação entre modelos de classificação e de regressão, que utiliza a variação predita pelo resultado do modelo de regressão para criar classes e assim obter métricas de classificação. Assim, é possível comparar o desempenho das duas abordagens. Além disso, este trabalho analisa o impacto que os parâmetros dia predito, jane la de tempo e limite têm nos resultados dos modelos criados, utilizando uma classificação em três classes, diferente de muitos estudos que utilizam apenas duas. Também são analisados os melhores hiperparâmetros, a fim de entender se há alguma configuração que seja consistentemente vantajosa, para os cenários testados.

Tabela 3.1: Trabalhos abordados neste capítulo.

Ano de publicação	Autores	Técnica de Aprendizado de Máquina	Abordagem	Resultados
2016	Ming-Wei <i>et al.</i> [20]	SVM, ANN e AR	Índices do mercado financeiro. Classificação binária e Regressão. Dados histórico e indicadores técnicos	Melhor modelo foi o SVM
2020	Nabipour <i>et al.</i> [3]	Árvore de Decisão, RF, XGBoost, SVC, <i>Naïve Bayes</i> , KNN, LR, ANN, RNN e LSTM	Ativos da bolsa de valores. Classificação binária, indicadores técnicos contínuos e binários.	Indicadores binários: 83% de acurácia RNN e LSTM: <i>F1-score</i> de 90%, acurácia de 86%
2022	Helfenstein, Rodolfo [21]	LSTM	Bitcoin. Classificação binária.	LSTM: Acurácia: 52,9% ROI: 122,82%
2022	Kanji <i>et al.</i> [4]	<i>LR</i> , <i>SVM</i> , <i>LSTM</i> , <i>GRU</i> , <i>LSTM-GRU</i> <i>SVM-GRU</i>	Bitcoin. Análise de sentimento. Classificação	<i>SVM-LSTM</i> e <i>SVM</i> : melhor revocação (60%). <i>SVM-GRU</i> : melhor acurácia (60,86%) e precisão (60%).
2022	Wang e Yan [7]	LSTM, GRU.	Classificação ternária. Varia limite, dia predito e janela de tempo. Usa dados históricos diários do Bitcoin e a taxa de mudança.	LSTM: 0,71% AUC.
2017	Jang e Lee [8]	BNN, LR, SVR.	Regressão do <i>Log</i> do preço e da volatilidade do Bitcoin. Dados históricos, da <i>blockchain</i> . Validação cruzada.	BNN: 0,0138 MAPE e 0,0039 RMSE para o <i>Log</i> do preço 0,5090 de MAPE e 0,2546 de RMSE para o log da volatilidade.
2022	Malhotra <i>et al.</i> [22]	Regressão linear, regressão Ridge, regressão LASSO, <i>LSTM</i> , <i>GRU</i> , <i>ARIMA</i> .	Regressão. Dados históricos do Bitcoin. Quatro meses de análise.	<i>ARIMA</i> : 0,02294 de <i>MAE</i> .
2019	Ji <i>et al.</i> [15]	<i>DNN</i> , <i>LSTM</i> , <i>CNN</i> , <i>DRN</i> .	Compara regressão e classificação binária por ROI. Dados históricos. Dados da <i>blockchain</i> . Validação Cruzada.	Classificação se mostrou mais rentável O melhor foi o DNN obteve acurácia de 53.06%, mas os outros modelos foram similares.
2022	Tejaswi <i>et al.</i> [6]	<i>RFR</i> , <i>SVR</i> , <i>KNN</i> , XGBoost, <i>LSTM</i> , <i>GRU</i> , <i>GRU-LSTM</i> .	Regressão. Dados históricos de preços e volatilidade do Ethereum.	<i>LSTM</i> :; <i>RMSE</i> : 253,683; <i>MSE</i> : 64355,564; <i>MAE</i> : 115,908; <i>R</i> ² : 0,965.
2021	Arezooji <i>et al.</i> [23]	Google Trends	Dados históricos de preços e da <i>blockchain</i> de Ethereum. Tendências no Google.	A análise auxilia na detecção de movimentações anormais, que podem indicar manipulação de preços ou bolhas.

Capítulo 4

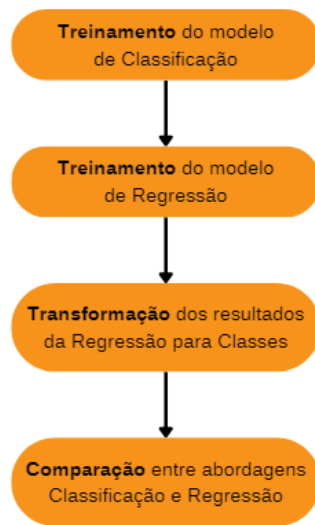
Metodologia

Neste capítulo, são detalhados os procedimentos empregados para conduzir os experimentos propostos neste trabalho. Esses procedimentos envolvem a previsão do preço futuro de negociação do Bitcoin, utilizando técnicas de regressão e de classificação. Além disso, exploram-se variações nos hiperparâmetros, do dia previsto, da janela de tempo e do limite. A comparação entre as técnicas é realizada para analisar qual deles apresenta o melhor desempenho na tarefa de predição durante o período de tempo estudado.

Para este trabalho, a Seção 4.1 aborda com mais detalhes a base de dados utilizada, que consiste em dados históricos do preço do Bitcoin e com seu volume de negociação em dólares. Já a Seção 4.2 demonstra como os dados foram preparados a fim de gerar rótulos e organizá-los para o uso do modelo LSTM. Também é apresentada a divisão da classificação, que é de 3 rótulos baseados na variação percentual predita pelo modelo, ou seja, se o modelo inferiu que o preço iria variar mais que um valor de limite superior, menos que um limite inferior ou ficar entre o limite superior e inferior. Nas Seções 4.3 e 4.4 há a explicação dos modelos de classificação e regressão implementados. Na Seção 4.5 há a comparação entre os modelos criados, como o desempenho de uma tarefa de regressão e de classificação não podem ser comparados diretamente, é necessário realizar uma alteração para compará-los. Ji *et al.* [15] utilizam o retorno sobre o investimento para comparar os modelos, mas este trabalho busca uma outra abordagem, que consiste em calcular, a partir do valor real estimado pela regressão, qual seria a variação percentual estimada e classificar de acordo com os rótulos anteriormente descritos. Por fim, a Seção 4.6 aborda sobre as métricas de desempenho utilizadas para avaliar os modelos de classificação e regressão, bem como a comparação entre eles.

A Figura 4.1 representa as etapas necessárias para a criação e aprimoramento dos modelos de classificação e regressão, bem como a análise de métricas para eles. Nela, há a etapa de treinamento do modelo de classificação e regressão, que seguem as etapas do fluxo de treinamento. É apresentada também a transformação dos resultados de Regressão

Etapas do Projeto



Treinamento

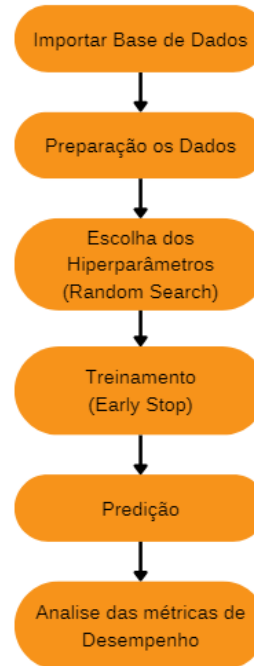


Figura 4.1: Esquema representativo das etapas abordadas no trabalho.

para classes, que é uma tarefa necessária para a futura comparação entre as abordagens.

4.1 Base de Dados

O Bitcoin é um ativo que é por natureza descentralizado, portanto não há uma base de dados oficial com todas as negociações de Bitcoins por dólares. No entanto, há dados oficiais de corretoras centralizadas que disponibilizam essas informações no mesmo formato que os ativos tradicionais do mercado financeiro. No caso deste trabalho, é utilizada a base de dados do Yahoo Finance [24]¹. Nela é possível acessar dados desde o dia 16/09/2014. Vale pontuar que datas muito anteriores à essa não possuem volume de negociações muito relevante, haja vista que o criptoativo vem ganhado cada vez mais conhecimento do público geral desde sua criação. Para esse trabalho, são utilizados dados de 16/09/2014 até 31/10/2023.

¹finance.yahoo.com

Date	Open	High	Low	Close	Volume
2014-09-17	465.864014	468.174011	452.421997	457.334015	21056800
2014-09-18	456.859985	456.859985	413.104004	424.440002	34483200
2014-09-19	424.102997	427.834991	384.532013	394.795990	37919700
2014-09-20	394.673004	423.295990	389.882996	408.903992	36863600
2014-09-21	408.084991	412.425995	393.181000	398.821014	26580100

Figura 4.2: Primeiros 5 elementos da base de dados.

Date	Open	High	Low	Close	Volume
2023-10-27	34156.500000	34238.210938	33416.886719	33909.800781	16418032871
2023-10-28	33907.722656	34399.390625	33874.804688	34089.574219	10160330825
2023-10-29	34089.371094	34743.261719	33947.566406	34538.480469	11160323986
2023-10-30	34531.742188	34843.933594	34110.972656	34502.363281	17184860315
2023-10-31	34500.078125	34719.253906	34083.308594	34667.781250	15758270810

Figura 4.3: Últimos 5 elementos da base de dados.

