



# Análise do Retorno do Investimento em Criptomoedas

Filipe Barreto de Oliveira

JUIZ DE FORA  
NOVEMBRO, 2020

# Análise do Retorno do Investimento em Criptomoedas

FILIPPE BARRETO DE OLIVEIRA

Universidade Federal de Juiz de Fora  
Departamento de Ciência da Computação  
Bacharelado em Sistemas de Informação

Orientador: Luciana Conceição Dias Campos

JUIZ DE FORA  
NOVEMBRO, 2020

# ANÁLISE DO RETORNO DO INVESTIMENTO EM CRIPTOMOEDAS

Filipe Barreto de Oliveira

MONOGRAFIA SUBMETIDA AO CORPO DOCENTE DO DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA, COMO PARTE INTEGRANTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE BACHAREL EM SISTEMAS DE INFORMAÇÃO.

Aprovada por:

Luciana Conceição Dias Campos  
DSc em Engenharia Elétrica PUC-Rio

Alex Borges Vieira  
DSc em Ciências da Computação UFMG

Leonardo Goliatt da Fonseca  
DSc em Modelagem Computacional LNCC

JUIZ DE FORA  
20 DE NOVEMBRO, 2020

## Resumo

Aprendizado de máquina e negociação assistida por inteligência artificial têm atraído interesse crescente nos últimos anos. Há um crescente interesse na utilização destas técnicas na previsão de preços no mercado financeiro, porém, as vantagens de tais técnicas em relação a modelos estocásticos ainda dependem de resultados empíricos.

Neste trabalho, usamos uma abordagem em redes LSTM para analisar o retorno do investimento feito nas criptomoedas Bitcoin e Ethereum. O modelo proposto visa auxiliar o investidor na análise de preço de transação e a rentabilidade esperada para as moedas virtuais, e não deve ser considerado uma previsão para o valor futuro das criptomoedas estudadas. Como objetivo secundário, é verificado se há relação entre os valores de transação da Bitcoin e o Dólar, como também entre a rentabilidade e o valor de transação para as duas criptomoedas utilizadas. Foi utilizado os dados diários de preço para a Bitcoin e para o Dólar em um período entre abril de 2013 e julho de 2020, e dados diários de preço para a Ethereum em um período entre agosto de 2015 e agosto de 2020.

# Abstract

Machine learning and artificial intelligence-assisted trading have attracted growing interest in recent years. There is a growing interest in the use of these techniques in the prediction of prices in the financial market, however, the advantages of such techniques in relation to stochastic models still dependent on empirical results.

In this paper, we use an approach in LSTM networks to analyze the return on investment made in Bitcoin and Ethereum cryptocurrencies. The proposed model aims to assist the investor in the analysis of the transaction price and the expected profitability for the virtual currencies, and should not be considered a forecast for the future value of the studied cryptocurrencies. As a secondary objective, it is checked whether there is a relationship between Bitcoin transaction values and Dollar, as well as between profitability and transaction value for the two cryptocurrencies used. Daily price data for Bitcoin and Dollar were used between April 2013 and July 2020, and daily price data for Ethereum between August 2015 and August 2020.

## **Agradecimentos**

Aos meus pais, pelo encorajamento e apoio.

À professora Luciana pela orientação, amizade e principalmente, pela persistência, sem a qual este trabalho não se realizaria.

Aos professores do Departamento de Ciência da Computação pelos seus imprescindíveis ensinamentos e aos funcionários do curso, que durante esses anos, contribuíram de algum modo para o nosso enriquecimento pessoal e profissional.

# Sumário

<b>Lista de Figuras</b>	<b>5</b>
<b>Lista de Tabelas</b>	<b>6</b>
<b>Lista de Abreviações</b>	<b>7</b>
<b>1 Introdução</b>	<b>8</b>
1.1 Apresentação do Tema . . . . .	8
1.2 Motivação . . . . .	10
1.3 Trabalhos relacionados . . . . .	11
1.4 Objetivos . . . . .	12
1.5 Organização do Trabalho . . . . .	12
<b>2 Fundamentação Teórica</b>	<b>13</b>
2.1 Séries Temporais . . . . .	13
2.2 Criptomoedas . . . . .	14
2.2.1 Bitcoin . . . . .	14
2.2.2 Ethereum . . . . .	15
2.2.3 <i>Blockchain</i> . . . . .	16
2.2.4 <i>Proof-of-work</i> . . . . .	17
2.2.5 Mineração . . . . .	17
2.3 Redes Neurais Artificiais . . . . .	18
2.3.1 Funções de Ativação . . . . .	21
2.3.2 Desempenho da Rede . . . . .	22
2.3.3 Redes Neurais Recorrentes . . . . .	22
2.3.4 Redes <i>Long Short-Term Memory</i> . . . . .	23
<b>3 Metodologia</b>	<b>26</b>
3.1 Base de Dados . . . . .	26
3.1.1 Pré-processamento dos Dados . . . . .	27
3.2 Modelo LSTM de Análise do Valor de Abertura da Bitcoin . . . . .	28
3.3 Modelo LSTM de Análise do Valor de Abertura da Bitcoin com o Dólar . . . . .	28
3.4 Modelo LSTM de Análise de Rentabilidade de Criptomoedas . . . . .	29
3.5 Modelo LSTM de Análise da Rentabilidade com o Valor de Abertura de Criptomoedas . . . . .	30
<b>4 Estudo de Caso</b>	<b>31</b>
4.1 Configurações do Ambiente . . . . .	31
4.2 Análise dos Experimentos . . . . .	31
<b>5 Conclusão e Trabalhos Futuros</b>	<b>40</b>
<b>Referências Bibliográficas</b>	<b>42</b>

## Lista de Figuras

2.1	Esquema de encadeamento de blocos da <i>Blockchain</i> (NAKAMOTO, 2008).	16
2.2	Esquema de um neurônio artificial (MCCULLOCH; PITTS, 1943).	19
2.3	Esquema de rede neural <i>Feedforward</i> (FACURE, 2020).	20
2.4	Gráfico das funções ReLU e <i>LeakyReLU</i> , respectivamente (SHARMA, 2020).	22
2.5	Esquema de rede neural recorrente (PERERVENKO, 2020)	23
2.6	Esquema de um neurônio da rede LSTM (COLAH, 2020)	24
3.1	Tabela representando a divisão da base de dados em conjuntos de treinamento e teste	27
3.2	Esquema do modelo LSTM com a série financeira da Bitcoin como entrada e 7 dias de janelamento - autor	28
3.3	Esquema do modelo LSTM com a série financeira da Bitcoin e do Dólar como entrada e 3 dias de janelamento - autor	29
3.4	Esquema do modelo LSTM com a rentabilidade intradiária como entrada e 7 dias de janelamento - autor	29
3.5	Esquema do modelo LSTM com a rentabilidade intradiária e o valor e abertura como entrada e 3 dias de janelamento - autor	30
4.1	Gráficos com o pior e melhor resultados do janelamento de 1 dia, respectivamente	32
4.2	Gráficos com o pior e melhor resultados do janelamento de 2 dias, respectivamente	33
4.3	Gráficos com o pior e melhor resultados do janelamento de 3 dias, respectivamente	33
4.4	Gráficos com o pior e melhor resultados do janelamento de 5 dias, respectivamente	34
4.5	Gráficos com o pior e melhor resultados do janelamento de 7 dias, respectivamente	35
4.6	Gráfico que representa a queda nos valores de títulos do tesouro nos EUA e na cotação da Bitcoin entre Setembro de 2019 e Junho de 2020, Trading-View (2020)	36
4.7	Gráfico com o melhor resultado encontrado com janelamento de 7 dias com a rentabilidade da Bitcoin	37
4.8	Gráfico com o melhor resultado encontrado com janelamento de 1 dia com a rentabilidade da Ethereum	37
4.9	Gráfico com o melhor resultado encontrado com janelamento de 3 dias com a rentabilidade e o preço de fechamento como entrada para a Bitcoin	38
4.10	Gráfico com o melhor resultado encontrado com janelamento de 3 dias com a rentabilidade e o preço de fechamento como entrada para a Ethereum	39

## Lista de Tabelas

4.1	Tabela com valores de MAPE de cada janelamento . . . . .	32
-----	--	----

## Lista de Abreviações

BTC	Bitcoin
ETH	Ethereum
RNA	Rede Neural Artificial
MLP	<i>MultiLayer Perceptron</i>
LSTM	<i>Long short-term memory</i>
SVM	<i>Support vector machine</i>
ARIMA	<i>Autoregressive integrated moving average</i>
MAPE	<i>Mean absolute percentage error</i>
ReLU	<i>Rectified linear unit</i>
ROI	<i>Return of investment</i>

# 1 Introdução

O presente capítulo apresenta a motivação, a justificativa e os objetivos da realização deste trabalho.

## 1.1 Apresentação do Tema

No ano de 2008, o artigo escrito por Nakamoto (2008) apresentava ao mundo o primeiro conceito de moeda totalmente virtual e regulada somente pelo usuário que a detém, a Bitcoin. A Bitcoin é a realização de décadas de estudos em criptografia e sistemas distribuídos e inclui 4 inovações tecnológicas que a transformaram hoje em um gigantesco mercado de quase 250 bilhões de dólares, segundo Coindesk (2020). A Bitcoin compreende as tecnologias de rede ponto-a-ponto; o uso de registro público virtual de transações, chamada *blockchain*; a emissão de moeda de maneira determinística e descentralizada; e um sistema de verificação de transações descentralizada.

Unidades da criptomoeda Bitcoin são usadas para guardar ou transferir valor entre os participantes da rede, que por sua vez se comunicam entre si primariamente pela Internet. A participação na rede Bitcoin é simples e fácil, se dá através de softwares livres de carteiras virtuais, que provam a posse do usuário por meio de uma chave criptográfica que deve ser resguardada por este.

Os programas clientes da Bitcoin podem ser de três modalidades:

- *Full client* ou *full node*: possui o histórico completo de transações da Bitcoin, gerencia a carteira virtual do usuário e é capaz de iniciar transações diretamente na rede Bitcoin;
- *Lightweight client*: gerencia a carteira virtual do usuário, porém acessa serviços de terceiros para acessar a rede da Bitcoin
- *Web client*: acessível através de navegadores e armazenam a carteira virtual do usuário em terceiros.

Contudo, a principal inovação advinda do trabalho de Nakamoto (2008) é a completa descentralização do sistema de dinheiro virtual, totalmente independente de um órgão central para emitir novas unidades da moeda e validar as transações. A *blockchain* é responsável por registrar cada movimentação de bitcoins já realizadas a partir de transações válidas. A validação destas transações é feita por um algoritmo, chamado de “*proof-of-work*”, que realiza uma espécie de votação entre os usuários a cada 10 minutos para escolher quais transações serão adicionadas à *blockchain*. Na prática, o algoritmo resolve o problema de gasto duplo, onde uma mesma unidade de moeda é gasta duas vezes, e ao validar o conjunto de transações, bonifica um dos usuários com novas bitcoins. O algoritmo também é capaz de se autoajustar para manter o tempo de 10 minutos para cada nova validação e emissão de moedas.

Conforme mais usuários participavam da “eleição” e utilizavam energia elétrica para manter seus computadores executando o “*proof-of-work*”, e sendo recompensados por isso, mais difícil acabava sendo “sequestrar” a rede e adicionar transações fraudulentas. Como já havia sido informado pelo artigo de Nakamoto (2008), a emissão de novas unidades, que começou com 50 bitcoins, está programada a ser reduzida pela metade a cada 4 anos, e em 2140 cessará totalmente sua emissão, o que dá à bitcoin um comportamento deflacionário em seu valor, numa analogia ao ouro, que é finito.

