

**CENTRO UNIVERSITÁRIO CARIOCA**  
**GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO**

**JOSÉ CARLOS DA SILVA FILHO**  
**MATHEUS MARQUES RODRIGUES**

**MINERAÇÃO E ANÁLISE DE DADOS PARA PREDIÇÕES EM**  
**CRIPTOATIVOS**

**RIO DE JANEIRO**

**2021**

**JOSÉ CARLOS DA SILVA FILHO**  
**MATHEUS MARQUES RODRIGUES**

**MINERAÇÃO E ANÁLISE DE DADOS PARA PREDIÇÕES EM  
CRIPTOATIVOS**

Trabalho de conclusão de curso  
apresentado ao Centro Universitário  
Carioca (UniCarioca), como requisito  
exigido parcial à obtenção do grau de  
Bacharel em Ciência da Computação.

Orientador (a): Prof.<sup>a</sup> Daisy Cristine  
Albuquerque da Silva, M.Sc.

RIO DE JANEIRO

2021

Silva Filho, José Carlos da.

Mineração e análise de dados para predições em criptoativos. / José Carlos da Silva Filho e Matheus Marques Rodrigues. Rio de Janeiro, 2021.  
59f.

Orientadora: Daisy Cristine Albuquerque da Silva.

Trabalho de Conclusão de Curso (Ciência da Computação) – Centro Universitário UniCarioca - Rio de Janeiro, 2021.

1. Regressão linear. 2. Bitcoin. 3. Traders. 4. Blockchain. 5. Halving. 6. Holders. 7. P2P. 8. Aprendizado de máquina. 9. KDD. I. Silva, Daisy Cristine Albuquerque da, prof. orient. II. Título.

CDD 005

**MINERAÇÃO E ANÁLISE DE DADOS PARA PREDIÇÕES EM  
CRIPTOATIVOS**

JOSÉ CARLOS DA SILVA FILHO

MATHEUS MARQUES RODRIGUES

Banca Examinadora:

---

Profª Daisy Cristine Albuquerque da Silva, M. Sc. - Orientadora  
Centro Universitário Carioca

---

Profº André Luiz Avelino Sobral, M. Sc. - Coordenador  
Centro Universitário Carioca

---

Profº Sérgio Assunção Monteiro, D. Sc. – Professor convidado  
Centro Universitário Carioca

JOSÉ CARLOS DA SILVA FILHO  
MATHEUS MARQUES RODRIGUES

**MINERAÇÃO E ANÁLISE DE DADOS PARA PREDIÇÕES EM  
CRIPTOATIVOS**

## **AGRADECIMENTOS**

A todos os professores do centro universitário carioca por todo ensinamento e experiências transmitidas.

A nós mesmos, como grupo, por toda dedicação depositada nesse projeto.

A nossa orientadora por todos os conselhos e direcionamentos para alcançarmos o objetivo deste projeto.

## RESUMO

A cada dia que passa o mercado de criptoativos se torna cada vez mais promissor e reconhecido. Com isso, há de se notar um aumento na quantidade e nos valores transacionados nesse novo ambiente não só por investidores antigos, mas também por novos usuários que buscam entrar nesse mundo de possibilidades.

Considerando esse cenário, nosso trabalho tomou como objetivo trazer um meio tecnológico para apoiar as decisões dos novos investidores trazendo um software com embasamento científico para entregar ao cliente final uma maior assertividade em seus aportes, fazendo com que seja analisado uma base de dados com todas as variações dentro de uma das maiores *exchanges* da América Latina.

Palavras-chave: Criptomoedas. Mineração de dados. Ciência de dados. Investimento. Regressão linear. Bitcoin. Traders. Blockchain. Halving. Holders. P2P. Aprendizado de máquina. KDD.

## **ABSTRACT**

With each passing day, the cryptoactives market becomes more and more promising and recognized. As a result, there is an increase in the amount and values traded in this new environment, not only for old investors, but also for new users seeking to enter this world of possibilities.

Considering this scenario, our work aimed to bring a technological means to support the decisions of new investors, bringing a software with a scientific basis to deliver to the end customer greater assertiveness in their contributions, making a database with all the variations within one of the largest exchanges in Latin America.

Keywords: Cryptocurrencies. Data mining. Data science. Investment. Linear regression. Bitcoin. Traders. Blockchain. Halving. Holders. P2P. Machine learning. KDD.



## LISTA DE ILUSTRAÇÕES

Figura 1 - Exemplificação da geração de hash.....	18
Figura 2 - Exemplificação do algoritmo de <i>hash</i> SHA-256.....	18
Figura 3 - Mercado <i>spot</i> da <i>exchange</i> Binance .....	20
Figura 4 - Mercado de margem da <i>exchange</i> Binance .....	21
Figura 5 - Mercado de investimentos da Binance .....	22
Figura 6 - Exemplo utilizando regressão linear .....	26
Figura 7 - Site principal da organização <i>Python</i> .....	32
Figura 8 - Tela principal do <i>Visual Studio Code</i> .....	34
Figura 9 - Tela inicial do programa Weka .....	35
Figura 10 - Tela principal do projeto no Github .....	36
Figura 11 - Seleções dos intervalos no site <i>CryptoDataDownload</i> .....	37
Figura 12 - Eliminação da primeira linha do csv .....	40
Figura 13 - Campos reconhecidos no Weka .....	41
Figura 14 - Ordenação decrescente do arquivo csv .....	41
Figura 15 - Configuração do Weka para criação do modelo matemático .....	42
Figura 16 - Resultado do treinamento do modelo .....	42
Figura 17 - Gráfico da métrica de coeficiente de correlação.....	43
Figura 18 - Visualização gráfica do modelo .....	45
Figura 19 - Resultado pós remoção de variáveis.....	46
Figura 20 - Resultado da execução com a base de testes .....	47
Figura 21 - Resultado da previsão da base de teste .....	48
Figura 22 - Visualização da classificação no Weka .....	49
Figura 23 - Conteúdo do arquivo de configurações .....	49
Figura 24 - Importação do csv obtido do sítio <i>CryptoDataDownload</i> .....	50
Figura 25 - Alocação dos eixos e criação do modelo matemático .....	50
Figura 26 - Retorno da API com dados atuais .....	51
Figura 27 - Previsão do fechamento e cálculo de variação .....	51
Figura 28 – Desenvolvimento do <i>front-end</i> .....	52
Figura 29 – Continuação do desenvolvimento do <i>front-end</i> .....	52
Figura 30 - Execução da aplicação.....	53
Figura 31 - Apresentação da previsão .....	54