Formato dos Dados

A base de dados utilizada são de 3332 linhas que representam os dias de negociação e 5 colunas que representam o preço de abertura (*open*), o preço de fechamento (*close*), o preço máximo negociado no dia (*high*), o preço mínimo no dia (*low*) e o volume total de negociações (*volume*). Todos os valores estão em dólares. A Figura 4.2 representa os primeiros 5 elementos da base de dados e a Figura 4.3, os últimos 5 elementos.

4.2 Preparação dos Dados

Nesta etapa são definidas as tarefas necessárias para preparar os dados da base escolhida para serem utilizados pelo modelo de aprendizado de máquina LSTM. Também é apresentado um exemplo dos dados gerados utilizando 30 dias de janela de tempo, 1 dia predito no futuro e 3% de limite. Nela são apresentados possíveis rótulos gerados para o conjunto dos 30 dias anteriores.

Séries Temporais

O problema de predição do preço futuro do Bitcoin utilizando LSTM necessita que os dados sejam agrupados, de forma que um número específico de dias seja utilizado para prever o dia futuro. A Seção 2.1.2 aborda sobre séries temporais. Neste trabalho, o de dias anteriores é chamado de **janela de tempo** ou *timestep*. Para os experimentos serão utilizados os valores arbitrários de 60 dias, 30, 15 ou 7. Já o dia a ser previsto no futuro ou *future date* é alterado entre o dia seguinte ($t + 1$), a semana seguinte ($t + 7$) ou até o mês seguinte ($t + 30$). Neste trabalho, ele é denominado **dia previsto**.

Um exemplo do processo realizado para o agrupamento dos dados é apresentado na Equação 4.1, que contém a definição das séries temporais ($f()$) e os resultados esperados na saída (x_6, x_7, x_8). Esse exemplo utiliza uma janela de tempo de 5 dias e 1 do dia previsto no futuro.

$$\begin{cases} x_6 = f(x_5, x_4, x_3, x_2, x_1) \\ x_7 = f(x_6, x_5, x_4, x_3, x_2) \\ x_8 = f(x_7, x_6, x_5, x_4, x_3) \end{cases} \quad (4.1)$$

Definição dos Rótulos

A tarefa de **regressão** se resume a utilizar a janela de tempo para prever o preço de fechamento (*close*) do dia futuro (x_{t+h}). Como a base de dados já possui esses valores, a definição de rótulos se resume a selecionar a data esperada real e agrupá-la à sua janela de tempo. Já para o problema de **classificação**, é necessário calcular a variação percentual do último dia da série (x_t) até o dia futuro (x_{t+h}). Então, é necessário escolher um limite superior e inferior a fim de criar as três classes que são abordadas no trabalho e exemplificadas na Equação 4.2. Nela, é possível observar que a classe 2 representa um aumento de preço acima do limite, a 1 representa que o preço permaneceu entre os dois limites e a 0 significa que diminuiu mais que o limite inferior. Os limiares 3%, 5% e 7% foram escolhidos de forma arbitrária para testar o modelo criado, baseados no trabalho de Wang e Yan [7]. Neste trabalho ele será tratado como **limite** ou *threshold*.

$$\begin{cases} 2 & \text{se } \frac{x_{t+h}-x_t}{x_t} > \text{limite} \\ 1 & \text{se } -\text{limite} < \frac{x_{t+h}-x_t}{x_t} < \text{limite} \\ 0 & \text{se } \frac{x_{t+h}-x_t}{x_t} < -\text{limite} \end{cases} \quad (4.2)$$

Como o Bitcoin é um ativo que se valorizou de forma expressiva ao longo dos anos, analisar o preço de forma nominal pode causar distorções ao modelo. Como o treinamento, validação e teste são realizados de forma sequencial, os valores do treinamento

	YClose	YLogClose	YChange	YVariation	YClass
2014-10-17	383.757996	2.584057	1.201996	0.314201	1
2014-10-18	391.441986	2.592667	7.683990	2.002301	1
2014-10-19	389.545990	2.590559	-1.895996	-0.484362	1
2014-10-20	382.845001	2.583023	-6.700989	-1.720205	1
2014-10-21	386.475006	2.587121	3.630005	0.948166	1

Figura 4.4: Dados gerados a partir da variação entre os dias da base de dados.

são historicamente muito menores do que os da validação e do teste. A fim de minimizar essa distorção, é utilizado o Log base 10 do preço de fechamento como rótulo da tarefa de regressão, similar ao trabalho de Jang e Lee [8].

Outro desafio é em relação às amostras da base de dados. É possível aumentar a quantidade delas, ao analisar em períodos de tempo menores ou maiores (segundos, semanas ou meses) porém os dados passados são fixos e não é possível gerar novos. Dependem do preço real de movimentação do Bitcoin. Como ele teve uma valorização expressiva ao longo dos anos, pode ser um fator que influencie o modelo. Além disso, ao definir um limite, automaticamente se modificam as quantidades de amostras. Quanto maior o limite, maior é o conjunto 1 em relação ao 2 e 0. Esse desbalanceamento de classes deve ser considerado na avaliação do modelo.

Os rótulos gerados estão exemplificados na Figura 4.4, em que YClose que representa o futuro preço de fechamento; YLogClose, o log base 10 do futuro preço de fechamento; YChange, a futura variação nominal entre o último dia do conjunto e o dia predito; YVariation, a variação percentual entre esses dias; YClass a classe de acordo com a Equação 4.2, utilizando um limite de 3%.

Divisão da Base de Dados

Então, o conjunto foi dividido de forma sequencial em treino, validação e teste, de forma que após o agrupamento dos dados de acordo com a janela de tempo o resultado esteja na proporção de 70%, 10% e 20%. A quantidade de elementos de cada grupo varia dependendo do cenário testado, mas a proporção se mantém. Essa divisão está representada no gráfico do preço de fechamento na Figura 4.5.

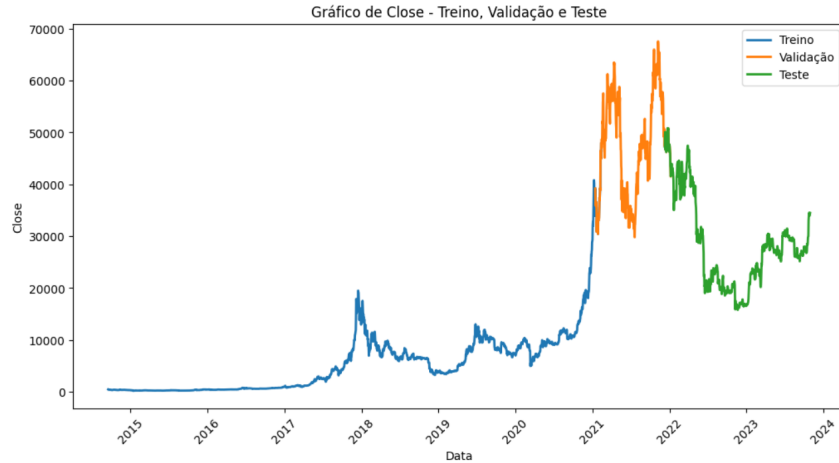


Figura 4.5: Gráfico com o preço de fechamento real do conjunto de treino, validação e teste pelo tempo.

Normalização dos Dados

Os dados de treino são normalizados de forma separada e os pesos definidos são utilizados nos conjuntos de validação e teste. Dessa forma o treino não é influenciado pelos outros conjuntos. Um exemplo dos dados normalizados está representados na Figura 4.6.

Agrupamento dos Dados

A fim de utilizar o modelo LSTM, é necessário agrupar o conjunto de dados de acordo com a janela de tempo (*timestep*) definida. Para um cenário com um *timestep* de 30 dias, 5 parâmetros (*Low*, *High*, *Open*, *Clone*, *Volume*) e 2270 conjuntos de 30 amostras, o formato dos dados de entrada ($x_{train.shape}$) é similar a (2270, 60, 5); já os dados da saída esperada ($y_{train.shape}$) têm um formato similar a (2270, 1).

4.3 Classificação

A tarefa de classificação busca utilizar os rótulos definidos baseados no limite para classificar se o preço de fechamento do Bitcoin no dia futuro escolhido subiu mais que o limite superior, caiu mais que o limite inferior ou ficou entre ambos os limites. A Equação 4.2 demonstra as classes subiu (2), lateral (1) e caiu (0). Para esta etapa, são testados 18 cenários distintos, apresentados na Tabela 4.1, variando os parâmetros de janela de tempo em 15, 30 e 60 dias; Dia predito no futuro em 1 e 7; Limite em 3%, 5% e 7%.

Dados de Treino (5 primeiros valores):					
	Open	High	Low	Close	Volume
0	0.006242	0.005315	0.006243	0.005850	0.000123
1	0.006048	0.005080	0.005369	0.005161	0.000232
2	0.005340	0.004479	0.004734	0.004540	0.000260
3	0.004704	0.004385	0.004853	0.004835	0.000251
4	0.004994	0.004159	0.004927	0.004624	0.000168
Dados de Validação (5 primeiros valores):					
	Open	High	Low	Close	Volume
0	0.802461	0.823901	0.815573	0.817268	0.515835
1	0.842026	0.815846	0.766479	0.767784	0.549447
2	0.791585	0.780337	0.788125	0.754224	0.467911
3	0.777371	0.756670	0.753361	0.746119	0.424556
4	0.769348	0.768626	0.771463	0.763692	0.401459
Dados de Teste (5 primeiros valores):					
	Open	High	Low	Close	Volume
0	1.025325	1.032160	1.041265	0.986046	0.251066
1	1.017170	1.020616	1.039457	1.030445	0.208977
2	1.062323	1.046867	1.079094	1.045861	0.177865
3	1.078737	1.036093	1.016177	0.975449	0.260803
4	1.005186	0.999335	1.027948	0.972833	0.280848

Figura 4.6: Figura representativa dos dados de treino, validação e teste.