A invenção da Bitcoin foi um marco na história e inspirou vários novos estudos em diversas áreas, como nos campos de computação distribuída e nas áreas de economia. A popularidade das criptomoedas cresceu exponencialmente em 2017 devido a consecutivos meses de crescimento da sua capitalização de mercado (ELBAHRAWY LAURA ALESSANDRETTI; BARONCHELLI, 2017), e atualmente já estamos em quase 4000 criptomoedas sendo comercializadas diariamente.

Dada sua alta valorização, vários estudos, como o de Hegazy e Mumford (2016) buscaram a existência de previsibilidade nos valores de transação da Bitcoin. Adicionado a isso, a Ethereum possui a segunda maior capitalização de mercado e suporta muito mais funcionalidades que a Bitcoin, e isso faz com que a previsibilidade desta difira substancialmente da Bitcoin.

Dados extraídos de Coindesk (2020) mostraram que entre 30 de agosto de 2015

e 19 de outubro de 2017 a Bitcoin teve uma volatilidade mensal em seu preço de 21.73%, já a Ethereum teve uma volatilidade mensal de 77.91%, muito acima dos 14% da SP 500, a bolsa de valores norte-americana que reúne os valores de ativos das 500 mais valiosas empresas listadas para troca de ações.

Resultados advindos do estudo de ElBahrawy Laura Alessandretti e Baronchelli (2017) mostraram que características de longo prazo do mercado de criptomoedas permaneceram estáveis entre 2013 e 2017 e são compatíveis com um cenário em que os investidores investem seu dinheiro de acordo com as participações de mercado da criptomoeda. Embora na maioria dos casos isso seja verdade, vários estudos têm se concentrado na análise e previsão de flutuações de preços, usando principalmente abordagens tradicionais para análise e previsão dos mercados financeiros (CIAIAN; RAJCANIOVA; KANCS, 2016). O sucesso das técnicas de aprendizado de máquina para previsão do mercado de ações (ENKE; THAWORNWONG, 2005) sugere que esses métodos podem ser eficazes também na previsão de preços de criptomoedas.

## 1.2 Motivação

O mercado de transações das criptomoedas Bitcoin se tornou extremamente volátil conforme foi se valorizando, devido à sua alta sensibilidade a fatores externos, como por exemplo, notícias de que um país começaria a reter suas reservas financeiras com a criptomoeda. Este comportamento propicia que investidores tenham um lucro acima do esperado em relação a outros investimentos, porém com um alto risco de perda associado. A motivação do trabalho está em aplicar um modelo de *machine learning* para aprender o comportamento volátil dos valores das criptomoedas analisadas, e assim agregar mais conhecimento para o investidor, possibilitando um aumento do capital investido.

A justificativa de se analisar os valores de uma criptomoeda se dá pelo alto desafio presente em investir e prosperar em um mercado extremamente volátil como o das criptomoedas, e técnicas capazes de se aproximar desse objetivo são o sonho de qualquer investidor. Neste trabalho é proposto um método de *machine learning* com redes LSTM para prever os valores de transação da Bitcoin e o retorno do investimento tanto da Bitcoin quanto da Ethereum, a partir de dados extraídos de suas séries temporais.

## 1.3 Trabalhos relacionados

Grande parte dos estudos acerca de predições de valores estão associados à bolsa de valores, como em Mithani, Machchhar e Jasdanwala (2016), onde é proposto um modelo modificado de rede neural recorrente para prever valor futuro e o máximo de ganho para o titular do ativo, e complementa comparando seu método com outros métodos utilizados para a mesma classe de problemas, como métodos de regressão e as máquinas de vetor suporte (SVM).

O estudo de Ryll e Seidens (2019) faz a comparação entre diversas aplicações de *machine learning* utilizadas para previsão de séries temporais financeiras e chega à conclusão de que modelos de *machine learning* têm desempenho melhor comparados a métodos estocásticos, e nisso que modelos de redes neurais recorrentes performam melhor que modelos *Feedforward*, presumidamente por trabalhar melhor com a dinâmica de relações temporais de um série histórica.

Outros estudos discutem a viabilidade de modelos de previsão em relação à séries temporais de valores de criptomoedas. Shah e Zhang (2014) propõe um método de regressão bayesiana, com o intuito de entender se a série histórica da Bitcoin guarda alguma informação relevante, capaz de ajudar a prever flutuações em seu preço; Stenqvist e Lönnö (2017) analisa o impacto que *tweets*<sup>1</sup> têm na tendência do preço da mais conhecida criptomoeda; e Chen, Narwal e Schultz (2019) propõem múltiplos modelos de classificação, como o SVM e o ARIMA, e os compara à LSTM em relação à acurácia de prever o valor da Ethereum, onde é afirmado que este possui uma maior volatilidade que o Bitcoin. Neste estudo, foi concluído que o modelo ARIMA obteve os melhores resultados.

A rede LSTM é uma arquitetura de rede neural recorrente implementada com a capacidade de aprender dependências temporais de longo prazo. Nelson (2017) faz uso da LSTM para atuar no problema de previsão de tendência de ações, utilizando tanto dados quantitativos (série temporal do valor da ação), quanto dados qualitativos (indicadores de análise técnica), obtendo resultados promissores em termos de retorno sobre o investimento. Já o estudo de McNally, Roche e Caton (2018) faz uma comparação de resultados ao prever os valores da Bitcoin entre um modelo LSTM e uma rede neural

---

<sup>1</sup>textos escritos por usuários da rede *Twitter*

---

recorrente com otimização *bayesiana*, chegando à conclusão de que o modelo LSTM foi mais preciso e obteve menor erro do que a rede neural recorrente.

## 1.4 Objetivos

O principal objetivo do estudo é o de propor um modelo capaz de trazer um retorno positivo ao investidor dentro de um período de 30 dias, e que o mesmo ilustre a tendência de valores das criptomoedas analisadas.

Como objetivo secundário, o trabalho traz a relação de quais fatores estão fortemente correlacionados entre si, por exemplo, se o volume de transações estando em uma crescente influencia o valor de compra subir.

## 1.5 Organização do Trabalho

O trabalho está organizado da seguinte forma, o capítulo 2 oferece referencial teórico para os métodos científicos utilizados no estudo e aborda as técnicas presentes na literatura para tratar o problema. O capítulo 3 apresenta a metodologia aplicada e o modelo LSTM utilizado na análise dos valores de abertura da Bitcoin e rentabilidade intradiária, tanto da Bitcoin como da Ethereum. O capítulo 4 apresenta o estudo de caso e o resultado dos experimentos realizados. E por fim, o capítulo 5 apresenta a conclusão e os trabalhos futuros sugeridos.

## 2 Fundamentação Teórica

O capítulo dispõe sobre os fundamentos teóricos relacionados ao trabalho. Na seção 2.1 é apresentado os fundamentos de séries temporais. Na Seção 2.2 é dada uma visão geral sobre criptomoedas, com mais profundidade na mais conhecida entre elas, a Bitcoin. Na Seção 2.3 está descrito os principais conceitos de Redes Neurais Artificiais, mais especificamente às redes neurais recorrentes.

### 2.1 Séries Temporais

De acordo com Brockwell, Davis e Fienberg (1991), uma série temporal é um conjunto de  $n$  observações realizadas ao longo do tempo. Sua principal característica é a de que as observações vizinhas são dependentes, o que torna a ordem dos dados observados fundamental. A série temporal é expressada na Equação 2.1

$$z_1, z_2, z_3, \dots, z_t, \dots, z_n \quad (2.1)$$

onde onde  $z_t$  é tido como a observação no instante  $t$ .

Na economia, Haavelmo (1944) assume que séries temporais são realizações de processos estocásticos, onde este representa uma variável aleatória para cada instante no tempo. O processo estocástico é apresentado pela Equação 2.2

$$X(t) : t \in T \quad (2.2)$$

sendo que  $X(t)$  é uma variável aleatória chamada *estado do processo* no instante  $t$  e  $T$  é o *espaço paramétrico* do processo. No caso de  $T$  ser um intervalo de números reais, definimos o processo estocástico como *processo contínuo*. Mas se  $T$  for um conjunto numerável de elementos, o processo é denominado *processo discreto*. Caso o processo

possa ser previsto, ele é considerado determinístico.

Segundo Adhikari e Agrawal (2013), séries temporais são caracterizadas por 4 fatores:

- Tendência: dita se o comportamento da série tende à estagnação, crescimento ou decrescimento;
- Variação cíclica: padrões da série que se repetem à médio prazo;
- Variação sazonal: padrões causados por fatores externos que se repetem ao longo do tempo;
- Variação irregular: comportamentos ocasionados por fenômenos imprevisíveis.

## 2.2 Criptomoedas

Criptomoedas são uma classe de moedas digitais que executam métodos de criptografia, utilizadas para a validação da transferência de fundos entre transações sem a participação de um Banco Central. Há um número finito de criptomoedas disponíveis na rede e que ainda serão emitidas, o que naturalmente eleva o seu preço. Além disso, seu valor se baseia na oferta e procura por suas unidades (LO; WANG, 2014).

De acordo com Heid (2013), sua estrutura é composta por arquivos pré-computados, compostos por pares de chaves pública e privada, gerados a partir de um algoritmo de criptografia. Cada par de chaves representa uma unidade da criptomoeda e assina a posse dela a um usuário em posse da chave privada. Esses pares são guardados em arquivos no disco rígido e funcionam como carteiras para as moedas digitais.

A primeira e mais conhecida delas, o Bitcoin, é detalhada na seção 2.2.1.

### 2.2.1 Bitcoin

“A moeda digital bitcoin ou BTC é definida por uma cadeia de assinaturas digitais, onde cada proprietário transfere a moeda assinando-a com um hash<sup>2</sup> da transação

---

<sup>2</sup>Algoritmo que mapeia dados de comprimento variável para dados de comprimento fixo

anterior e a chave pública do próximo proprietário, adicionando estes ao final da moeda” (NAKAMOTO, 2008).

A cadeia de assinaturas digitais, chamada de *blockchain*, faz com que a rede não necessite de uma instituição mediadora que resguarde os dois lados da transação em relação a fraudes. A moeda Bitcoin tem diversas vantagens em relação às moedas governamentais, sendo elas, como disseram Lo, Wang (2014): a Bitcoin é menos suscetível a esquemas fraudulentos e não há uma taxa de cobrança direta nas transações. Estes fatores resultam em menores custos de transação no geral. Adicionalmente, essa nova forma de transação de valores impede que uma possível falha relacionada às moedas digitais ocorra, o gasto duplo.<sup>3</sup>

### 2.2.2 Ethereum

A Ethereum, criada em julho de 2015, expandiu a tecnologia *blockchain* para mais do que uma moeda digital, tornando-a em uma plataforma de software descentralizada. Segundo Reiff (2020), a principal diferença da Ethereum em relação à Bitcoin é a capacidade de implantação de contratos inteligentes e aplicativos descentralizados sendo executados sem qualquer tempo de inatividade, fraude, controle ou interferência de terceiros. A Ethereum possui sua própria linguagem de programação que roda em uma *blockchain*, permitindo que os desenvolvedores criem e executem aplicativos distribuídos.

Todas essas aplicações do Ethereum são alimentadas por seu token criptográfico nativo, *ether* (ou ETH). Ether é o combustível para executar comandos na plataforma Ethereum e é usado por desenvolvedores para construir e executar aplicativos na plataforma, e também pode ser negociado como uma moeda digital nas bolsas da mesma maneira que outras criptomoedas.