## LISTA DE SIGLAS E ABREVIATURAS

API: *Application programming interface*

APY: *Annual percentage yield*

BTC: Bitcoin

ETH: *Ethereum*

GUI: *Graphic user interface*

IA: Inteligência artificial

INI: *Initialization file*

KDD - *Knowledge-discovery in databases*

OHLC: *Open, high, low e close*

P2P: *Peer-to-peer*

POS: *Proof of stake*

URL: *Uniform resource locator*

USD: *United states dólar*

UTC: *Coordinated universal time*

WEB: *World wide web*

## SUMÁRIO

<b>1. INTRODUÇÃO .....</b>	<b>10</b>
<b>1.1 Motivação e justificativa .....</b>	<b>11</b>
<b>1.2 Objetivos .....</b>	<b>11</b>
1.2.1 Objetivo geral .....	11
1.2.2 Objetivo sumarizado .....	12
<b>1.3 Organização do trabalho .....</b>	<b>12</b>
<b>2. CRIPTOMOEDAS .....</b>	<b>14</b>
<b>2.1 Bitcoin .....</b>	<b>15</b>
<b>2.2 Altcoin .....</b>	<b>15</b>
<b>2.3 Blockchain .....</b>	<b>16</b>
<b>2.4 Mineração de criptomoedas .....</b>	<b>17</b>
2.4.1 Proof of work.....	17
2.4.2 Proof of stake.....	19
<b>2.5 Exchanges .....</b>	<b>19</b>
<b>2.6 Tipos de operações.....</b>	<b>22</b>
2.6.1 Day trade .....	22
2.6.2 Swing trade.....	22
2.6.3 Buy hold.....	23
2.6.4 Scalper trade.....	23
<b>3. A DESCOBERTA DE CONHECIMENTO EM BASES DE DADOS .....</b>	<b>24</b>
<b>3.1 As fases do KDD .....</b>	<b>25</b>
3.1.1 Mineração de dados.....	26
3.1.2 Tarefas de mineração de dados .....	26
3.1.3 Infraestrutura.....	27
3.1.4 Usabilidade e relevância .....	28

3.2	Aprendizado de máquina.....	28
3.3	Regressão linear .....	30
4.	FERRAMENTAS TÉCNICAS .....	32
4.1	<i>Python</i> .....	32
4.1.1	Sklearn.....	33
4.1.2	Pandas.....	33
4.1.3	Json .....	33
4.1.4	Tkinter .....	34
4.2	<i>Visual Studio Code</i> .....	34
4.3	Weka .....	35
4.4	Github.....	36
5.	BITCOIN <i>PREDICTION</i> .....	37
5.1	Definição da base de dados .....	37
5.2	Manipulação da base de dados.....	39
5.3	Mineração da base de dados.....	41
5.4	Validação da base de dados.....	46
5.5	Importação da base de dados .....	49
5.6	Criação do modelo .....	50
5.7	Obtenção dos dados atuais.....	50
5.8	Previsão do fechamento .....	51
5.9	Apresentação do resultado .....	51
6.	RESULTADOS .....	55
7.	CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS .....	56
8.	REFERÊNCIAS BIBLIOGRÁFICAS .....	57

## 1. INTRODUÇÃO

Criptomoedas são ativos digitais que tem como base a produção descentralizada e a utilização de tecnologias criptográficas e de *blockchain* para garantir a autenticidade das transações dos seus usuários, tornando as operações seguras e anônimas.

Por conta dessas características, cada vez mais pessoas tem aderido e acreditado nessa revolução cambial, fugindo assim do controle governamental em cima das moedas fiduciárias, onde ele tem total controle da produção e distribuição, fazendo com que seja manipulada de acordo com interesses políticos e econômicos de determinados países.

Em consequência da falta de controle central, as criptomoedas, em sua maioria (existem criptomoedas centralizadas e que também possuem controle de emissão), possuem uma alta volatilidade de preços e são muito utilizadas por investidores para operações a curto e médio prazo. Os *traders*, estão sempre em busca das melhores estratégias para realizar operações lucrativas. Assim, utilizaremos algumas ferramentas de mineração e análise de dados para criar uma aplicação capaz de prever os valores do preço de fechamento dentro de um intervalo de tempo ideal para realizar operações mais lucrativas.

Com a ajuda da ferramenta Weka, vamos entender as variáveis que envolvem uma operação de *trade* e filtraremos quais delas são realmente importantes para que seja gerado um modelo capaz de prever os valores de fechamento de uma criptomoeda. E para isso, também precisamos entender o processo de mineração e análise de dados, passando pelas etapas de seleção, pré-processamento, formatação, mineração e validação dos dados, antes de propriamente utilizar um algoritmo de análise de dados em uma aplicação.

Para nossa aplicação em si, veremos como o *Python* é uma linguagem extremamente útil para um cientista de dados, contando com diversas bibliotecas de análise e mineração de dados. Além disso, a linguagem possui uma biblioteca nativa para o desenvolvimento de uma interface mais amigável para o usuário.

Ao fim deste trabalho de conclusão de curso, teremos um conhecimento abrangente sobre o mercado de criptomoedas, suas principais tecnologias e as ferramentas necessárias para criar uma aplicação capaz de realizar predições de fechamento do Bitcoin em um determinado intervalo com análise e mineração de dados.

## **1.1 Motivação e justificativa**

Com o aumento do valor de mercado e do crescimento de comercialização com pagamento em criptomoedas, cada vez mais pessoas estão entrando no mundo das cibermonedas. De acordo com a pesquisa realizada pelo Patrick Bucquet, Marie Lermite e Ally Jo (2019) o número de *traders* ativos no mundo inteiro foi de cerca de 42,9 milhões de usuários. Já em abril de 2021, foi registrado o número de 72 milhões de investidores segundo levantamento do site *BuyBitcoinWorldwide*, o que representa um aumento de aproximadamente 68% em quase 2 anos.

Com base nesses dados, e no artigo levantado por alunos da Fundação Getúlio Vargas, Chague, Fernando e Giovannetti, Bruno (2019) apenas 9% de todos os investidores de *day trade* obtiveram lucro no Brasil.

Sendo assim, elaboramos um projeto a fim de auxiliar investidores na tomada de decisões em suas operações, agregando um dado tecnológico sobre a possível variação de determinada criptomoeda.

## **1.2 Objetivos**

### **1.2.1 Objetivo geral**

Este trabalho tem como objetivo auxiliar na tomada de decisão do usuário final, efetuando previsão com base no intervalo de maior assertividade estatística utilizando algoritmos que geram modelos matemáticos a fim de determinar atual e iminente tendência de mercado, garantindo um maior retorno financeiro do valor aportado.

A base de dados será estudada através da função de regressão linear, que tem como princípio a definição de uma trajetória retilínea sob os valores apresentados, resultando em um número que consiga referenciar e se aproximar, da melhor maneira possível, do valor desejado.

Todo o algoritmo foi desenvolvido através da linguagem de programação *Python* com a importação da biblioteca *sklearn*, dedicada ao estudo de *machine learning*.

### 1.2.2 Objetivo resumido

- Estudar todas as variações, considerando o intervalo com maior aproveitamento estatístico, da criptomoeda Bitcoin.
- Gerar um modelo matemático através do uso de regressão linear.
- Apresentar ao usuário final um valor previsto do possível valor do próximo *close* da determinada moeda.

## 1.3 Organização do trabalho

O trabalho será estruturado da seguinte forma:

- **Capítulo 2 – Criptomoedas:**

Apresentação do mercado de criptomoedas com explicação técnica sobre o Bitcoin e as altcoins, fundamentação conceitual da tecnologia de *blockchain*, as formas de mineração das moedas e o qual é o impacto na rede, as casas de câmbio (*exchanges*) e quais são as possíveis operações assim como as classificações dos investidores com base na frequência de operações.

- **Capítulo 3 – A descoberta de conhecimento em bases de dado:**

Aprofundamento de todo o embasamento teórico para compreensão da descoberta do conhecimento, desde o entendimento das necessidades das organizações até o conceito da técnica e algoritmo de regressão linear utilizados para a validação e utilização na aplicação.