Tabela 4.1: Cenários testados para a Classificação.

Número do Experimento	Janela de Tempo <i>Timestep</i>	Dia predito <i>Future date</i>	Limite <i>Threshold</i>
1	30	1	3%
2	30	1	5%
3	30	1	7%
4	15	1	3%
5	15	1	5%
6	15	1	7%
7	30	7	3%
8	30	7	5%
9	30	7	7%
10	15	7	3%
11	15	7	5%
12	15	7	7%
13	60	1	3%
14	60	1	5%
15	60	1	7%
16	60	7	3%
17	60	7	5%
18	60	7	7%

Tabela 4.2: Configurações de Rede Neural para Classificação.

Configuração	Valores
Células em cada camada	16 a 128, com incrementos de 16
Camada de <i>dropout</i>	Valor entre 0 e 0,4
Taxa de aprendizado (Adam)	Entre 0,001 e 0,1

Seleção dos Hiperparâmetros

Em cada cenário avaliado, a abordagem de busca aleatória, conhecida como *Random Search*, foi empregada para selecionar os melhores parâmetros, conforme detalhado na Tabela 4.2. Isso demanda a importação dos dados, a etapa de preparação, a seleção de rótulos e um estágio de pré-treinamento para identificar os hiperparâmetros que resultaram na menor perda, considerando o cenário em análise. Foram testados 100 modelos, cada um treinado ao longo de 20 épocas para cada cenário. O modelo LSTM foi configurado com uso de uma camada, a função de ativação *softmax*, a função de perda (*loss*) `sparse_categorical_crossentropy` e otimizador Adam.

Treinamento Definitivo

Após a seleção do melhor conjunto de hiperparâmetros para o cenário definido, é necessário treinar o modelo mais vezes a fim de aprimorá-lo. Neste trabalho, foi utilizado o *EarlyStop*, ou parada antecipada, para salvar o melhor estado do modelo em um máximo de 500 épocas. O parâmetro de *patience* foi utilizado com o valor 100, ou seja, se não melhorar em 100 épocas, o treinamento é parado.

4.4 Regressão

A tarefa de regressão busca utilizar o *log* base 10 do preço de fechamento real do dia previsto como rótulo para treinar cada conjunto de uma série temporal. O resultado da previsão é uma estimativa de qual seria o *log* preço de fechamento do Bitcoin. A partir dele é possível encontrar o preço de fechamento estimado. Para esta etapa, são testados 12 cenários distintos, apresentados na Tabela 4.3. Foram variados os parâmetro da janela de tempo em 7, 15, 30 e 60 dias; Dia predito no futuro em 1, 7 e 30.

Seleção dos Hiperparâmetros

Em cada cenário avaliado, a abordagem de busca aleatória, conhecida como *Random Search*, foi empregada para selecionar os melhores parâmetros, conforme detalhado na Tabela 4.4. Isso demanda a importação dos dados, a etapa de preparação, a seleção de

Tabela 4.3: Cenários testados para a Regressão.

Número do Experimento	Janela de Tempo <i>Timestep</i>	Dia predito <i>Future date</i>
1	30	1
2	30	7
3	30	30
4	15	1
5	15	7
6	15	30
7	7	1
8	7	7
9	7	30
10	60	1
11	60	7
12	60	30

Tabela 4.4: Configurações de Rede Neural para Regressão.

Configuração	Valores
Células em cada camada	16 a 128, com incrementos de 16
Camada de <i>dropout</i>	Valor entre 0 e 0,4
Taxa de aprendizado (Adam)	Entre 0,001 e 0,1
Função de ativação	'tanh' ou 'relu'

rótulos e um estágio de pré-treinamento para identificar os hiperparâmetros que resultaram na menor perda, considerando o cenário em análise. Foram testados 100 modelos, cada um treinado ao longo de 25 épocas para cada cenário. O modelo LSTM foi configurado com uso de uma camada, a função de perda (*loss*) *mean_squared_error* e otimizador Adam.

Treinamento Definitivo

Após a seleção do melhor conjunto de hiperparâmetros para o cenário definido, é necessário treinar o modelo mais vezes a fim de aprimorá-lo. Neste trabalho, foi utilizado o *EarlyStop*, ou parada antecipada, para salvar o melhor estado do modelo em um máximo de 500 épocas. O parâmetro de *patience* foi utilizado com o valor 100, ou seja, se não melhorar em 100 épocas, o treinamento é parado.

4.5 Comparação entre a Classificação e a Regressão

As métricas de desempenho dos modelos de classificação e regressão são diferentes e não há como compará-los diretamente. Neste trabalho, o resultado da regressão foi transformado em classes para gerar métricas de classificação para uma comparação. Como a regressão estima o log do preço de fechamento no dia futuro analisado, é possível elevar 10 ao número estimado ($10^{X_{t+h}}$) a fim de encontrar o preço de fechamento estimado. Com esse valor, é possível medir a variação percentual estimada pelo modelo, ou seja, se o valor anterior (x_t) era de 10 dólares e o modelo estima que no dia $t+h$ será 15 dólares, significa que a variação percentual estimada é de +50%. A partir da variação e da definição de um limite, é possível definir as classes da mesma forma que o modelo de classificação, que está representado na Equação 4.2. A definição das classes do modelo, permite que sejam geradas as métricas de classificação, levando em consideração a variação real contra a variação estimada, ambos em classes. Por fim, é possível comparar as duas abordagens, regressão e classificação.

4.6 Análise dos Resultados

A avaliação da classificação foi conduzida usando métricas de desempenho como Acurácia, Precisão média, *Recall* médio, *F1-Score* médio e AUC médio geradas a partir do conjunto de teste. As médias foram calculadas analisando os resultados de forma agregada, tratando um conjunto global de predições verdadeiras positivas, falsos positivos e falsos negativos. Já os resultados da previsão dos modelos de regressão foram desnormalizados antes da análise das métricas de desempenho MAE, MSE, RMSE, MAPE, R^2 , que estão apresentadas na Seção 2.5.1.

A comparação entre os cenários criados para cada tarefa foi realizada pela ótica geral e pela ótica específica. A primeira se baseia em uma observação dos resultados dos cenários de forma geral, já a segunda, na análise do impacto das variáveis limite, janela de tempo e dia predito nos resultados. Então é analisada a comparação entre os resultados da tarefa de classificação e da tarefa de regressão, após sua transformação para classes. Para isso, são utilizadas as métricas da etapa de classificação a fim de encontrar qual é a tarefa que obteve o melhor desempenho de forma geral.

Capítulo 5

Resultados Experimentais

Este capítulo detalha as informações relativas à cada experimento realizado. São abordados os resultados e análises sobre o conjunto de dados utilizado, a normalização, escolha dos melhores hiperparâmetros e a avaliação dos modelos preditivos. Os experimentos foram divididos de acordo com a tarefa, classificação ou regressão.

Ambiente de Desenvolvimento

Os experimentos foram realizados através da infraestrutura online do Google Colab¹, que usa o *framework* Jupyter². Ele permite a escrita e execução código Python pelo navegador. É ideal para aprendizado de máquina, análise de dados. Não depende de configuração específica da máquina do usuário e nem de instalação de programas. As métricas e a normalização dos dados foi feita utilizando as funções disponibilizadas pela biblioteca *sklearn*³. Já o código criado está disponível no github⁴, junto com mais informações sobre cada experimento de forma específica.

Preparação dos Dados

A preparação dos dados seguiu o estipulado na Seção 4.2. Foi realizado o agrupamento dos dados em séries temporais, com os rótulos definidos de acordo com a tarefa (classificação ou regressão), foi efetuada a divisão da base de dados, a normalização dos dados e o agrupamento de acordo com o cenário a ser testado.

¹colab.research.google.com

²jupyter.org

³<https://scikit-learn.org/>

⁴github.com/Brunocmarino/TCC-2023

Tabela 5.1: Número de amostras de cada classe para cada cenário de Classificação

Cenário	Queda (0)	Manutenção (1)	Alta (2)
1	414	2401	487
2	212	2854	236
3	98	3077	127
4	417	2409	491
5	213	2868	236
6	99	3091	127
7	913	1162	1221
8	683	1668	945
9	502	2044	750
10	915	1167	1229
11	684	1675	952
12	503	2052	756
13	408	2379	485
14	208	2830	234
15	98	3048	126
16	897	1158	1211
17	668	1663	935
18	491	2032	743

5.1 Classificação da Tendência do Bitcoin

Esta seção detalha sobre os experimentos realizados para a tarefa de classificação, descrita na Seção 4.3. Nela está especificado também os resultados da escolha dos melhores hiperparâmetros *Random Search* e do treinamento utilizando *EarlyStop*. A Seção 4.3 contém a metodologia aplicada, incluindo a configuração utilizada para o treinamento de cada cenário.