A seção 2.2.3 trata da estrutura de dados que possibilitou o surgimento das criptomoedas.

---

<sup>3</sup>Diferente de moedas físicas, unidades de criptomoedas podem ser duplicadas, ou seja, o ato de gastar a criptomoeda não implica na transferência da sua posse. O gasto duplo se dá quando o usuário consegue gastar mais de uma vez a mesma unidade da criptomoeda

### 2.2.3 *Blockchain*

A principal tecnologia que sustenta a Bitcoin e as demais criptomoedas é o esquema de *blockchain*. Sua função é agrupar todas as transações da criptomoeda realizadas e ser pública para todos os usuários, o que garante que toda informação de transação seja transmitida aos demais usuários da criptomoeda de maneira transparente e descentralizada.

A *blockchain* é incrementada da seguinte forma: para participar em transações, o usuário recebe uma cópia completa da *blockchain* e é capaz de validar transações e blocos. A medida que as transações são validadas, elas são agrupadas em blocos. Ao final de cada bloco é gerado um *hash* com o registro de todas as transações do bloco. Blocos subsequentes recebem a informação do *hash* do bloco anterior, como mostra a Figura 2.1.

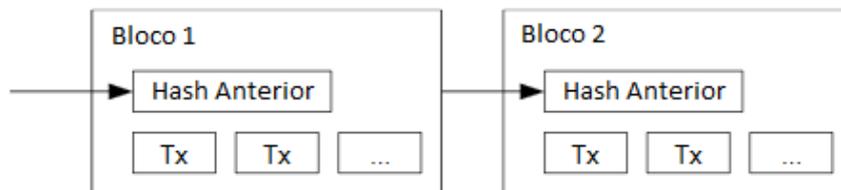


Figura 2.1: Esquema de encadeamento de blocos da *Blockchain* (NAKAMOTO, 2008).

Caso o conjunto de blocos que foi adicionado seja aceito pela maioria dos nós da rede, através do *proof-of-work* detalhada na seção 2.2.4, a *blockchain* resultante é considerada válida e retransmitida aos outros nós da rede.

A *Blockchain* envolve 5 princípios básicos, como explicado por Steler e Cerqueira (2017):

1. rede ponto-a-ponto: as transações são feitas em um canal direto entre os nós;
2. ausência de uma autoridade central para certificar a autenticidade das transações: este papel é realizado pelos nós da rede;
3. *proof-of-work* ou prova-de-esforço: inviabiliza a ação de usuários com a intenção de controlar a rede e alterar os registros de transação. Seu conceito será explicado na seção 2.2.4;

4. consenso da maioria: mantém a rede confiável através de um algoritmo de consenso, que permite que a transação seja validada através de um consenso entre todos os nós da rede, substituindo a necessidade de um órgão central validador;
5. sincronização entre os nós: os usuários na rede possuem o mesmo registro de transações.

#### 2.2.4 *Proof-of-work*

O conceito de *proof-of-work* idealizado por Back (2002) garante o consenso na rede de transações, através da solução do problema criptográfico do *hash*. Quando a solução do problema é encontrada um novo bloco é criado, significando que este bloco foi “minerado”. O processo conhecido como mineração é detalhado na seção 2.2.5.

O *proof-of-work* é implementado da seguinte forma: seu algoritmo adiciona um número arbitrário, conhecido como *nonce*, em cada bloco. O minerador, utilizando o poder de processamento da sua máquina, procura um valor desse nonce que, ao ser adicionado ao registro das transações do bloco, gere um *hash* com um certo número de zeros no seu início. A *blockchain* autoajusta a dificuldade para se encontrar o *hash* dependendo do processamento total de seus nós, tal que um novo bloco seja criado, em média, a cada 10 minutos (BINANCE, 2020).

O *proof-of-work* também faz com que a representação da maioria na rede seja determinada pelo maior encadeamento de blocos. Pois sendo a maioria do poder de processamento da rede composta por nós honestos, a cadeia com transações válidas crescerá mais rapidamente do que qualquer outra, o que inibe a ação de nós desonestos que têm a intenção de adicionar transações falsas à *blockchain*.

#### 2.2.5 Mineração

Mineração é o ato de utilizar o processamento de máquina para competir com outros mineradores, para ser o primeiro a solucionar o *proof-of-work*, e assim gerar novos blocos na *blockchain*. Por convenção, a primeira transação de cada novo bloco da *blockchain* é especial, e confere ao minerador que solucionou o *proof-of-work* primeiro

uma recompensa em bitcoins, além de uma taxa para cada transação que ocorra a partir daquele bloco (INVESTOPEDIA, 2020).

Os bitcoins ganhos pelos mineradores, além de serem o maior incentivo da mineração, são novas emissões da moeda, assim tornando desnecessária a criação de uma entidade reguladora que gere mais unidades da criptomoeda.

## 2.3 Redes Neurais Artificiais

“Uma rede neural é um processador maciçamente paralelamente distribuído constituído de unidades de processamento simples, que têm a propensão natural para armazenar conhecimento experimental e torná-lo disponível para o uso” (HAYKIN, 2001).

As redes neurais artificiais, ou RNAs, são redes conexionistas programadas para simular a maneira com que o cérebro biológico lida com uma tarefa ou função de interesse, e apresentam semelhanças entre si em dois aspectos: todo conhecimento adquirido se dá através de um processo de aprendizagem e conexões entre neurônios, chamados pesos sinápticos que armazenam o conhecimento adquirido. Em termos matemáticos, o neurônio artificial é descrito pelas equações (2.3) e (2.4) e é representado pela Figura 2.2.

$$u_k = \sum_{j=1}^p w_{kj} x_j + b_k \quad (2.3)$$

$$y_k = \varphi(u_k) \quad (2.4)$$

Os elementos usados nas equações (2.3), (2.4) e na Figura 2.2 são definidos em:

- Sinais de entrada  $x_j$ : dados que representam características extraídas do problema, como valores discretos, do conjunto  $\{0,1\}$ ,  $\{-1,1\}$  ou números reais;
- Conjuntos de pesos sinápticos  $w_{kj}$ : parâmetros do modelo ajustados durante a fase de treinamento da rede;

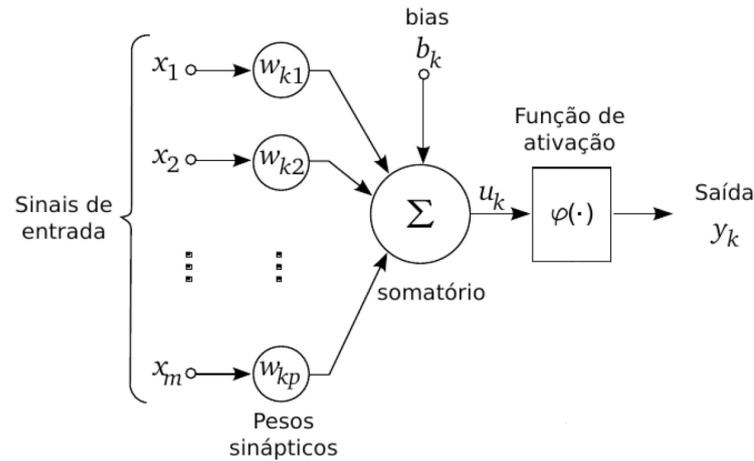


Figura 2.2: Esquema de um neurônio artificial (MCCULLOCH; PITTS, 1943).

- Nível de ativação  $\Sigma$ : determina a força cumulativa dos sinais de entrada escalado por cada peso sináptico correspondente, resultando na soma ponderada  $\sum w_j x_j$ ;
- Bias  $b_k$ : variável adicionada com o intuito de aumentar o grau de liberdade da função de ativação e ajustado da mesma forma que os pesos sinápticos. Importante pois possibilita ao neurônio saídas não nulas mesmo que todas suas entradas sejam nulas;
- Função de ativação  $\varphi(\cdot)$ : calcula o estado final do neurônio evitando o acréscimo dos valores de saída ao longo das camadas da rede. Na seção 2.3.1 é detalhado algumas das funções mais utilizadas na literatura e a que será utilizada neste trabalho.

Os neurônios artificiais ficam dispostos em camadas dentro da rede, onde as que recebem os sinais de entrada e as que se extraem os sinais de saída são denominadas camadas visíveis. As demais camadas intermediárias são denominadas camadas ocultas.

A distinção de cada tipo existente de RNA, segundo Haykin (2001) se dá em relação à três características: a função de ativação do neurônio artificial; a topologia da rede neural; e por fim, a regra de aprendizagem da RNA. A característica da topologia da rede define dois dos grupos mais utilizados na área de redes neurais artificiais: as RNAs não recorrentes ou *Feedforward*, excelentes em problemas de reconhecimento de padrões; e as RNAs recorrentes, que possuem realimentação de suas saídas para as entradas. Tais estruturas de rede possibilitam que as sinapses entre neurônios se interliguem de maneira não consecutiva e entre a mesma camada, tornando suas interconexões mais complexas

em relação às redes *Feedforward*. O esquema de rede neural *Feedforward* é apresentado na Figura 2.3

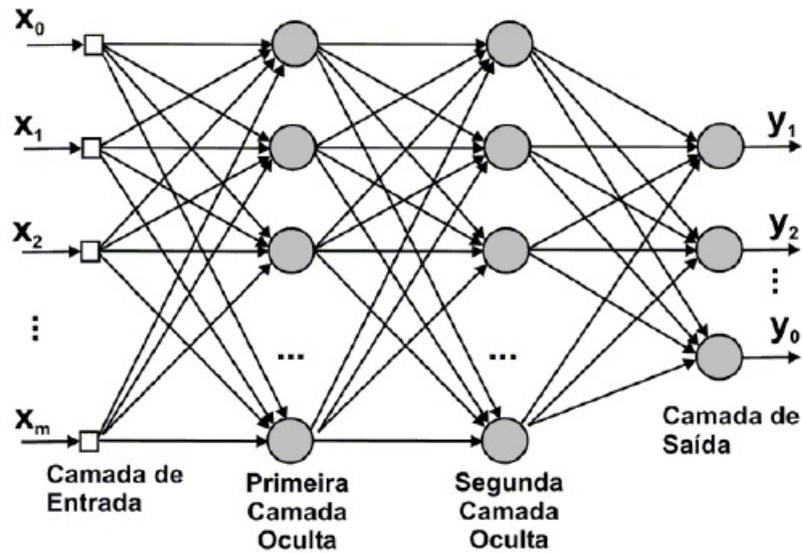


Figura 2.3: Esquema de rede neural *Feedforward* (FACURE, 2020).

A Figura 2.3 mostra o fluxo unidirecional das redes *Feedforward*, semelhante a um grafo acíclico, o que faz com que a rede seja uma função de sua entrada, não possuindo outro estado interno além dos pesos em si.

Já o processo de aprendizagem se dá através de um algoritmo que modifica os pesos sinápticos a fim de obter a maior generalização e reduzir erros quando confrontada com exemplos desconhecidos. A modificação dos pesos sinápticos faz parte da fase de treinamento da rede, mas é possível também que a rede neural altere sua própria topologia, através de decaimento dos pesos sinápticos, ou gerando novas conexões, de forma análoga a plasticidade cerebral<sup>4</sup>. Cada treinamento necessita de um vetor de entrada, porém a presença ou não de um vetor de saída divide o processo de treinamento em duas classes:

- Treinamento supervisionado;
- Treinamento não-supervisionado.