- **Capítulo 4 – Ferramentas Técnicas**

Apresentação das ferramentas e tecnologias utilizadas na criação do algoritmo de previsão, comentando principalmente sobre a linguagem de programação definida e as principais bibliotecas trabalhadas.

- **Capítulo 5 – Bitcoin *Prediction***

Apresentação de todo o desenvolvimento do trabalho, passando pela decisão da fonte de dados e as motivações, assim como as manipulações necessárias para se trabalhar e gerar o modelo matemático. Também será transparecido o desenvolvimento *back-end* e a apresentação do *front-end* ao usuário final.

- **Capítulo 6 – Resultados:**

Apresentação sucinta dos resultados obtidos após a validação do modelo de regressão linear através do programa Weka, informando se o projeto atingiu seu objetivo e se pode ser considerado nas operações dos investidores.

- **Capítulo 7 – Considerações Finais:**

Consolidação do trabalho, apresentando um resumo do desenvolvimento, dos seus resultados e sugestões para trabalhos futuros baseados no atual projeto.



## 2. CRIPTOMOEDAS

Criptomoeda é um tipo de dinheiro, uma moeda de troca como as quais estamos acostumados, com a diferença de serem totalmente digitais. E não são emitidas por nenhum governo, como é o caso da nossa moeda, o real.

Para explicar como é possível, o autor do livro “Bitcoin: A moeda na era digital”, Fernando Ulrich fez a seguinte analogia: “O que o e-mail fez com a informação, o Bitcoin fará com o dinheiro” (Ulrich, 2014). Antes da internet, as pessoas utilizavam cartas para enviar mensagens a quem estivesse em outro lugar, dependiam dos correios. A necessidade de um intermediário para entregá-las fisicamente é algo que passa longe do imaginário de quem tem acesso a e-mail e outros serviços de mensageria (Infomoney, 2021).

Segundo o site Bitcoin.org, em 1998 Wei Dai descreveu o conceito de criptomoedas sugerindo usar a criptografia para controlar a emissão e as transações realizadas com um novo tipo de dinheiro, dispensando a necessidade de uma autoridade central, como acontece com as moedas convencionais (Infomoney, 2021).

As criptomoedas podem ser utilizadas com as mesmas finalidades do dinheiro físico. Suas principais funções são: meio de troca, facilitar as transações comerciais; reserva de valor, preservar poder de compra futuro; e também unidade de conta, quando os produtos são precificados e o cálculo econômico é realizado em função dela (Infomoney, 2021).

Segundo Ulrich (2014), moedas como BTC ainda não adquiriram o status de unidade de conta, devido à sua grande volatilidade de preços. Esta volatilidade é muito explorada por *traders*, investidores do mercado financeiro que visam obter lucros com operações de curto prazo (Infomoney, 2021).

A variação na cotação das criptomoedas segue a boa e velha lei da oferta e demanda. Ou seja, é normal que em épocas em que ganham mais atenção e visibilidade seus preços subam. “Há somente um número limitado de Bitcoin em circulação e novos Bitcoins são criados em uma taxa previsível e decrescente, o que significa que a demanda deva seguir este nível para manter seu preço estável.” (Bitcoin.org, 2021).

Assim como ocorre com ações na bolsa de valores e outros tipos de investimentos onde há a possibilidade de obter lucros com operações no curto prazo, há muitas pesquisas e estudos acerca de técnicas para *traders*, mineração e análise de dados, algoritmos de automação e diversas outras ferramentas e pesquisas que buscam entender o comportamento do mercado de criptomoedas.

## 2.1 Bitcoin

O conceito do Bitcoin surgiu numa publicação de 2008 por Satoshi Nakamoto. No ano de 2009, aplicou o código e “garimpou” as primeiras moedas digitais, dando início ao Bitcoin (The Capital Advisor, 2021).

A mais famosa das criptomoedas usa um código complexo, que não pode ser alterado, e suas transações são protegidas por criptografia. As transações são validadas por um grupo de pessoas, os mineradores, que através de seus computadores, gravam estas transações na *blockchain* (Infomoney, 2021).

Nesse processo de mineração, se criam as moedas. Conforme os mineradores validam as operações, eles completam “blocos” do sistema (por isso o nome *blockchain*). Por cada bloco terminado, eles ganham frações de novos Bitcoins como recompensa (os *satoshis*, a menor unidade que se pode fracionar o Bitcoin), sendo que a cada 210 mil blocos, a recompensa cai pela metade, no processo de *halving* (Infomoney, 2021).

Todas estas características tornam o Bitcoin uma moeda altamente confiável e seguro para *holders*, que utilizam o Bitcoin como investimento a longo prazo, visando lucros a partir de sua extrema escassez e aumento de demanda e para *traders*, que aproveitam a alta volatilidade da moeda para lucrar com operações de curto prazo.

## 2.2 Altcoin

Atualmente existe uma extensa quantidade de aplicações e outras moedas virtuais que agregam ainda mais valor para o mercado de criptomoedas (Bitcoin Trade, 2019).

As altcoins são criptomoedas alternativas, lançadas após o Bitcoin, se beneficiando do mundo de possibilidades que o Bitcoin criou, essas moedas virtuais traçam objetivos e propostas diferentes, buscando atender diversas necessidades que não foram supridas pelo BTC e criando horizontes de possibilidades e soluções (Bitcoin Trade, 2019).

Enquanto algumas altcoins surgiram da motivação de trazer melhorias para as funcionalidades do Bitcoin, outras resolveram aproveitar a oportunidade de melhorar serviços fora do mundo das criptomoedas e não serem apenas uma moeda de troca através de outras funcionalidades (Bitcoin Trade, 2019). Um exemplo é o *Ethereum* (ETH), moeda utilizada em transações ou execução de contratos inteligentes (*smart contracts*) na rede *blockchain* descentralizada chamada *Ethereum*, onde os usuários da rede armazenam e processam essas informações (Coinext, 2021).

## 2.3 Blockchain

A *blockchain* foi introduzida através do Bitcoin sendo uma tecnologia utilizada para evitar a duplicidade de mesmas transações, o termo *blockchain* se refere a uma corrente de blocos que são distribuídos entre diversos sistemas computadorizados, armazenando informações de maneira imutável, onde nenhum usuário tem a possibilidade de corromper ou a alterar depois que um registro for compartilhado.

Embora as transações sejam compartilhadas com todos os computadores da rede de maneira transparente, todos os dados dos seus usuários são assegurados através de criptografia. Todas as movimentações registradas podem ser verificadas, tornando a *blockchain* à prova de violação.

Quando uma transação é efetuada, ela recebe uma assinatura digital que contém o *timestamp* (data e hora da transação), o *hash do bloco atual* e o *hash* do último bloco, essas informações são verificadas pela rede e só é escrita após

aprovação da própria que passa por um consenso, onde será necessário aprovação de 50% da rede mais 1 participante.

## 2.4 Mineração de criptomoedas

A mineração de criptomoedas é o ato voluntário dos usuários que se dispõem a serem validadores de novos blocos toda vez que há uma transação a ser registrada na *blockchain*, com isso eles recebem em troca algumas frações de moedas.

Esse processo também é responsável pela introdução de novas moedas em circulação no mercado, que depende do protocolo do criptoativos.