Cenários Avaliados

A Tabela 5.1 contém a quantidade de amostras de cada classe para cada um dos 18 cenários testados, que são apresentados na Seção 4.3 na Tabela 4.1. A classe 0 é apresentada como **Queda** do preço para baixo do limite inferior, a classe 1 representa **Manutenção** da faixa de preço, já a classe 2 representa a **Alta** do preço acima do limite superior. É possível observar que a escolha de parâmetros influencia diretamente na quantidade de elementos de cada classe.

Tabela 5.2: Resultado da escolha dos hiperparâmetros com *Random Search*.

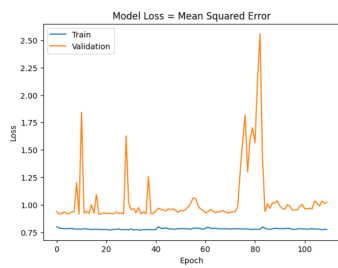
Cenário	Núm. Células	<i>Dropout</i>	Taxa de aprendizado	<i>Loss</i>
1	32	0.22547	0.07191	0.60366
2	16	0.28848	0.04773	0.90897
3	16	0.12556	0.06224	0.34820
4	32	0.33636	0.03373	0.91558
5	32	0.19440	0.08961	0.60597
6	16	0.16830	0.07964	0.35728
7	80	0.28050	0.09275	1.08890
8	128	0.32843	0.07047	1.07006
9	112	0.13379	0.02468	1.01053
10	128	0.38047	0.04484	1.08360
11	96	0.23029	0.01622	1.07721
12	64	0.06110	0.06985	1.00937
13	112	0.24092	0.08976	0.90945
14	112	0.23585	0.09277	0.59163
15	16	0.18331	0.00152	0.34193
16	32	0.25371	0.01289	1.07369
17	112	0.15800	0.00818	1.07501
18	112	0.36054	0.01655	1.00335

Avaliação Hiperparâmetros para classificação

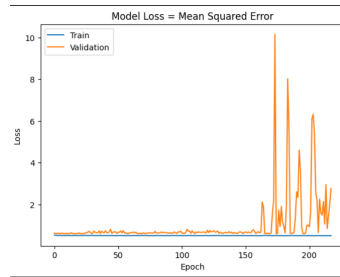
A fim de definir os melhores hiperparâmetros para cada cenário testado, foram testados 100 modelos com 20 épocas cada e escolhidas as melhores combinações para prosseguir com o experimento. Elas estão disponíveis na Tabela 5.2. A partir da análise dos melhores cenários e seus hiperparâmetros, pode-se perceber que o melhor número de camadas variou de 16 até 128, que é o máximo estipulado no teste. Além disso, o valor que mais se repetiu foi 112, com 5 ocorrências, depois 16 e 32, com 4, e então 128, com duas, 80, 96 e 64 com uma ocorrência cada. Em relação à taxa de *dropout*, o menor valor foi 0.06110 e o maior, 0.38047. Já, para a taxa de aprendizado, o menor valor foi 0.00152 e o maior, 0.09277.

Análise dos Gráficos de Perda por Época

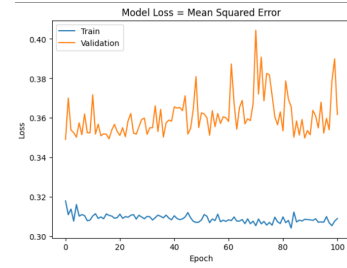
Após o treinamento de cada cenário por até 500 épocas com um *EarlyStop* com *patience* de 100, foram gerados gráficos de perda (*loss*) por épocas, representados pelas Figuras 5.1 e 5.2. É possível observar que não há um padrão entre eles. Há cenários que reduzem a perda ao longo das épocas, como o 17, mas também há aqueles que aumentam, como o 7, 9, 12, 11, 12, 15. Também há cenários que permanecem com uma perda relativamente estável, como o 3, 5, 6.



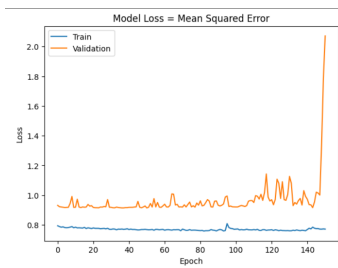
(a) Cenário 1



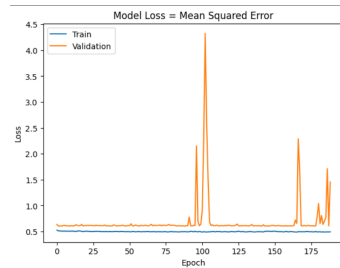
(b) Cenário 2



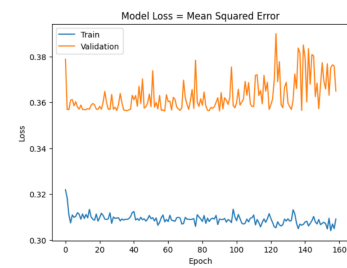
(c) Cenário 3



(d) Cenário 4



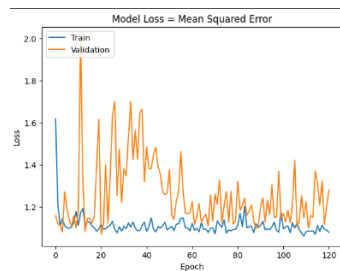
(e) Cenário 5



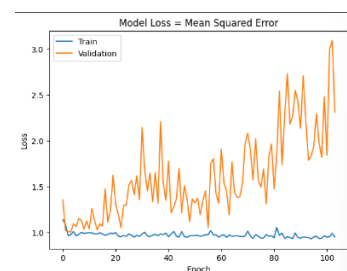
(f) Cenário 6



(g) Cenário 7



(h) Cenário 8

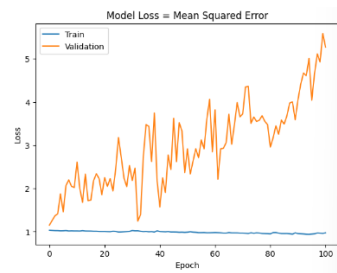


(i) Cenário 9

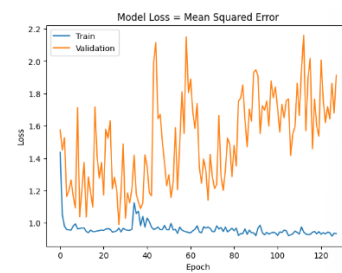
Figura 5.1: Gráficos de perda por época de cada cenário de classificação (1 a 9).



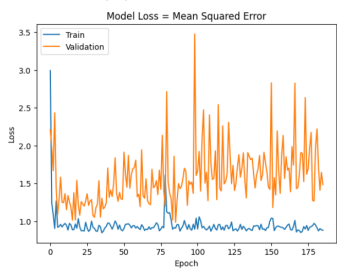
(a) Cenário 10



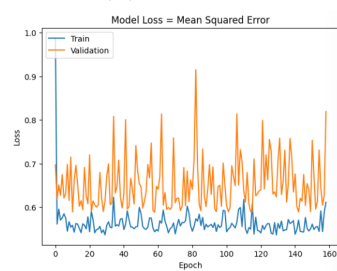
(b) Cenário 11



(c) Cenário 12



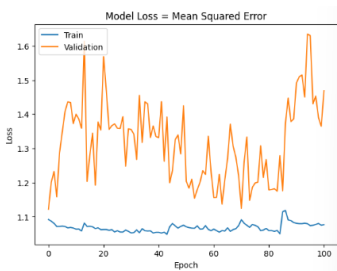
(d) Cenário 13



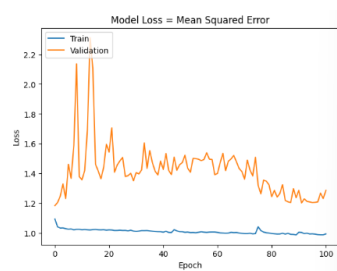
(e) Cenário 14



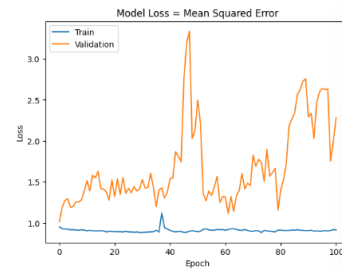
(f) Cenário 15



(g) Cenário 16



(h) Cenário 17



(i) Cenário 18

Figura 5.2: Gráficos de perda por época de cada cenário de classificação (10 a 18).

Tabela 5.3: Resultado da escolha dos hiperparâmetros com *Random Search* para a regressão.

Cenário	Núm. Células	<i>Dropout</i>	Taxa de aprendizado	Ativação	<i>Loss</i>
1	16	0.09107	0.00405	tanh	0.00047
2	48	0.09547	0.03089	tanh	0.00186
3	96	0.16707	0.08065	tanh	0.00640
4	64	0.15907	0.03382	tanh	0.00046
5	48	0.10789	0.04053	tanh	0.00194
6	48	0.35809	0.07588	tanh	0.00610
7	96	0.06207	0.06404	tanh	0.00040
8	80	0.26517	0.03925	tanh	0.00205
9	48	0.22036	0.07573	tanh	0.00653
10	64	0.18449	0.02985	tanh	0.00034
11	48	0.25356	0.07598	tanh	0.00182
12	32	0.09553	0.04160	tanh	0.00617

5.2 Regressão do *Log* do Preço do Bitcoin

Esta seção detalha sobre os experimentos realizados para a tarefa de regressão. Nela está especificado também os resultados da escolha dos melhores hiperparâmetros *Random Search* e do treinamento utilizando *EarlyStop*. A Seção 4.4 contém a metodologia aplicada, incluindo a configuração utilizada para o treinamento de cada cenário.