No treinamento não supervisionado, a rede não possui vetor alvo para as saídas, assim não realiza comparações para chegar a resposta ideal, seu conjunto de treinamento

<sup>4</sup>É a propriedade do sistema nervoso que permite o desenvolvimento de alterações estruturais em resposta à experiência, e como adaptação a condições mutantes e a estímulos repetidos.

modifica os pesos de forma a produzir saídas consistentes. O treinamento supervisionado, que é o utilizado neste trabalho, necessita do vetor alvo para comparação com a saída da rede. Assim, o erro encontrado entre os dois é realimentado para a rede e os pesos são atualizados a fim de minimizar este erro. O processo se repete até que o erro alcance um valor pré-determinado.

A arquitetura de rede usada neste trabalho é a rede neural recorrente (RNN), descrita na seção 2.3.3. A escolha da RNN justifica-se pois é considerado o tipo de rede com resultados estado-da-arte para problemas de previsão de valores no mercado financeiro, de acordo com Hansson (2017), o que viabiliza este modelo para o ambiente de transações de criptomoedas.

### 2.3.1 Funções de Ativação

As funções de ativação possuem um dos principais papéis no funcionamento de uma RNA, pois é decidido por elas se a informação transmitida no neurônio é considerada relevante e deve ser passada adiante, ou deve ser ignorada (SHARMA, 2020).

Existe uma pluralidade de funções, cada qual sendo indicada para cada tipo de problema. Por exemplo, as funções *step* e linear são simples e suficientes para criar um classificador binário, porém são incapazes de retropropagar o erro para ensinar a rede. Já as funções sigmóide e tangente hiperbólica, que são funções não-lineares, permitem ao modelo criar mapeamentos complexos entre as entradas e saídas da rede, o que se torna essencial ao modelar dados mais complexos, como por exemplo imagens, vídeos e conjuntos de dados não-lineares ou com alta dimensionalidade (SHARMA, 2020).

A função ReLU (*Rectified Linear Unit*) e a sua versão melhorada, a *Leaky*ReLU, que são utilizadas neste trabalho, são fáceis de otimizar, pois se assemelham à função identidade na metade positiva de seus domínios, como mostrado na Figura 2.4. Porém, a ReLU possui a desvantagem de produzir no neurônio valores zeros caso a soma ponderada se torne negativa, fazendo com que a fase de treinamento pare de surtir efeito na rede. Então Maas, Hannun e Ng (2013) propuseram a inserção de uma pequena inclinação  $\alpha$  para a parte negativa do domínio da ReLU. Esse  $\alpha$  é responsável por determinar a região positiva, denominada vazamento, da derivada da região negativa da *Leaky*ReLU.

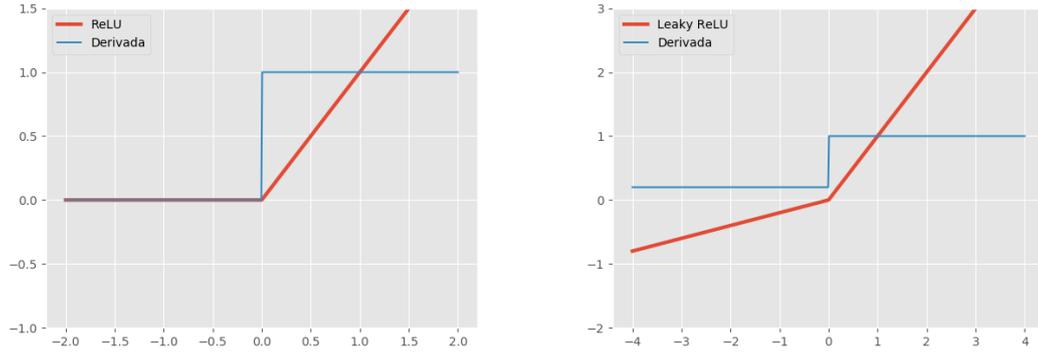


Figura 2.4: Gráfico das funções ReLU e *Leaky*ReLU, respectivamente (SHARMA, 2020).

### 2.3.2 Desempenho da Rede

Há diversas formas de se medir o desempenho de uma rede neural, sendo que algumas destas formas dependem do tipo de algoritmo de aprendizado. Um exemplo ao ser utilizado um algoritmo de aprendizado supervisionado é o de realizar a verificação de quantas amostras a rede acerta em relação às classes inseridas como parâmetro na entrada (CARUANA; NICULESCU-MIZIL, 2006).

Para o presente trabalho é definido a medida de desempenho através de correção de erros, que faz a avaliação da diferença entre o valor estimado e o valor real amostrado. Dos métodos de correção de erros, é utilizado o Erro Médio Percentual Absoluto (MAPE) (REN; GLASURE, 2009), que é definido na Equação (2.5)

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{Y_i - F_i}{F_i} \right| \quad (2.5)$$

onde  $n$  é o número total de amostras,  $i$  é o instante,  $Y_i$  representa o valor real a ser alcançado no instante  $i$  e  $F_i$  a previsão de  $Y_i$ .

### 2.3.3 Redes Neurais Recorrentes

As redes recorrentes (RNN) têm sido amplamente empregadas em problemas que envolvem memórias associativas bem como em problemas de otimização, devido à sua capacidade de manter informações das iterações anteriores, influenciando o resultado da

iteração atual. Esta capacidade pode ser interpretada como uma memória de curto prazo da rede. Somado a isso, o aprendizado por treinamento pode ser considerada memória de longo prazo da rede, o que resulta em modelos mais complexos, capazes de resolver uma gama maior de problemas (HAYKIN, 2001).

Um exemplo de rede neural artificial recorrente é apresentado na Figura 2.5.

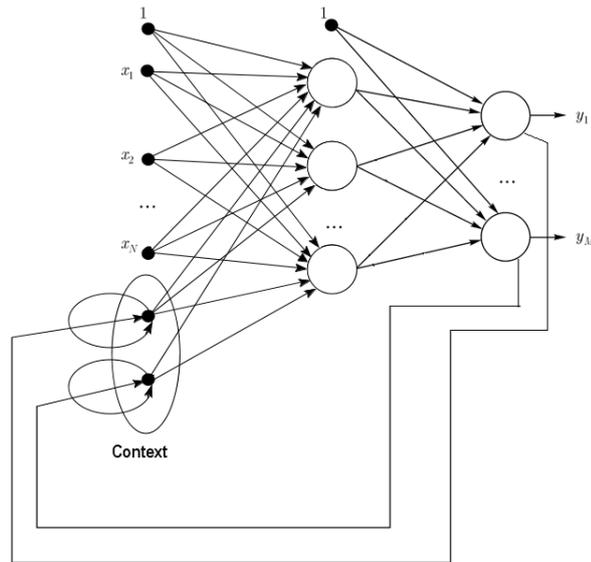


Figura 2.5: Esquema de rede neural recorrente (PERERVENKO, 2020)

A partir da Figura 2.5 é possível notar como as saídas se conectam com camadas anteriores através de uma camada denominada *context*, que serve para realimentar as camadas de neurônios intermediárias. Essa realimentação dos sinais em camadas anteriores e até na mesma camada produz um efeito de memória de curto prazo na rede, que é responsável pelo grande ganho dessas redes nas resoluções de problemas com grandes volumes de dados.

### 2.3.4 Redes *Long Short-Term Memory*

A rede *long short-term memory* (LSTM) é um modelo recorrente e profundo de rede neural, “capaz de aprender e interligar intervalos de tempo de mais de 1000 períodos, sem que haja perda na capacidade de memória de curto prazo. Isso se tornou possível através de um algoritmo eficiente baseado em gradiente para um fluxo de erro constante.” (HOCHREITER; SCHMIDHUBER, 1997). Sua motivação foi em resolver o “pro-

blema de desaparecimento de gradiente” que redes recorrentes possuem ao lidar com grande volume de dados, e acontece devido ao gradiente desaparecer ao ser retropropagado ao longo do tempo (HOCHREITER, 1998).

Em resumo, mantendo o fluxo de erro constante através de filtros no interior do neurônio foi possível resolver o problema do desaparecimento de gradiente, ao permitir o ajuste de pesos e truncar a sequência quando a informação é desnecessária. Este efeito pode ser interpretado como um esquecimento. Na Figura 2.3.4 é mostrado a organização interna de um neurônio da rede LSTM.

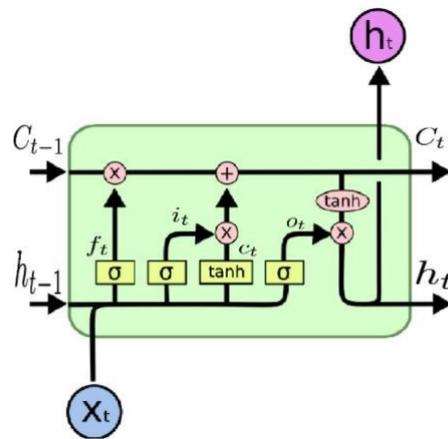


Figura 2.6: Esquema de um neurônio da rede LSTM (COLAH, 2020)

Os elementos do neurônio LSTM da Figura 2.3.4 são descritos em:

- Filtro  $f_t$ : responsável por decidir qual informação, advinda do estado do neurônio anterior  $h_{t-1}$ , é pertinente para a rede ou deve ser ignorada;
- Filtro  $i_t$ : possui um “portão de entrada” onde são calculados novos valores, que são adicionados ao estado do neurônio caso sejam relevantes;
- Filtro  $C_t$ : calcula um novo estado baseado no estado do neurônio anterior  $C_{t-1}$ , que é usado para calcular a saída do neurônio nesse passo de tempo, e também repassado para o próximo neurônio;
- Filtro  $o_t$ : faz uso de todas as informações passadas pelos filtros anteriores para gerar a saída, composta por este e vários passos de tempo anteriores.

---

Devido tais capacidades da rede LSTM de guardar e “esquecer” memória de curto prazo ao realizar treinamento com grandes entradas, ela se ajusta bem ao problema de previsão de valores (HANSSON, 2017).

## 3 Metodologia

O capítulo apresenta a base de dados utilizada, o pré-processamento realizado em cima destes dados e os detalhes do modelo LSTM que auxilia na análise da rentabilidade da Bitcoin e da Ethereum.

### 3.1 Base de Dados

Seguindo os trabalhos de Hegazy e Mumford (2016) e Chen, Narwal e Schultz (2019), são utilizados os valores diários de transação da Bitcoin em um dos experimentos e da rentabilidade intradiária tanto da Bitcoin como da Ethereum nos últimos anos.

O estudo faz o uso dos valores de abertura<sup>5</sup> e fechamento<sup>6</sup> durante o período de abril de 2013 e agosto de 2020 para a Bitcoin e para a série temporal do Dólar, e agosto de 2015 a agosto de 2020 para a Ethereum. As bases das criptomoedas foram extraídas de Coindesk (2020), e a do Dólar foi retirada de Investing.com (2020).

O valor da rentabilidade intradiária ( $ROI$ ) (THEODORO, 2020), tanto da Bitcoin como da Ethereum, é calculada pela Equação (3.1):

$$ROI_{(i)} = \frac{P_{a(i)} - P_{f(i-1)}}{P_{f(i-1)}} \quad (3.1)$$

onde  $P_{a(i)}$  representa o preço de abertura no dia  $i$  e  $P_{f(i-1)}$  representa o preço de fechamento no dia  $i-1$ .

A base de dados da Bitcoin reúne valores diários de 29/04/2013 a 24/08/2020, sendo que de 2013 até 25/07/2020 os valores são inseridos a um conjunto de treinamento e de 26/07/2020 a 24/08/2020 (30 dias) são inseridos em um conjunto de teste.

A base de dados da Ethereum reúne valores diários de 08/08/2015 a 24/08/2020, sendo que de 2015 até 25/07/2020 os valores são inseridos a um conjunto de treinamento

---

<sup>5</sup>Valor da primeira transação realizada no dia

<sup>6</sup>Valor da última transação realizada no dia

e de 26/07/2020 a 24/08/2020 (30 dias) são inseridos em um conjunto de teste.