A mineração acontece utilizando um mecanismo de consenso, que é definido como um “procedimento através do qual todos os pares da rede blockchain chegam a um acordo comum sobre o estado atual do livro contábil distribuído.” (Criptofácil, 2019).

Os dois principais e mais utilizados mecanismos são: *proof of work* e *proof of stake*.

### 2.4.1 Proof of work

O mecanismo de *proof of work* (prova de trabalho), utilizado atualmente na rede Bitcoin, consiste na utilização de recurso computacional onde os mineradores competem entre si para encontrar uma solução para um “quebra-cabeças” matemático com bastante complexidade. A dificuldade do trabalho nem sempre é a mesma, sendo ajustado a cada 10 minutos para geração de novos blocos.

A função do minerador nesse mecanismo é adivinhar um número aleatório que forneça o *hash* certo para o bloco trabalhado. Existem duas informações que precisamos entender:

- *Hash*: “é qualquer algoritmo que mapeie dados grandes e de tamanho variável para pequenos dados de tamanho fixo.” (Techtudo, 2012).

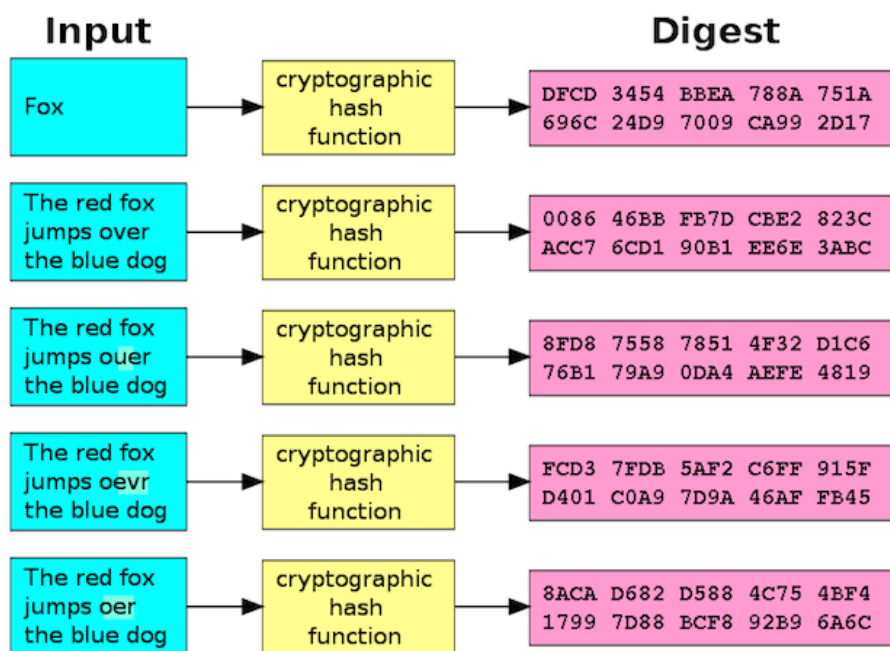


Figura 1 - Exemplificação da geração de *hash*

Fonte: Wikimedia Commons

- *Nonce*: Um número aleatório utilizado uma única vez.

Quando se deseja verificar um próximo bloco da *blockchain*, o minerador pega o *hash* do último bloco, adiciona o atual bloco de transações e depois adiciona o *nonce* ao final do bloco. Com essas informações é possível iniciar os cálculos utilizando a função *hash* para obter a sequência que tenha um certo número de zero à sua frente.

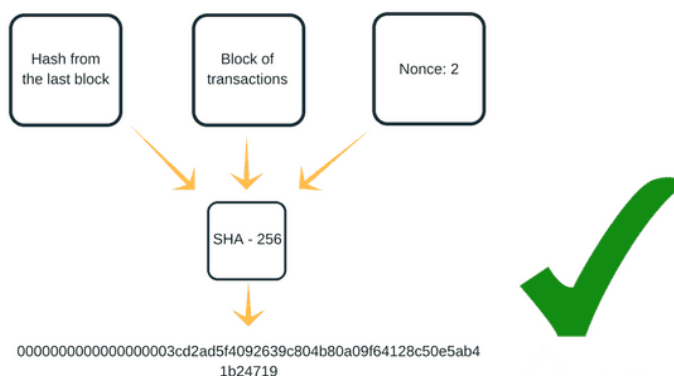


Figura 2 - Exemplificação do algoritmo de *hash* SHA-256

Fonte: Criptofacil

O primeiro minerador a descobrir tal hash recebe a recompensa pelo trabalho.

#### 2.4.2 *Proof of stake*

O mecanismo de *proof of stake* (prova de aposta) foi criado para ser alternativo ao *proof of work*, propondo solucionar alguns problemas deste consenso: o alto custo energético e a alta demanda por tecnologias cada vez mais rápidas. Embora os dois possuam a mesma finalidade, que é validar as transações dos usuários na *blockchain*, o processo é muito diferente que consiste numa seleção pseudoaleatória para definição o nó validará o próximo bloco da rede. A seleção é intitulada desta forma pois há alguns fatores que podem aumentar a probabilidade da seleção de um determinado nó, como por exemplo: o tempo que o usuário dispõe da moeda ou a quantidade que possui. Esses fatores podem mudar dependendo da implementação do algoritmo de POS.

Nesse consenso, os blocos não são minerados de fato e sim “forjados”, o usuário dispõe de uma determinada quantidade das moedas onde elas serão presas na rede como parte da comprovação da sua participação.

Quando o nó é escolhido, ele será responsável por verificar a validade da transação no bloco, e se for positivo, é introduzido na *blockchain*. E como forma de incentivar e recompensar o participante é entregue o valor das taxas de transações que foram associadas ao bloco.

Caso o investidor tenha a pretensão de sair do consenso ele precisará aguardar um período para que a rede valide que não houve nenhum bloco fraudulento adicionado na *blockchain* pelo usuário.

### 2.5 *Exchanges*

*Exchanges* estão para criptomoedas assim como a bolsa de valores estão para o mercado de ações. Ou seja, são plataformas digitais que facilitam a compra, venda

e a troca de moedas digitais como bitcoins e altcoins. Existem as que negociam apenas BTC e outras que fornecem uma grande variedade de criptomoedas (Bitcoin Trade, 2021).

As *exchanges* conectam compradores e vendedores, garantindo uma transação prática e segura. Embora não haja nenhum impedimento entre a troca direta de criptomoedas (P2P), as exchanges são o ambiente perfeito para um *trader*, que terá à sua disposição uma plataforma segura onde poderá realizar operações de forma instantânea, sem precisar de um contato direto com o vendedor/comprador. E assim como as corretoras tradicionais, elas também podem realizar vendas diretas de criptoativos, onde acumulam diferentes moedas compradas a baixo custo e vendem pela cotação atual mais as taxas de serviço (Bitcoin Trade, 2021).

Algumas plataformas dispõem de grandes variedades de opção de investimentos, podendo ter:

- Mercado *spot*: A principal utilização, que reúne diretamente os comprados com os vendedores onde a cotação é definida pela última transação de venda mais barata disponível.