Cenários Avaliados

A Tabela 4.3 apresenta os 12 cenários que são testados de acordo com a metodologia apresentada na Seção 4.4.

Avaliação dos Hiperparâmetros para Regressão

A fim de definir os melhores hiperparâmetros para cada cenário testado, foram testados 100 modelos com 25 épocas cada e escolhidas as melhores combinações para prosseguir com o experimento. Elas estão disponíveis na Tabela 5.3. A partir da análise dos melhores cenários e seus hiperparâmetros, pode-se perceber que o melhor número de camadas variou de 16 até 96. Além disso, o valor que mais se repetiu foi 48, com 5 ocorrências, depois 96 e 64, com duas, 16, 32 e 80 com uma ocorrência cada. Em relação à taxa de *dropout*, o menor valor foi 0.06207 e o maior, 0.35809. Já, para a taxa de aprendizado, o menor valor foi 0.00405 e o maior, 0.08065. Pode-se observar também que a função de ativação 'tanh' se mostrou ser mais adequada ao problema, se comparada com a 'relu', haja vista que nas melhores combinações de cada cenário ela esteve presente.

Gráficos de Perda por Época

Após o treinamento de cada cenário por até 500 épocas com *EarlyStop* com *patience*, foram gerados gráficos de perda (*loss*) por épocas, representados pela Figura 5.3. É possível observar que não há um padrão claro entre eles. Nos cenários 1, 2, 3, 4, 6, 8, 9 e 10 a perda começou alta nos casos de treino, mas logo caíram para uma faixa que é próxima à perda do conjunto de validação. Já nos casos 5, 7 e 11 o conjunto de teste não obteve grandes erros no início do treinamento, se comparados às épocas subsequentes.

5.3 Transformação em Classes

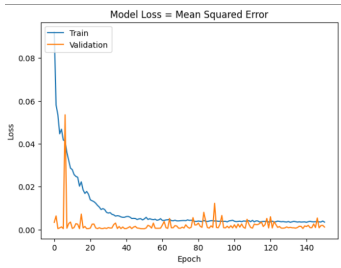
A fim de comparar os resultados do modelo de regressão com o de classificação, é necessário realizar os passos descritos na Seção 4.5. Após elevar 10 ao valor predito (log preço futuro), foi possível obter o valor nominal predito para o preço de fechamento futuro. Assim, a partir dele, é possível calcular como modelo previu que o preço se comportaria de acordo com as classes definidas na Seção 4.2. As Tabelas 5.4 e 5.5 contém a quantidade de amostras de cada classe para cada cenário de Regressão. A classe 0 é apresentada como Queda do preço para baixo do limite inferior, a classe 1 representa a Manutenção da faixa de preço, já a classe 2 representa a Alta do preço acima do limite superior.

5.4 Análise das Métricas de Desempenho

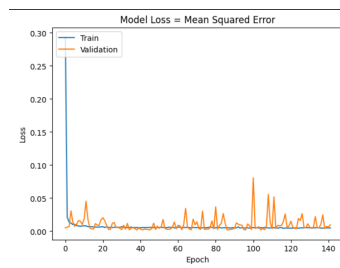
Esta seção analisa as métricas de desempenho para o conjunto de teste, descritas na Seção 4.6. São apresentadas as métricas da tarefa de classificação, na Seção 5.4.1 e de regressão na Seção 5.4.2. A Seção 5.4.3 aborda sobre os resultados da transformação da predição obtida pela Regressão em classes. Já a 5.4.4 compara os resultados obtidos pelas duas tarefas utilizando métricas de classificação.

5.4.1 Análise das Métricas de Classificação

A fim de analisar a influência que a alteração das variáveis dos cenários tem nas métricas de desempenho, são apresentadas três Tabelas que possuem os mesmos dados, mas são agrupados de forma facilitar a comparação dos resultados variando-se um dos parâmetros de cada experimento. A variação do limite é destacado na Tabela 5.6, já a alteração do dia predito é evidenciada na Tabela 5.7, a janela de tempo, por sua vez, é ressaltada na Tabela 5.8. Em negrito estão marcados os dados que apresentaram os melhores resultados para cada grupo. É importante observar que a quantidade de elementos de cada classe muda de acordo com o cenário e está representada pela Tabela 5.1.



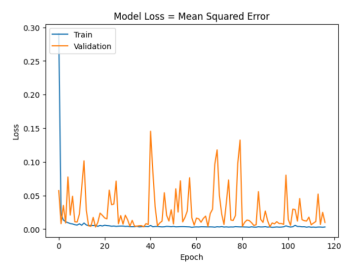
(a) Cenário 1



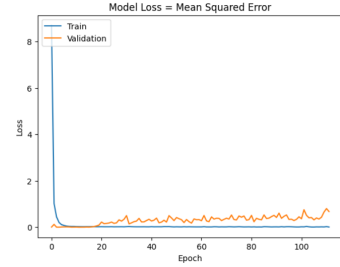
(b) Cenário 2



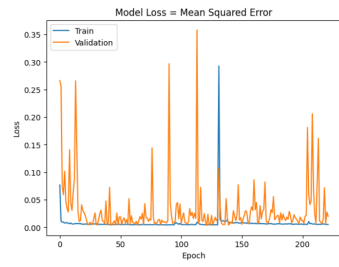
(c) Cenário 3



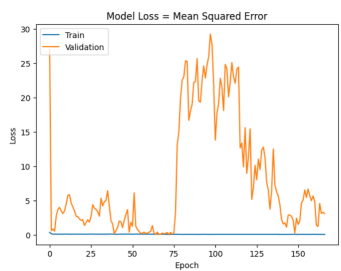
(d) Cenário 4



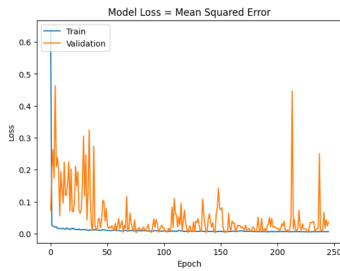
(e) Cenário 5



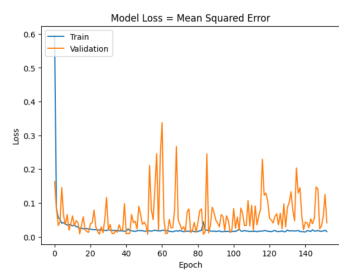
(f) Cenário 6



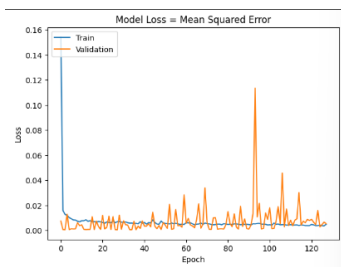
(g) Cenário 7



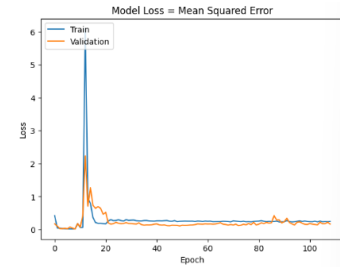
(h) Cenário 8



(i) Cenário 9



(j) Cenário 10



(k) Cenário 11



(l) Cenário 12

Figura 5.3: Gráficos de perda por época de cada cenários de regressão.

Tabela 5.4: Número de amostras de cada classe para cada cenário do modelo de Regressão (1 a 6).

Cenário	limite	Queda (0)	Manutenção (1)	Alta (2)
1	3%	414	2401	487
1	5%	212	2854	236
1	7%	98	3077	127
2	3%	913	1162	1221
2	5%	683	1668	945
2	7%	502	2044	750
3	3%	1192	437	1644
3	5%	1032	733	1508
3	7%	917	1004	1352
4	3%	417	2409	491
4	5%	213	2868	236
4	7%	99	3091	127
5	3%	915	1167	1229
5	5%	684	1675	952
5	7%	503	2052	756
6	3%	1196	444	1648
6	5%	1036	743	1509
6	7%	920	1015	1353

Tabela 5.5: Número de amostras de cada classe para cada cenário do modelo de Regressão (7 a 12).

Cenário	limite	Queda (0)	Manutenção (1)	Alta (2)
7	3%	418	2415	492
7	5%	214	2875	236
7	7%	99	3099	127
8	3%	923	1167	1229
8	5%	692	1675	952
8	7%	511	2052	756
9	3%	1204	444	1648
9	5%	1044	743	1509
9	7%	927	1016	1353
10	3%	408	2379	485
10	5%	208	2830	234
10	7%	98	3048	126
11	3%	897	1158	1211
11	5%	668	1663	935
11	7%	491	2032	743
12	3%	1183	431	1629
12	5%	1026	721	1496
12	7%	912	990	1341

Tabela 5.6: Resultado das métricas de Classificação agrupadas variando o limite.