Dados de Entrada				Dados de Saída	
$x_{(n-s)}$	$x_{(n-1)}$	...	$x_{(i-1)}$	$x_{(i)}$	
					Conjunto de Treinamento
					Conjunto de Teste

Figura 3.1: Tabela representando a divisão da base de dados em conjuntos de treinamento e teste

### 3.1.1 Pré-processamento dos Dados

É feita a normalização Min-Max (LOUKAS, 2020), técnica aplicada como parte da preparação dos dados para inserção em um modelo de aprendizagem de máquina, e tem como objetivo a alteração dos valores de entrada para uma escala única, sem distorcer as diferenças nos intervalos de valores ou perder informação. Os valores dos coeficientes de escala (mínimo e máximo) são calculados no conjunto de dados de treinamento e aplicados aos conjuntos de dados de treinamento e de teste. Assim os dados de entrada são reescalados para um intervalo fechado entre 0 e 1 conforme Equação (3.2).

$$x_{norm} = \frac{x_i - x_{min}}{x_{max} - x_{min}} \quad (3.2)$$

onde  $x_{norm}$  é o novo valor  $x$  no instante  $i$  após a normalização.

Os modelos foram baseados em códigos extraídos da plataforma Kaggle (2020), e são compostos por uma camada oculta LSTM com 96 neurônios e *batch* de tamanho 32, foi utilizada em um primeiro momento a função de ativação ReLU, porém esta foi substituída pela função de ativação *LeakyReLU*. A camada de saída utiliza a função de ativação linear. É utilizado o otimizador “ADAM” com 30 iterações do algoritmo em cada execução, a fim de que o modelo não fique dependente da inicialização dos pesos.

## 3.2 Modelo LSTM de Análise do Valor de Abertura da Bitcoin

A Figura 3.2 representa o modelo LSTM que recebe em sua entrada os valores de abertura diários da criptomoeda Bitcoin. A fim de melhorar o treinamento, são utilizados diferentes janelamentos. Para o valor de abertura da Bitcoin é feito o treinamento com 1, 2, 3, 5 e 7 dias de janelamento.

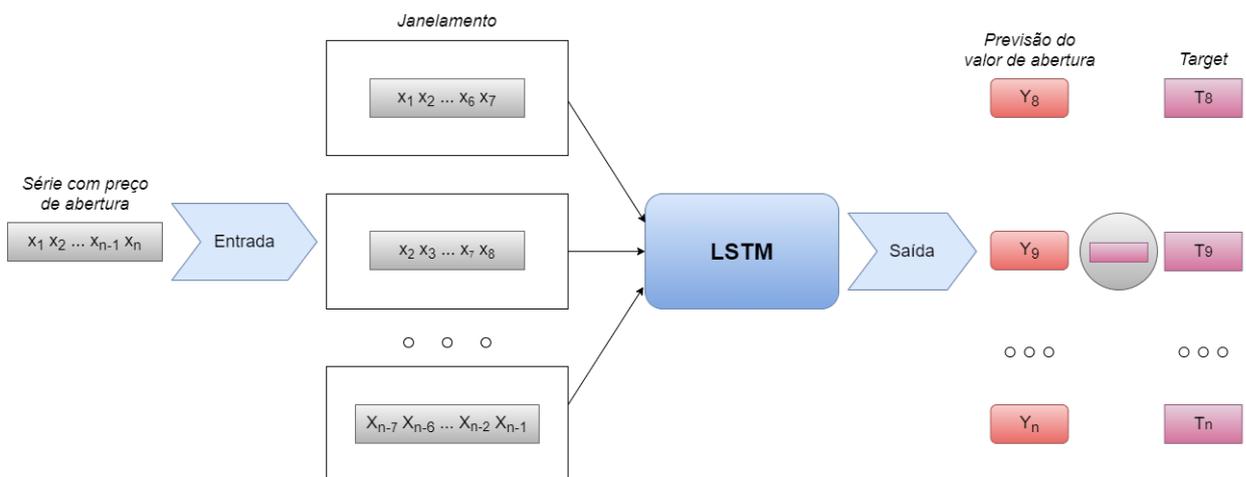


Figura 3.2: Esquema do modelo LSTM com a série financeira da Bitcoin como entrada e 7 dias de janelamento - autor

## 3.3 Modelo LSTM de Análise do Valor de Abertura da Bitcoin com o Dólar

A Figura 3.3 representa o modelo LSTM que recebe em sua entrada os valores de abertura diários da criptomoeda Bitcoin em conjunto aos valores de abertura diários do Dólar. Este experimento foi realizado para avaliar a hipótese de que o valor do Dólar está correlacionado ao valor da Bitcoin, e foi feito com janelamento de 1 e 3 dias.

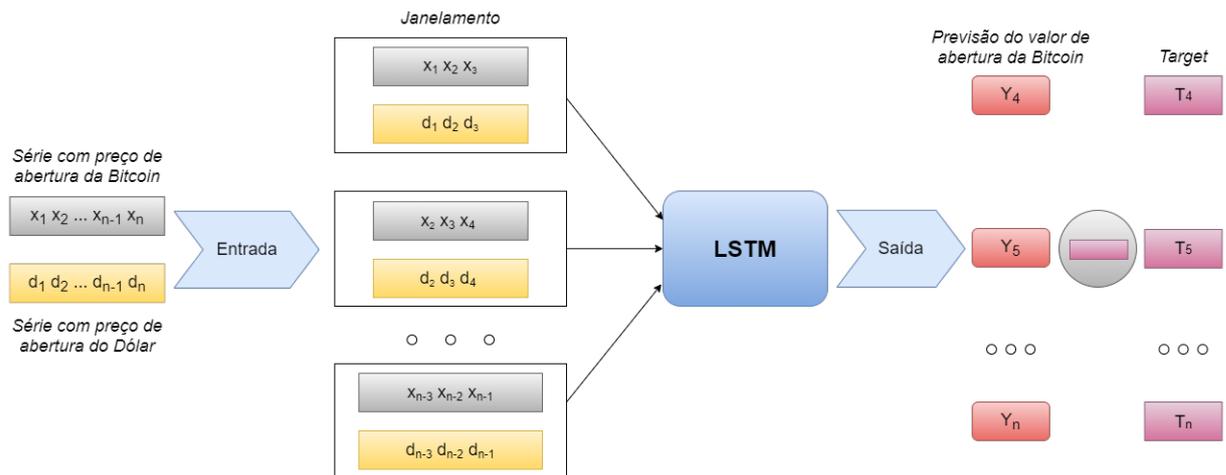


Figura 3.3: Esquema do modelo LSTM com a série financeira da Bitcoin e do Dólar como entrada e 3 dias de janelamento - autor

### 3.4 Modelo LSTM de Análise de Rentabilidade de Criptomoedas

A Figura 3.4 representa o modelo LSTM que recebe os valores de rentabilidade intradiária da Bitcoin ou da Ethereum em sua entrada. Este experimento é baseado no trabalho de ElBahrawy Laura Alessandretti e Baronchelli (2017), que evidencia a rentabilidade como uma característica significativa do crescimento da capitalização de mercado de uma criptomoeda. Para o modelo é realizado o treinamento com 1, 3 e 7 dias de janelamento.

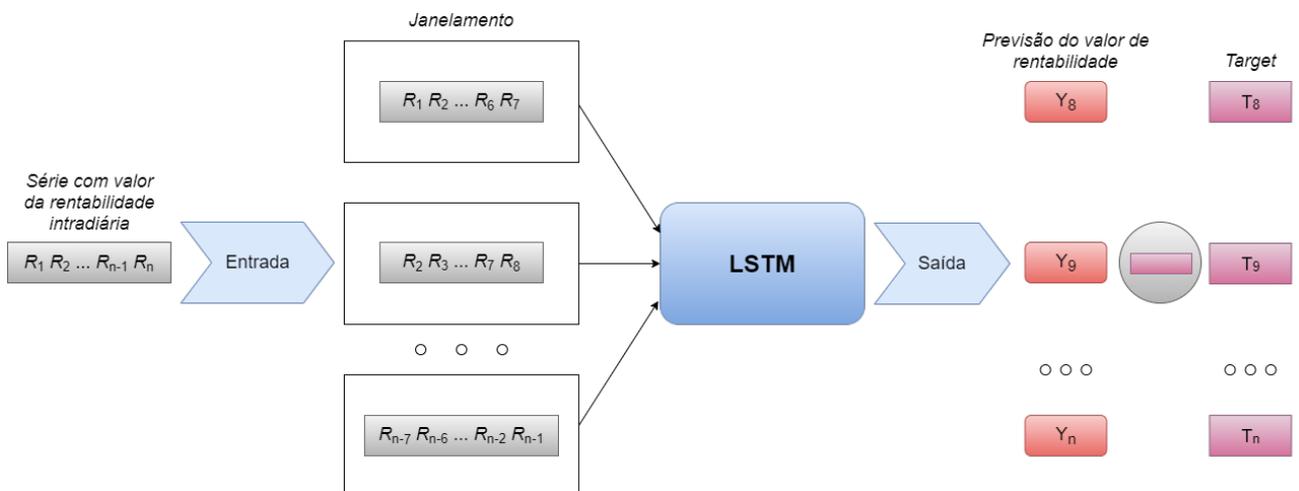


Figura 3.4: Esquema do modelo LSTM com a rentabilidade intradiária como entrada e 7 dias de janelamento - autor

## 3.5 Modelo LSTM de Análise da Rentabilidade com o Valor de Abertura de Criptomoedas

A Figura 3.5 representa o modelo LSTM que recebe tanto os valores de rentabilidade intradiária da Bitcoin ou da Ethereum com seus respectivos valores de abertura como entrada. Este treinamento busca avaliar se as duas séries em conjunto melhoram a previsão da rentabilidade futura que o modelo tenta aprender. Neste modelo é realizado o treinamento com 1 e 3 dias de janelamento.

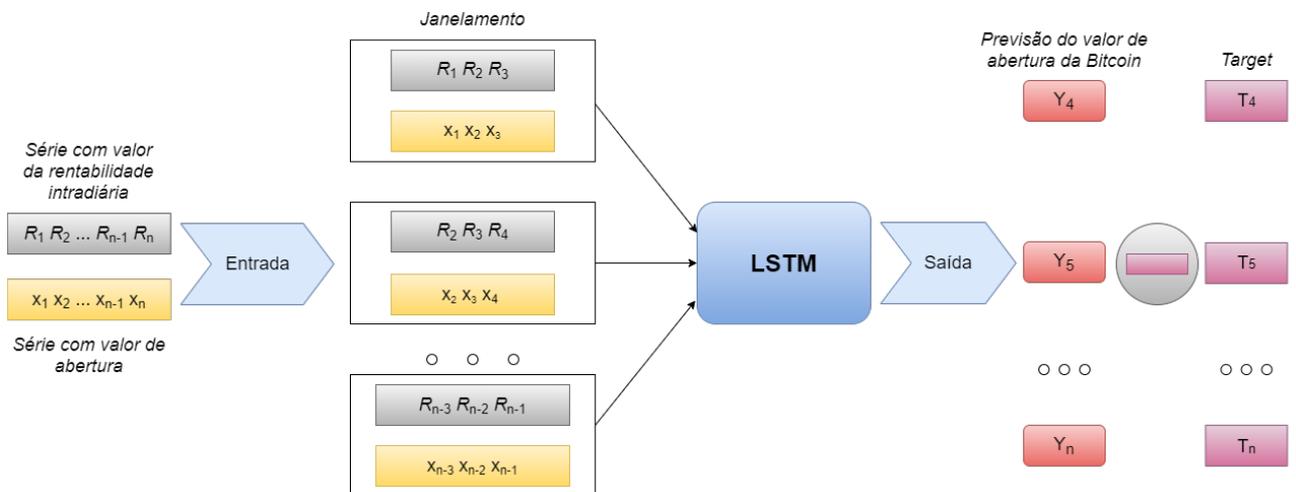


Figura 3.5: Esquema do modelo LSTM com a rentabilidade intradiária e o valor e abertura como entrada e 3 dias de janelamento - autor

## 4 Estudo de Caso

O capítulo aborda as configurações da máquina onde os experimentos foram realizados e as análises de cada um dos experimentos realizados.