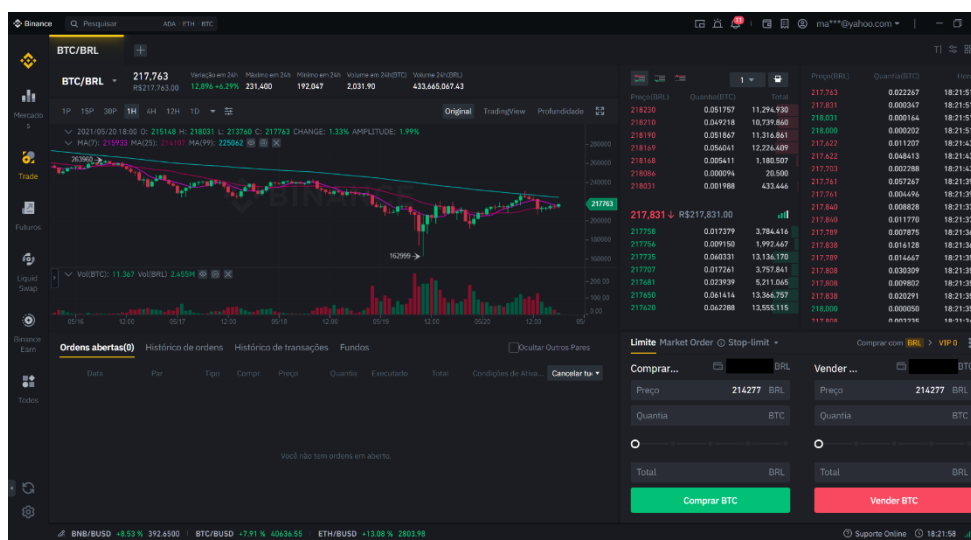


Figura 3 - Mercado *spot* da exchange Binance

Fonte: Os autores (2021)

- Mercado de Margem: É o mercado que utiliza fundos de terceiros, onde permite que os *traders* tenham acesso a maior quantia de capital, permitindo

a alavancagem (técnica de multiplicar a rentabilidade através de endividamento, necessitando uma margem de garantia para sinalizar que, caso haja prejuízos, você terá condições de pagá-los). O mercado de margem pode ser de dois tipos: a isolada e a cruzada.

- Margem isolada: Cada par de *trade* é independente, isso é, caso haja ganhos ou percas elas serão relativas só a posição atual do investidor.
- Margem cruzada: Os fundos da conta de margem cruzada são compartilhados com todas as posições, uma vez que uma liquidação acontece, todas as posições são fechadas.

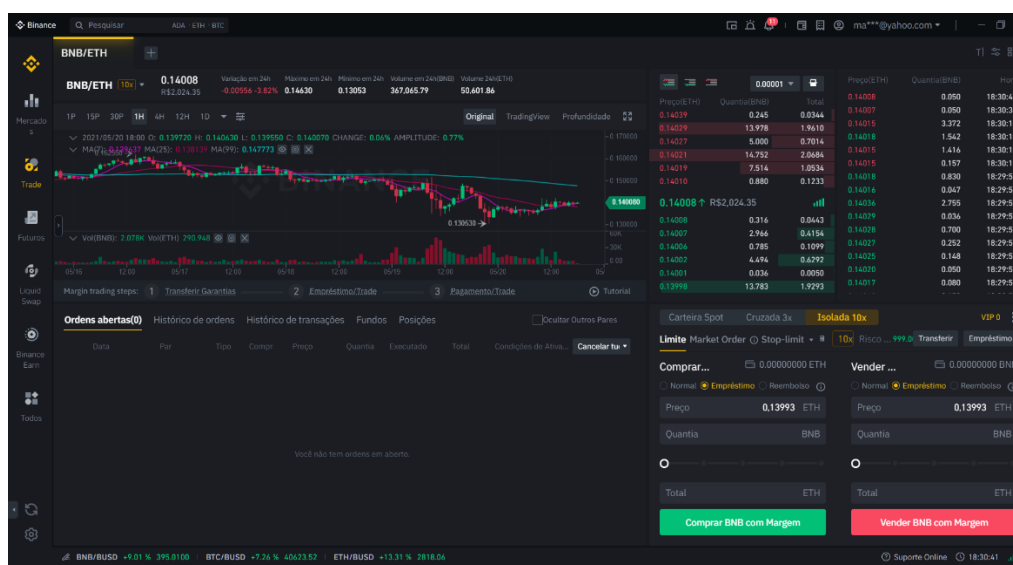


Figura 4 - Mercado de margem da exchange Binance

Fonte: Os autores (2021)

- Mercado de investimentos: Onde o usuário coloca seus fundos para que seja gerado rendimento com base nos mesmos, podendo ser similar à uma poupança. As informações dos investimentos são apresentadas no formato de APY (rendimento anualizado).



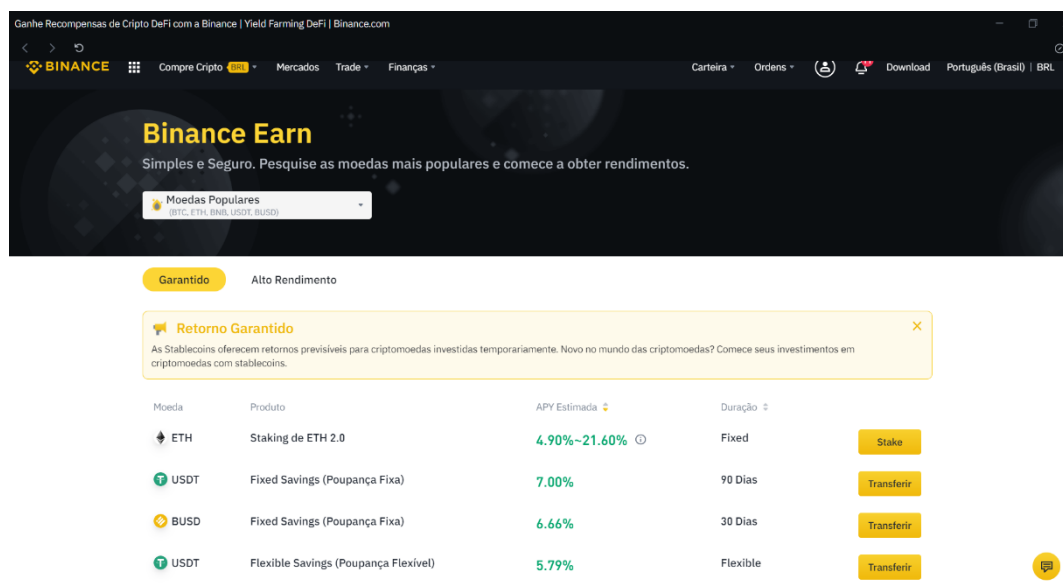


Figura 5 - Mercado de investimentos da Binance

Fonte: Os autores (2021)

## 2.6 Tipos de operações

Os investidores do mercado de criptomoedas, assim como no mercado de ações, são classificados de acordo com a frequência de seus investimentos, podendo ser considerados: *day trader*, *swing trader*, *buy hold* e *scalper trade*.

### 2.6.1 Day trade

É uma operação de curto prazo, utilizado por *traders* com grande conhecimento de mercado, leitura de gráficos e de indicadores. A principal característica é a realização de operações de compra e venda no mesmo dia, com uma frequência que varia entre 1 transação até 10.

### 2.6.2 Swing trade

Diferente da operação de *day trade*, o *swing trade* utiliza uma estratégia de curto a médio prazo, procurando um melhor retorno em relação ao tempo. A principal característica deste investidor é a paciência e a disciplina para operar com cautela.