Cenário	Acurácia	Precisão	revocação	<i>F1-Score</i>	AUC Médio
1	0.80635	0.80635	0.80635	0.80635	0.85476
2	0.91679	0.91679	0.91679	0.91679	0.93759
3	0.95763	0.95763	0.95763	0.95763	0.96822
4	0.80451	0.80451	0.80451	0.80451	0.85338
5	0.91578	0.91578	0.91578	0.91578	0.93684
6	0.95789	0.95789	0.95789	0.95789	0.96842
7	0.31969	0.31969	0.31969	0.31969	0.48977
8	0.62121	0.62121	0.62121	0.62121	0.71590
9	0.71818	0.71818	0.71818	0.71818	0.78863
10	0.27752	0.27752	0.27752	0.27752	0.45814
11	0.34992	0.34992	0.34992	0.34992	0.51244
12	0.66063	0.66063	0.66063	0.66063	0.74547
13	0.57099	0.57099	0.57099	0.57099	0.67824
14	0.91603	0.91603	0.91603	0.91603	0.93702
15	0.95725	0.95725	0.95725	0.95725	0.96793
16	0.27828	0.27828	0.27828	0.27828	0.45871
17	0.56422	0.56422	0.56422	0.56422	0.67316
18	0.71712	0.71712	0.71712	0.71712	0.78784

Análise do Limite

Ao analisar o limite, é possível perceber que sua escolha teve uma influência direta no desempenho do modelo. Conforme o limite aumenta, há mais dados da classe 1, que representam um mercado lateral. Isso promove um maior desbalanceamento de classes. É possível observar que todos os cenários com limite de 7% superaram os resultados das métricas que utilizam 5% que, por sua vez, superaram os de 3%, para uma janela de tempo de dia predito constantes.

Análise do Dia Predito

Ao analisar o dia predito, é possível perceber que sua escolha teve uma influência direta no desempenho do modelo. Conforme o dia predito aumenta, há menos dados da classe 1, que representam um mercado lateral. Isso promove um maior desbalanceamento de

Tabela 5.7: Resultado das métricas de Classificação agrupadas variando o dia predito.

Cenário	Acurácia	Precisão	revocação	<i>F1-Score</i>	AUC Médio
1	0.80635	0.80635	0.80635	0.80635	0.85476
7	0.31969	0.31969	0.31969	0.31969	0.48977
2	0.91679	0.91679	0.91679	0.91679	0.93759
8	0.62121	0.62121	0.62121	0.62121	0.71590
3	0.95763	0.95763	0.95763	0.95763	0.96822
9	0.71818	0.71818	0.71818	0.71818	0.78863
4	0.80451	0.80451	0.80451	0.80451	0.85338
10	0.27752	0.27752	0.27752	0.27752	0.45814
5	0.91578	0.91578	0.91578	0.91578	0.93684
11	0.34992	0.34992	0.34992	0.34992	0.51244
6	0.95789	0.95789	0.95789	0.95789	0.96842
12	0.66063	0.66063	0.66063	0.66063	0.74547
13	0.57099	0.57099	0.57099	0.57099	0.67824
16	0.27828	0.27828	0.27828	0.27828	0.45871
14	0.91603	0.91603	0.91603	0.916030	0.93702
17	0.56422	0.56422	0.56422	0.56422	0.67316
15	0.95725	0.95725	0.95725	0.95725	0.96793
18	0.71712	0.71712	0.71712	0.71712	0.78784

classes. É possível observar que todos os cenários com o dia predito em 1 superam os resultados das métricas que utilizam 7.

Análise da Janela de Tempo

Ao analisar a Janela de Tempo, os cenários 1, 2, 6, 7, 8, 9 obtiveram as melhores métricas. Em geral, aqueles que utilizam uma janela de tempo de 30 dias se mostraram ligeiramente superiores, a exceção é o cenário 6, que utilizou 15 dias e foi o melhor de todos. Também é relevante que os casos 1, 2, 7, 8 e 9 performaram melhor mesmo não possuindo a maior concentração de elementos na classe 1, o que foi observado na análise do limite e do dia predito.

Tabela 5.8: Resultado das métricas de Classificação agrupadas variando a janela de tempo.

Cenário	Acurácia	Precisão	revocação	<i>F1-Score</i>	AUC Médio
4	0.80451	0.80451	0.80451	0.80451	0.85338
1	0.80635	0.80635	0.80635	0.80635	0.85476
13	0.57099	0.57099	0.57099	0.57099	0.67824
5	0.91578	0.91578	0.91578	0.91578	0.93684
2	0.91679	0.91679	0.91679	0.91679	0.93759
14	0.91603	0.91603	0.91603	0.916030	0.93702
6	0.95789	0.95789	0.95789	0.95789	0.96842
3	0.95763	0.95763	0.95763	0.95763	0.96822
15	0.95725	0.95725	0.95725	0.95725	0.96793
10	0.27752	0.27752	0.27752	0.27752	0.45814
7	0.31969	0.31969	0.31969	0.31969	0.48977
16	0.27828	0.27828	0.27828	0.27828	0.45871
11	0.34992	0.34992	0.34992	0.34992	0.51244
8	0.62121	0.62121	0.62121	0.62121	0.71590
17	0.56422	0.56422	0.56422	0.56422	0.67316
12	0.66063	0.66063	0.66063	0.66063	0.74547
9	0.71818	0.71818	0.71818	0.71818	0.78863
18	0.71712	0.71712	0.71712	0.71712	0.78784

Análise Geral

Os cenários 3, 6 e 15 foram os que obtiveram os melhores resultados nas métricas gerais, com a diferença estando na quarta casa decimal. Os três possuem dia predito e limite iguais a 1 dia e 7%, apenas variam a janela de tempo de 30, 15 e 60, respectivamente. Deles, o 6º cenário teve o melhor desempenho e utilizou os últimos 15 dias para prever se o valor do próximo dia seria maior que 7% ou menor que -7% em relação ao dia anterior.

5.4.2 Análise das Métricas de Regressão

A fim de analisar a influência que a alteração das variáveis dos cenários tem nas métricas de desempenho, são apresentadas duas Tabelas, 5.9 e 5.10. Em negrito estão marcados os dados que apresentaram os melhores resultados para cada grupo. As tabelas possuem os mesmos dados, mas são agrupados de forma facilitar a comparação dos resultados variando-se um dos parâmetros de cada experimento: dia predito e janela de tempo, respectivamente.

Análise do Dia Predito

Ao analisar o dia predito, é possível perceber que sua escolha teve uma influência no desempenho do modelo. É possível observar que todos os cenários em que era 1 dia superaram os resultados das métricas que utilizam 7 dias que, por sua vez, superaram os de 30 dias, para uma janela de tempo constante. Como a probabilidade de um valor variar muito em um dia é menor do que em 7 ou 30, é possível que o modelo esteja com dificuldade de prever, com assertividade, maiores movimentações no preço.

Análise da Janela de Tempo

Ao analisar a Janela de Tempo, os cenários 1, 2 e 12 obtiveram as melhores métricas. Deles, os dois primeiros utilizam uma janela de tempo de 30 dias e mostraram superiores. Já o último, utilizou 15 dias e foi melhor que os seus pares.

Análise Geral

Os cenários 1 e 10 foram os que obtiveram os melhores resultados nas métricas gerais, se distanciando dos demais resultados. Ambos possuem dia previsto igual a 1 dia e apenas variam a janela de tempo de entre 30 e 60, respectivamente, sendo o primeiro experimento o que obteve as melhores marcas.

Tabela 5.9: Resultado das métricas de Regressão agrupadas variando o dia predito.

Cenário	MAE	MSE	RMSE	MAPE (%)	R ²
7	0.03436	0.00197	0.04446	0.77737	0.86307
8	0.05065	0.00370	0.06088	1.15241	0.74209
9	0.06849	0.00747	0.08645	1.56276	0.47207
4	0.03397	0.00176	0.04203	0.76949	0.87707
5	0.04113	0.00286	0.05354	0.93876	0.79914
6	0.12630	0.02113	0.14537	2.83001	-0.49694
1	0.01274	0.00027	0.01667	0.28828	0.98042
2	0.03266	0.00183	0.04286	0.74033	0.87023
3	0.09415	0.01299	0.11400	2.14223	0.06622
10	0.01347	0.00032	0.01810	0.30666	0.97645
11	0.04371	0.00298	0.05461	0.99327	0.78495
12	0.06448	0.00695	0.08337	1.46710	0.49050

Tabela 5.10: Resultado das métricas de Regressão agrupadas variando a janela de tempo.

Cenário	MAE	MSE	RMSE	MAPE (%)	R ²
7	0.03436	0.00197	0.04446	0.77737	0.86307
4	0.03397	0.00176	0.04203	0.76949	0.87707
1	0.01274	0.00027	0.01667	0.28828	0.98042
10	0.01347	0.00032	0.01810	0.30666	0.97645
8	0.05065	0.00370	0.06088	1.15241	0.74209
5	0.04113	0.00286	0.05354	0.93876	0.79914
2	0.03266	0.00183	0.04286	0.74033	0.87023
11	0.04371	0.00298	0.05461	0.99327	0.78495
9	0.06849	0.00747	0.08645	1.56276	0.47207
6	0.12630	0.02113	0.14537	2.83001	-0.49694
3	0.09415	0.01299	0.11400	2.14223	0.06622
12	0.06448	0.00695	0.08337	1.46710	0.49050

Tabela 5.11: Resultado das métricas de classificação para os resultados do modelo de Regressão (1 a 6).