### 4.1 Configurações do Ambiente

Os experimentos foram realizados em um máquina com processador Intel Core i7 de 3ª Geração de 2,1 GHz (4 núcleos e 8 threads) e 8 GB de memória RAM DDR3 1600MHz. A linguagem Python e as bibliotecas TensorFlow<sup>7</sup>, NumPy<sup>8</sup>, Pandas<sup>9</sup> e Keras<sup>10</sup> foram utilizadas na implementação do modelo e a biblioteca Matplotlib<sup>11</sup> foi utilizada na criação dos gráficos que auxiliaram na análise dos resultados deste trabalho.

### 4.2 Análise dos Experimentos

Cada modelo apresentado na metodologia foi treinado utilizando, de forma empírica, 300 épocas e 30 iterações do algoritmo. Ao final de cada iteração, é guardado o modelo e registrado seu MAPE.

Inicialmente a função de ativação ReLU foi escolhida, porém foi notado que frequentemente algumas iterações do algoritmo faziam com que o modelo não aprendesse e zerasse totalmente sua saída. Como forma de contornar este problema foi empregada a função de ativação *LeakyReLU*.

A Tabela 4.1 mostra os valores do pior e melhor MAPE resultantes de cada modelo treinado com o valor de abertura da Bitcoin, o MAPE médio de todas as iterações e a quantidade de iterações que resultaram em um MAPE abaixo de 2% para cada janela-mento.

---

<sup>7</sup>tensorflow.org

<sup>8</sup>numpy.org

<sup>9</sup>pandas.pydata.org

<sup>10</sup>keras.io

<sup>11</sup>matplotlib.org

Tamanho da janela	Melhor MAPE (%)	Pior MAPE (%)	MAPE total médio (%)	Iterações com MAPE < 2%
1 dia	1.17	3.34	2.10	13
2 dias	0.55	3.73	2.39	9
3 dias	0.73	4.07	2.39	11
5 dias	0.88	4.21	2.11	15
7 dias	0.94	5.20	2.24	13

Tabela 4.1: Tabela com valores de MAPE de cada janelamento

A Figura 4.1 apresenta uma comparação da série predita com a série histórica utilizando o modelo que forneceu o pior MAPE (gráfico à direita) e a topologia que forneceu o melhor MAPE (gráfico à esquerda) com o janelamento de 1 dia.

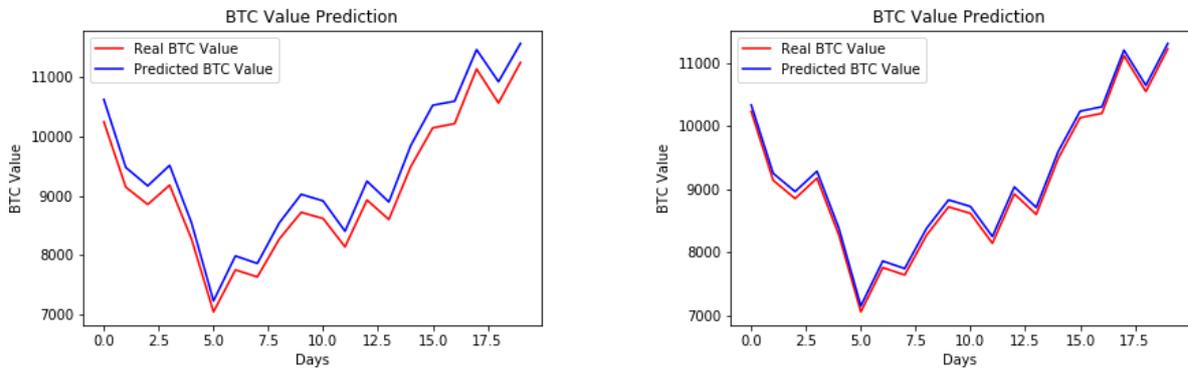


Figura 4.1: Gráficos com o pior e melhor resultados do janelamento de 1 dia, respectivamente

Observe que o modelo com janelamento de 1 dia conseguiu capturar, até no seu pior caso, o comportamento praticamente idêntico à série real da Bitcoin, e obteve o menor MAPE médio, com valor de 1.17%, porém também o maior valor de MAPE em seu melhor resultado comparado a todos os melhores resultados dos outros janelamentos apresentados na Tabela 4.1. O MAPE médio, somado de todas as iterações, obteve o valor de 2.10%, o menor entre todos os valores de MAPE médios, e que pode ser explicado pelo número de iterações que obtiveram MAPE menor que 2%, o segundo maior com 13 iterações.

A Figura 4.2 apresenta uma comparação da série predita com a série histórica utilizando o modelo que forneceu o pior MAPE (gráfico à direita) e a topologia que forneceu o melhor MAPE (gráfico à esquerda) com o janelamento de 2 dias.

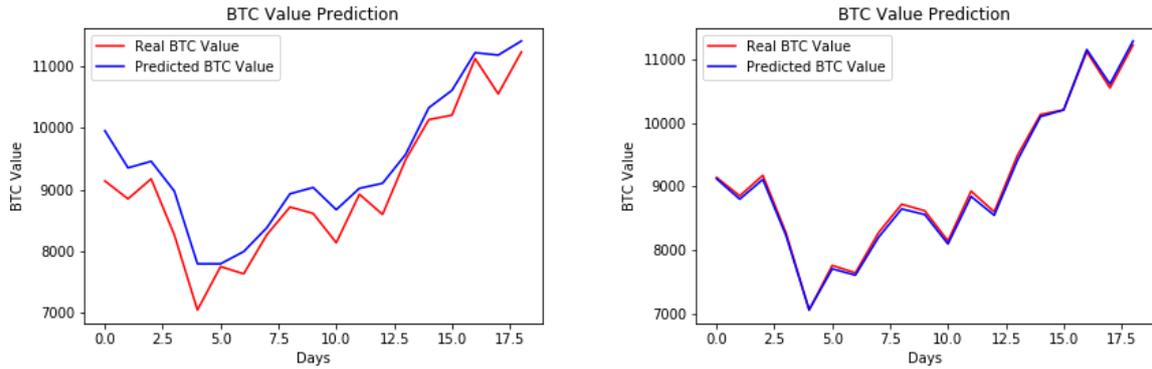


Figura 4.2: Gráficos com o pior e melhor resultados do janelamento de 2 dias, respectivamente

É possível notar que o janelamento de 2 dias, em seu pior caso, não conseguiu obter um bom resultado do treinamento, notado entre os dias 8 e 9, em que a série predita não seguiu o comportamento de queda da série real e previu a alta do valor. Contudo, obteve o melhor resultado com MAPE de 0.55% e com diferença quase imperceptível entre a série predita e a série real. Observe na Tabela 4.1 que este é o melhor resultado em relação aos outros janelamentos realizados, porém seu MAPE médio foi de 2.39%, ficando entre os piores valores de MAPE médio encontrados nos outros janelamentos, e que pode ser explicado por ter somente 9 iterações entre os valores de MAPE menor que 2%.

A Figura 4.3 apresenta uma comparação da série predita com a série histórica utilizando o modelo que forneceu o pior MAPE (gráfico à direita) e a topologia que forneceu o melhor MAPE (gráfico à esquerda) com o janelamento de 3 dias.

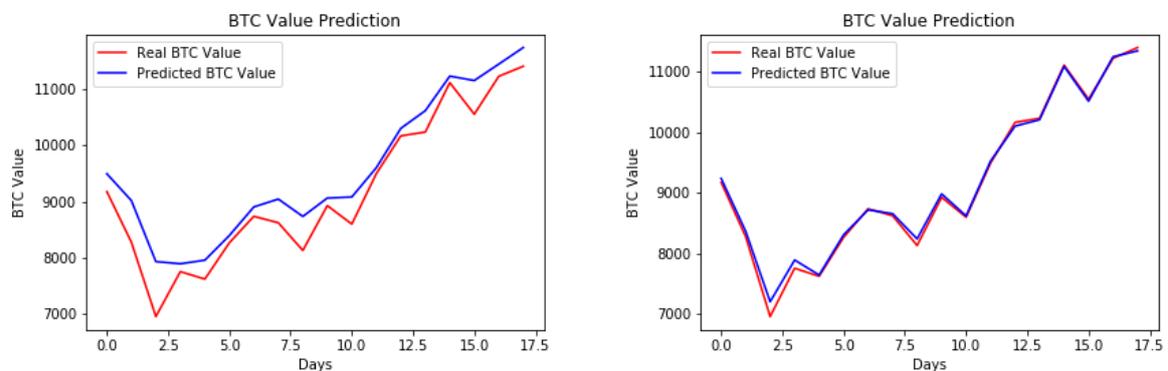


Figura 4.3: Gráficos com o pior e melhor resultados do janelamento de 3 dias, respectivamente

Se observa que os resultados do janelamento de 3 dias apresentaram uma piora em relação ao janelamento de 2 dias, sendo que a série predita no modelo com pior MAPE também não conseguiu aprender o comportamento do valor entre os dias 6 e 7. O modelo obteve um MAPE médio de suas iterações semelhante ao do janelamento de 2 dias, de 2.39%, porém com valores, tanto do melhor MAPE quanto do pior, acima dos encontrados no janelamento de 2 dias.

A Figura 4.4 apresenta uma comparação da série predita com a série histórica utilizando o modelo que forneceu o pior MAPE (gráfico à direita) e a topologia que forneceu o melhor MAPE (gráfico à esquerda) com o janelamento de 5 dias.

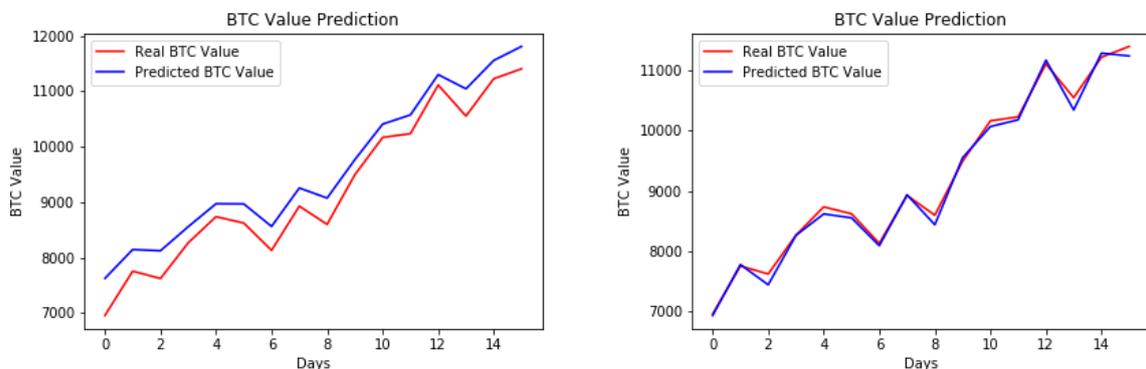


Figura 4.4: Gráficos com o pior e melhor resultados do janelamento de 5 dias, respectivamente

É possível observar para o janelamento de 5 dias que no gráfico com pior MAPE a série predita seguiu o comportamento da série real, porém com um distanciamento nítido entre os valores no mesmo dia. Já para o gráfico com melhor MAPE, o MAPE encontrado foi maior comparado aos janelamentos anteriores. Contudo, observe que o seu MAPE médio, soma de todas as suas iterações, foi de 2.11%, valor bem próximo do MAPE médio encontrado no janelamento de 1 dia, consequência das 15 iterações que obtiveram MAPE menor que 2%.