### 2.6.3 *Buy hold*

São investidores que focam no médio e ao longo prazo, podendo levar semanas ou até anos para a venda de qualquer ativo. É uma característica que não depende de muito conhecimento técnico sobre operações, é muito utilizada por pessoas que investem em moedas recém-criadas, onde aguardam a valorização temporal conforme a notoriedade do ativo aumenta.

### 2.6.4 *Scalper trade*

Este tipo de operação é semelhante ao *day trader*, efetuando transações no mesmo dia, o que difere da outra modalidade é quantidade de operações, que variam entre 20 e 70 operações diárias. O foco dessa operação é ganhar dinheiro em pouco tempo, mesmo que a rentabilidade por negociação seja menor.

### 3. A DESCOBERTA DE CONHECIMENTO EM BASES DE DADOS

O avanço na coleta e armazenamento de dados, possibilitou que as organizações acumulassem uma enorme quantidade de informações em bases de dados. Porém, extrair conhecimento a partir desses dados, tem sido um grande desafio. Devido ao conjunto de dados ser muito grande, as técnicas tradicionais de análise de dados não podem ser utilizadas. Em outras situações, a não trivialidade dos dados quer dizer que abordagens tradicionais não podem ser aplicadas (Brito, 2012).

Neste capítulo apresentar todo o embasamento teórico para que seja possível entender o processo de descoberta do conhecimento, desde o entendimento do conceito das necessidades das organizações até a técnica e algoritmo utilizados para a escolha e validação da abordagem utilizada na aplicação.

Muitas pesquisas têm sido direcionadas para o desenvolvimento de técnicas com o intuito de extrair informações a partir de um grande volume de dados e transformar estas informações em conhecimento utilizável. Esta área é conhecida como KDD (*knowledge-discovery in databases*) (Brito, 2012).

“KDD é um processo, de várias etapas, não trivial, interativo e iterativo, para identificação de padrões compreensíveis, válidos, novos e potencialmente úteis a partir de grandes conjuntos de dados” (FAYYAD, 1996).

A expressão “não trivial”, demonstra a complexidade na execução e persistência dos processos de KDD, “interativo” diz respeito à relevância de ter um elemento controlador do processo, o termo “iterativo” sugere a possibilidade de repetições em qualquer etapa do processo e por fim o “conhecimento útil” que é aquele onde o objetivo foi alcançado, trazendo os benefícios as aplicações de KDD (Brito, 2012).

Segundo FAYYAD et al. (1996), o processo de KDD possui diversas fases, e se inicia na análise do domínio da aplicação e dos objetivos a serem alcançados, sendo dividido em 5 fases:

1. Seleção de dados;
2. Pré-processamento;

3. Transformação;
4. Mineração;
5. Análise e assimilação dos resultados.

### **3.1 As fases do KDD**

Na primeira parte do processo, busca-se escolher a base de dados a ser minerada, podendo ser subconjunto de variáveis, amostras de dados até grandes massas de dados. A fase de pré-processamento tem o objetivo a limpeza da base, eliminando os ruídos, tuplas vazias e valores inválidos. A etapa de transformação varia de acordo com objetivo da busca e do algoritmo que será utilizado, pois é ele quem possui as limitações a serem impostas na base de dados. A melhoria na qualidade dos dados é imprescindível para que ocorra um melhor resultado, garantindo assim uma melhor qualidade nos padrões descobertos (Brito, 2012).

A próxima etapa é a etapa de mineração dos dados. Essa é a fase mais importante do processo de descoberta de conhecimento KDD, é onde se aplica o algoritmo que utiliza uma determinada técnica, e que tem como objetivo elaborar um modelo para representar um conjunto de dados (Brito, 2012).

A interpretação ou pós-processamento é a fase da identificação, entre os padrões extraídos na etapa de mineração de dados. Esta realiza-se uma avaliação criteriosa dos resultados obtidos, envolvendo todos os participantes. É crucial interpretar os padrões minerados, provavelmente retornando a qualquer uma das fases anteriores para novas iterações, visando o refinamento do modelo e resultado gerado de acordo com a necessidade, com o objetivo de apresentar conhecimento descoberto ao usuário ao usuário. A figura 6 apresenta as atividades que constituem o processo de KDD (Brito, 2012).

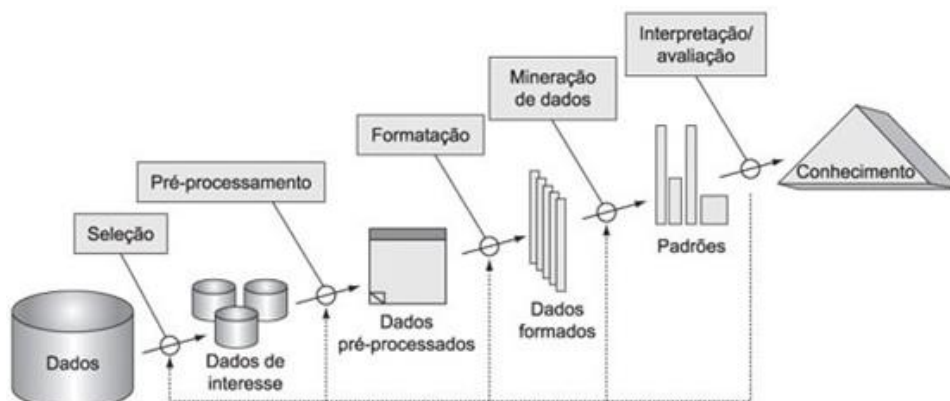


Figura 6 - Etapas do KDD

Fonte: FAYYAD et al. (1996)

### 3.1.1 Mineração de dados

A mineração de dados ou *data mining* é a principal etapa do processo da fase de KDD, com base em técnicas da estatística, inteligência artificial, computação paralela, máquina de estado e aprendizado de máquina, ela constrói um longo histórico de pesquisas relacionadas a estas áreas. Buscando padrões, relação entre dados, anomalias e regras, com o objetivo de encontrar informações implícitas, que possuem relevância na tomada de decisão e/ou avaliação de resultados (Brito, 2012).

Com aplicações em vários campos, como ciência e pesquisa, com mineração de dados as empresas podem aprender mais sobre seus clientes e desenvolver estratégias mais eficazes relacionadas a muitas funções de negócio, assim alavancar recursos de maneira otimizada e perspicaz. Afinal, com o entendimento mais aprofundado dos dados, as empresas ficam mais próximas dos seus objetivos e tomam decisões melhores (Infonova, 2019).

### 3.1.2 Tarefas de mineração de dados

As tarefas de mineração de dados são os tipos de descoberta que se busca realizar em uma base de dados, ou seja, são informações que se deseja obter. É

necessário ter um bom conhecimento do domínio da aplicação e saber o tipo de informação que se quer obter para definir qual tarefa a ser resolvida (Brito, 2021):

A definição da técnica de mineração escolhida está ligada intimamente com a tarefa de mineração que se deseja executar, visto que essa tarefa define a relação entre os dados, ou seja, o modelo. Existem diversas tarefas de mineração de dados, entre elas:

- Classificação - associa ou classifica objetos a determinadas classes, ela busca prever uma classe de um novo dado de forma automática;
- Clusterização - também denominada agrupamento ou segmentação, é utilizada para dividir os dados em grupos (*clusters*) com o objetivo de que os objetos dentro de um grupo sejam semelhantes e diferentes de outros objetos de outros grupos;
- Associação - tem como principal objetivo encontrar padrões do tipo  $X \rightarrow Y$ , ou seja, o quanto  $X$  implica em  $Y$  onde  $X$  e  $Y$  são conjuntos distintos.
- Regressão - consiste em uma equação matemática tendo a finalidade de tratar um valor  $Y$  que não consegue se estimar inicialmente com base nos valores de outras variáveis  $X$ . Como veremos nos capítulos seguintes, é ideal para previsão de valores com alta correlação linear.