Cenário	limite	Acurácia	Precisão	revocação	<i>F1-Score</i>	AUC Médio
1	3%	0.79727	0.79727	0.79727	0.79727	0.84795
1	5%	0.90468	0.90468	0.90468	0.90468	0.92851
1	7%	0.94553	0.94553	0.94553	0.94553	0.95915
2	3%	0.30909	0.30909	0.30909	0.30909	0.48181
2	5%	0.35000	0.35000	0.35000	0.35000	0.51250
2	7%	0.39090	0.39090	0.39090	0.39090	0.54318
3	3%	0.35572	0.35572	0.35572	0.35572	0.51679
3	5%	0.31908	0.31908	0.31908	0.31908	0.48931
3	7%	0.26106	0.26106	0.26106	0.26106	0.44580
4	3%	0.37593	0.37593	0.37593	0.37593	0.53195
4	5%	0.36842	0.36842	0.36842	0.36842	0.52631
4	7%	0.35939	0.35939	0.35939	0.35939	0.51954
5	3%	0.37254	0.37254	0.37254	0.37254	0.52941
5	5%	0.36349	0.36349	0.36349	0.36349	0.52262
5	7%	0.35746	0.35746	0.35746	0.35746	0.51809
6	3%	0.45523	0.45523	0.45523	0.45523	0.59142
6	5%	0.36874	0.36874	0.36874	0.36874	0.52655
6	7%	0.33080	0.33080	0.33080	0.33080	0.49810

5.4.3 Análise das Métricas de Classificação para os Resultados do Modelo de Regressão

A partir da predição dos modelos de Regressão, foi possível gerar as métricas de desempenho gerais utilizadas para a etapa de classificação, como representado na Seção 4.5. As Tabelas 5.11 e 5.11 representam os resultados obtidos para os limites de 3%, 5% e 7%.

Análise do Limite

É possível perceber que sua escolha teve uma influência no desempenho do modelo. Nos cenários 3, 4, 5, 6, 8, 9 e 11, o melhor utilizou limite de 3%, enquanto que em nenhum dos casos o cenário de 5% foi o melhor. Já, o melhor cenário dos experimentos 1, 2, 7, 10 e 12 utilizou 7%.

Tabela 5.12: Resultado das métricas de classificação para os resultados do modelo de Regressão (7 a 12).

Cenário	limite	Acurácia	Precisão	revocação	<i>F1-Score</i>	AUC Médio
7	3%	0.34984	0.34984	0.34984	0.34984	0.51238
7	5%	0.37537	0.37537	0.37537	0.37537	0.53153
7	7%	0.37837	0.37837	0.37837	0.37837	0.53378
8	3%	0.29924	0.29924	0.29924	0.29924	0.47443
8	5%	0.23759	0.23759	0.23759	0.23759	0.42819
8	7%	0.21654	0.21654	0.21654	0.21654	0.41240
9	3%	0.39090	0.39090	0.39090	0.39090	0.54318
9	5%	0.37878	0.37878	0.37878	0.37878	0.53409
9	7%	0.33939	0.33939	0.33939	0.33939	0.50454
10	3%	0.72671	0.72671	0.72671	0.72671	0.79503
10	5%	0.81526	0.81526	0.81526	0.81526	0.86145
10	7%	0.85343	0.85343	0.85343	0.85343	0.89007
11	3%	0.31192	0.31192	0.31192	0.31192	0.48394
11	5%	0.29204	0.29204	0.29204	0.29204	0.46903
11	7%	0.27981	0.27981	0.27981	0.27981	0.45986
12	3%	0.18181	0.18181	0.18181	0.18181	0.38636
12	5%	0.23112	0.23112	0.23112	0.23112	0.42334
12	7%	0.25731	0.25731	0.25731	0.25731	0.44298

Tabela 5.13: Comparação dos resultados das métricas de Classificação para os modelos criados, com um limite de 7%.

Tarefa	Dia	Limite	Acurácia	Precisão	Revocação	F1-Score	AUC Médio
R	1	30	0.94553	0.94553	0.94553	0.94553	0.95915
C	1	30	0.95763	0.95763	0.95763	0.95763	0.96822
R	1	15	0.35939	0.35939	0.35939	0.35939	0.51954
C	1	15	0.95789	0.95789	0.95789	0.95789	0.96842
R	1	60	0.85343	0.85343	0.85343	0.85343	0.89007
C	1	60	0.95725	0.95725	0.95725	0.95725	0.96793

Análise Geral

Ao observar os resultados, pode-se perceber que os melhores cenários se mantiveram sendo o 1 e o 10, obtendo uma acurácia acima de 80%, sendo o primeiro o melhor de forma geral. Ambos possuem dia previsto igual a 1 dia e limite de 7%. Apenas variam a janela de tempo de entre 30 e 60, respectivamente.

5.4.4 Comparação entre as Tarefas Regressão e Classificação

Com a construção do modelo de classificação e do modelo de regressão, é possível compará-los a fim de entender qual foi o mais eficiente para a tarefa de predição da tendência do preço do Bitcoin em 3 classes baseados em um limite superior e inferior. A Tabela 5.13 apresenta uma comparação dos modelos de acordo com as métricas de Classificação geradas. Foram escolhidos os cenários que obtiveram os melhores resultados para a comparação, que são os cenários 3, 6 e 15 de classificação e os cenários 1 e 10 de regressão. Nas tabelas eles estão apresentados com R para modelo de regressão e C para modelo de classificação. Os cenários já estão apresentados de forma a mostrar os parâmetros.

A partir da observação da Tabela 5.13 é possível perceber que a tarefa de classificação se mostrou mais eficaz para o objetivo deste trabalho, com o melhor cenário sendo o de dia futuro igual a 1 e janela de tempo de 30 dias usando um limite de 7%.

5.4.5 Reflexões sobre os Resultados Obtidos

Nesta seção, são feitas reflexões sobre os resultados obtidos e pontuando possíveis melhorias para trabalhos futuros.

Reflexão sobre os Hiperparâmetros

A escolha dos melhores hiperparâmetros para cada cenário de cada tarefa, escolhidos nas Seções 5.1 e 5.2 se mostrou uma atividade necessária. Em geral, não houve um número de células, *dropout* ou taxa de aprendizado ideal e constante para todos os cenários. Assim, foi positivo utilizar *Random Search* para essa escolha. O único hiperparâmetro que se mostrou constante foi a função de ativação *tanh*, para a tarefa de Regressão, se mostrando superior à função *relu* para os experimentos.

Reflexão sobre as Métricas de Desempenho

A partir dos resultados obtidos, é possível entender a influência que os parâmetros dos cenários tiveram nos modelos criados. Para a Classificação, o limite e o dia predito tiveram uma influência direta no desempenho do modelo. Isso pode ter sido influenciado pelo desbalanceamento das classes que aumenta quando o limite é maior ou quando o dia predito é menor, resultando em um aumento significativo da quantidade de amostras da classe 1, que representa um mercado lateral. Isso afeta o treinamento, a validação e os resultados das métricas para o conjunto de teste. Por esse motivo, os melhores cenários utilizaram 7% de limite e 1 de dia predito, sendo que melhor deles apresentou também a maior concentração de amostras pertencentes à classe 1, se comparada aos outros cenários testados. Em relação à janela de tempo utilizada, os cenários com 30 dias se mostraram superiores, apesar de não possuírem o maior desbalanceamento, se comparados aos que utilizaram os mesmos parâmetros de dia predito e limite. Isso apenas não ocorreu para o cenário 6, que foi o de melhor desempenho geral e utilizou 15 dias. Ao observar o resultado da predição de cada classe separadamente, o modelo de Classificação não obteve constância. Apesar de acertar muitos resultados de forma agregada, não conseguiu distinguir as classes de forma satisfatória, provavelmente devido à concentração de amostras de um mesmo rótulo.

Os resultados da Regressão, por sua vez, mostraram que o dia predito teve relação direta com o melhor cenário. Obteve uma dificuldade para prever, com muita assertividade, valores com variações altas, se mostrando melhor para prever valores em um curto período de tempo. Da mesma forma que na etapa anterior, 30 dias de janela de tempo se mostraram promissores para tarefa, ao utilizar 1 e 7 para o dia predito. No caso de prever 30 dias no futuro, o cenário que utiliza 60 dias de janela de tempo conseguiu melhor desempenho. Assim, é possível entender que para se prever um prazo maior de tempo, é interessante também usar uma janela temporal maior.

Em relação ao limite utilizado para os resultados da Regressão, após a transformação em classes, é possível perceber que em nenhum caso o limite de 5% foi o mais efetivo.

Apesar disso, os melhores cenários ficaram divididos entre 3%, e 7%, com 7 e 5, respectivamente. Já o melhor cenário para a tarefa de regressão possui a terceira maior concentração de amostras da classe 1. Ao levar em consideração a assertividade do modelo, ela foi baixa. Em 15 cenários obteve uma acurácia abaixo de 40%. Isso pode ser devido à movimentação do Bitcoin específica no espaço de tempo analisado pelo conjunto de teste.

Reflexão sobre Aprimoramento do Modelo

Os resultados apresentaram melhores informações sobre como cada parâmetro se comporta para a predição do preço e da tendência do Bitcoin. A partir dessas informações é possível evoluir o estudo sobre a movimentação do ativo ao utilizar formas de variar o conjunto de teste, com técnicas de validação cruzada, a fim de treinar o modelo em diferentes períodos históricos e evitar que a movimentação específica de um período cause um forte impacto no treinamento do modelo e nas suas métricas de desempenho. Uma ideia é analisar em um espaço de tempo bem reduzido, como dia ou semana, haja vista que nesse período o preço pode se comportar de forma menos sensível à sentimentos e notícias. Realizar um estudo considerando amostras de minutos ao invés de dias aumenta muito a quantidade de elementos e pode ser uma abordagem que contribui positivamente para o treinamento do modelo, principalmente ao se utilizar validação cruzada. Além disso, este trabalho optou por utilizar apenas os dados históricos de preços e o volume. É interessante também incluir outras métricas a fim de testar se a assertividade do modelo é melhorada, como dados da *blockchain*, macroeconômicos, indicadores técnicos (binários ou numéricos). Finalmente, a exploração dos modelos de *ensemble* pode se revelar uma solução intrigante. Cada modelo pode aproveitar um cenário distinto para realizar suas previsões, e o agrupamento desses cenários culmina no resultado final.