A Figura 4.5 apresenta uma comparação da série predita com a série histórica utilizando o modelo que forneceu o pior MAPE (gráfico à direita) e a topologia que forneceu o melhor MAPE (gráfico à esquerda) com o janelamento de 7 dias.

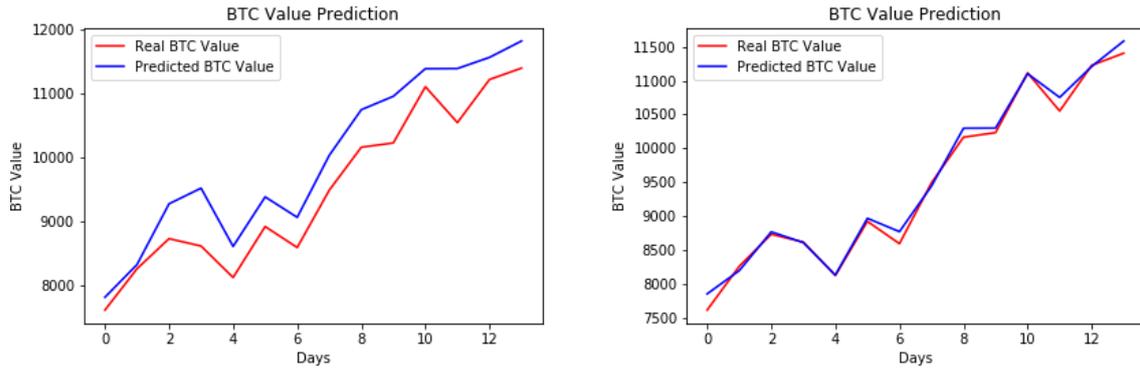


Figura 4.5: Gráficos com o pior e melhor resultados do janelamento de 7 dias, respectivamente

A partir dos gráficos da Figura 4.5 é possível observar que na pior iteração, o modelo não conseguiu aprender o comportamento da série real entre os dias 2 e 3 e se manteve distante do valor real em toda a extensão dos valores. Em relação ao gráfico do melhor MAPE encontrado, novamente houve um aumento no MAPE em relação aos janelamentos anteriores, com exceção ao janelamento de 1 dia. Ademais, seu valor de MAPE médio, somado de todas as iterações foi de 2.24%, valor menor do que os janelamentos de 2 e 3 dias, porém pior que os janelamentos de 1 e 5 dias.

Em resumo, o modelo com janelamento de 5 dias apresentou o maior número de iterações abaixo de 2% em valor de MAPE, o que resulta em um MAPE médio baixo, e o torna mais propenso em apresentar bons resultados em relação aos outros janelamentos. Um adendo ao janelamento de 2 dias, que obteve o menor MAPE entre todos os janelamentos, porém não foi escolhido por ter tido o pior resultado em relação às iterações, com apenas 9 iterações a baixo de 2%.

Contudo, Eroglu (2020) faz uma menção em relação aos métodos de *machine learning* empregados às séries financeiras, chegando à conclusão que modelos treinados para este problema não generalizam o conhecimento e estão somente memorizando regras básicas, e para que realmente as técnicas de *machine learning* possam ser aproveitadas, são necessários modelos mais especializados para a tarefa.

Para os experimentos em que a série temporal da Bitcoin e a do Dólar foram treinadas em conjunto, o modelo não foi capaz de aprender o comportamento da série em grande parte das iterações. Isso pode ser explicado por não existir relação entre,

por exemplo, a alta do Dólar e a valorização da Bitcoin, mesmo que ainda assim grandes quedas, como as ocorridas no terceiro trimestre de 2020, sejam sentidas pelos mais diversos ativos. A Figura 4.6 mostra tal comportamento, onde em azul está a taxa dos títulos do tesouro de 5 anos nos EUA e em laranja o preço do Bitcoin em Dólar.



Figura 4.6: Gráfico que representa a queda nos valores de títulos do tesouro nos EUA e na cotação da Bitcoin entre Setembro de 2019 e Junho de 2020, TradingView (2020)

No trabalho de ElBahrawy Laura Alessandretti e Baronchelli (2017), há um interesse na performance de curto prazo e rentabilidade da Bitcoin dada a sua rentabilidade. Assim, este trabalho inspirou o uso da rentabilidade como entrada do nosso modelo.

No entanto, os valores de rentabilidade encontrados são muito próximos a zero e o MAPE possui desvantagens ao tratar tais valores que tendem a zero Kim e Kim (2016), seu valor não foi computado para este experimento. Abaixo seguem os gráficos para comparação dos melhores resultados do modelo treinado e seu janelamento.

O gráfico do melhor resultado para o modelo treinado somente com os valores de rentabilidade da Bitcoin foi com janelamento de 7 dias, e está representado na Figura 4.7. O experimento com este janelamento obteve em sua maioria valores muito próximos de uma reta, evidenciando uma dificuldade do modelo identificar mudanças bruscas de valor na rentabilidade.

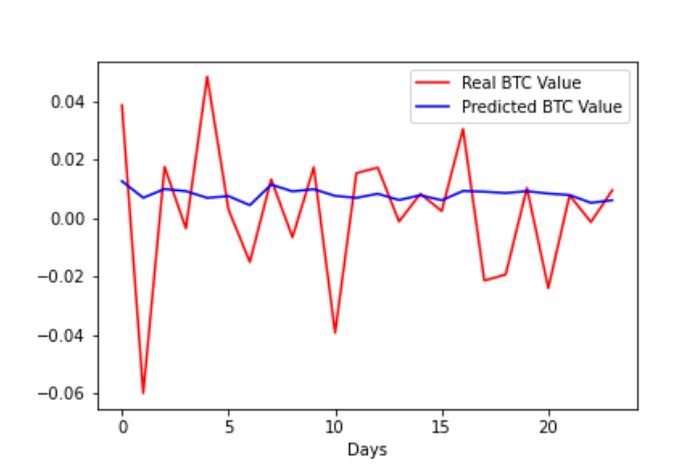


Figura 4.7: Gráfico com o melhor resultado encontrado com janelamento de 7 dias com a rentabilidade da Bitcoin

É possível observar pela Figura 4.7 que o modelo não conseguiu acompanhar a forte variação da rentabilidade da Bitcoin, praticamente traçando uma linha reta com poucos indícios de aprendizado do comportamento da série.

O gráfico do melhor resultado para o modelo treinado somente com os valores de rentabilidade da Ethereum foi com janelamento de 1 dia, e é representado na 4.8. O experimento com este janelamento foi o que obteve os resultados mais semelhantes aos experimentos com o valor de abertura da Bitcoin.

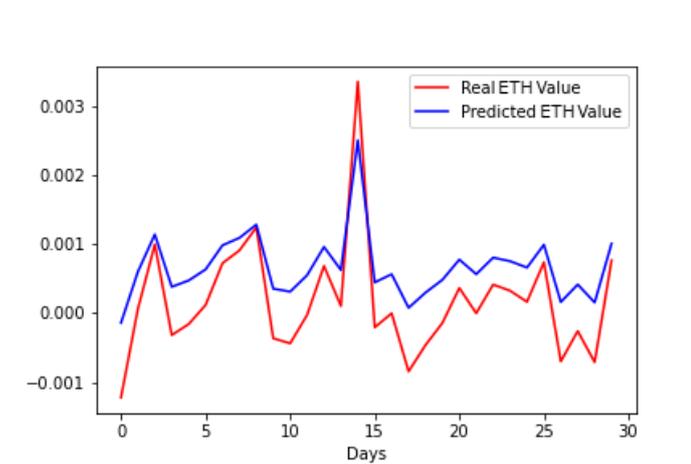


Figura 4.8: Gráfico com o melhor resultado encontrado com janelamento de 1 dia com a rentabilidade da Ethereum

Observe que o modelo obteve uma boa aproximação e foi capaz de acompanhar o comportamento da série de valores reais da Ethereum. Foi o experimento com uso da rentabilidade que mais se aproximou dos resultados dos experimentos do valor de abertura,

o que pode evidenciar a mesma situação citada por Eroglu (2020).

O gráfico do melhor resultado para o modelo treinado com os valores de rentabilidade e preço de fechamento da Bitcoin foi com janelamento de 3 dias, e é representado na 4.9. A grande maioria das iterações deste treinamento não foi eficaz, o que resultou em um gráfico de linha para estes resultados.

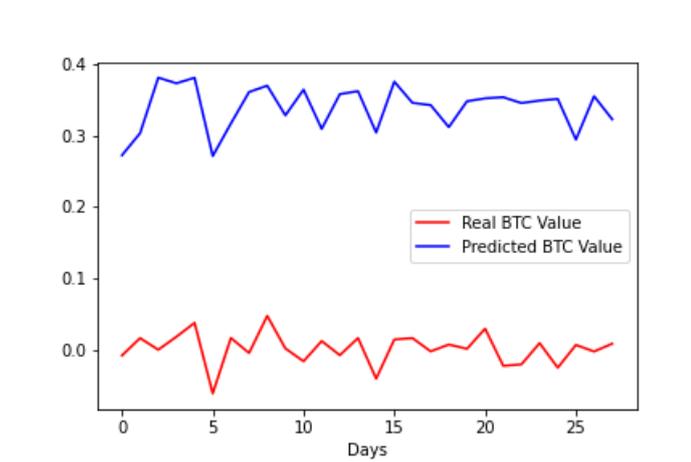


Figura 4.9: Gráfico com o melhor resultado encontrado com janelamento de 3 dias com a rentabilidade e o preço de fechamento como entrada para a Bitcoin

Na Figura 4.9 há sinais de que o modelo foi capaz de aprender parte do comportamento da série real, porém não obteve sucesso em aproximar os valores preditos, tendo o pior resultado entre os 4 modelos comparados no experimento. Tal resultado implica que o conjunto rentabilidade e valor de abertura da Bitcoin atrapalharam o treinamento e pioraram o resultado ao serem utilizados em conjunto.

O gráfico do melhor resultado para o modelo treinado com os valores de rentabilidade e preço de fechamento da Ethereum foi com janelamento de 3 dias, e é representado na 4.10.

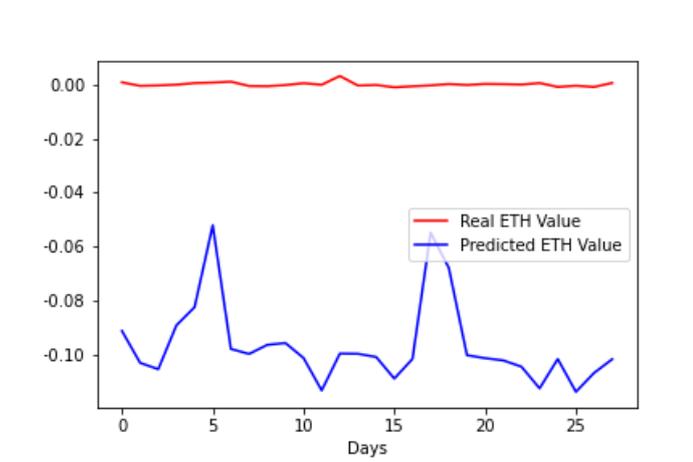


Figura 4.10: Gráfico com o melhor resultado encontrado com janelamento de 3 dias com a rentabilidade e o preço de fechamento como entrada para a Ethereum

Pela Figura 4.10 está nítida a incapacidade do modelo em aprender o comportamento da série de rentabilidade da Ethereum em conjunto com o valor de abertura, mesmo estando a um intervalo menor de distância do valor real comparado ao modelo análogo a Bitcoin.