### 3.1.3 Infraestrutura

Para minerar dados é necessário ter uma certa infraestrutura tecnológica:

- Tamanho do banco de dados: para criar sistemas mais robustos, mais dados serão necessários para serem processados e mantidos (Infonova, 2019);
- Complexidade da consulta: É importante definir a complexidade de cada consulta, pois quanto maior o número de consultas, mais poderoso deve ser o sistema utilizado (Infonova, 2019).

### 3.1.4 Usabilidade e relevância

Técnicas de mineração de dados são muito utilizadas em projetos de pesquisa, como matemática, cibernética, genética e marketing. Com *data mining*, um varejista pode gerenciar e usar registros de pontos de vendas de compras de clientes para enviar promoções direcionadas baseadas em dados demográficos de mineração de comentários ou cartões de garantia (Infonova, 2019).

Conforme explicado, a mineração de dados é um processo de detecção de padrões relevantes em bancos de dados. Por exemplo, a probabilidade de inadimplência em empréstimo de um cliente pode ser indicada por um padrão de salários médios anuais, onde os salários mais baixos indicam alta probabilidade de inadimplência. Com essas informações, um gerente pode elaborar uma estratégia de marketing mais eficaz para futuros clientes (Infonova, 2019).

Para prever o comportamento de um cliente, a mineração de dados faz uso de técnicas estatísticas e de aprendizado de máquina. O mais popular entre eles é a técnica da árvore de decisão, usada tanto para análise exploratória quanto para modelagem preditiva e existem vários métodos que podem manipular qualquer forma de dado (Infonova, 2019).

## 3.2 Aprendizado de máquina

Em 1959, Arthur Samuel, que desenvolveu um dos primeiros programas de computador para jogar dama, definiu aprendizado de máquina como tecnologia que dá aos “computadores a capacidade de aprender sem serem explicitamente programados” (Silva, 2020).

Aprendizagem de máquina é um subconjunto de IA, um segmento da Ciência da Computação que visa criar computadores que pensam da maneira como humanos pensam. Ou seja, sistemas de aprendizado de máquina são sistemas de IA, mas nem todos os sistemas de IA possuem aprendizado de máquina (Silva, 2020).

Podemos subdividir a aprendizagem de máquina em várias categorias diferentes:

- Aprendizagem supervisionada – requer exemplos de quais entradas se alinham com o resultado. Por exemplo, para ensinar um computador a reconhecer fotos de gatos com o aprendizado supervisionado, é necessário fornecer um conjunto de imagens rotuladas como “gatos” e um outro conjunto de imagens rotuladas como “não gatos”. Assim os algoritmos de aprendizagem de máquina ajudariam o sistema a aprender a generalizar os conceitos para que ele pudesse identificar novos conjuntos de dados não rotulados (Silva, 2020);
- Aprendizagem não supervisionada – A partir de um conjunto de dados não rotulados, o próprio sistema desenvolve suas conclusões. Por exemplo, para um grande conjunto de dados de vendas online, o aprendizado sem supervisão busca *clusters* ou associações entre esses dados para corroborar com o marketing. Assim, é possível evidenciar, por exemplo, que o usuário que compra leite, também compra fraldas (Silva, 2020);
- Aprendizagem semi-supervisionada - É uma combinação de aprendizado supervisionado e não supervisionado. Utilizando o mesmo exemplo do gato, com muitas imagens, algumas rotuladas como “gato” e “não gato” e outras sem rótulos. Um sistema de aprendizado semi-supervisionado utiliza as imagens rotuladas para realizar inferências sobre quais imagens não marcadas incluem gatos. As melhores suposições são devolvidas ao sistema para ajudá-lo a melhorar suas capacidades e ciclos (Silva, 2020);
- Aprendizagem por reforço – Se refere a um sistema que recebe *feedback* análogo a punições e recompensas. Um exemplo muito utilizado é um agente aprendendo a jogar um jogo, de acordo com erros e acerto, o agente vai sendo recompensado ou punido até atingir o objetivo final (Silva, 2020).

Muitas organizações de uma extensa variedade de indústrias já experimentam e implementam a aprendizagem de máquina. Em alguns casos, incorporam o aprendizado de máquina em ferramentas usadas para um propósito específico e, em outros casos, os usuários aplicações de aprendizado de máquina de uso geral para suas próprias necessidades ou desenvolvem aplicações analíticas para si com a ajuda de cientistas de dados. Entre os principais casos de uso de *machine learning* temos a detecção de fraudes, sistemas de recomendação, mecanismo de busca, sistemas de



vigilância em vídeo, reconhecimento de manuscrito, processamento de linguagem natural, *bots* de serviço ao cliente e segurança de TI (Silva, 2020).

### 3.3 Regressão linear

Regressão linear é uma técnica de análise de dados que consiste em uma equação matemática tendo a finalidade de tratar um valor que não consegue se estimar inicialmente. A fórmula procura determinar uma relação, traçando uma reta retilínea entre duas variáveis.

Com o uso dessa técnica é possível determinar, por exemplo: através do valor da altura a estimativa do peso de uma pessoa, os filmes que lhe agradam através da idade ou também o consumo do seu carro sabendo quantos cavalos de potência ele possui.

Assim, podemos utilizar regressão linear para prever valores de fechamento de uma criptomoeda em um intervalo de tempo determinado com base nas variáveis de abertura, baixa e alta previamente conhecidas e que possuem correlação linear suficiente para gerar uma previsão aceitável.

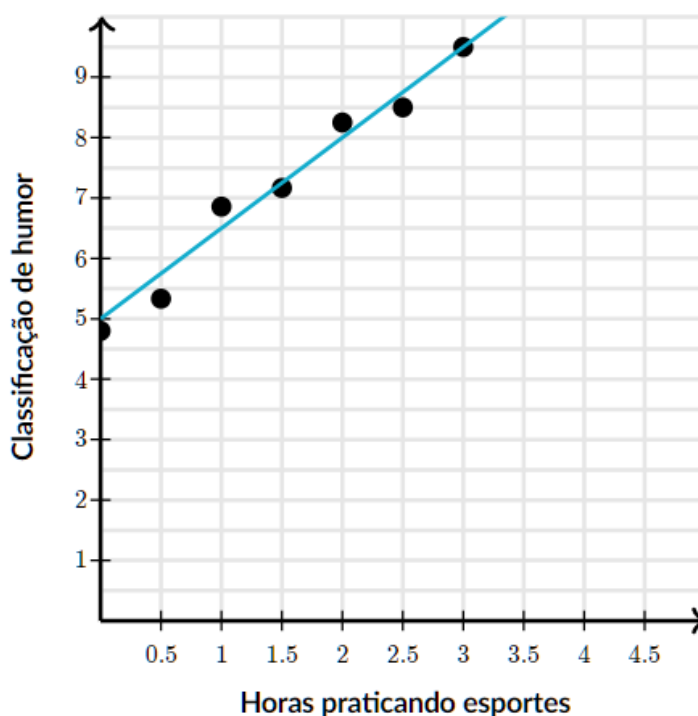


Figura 6 - Exemplo utilizando regressão linear

Fonte: Khan Academy

## 4. FERRAMENTAS TÉCNICAS

Aqui apresentaremos as ferramentas utilizadas para que todos os objetivos deste trabalho se concretizem em uma aplicação funcional para o usuário, devidamente testada. Toda a codificação foi feita de modo a garantir a constante progressão do processo ao fim de cada etapa do desenvolvimento.