Capítulo 6

Trabalhos Futuros

Este trabalho propôs uma análise da predição da tendência e do preço do Bitcoin por meio de técnicas de regressão e classificação, apresentando uma forma de transformar o valor encontrado pelo modelo de regressão em classes a fim de comparar as abordagens. A partir dos resultados apresentados neste trabalho, o melhor cenário para **classificação** utilizou uma janela de tempo de 15 dias, previu a classe da tendência do preço para 1 dia no futuro e utilizou um limite de 7%. Ele obteve um desempenho de 0,95789 para acurácia, precisão, revocação e *f1-score*; e um AUC médio de 0,96842. Em relação ao modelo de **regressão**, o melhor cenário utilizou uma janela de tempo de 30 dias, previu o log do preço para 1 dia no futuro. Ele obteve um desempenho de 0,01274 para o MAE, 0,00027 para MSE, 0,01667 para RMSE, 0,28828% para MAPE e 0,98042 para R^2 . Posteriormente, o resultado foi analisado pela ótica da variação percentual predita e então separado pelas mesmas classes definidas na etapa do modelo de classificação. Assim, o melhor modelo obteve um desempenho para um limite de 7% de 0,94553 para acurácia, precisão, revocação e *f1-score*; e um AUC médio de 0,95915. Ao comparar as métricas geradas de cada abordagem, o modelo de classificação mostrou superior ao de regressão.

A classificação demonstrou ser sensível ao desbalanceamento de amostras, apresentando os melhores cenários com uso do maior limite e o menor dia predito. Isso que colabora para aumentar o desbalanceamento para a classe 1, que significa o movimento lateral do ativo. Já a Regressão mostrou ter menos influência do limite, haja vista que, ao comparar os cenários, aquele com limite 7% não necessariamente foi o com melhor desempenho. Vale pontuar que apesar do valor alto do AUC médio, os modelos mostram dificuldade para prever as classes isoladas. Notavelmente, a tarefa de Regressão demonstrou vantagem sobre a Classificação nesse aspecto, conseguindo discernir melhor as classes individualmente em mais cenários.

Assim, foi possível perceber a importância de se observar várias métricas a fim de formular uma conclusão sobre um problema a fim de ter uma visão completa sobre os

resultados observados, porém algumas limitações foram encontradas. Uma delas foi o desbalanceamento dos rótulos, que mesmo com uma alta assertividade geral para a classificação, os modelos podem ser aprimorados a fim de melhorarem os resultados individuais das classes. Somado a isso, o trabalho não buscou aumentar ou diminuir a quantidade de dados utilizados (3332 dias), possivelmente aumentar o número de amostras possa melhorar o modelo. Aliado à essa mudança, é interessante utilizar validação cruzada a fim de entender se é uma técnica capaz de melhorar os resultados obtidos e com quantas amostras ela começa a se tornar eficaz. O uso de variações de tempo diferentes, como analisar dados por horas, minutos ou até segundos pode ser positivo, além de alterar o limite utilizado para valores que priorizem esse balanceamento das classes.

A abordagem proposta não analisa o retorno sobre o investimento, caso um investidor siga o modelo proposto como auxílio na tomada de decisão. Essa é que é uma análise interessante a ser estudada, principalmente se aliada à comparação com técnicas de investimentos tradicionais. Outro ponto de melhoria é a adição de novas informações ao modelo, não só as relacionadas a preços históricos e volatilidade. Uma proposta é incluir dados da *blockchain*, sentimentos expressos em redes sociais, indicadores econômicos e técnicos, o que pode se mostrar vantajoso para a construção de um modelo mais completo e assertivo. Somado a isso, a utilização de métodos de *ensemble* com o uso de diferentes cenários pode unir os resultados das duas abordagens em uma única predição.

Referências

- [1] Phi, Michael: *Illustrated Guide to LSTM's and GRU's: A step by step explanation.* <https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21>, acesso em 2023-24-10. ix, 8
- [2] Lo, Andrew W e A Craig MacKinlay: *Stock market prices do not follow random walks: Evidence from a simple specification test.* The Review of Financial Studies, 1(1):41–66, 1988. 2
- [3] Nabipour, Mojtaba, Pooyan Nayyeri, Hamed Jabani, S Shahab e Amir Mosavi: *Predicting stock market trends using machine learning and deep learning algorithms via continuous and binary data; a comparative analysis.* IEEE Access, 8:150199–150212, 2020. 2, 14, 19
- [4] Kanji, Abdul Mannan, Ishita Chaudhary, Rithika Lakshmi Shankar e Gowri Srinivasa: *Predicting the price direction of bitcoin using twitter data and machine learning.* Em *2022 IEEE 2nd International Conference on Data Science and Computer Application (ICDSCA)*, páginas 46–52. IEEE, 2022. 2, 15, 19
- [5] Verma, Satya, Satya Prakash Sahu e Tirath Prasad Sahu: *Stock market forecasting with different input indicators using machine learning and deep learning techniques: A review.* Engineering Letters, 31(1), 2023. 2, 4, 10, 18
- [6] Tejaswi, Dronavalli Krishna, Himanshi Chauhan, T Jaya Lakshmi, Rachakonda Swetha e Nallamothe Navya Sri: *Investigation of ethereum price trends using machine learning and deep learning algorithms.* Em *2022 2nd International Conference on Intelligent Technologies (CONIT)*, páginas 1–5. IEEE, 2022. 2, 17, 19
- [7] Wang, Yimeng e Keyue Yan: *Prediction of significant bitcoin price changes based on deep learning.* Em *2022 5th International Conference on Data Science and Information Technology (DSIT)*, páginas 1–5. IEEE, 2022. 2, 10, 13, 15, 18, 19, 23
- [8] Jang, Huisu e Jaewook Lee: *An empirical study on modeling and prediction of bitcoin prices with bayesian neural networks based on blockchain information.* IEEE Access, 6:5427–5437, 2017. 2, 6, 16, 19, 24
- [9] Jiang, Weiwei: *Applications of deep learning in stock market prediction: recent progress.* Expert Systems with Applications, 184:115537, 2021. 4

- [10] MATSURA, EDUARDO KOITI: *Comprar ou vender?: como investir na bolsa utilizando análise gráfica*. Saraiva Educação SA, 2017. 5
- [11] Han, Zhongyang, Jun Zhao, Henry Leung, King Fai Ma e Wei Wang: *A review of deep learning models for time series prediction*. IEEE Sensors Journal, 21(6):7833–7848, 2019. 5
- [12] Nakamoto, Satoshi: *A peer-to-peer electronic cash system*, Nov 2008. <https://bitcoin.org/en/bitcoin-paper>. 5
- [13] Sheth, Harsh e Janvi Dattani: *Overview of blockchain technology*. Asian Journal For Convergence In Technology (AJCT) ISSN-2350-1146, 2019. 5
- [14] IBM: *O que é machine learning?* <https://www.ibm.com/br-pt/topics/machine-learning>, acesso em 2023-24-10. 6
- [15] Ji, Suhwan, Jongmin Kim e Hyeonseung Im: *A comparative study of bitcoin price prediction using deep learning*. Mathematics, 7(10):898, 2019. 7, 16, 19, 20
- [16] Zakaria, Magdi, AS Mabrouka e Shahenda Sarhan: *Artificial neural network: a brief overview*. Neural Networks, 1:2, 2014. 7
- [17] Pascanu, Razvan, Tomas Mikolov e Yoshua Bengio: *On the difficulty of training recurrent neural networks*. Em *International Conference on Machine Learning*, páginas 1310–1318. Pmlr, 2013. 8
- [18] Hochreiter, Sepp e Jürgen Schmidhuber: *Long short-term memory*. Neural Computation, 9(8):1735–1780, 1997. 8
- [19] Fawcett, Tom: *An introduction to roc analysis*. Pattern Recognition Letters, 27(8):861–874, 2006. 13
- [20] Hsu, Ming Wei, Stefan Lessmann, Ming Chien Sung, Tiejun Ma e Johnnie EV Johnson: *Bridging the divide in financial market forecasting: machine learners vs. financial economists*. Expert Systems with Applications, 61:215–234, 2016. 14, 19
- [21] Helfenstein, Rodolfo: *Um modelo de rede neural de classificação para predição da tendência do preço bitcoin*, 2022. 15, 19
- [22] Malhotra, Bhavay, Chittaranjan Chandwani, Pratham Agarwala e Suman Mann: *Bitcoin price prediction using machine learning and deep learning algorithms*. Em *2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)*, páginas 1–6. IEEE, 2022. 16, 19
- [23] Arezooji, Dorsa Mohammadi: *A big data analysis of the ethereum network: from blockchain to google trends*. arXiv preprint arXiv:2104.01764, 2021. 17, 19
- [24] Yahoo: *Banco de dados históricos dos preços do bitcoin*, 2023. <https://finance.yahoo.com/quote/BTC-USD>, acesso em 2023-06-10. 21