Com os experimentos realizados com o valor de abertura da Bitcoin e do Dólar, podemos concluir que o fato do modelo, que recebeu na entrada somente os valores de abertura da Bitcoin, ter apresentado valores de MAPE tão baixos nos mostra que o treinamento foi eficaz e sugere a existência de um padrão de comportamento dos preços da série financeira da Bitcoin. Contudo, ao utilizarmos a mesma série da Bitcoin em conjunto com a do Dólar, foi observado o contrário, e a série do Dólar piorou os resultados do modelo.

Já os experimentos realizados com a rentabilidade de cada criptomoeda mostraram que a rentabilidade tornou mais difícil o aprendizado comparado ao valor de abertura, levando a resultados piores para a maioria dos treinamentos realizados. A piora nos resultados pode ser decorrente da significativa diminuição de grandeza do valor da rentabilidade comparada ao valor de abertura e um sinal de que o modelo necessita ser mais especializado.

## 5 Conclusão e Trabalhos Futuros

A literatura evidencia a efetividade dos modelos LSTM em aprender dependências de longo prazo, mesmo em bases de dados massivas. Porém, no caso de séries financeiras que não possuem qualquer tipo de sazonalidade perceptível e sequer uma base padronizada de características (RYLL; SEIDENS, 2019), a tarefa se torna complexa.

Por isto, neste trabalho foi desenvolvido um modelo LSTM para atacar o problema por meio de uma análise de retorno em investimentos feitos na Bitcoin e na Ethereum, tanto no âmbito dos valores de transação quanto na rentabilidade de cada criptomoeda. O modelo inicialmente utilizava a função de ativação ReLU com janelamento de 1 dia. Devido a certas iterações do treinamento não conseguirem aprender os valores, foi estudado o uso da função de ativação *LeakyReLU* e a escolha de diferentes janelamentos de forma empírica. A mudança na função de ativação foi capaz de corrigir as iterações em que o modelo zerava sua saída, e o janelamento melhorou o valor do MAPE médio do resultado.

Os experimentos com uso do valor de abertura apresentaram ótimos resultados, o que levantaram suspeitas da real eficácia destes modelos, e que foram abordadas por Eroglu (2020), contudo concluímos que o modelo com janelamento de 5 dias é o escolhido entre todos os outros janelamentos, devido a ter um dos melhores valores de MAPE médio e maior número de iterações ótimas abaixo de 2%, podendo oferecer melhor assistência a um investidor que procura uma ferramenta auxiliar de análise do mercado de criptomoedas, e não uma máquina que faça suas escolhas. Já os modelos que realizaram o treinamento com a rentabilidade, se faz necessário um aprimoramento do modelo para atribuição de mais características que possam melhorar seus resultados, e não que os piorem.

Como sugestão de trabalhos futuros que possam aprimorar o presente estudo, está o uso da técnica de *Multitask learning* fazendo o uso de séries históricas da mesma criptomoeda só que em diferentes *exchanges*<sup>12</sup>, como também dedicar mais tempo no

---

<sup>12</sup>Plataforma especializada na comercialização de criptomoedas

início dos experimentos ao chamado *Feature engineering* (NG, 2013) para a escolha das características mais significativas de uma série financeira de criptomoeda. Tal estratégia resultaria tanto em resultados menos dispersos entre uma iteração e outra, e diminuiria as chances de *overfitting*<sup>13</sup> no treinamento do modelo.

---

<sup>13</sup>Diminuição da generalização do modelo resultante da sobrecarga de treinamento

## Referências Bibliográficas

ADHIKARI, R.; AGRAWAL, R. K. An introductory study on time series modeling and forecasting. *arXiv preprint arXiv:1302.6613*, 2013.

BACK, A. *Hashcash - a denial of service counter-measure*. 2002. Disponível em: <http://www.hashcash.org/papers/hashcash.pdf>.

BINANCE. *What Is Proof of Work (PoW)?* 2020. Accessed: 2020-10-21. Disponível em: <https://academy.binance.com/en/articles/proof-of-work-explained>.

BROCKWELL, P. J.; DAVIS, R. A.; FIENBERG, S. E. *Time Series: Theory and Methods: Theory and Methods*. [S.l.]: Springer Science & Business Media, 1991.

CARUANA, R.; NICULESCU-MIZIL, A. An empirical comparison of supervised learning algorithms. In: *Proceedings of the 23rd international conference on Machine learning*. [S.l.: s.n.], 2006. p. 161–168.

CHEN, M.; NARWAL, N.; SCHULTZ, M. Predicting price changes in ethereum. *International Journal on Computer Science and Engineering (IJCSE) ISSN*, p. 0975–3397, 2019.

CIAIAN, P.; RAJCANIOVA, M.; KANCS, d. The economics of bitcoin price formation. *Applied Economics*, Taylor & Francis, v. 48, n. 19, p. 1799–1815, 2016.

COINDESK. *Bitcoin (USD) Price*. 2020. 2020-10-21. Disponível em: <https://www.coindesk.com/price/>.

COLAH. *LSTM Node*. 2020. 2020-10-21. Disponível em: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>.

ELBAHRAWY LAURA ALESSANDRETTI, A. K. R. P.-S. A.; BARONCHELLI, A. *Evolutionary dynamics of the cryptocurrency market*. 2017. 2017-11-15. Disponível em: <https://royalsocietypublishing.org/doi/full/10.1098/rsos.170623>.

ENKE, D.; THAWORNWONG, S. The use of data mining and neural networks for forecasting stock market returns. *Expert Systems with applications*, Elsevier, v. 29, n. 4, p. 927–940, 2005.

EROGOL. *Why mere Machine Learning cannot predict Bitcoin price*. 2020. Accessed: 2020-10-21. Disponível em: <https://erogol.com/cannot-predict-bitcoin-price-vanilla-machine-learning/>.

FACURE. *Redes Neurais Feedforward Densas*. 2020. 2020-10-21. Disponível em: <https://matheusfacure.github.io/2017/05/15/deep-ff-ann/>.

HAAVELMO, T. The probability approach in econometrics. *Econometrica: Journal of the Econometric Society*, JSTOR, p. iii–115, 1944.

HANSSON, M. On stock return prediction with lstm networks. 2017.

- HAYKIN, S. *Redes neurais: princípios e prática*. [S.l.]: Bookman Editora, 2001.
- HEGAZY, K.; MUMFORD, S. Comparative automated bitcoin trading strategies. *CS229 Project*, v. 27, 2016.
- HEID, A. Analysis of the cryptocurrency marketplace. *Retrieved February*, v. 15, p. 2014, 2013.
- HOCHREITER, S. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, World Scientific, v. 6, n. 02, p. 107–116, 1998.
- HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. *Neural computation*, MIT Press, v. 9, n. 8, p. 1735–1780, 1997.
- INVESTING.COM. *Dollar (BRL) Price*. 2020. 2020-10-21. Disponível em: <https://www.investing.com/currencies/usd-brl-historical-data>.
- INVESTOPEDIA. *How Does Bitcoin Mining Work?* 2020. Accessed: 2020-10-21. Disponível em: <https://www.investopedia.com/tech/how-does-bitcoin-mining-work/>.
- KAGGLE. *RNN for predicting closing price (using Keras)*. 2020. Accessed: 2020-10-21. Disponível em: <https://www.kaggle.com/daiearth22/rnn-for-predicting-closing-price-using-keras>.
- KIM, S.; KIM, H. A new metric of absolute percentage error for intermittent demand forecasts. *International Journal of Forecasting*, Elsevier, v. 32, n. 3, p. 669–679, 2016.
- LO, S.; WANG, J. C. Bitcoin as money? 2014.
- LOUKAS, S. *Everything you need to know about Min-Max normalization: A Python tutorial*. 2020. Accessed: 2020-10-21. Disponível em: <https://towardsdatascience.com/everything-you-need-to-know-about-min-max-normalization-in-python-b79592732b79>.
- MAAS, A. L.; HANNUN, A. Y.; NG, A. Y. Rectifier nonlinearities improve neural network acoustic models. In: *Proc. icml*. [S.l.: s.n.], 2013. v. 30, n. 1, p. 3.
- MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, Springer, v. 5, n. 4, p. 115–133, 1943.
- MCNALLY, S.; ROCHE, J.; CATON, S. Predicting the price of bitcoin using machine learning. In: IEEE. *2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP)*. [S.l.], 2018. p. 339–343.
- MITHANI, F.; MACHCHHAR, S.; JASDANWALA, F. A modified bpn approach for stock market prediction. In: IEEE. *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICIC)*. [S.l.], 2016. p. 1–4.
- NAKAMOTO, S. Bitcoin: A peer-to-peer electronic cash system. 2008.
- NELSON, D. M. Q. *Uso de redes neurais recorrentes para previsão de séries temporais financeiras*. UFMG, 2017.
- NG, A. Machine learning and ai via brain simulations. *Accessed: May*, v. 3, p. 2018, 2013.

PERERVENKO, V. *REDES NEURAIS DE TERCEIRA GERAÇÃO: REDES PROFUNDAS*. 2020. 2020-10-21. Disponível em: <https://www.mql5.com/pt/articles/1103>.

REIFF. *Bitcoin vs. Ethereum: What's the Difference?* 2020. 2020-10-24. Disponível em: <https://www.investopedia.com/articles/investing/031416/bitcoin-vs-ethereum-driven-different-purposes.asp>.

REN, L.; GLASURE, Y. Applicability of the revised mean absolute percentage errors (mape) approach to some popular normal and non-normal independent time series. *International Advances in Economic Research*, Springer, v. 15, n. 4, p. 409, 2009.

RYLL, L.; SEIDENS, S. Evaluating the performance of machine learning algorithms in financial market forecasting: A comprehensive survey. *arXiv preprint arXiv:1906.07786*, 2019.

SHAH, D.; ZHANG, K. Bayesian regression and bitcoin. In: IEEE. *Communication, Control, and Computing (Allerton), 2014 52nd Annual Allerton Conference on*. [S.l.], 2014. p. 409–414.

SHARMA. *Activation functions*. 2020. 2020-10-21. Disponível em: <https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6>.

STELER, F. W.; CERQUEIRA, A. H. *Cinco princípios básicos do Blockchain*. 2017. Disponível em: <http://cio.com.br/tecnologia/2017/03/06/cinco-principios-basicos-do-blockchain/>.

STENQVIST, E.; LÖNNÖ, J. *Predicting Bitcoin price fluctuation with Twitter sentiment analysis*. 2017.

THEODORO, D. *Cálculo de VaR para uma carteira de ações: Sistema informatizado para ações negociadas na Bovespa*. 2020. Accessed: 2020-10-21. Disponível em: [http://modelosfinanceiros.com.br/assets/documentos/calculo\\_de\\_var.pdf](http://modelosfinanceiros.com.br/assets/documentos/calculo_de_var.pdf).

TRADINGVIEW. *BTCUSD Crypto Chart*. 2020. Accessed: 2020-10-21. Disponível em: <https://www.tradingview.com/symbols/BTCUSD>.