### 4.1 Python

Lançado em 1991, *Python* é uma linguagem de programação de alto nível, ou seja, sua sintaxe simples se aproxima muito da linguagem humana. Utilizada nos mais diversos tipos de aplicações como *desktop*, *web*, servidores e ciência de dados (Tecnoblog, 2021).

Além de uma grande comunidade, *Python* conta com uma extensa variedade de bibliotecas de ciência de dados, que crescem cada vez mais (Ciência e Dados, 2021).



Figura 7 - Site principal da organização *Python*

Fonte: Os autores (2021)

Abordaremos de maneira sucinta as principais bibliotecas utilizadas no projeto.

#### 4.1.1 Sklearn

A biblioteca sklearn é a principal importação do projeto, é uma ferramenta simples, eficiente e de código aberto. Possui uma vasta opção de API, abordando tópicos como aprendizado supervisionado, aprendizado não supervisionado, seleção de modelo e validação, inspeções, validações, transformações de *datasets*, entre outros. É através dela que criamos o modelo baseado em regressão linear e sustenta as previsões.

#### 4.1.2 Pandas

Pandas é a biblioteca utilizada para manipulação, análise e ciência de dados em *Python*, sendo seu código *open source*.

Ela conta com uma grande documentação e referências de API, tendo sua utilização prática no projeto na manipulação da base de dados na extensão csv, onde ela transforma a variável que recebe a fonte de dados em objeto do tipo pandas, onde facilita a manipulação.

#### 4.1.3 Json

O json faz parte das bibliotecas padrões do *Python*, onde é especializada na manipulação dos arquivos que levam o nome da biblioteca (json).

Através da utilização deste recurso nós conseguimos trabalhar de maneira mais prática com o retorno da API da *exchange* e operar em conjunto com o valor previsto do fechamento para quantificar a variação cambial do período.

#### 4.1.4 Tkinter

Para a criação de uma interface gráfica do usuário (GUI), o *Python* conta com Tkinter, uma biblioteca que acompanha a instalação padrão para criação de interfaces gráficas. Ou seja, qualquer computador que tenha o interpretador *Python* instalado é capaz de criar interfaces gráficas usando o Tkinter, exceto por algumas distribuições Linux, exigindo a instalação do módulo separadamente (Otávio, 2016).

Com uma grande quantidade de recursos disponíveis e uma grande facilidade de uso, o Tkinter pode ser uma ferramenta muito útil para quem visa o desenvolvimento de uma interface gráfica de forma rápida e eficiente. E por ser nativo da linguagem *Python*, só precisamos importá-lo na hora do uso (Otávio, 2016).

## 4.2 Visual Studio Code

O *Visual Studio* é um editor de código-fonte de uso gratuito, com diversas funcionalidades nativas como debug, completude de código, terminal integrado e muitos outros. Embora nativamente o VSCode suporte JavaScript, TypeScript e NodeJS, ele possui diversas extensões para outras linguagens como Java, C#, PHP, *Python* etc. Além disso, conta com diversos plug-ins e extensões para facilitar o desenvolvimento dos projetos.

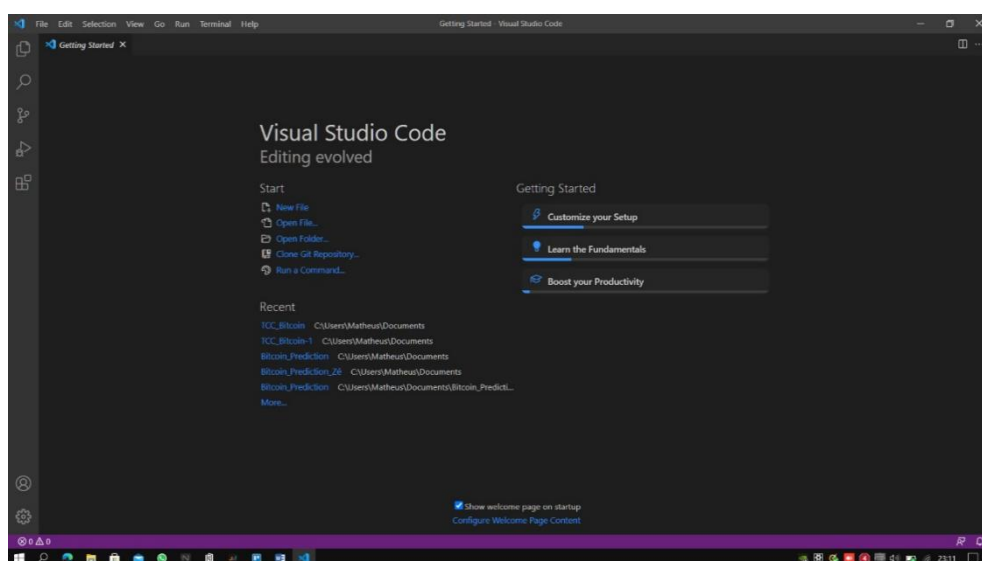


Figura 8 - Tela principal do *Visual Studio Code*

Fonte: Os autores (2021)

### 4.3 Weka

O Weka é um software de aprendizado de máquina de código aberto que pode ser acessado através de uma interface gráfica de usuário, aplicativos de terminal padrão ou uma API Java. É amplamente utilizado para ensino, pesquisa e aplicações industriais, contendo uma infinidade de ferramentas incorporadas para tarefas de aprendizado de máquina, além disso dá acesso transparente a caixas de ferramentas bem conhecidas, como scikit-learn, Re Deeplearning4j (Waikato, 2021).

É uma ferramenta mais adequada para análise de dados e modelagem preditiva por conter algoritmos e ferramentas de visualização que suportam o aprendizado de máquina (Infonova, 2019).

O Weka também suporta as principais tarefas de mineração de dados, incluindo mineração de dados, processamento, visualização, regressão, etc. Porém, a ferramenta trabalha com a premissa de que os dados estão disponíveis na forma de um arquivo simples (Infonova, 2019).



Figura 9 - Tela inicial do programa Weka

Fonte: Os autores (2021)

## 4.4 Github

Github é o mais famoso serviço baseado em nuvem que hospeda um sistema de controle de versão chamado Git. Ele permite que os desenvolvedores colaborem e realizem mudanças em projetos compartilhados enquanto mantêm um registro detalhado de todo o progresso (Hostinger, 2021).

Atualmente acomoda mais de 25 milhões de usuários e é considerado uma ferramenta essencial para um engenheiro de software (Hostinger, 2021).

E este projeto já se encontra no GitHub, onde estará aberto para estudos e contribuições futuras.

Acesse através do *link*: [https://github.com/MarquesRodrigues/TCC\\_Bitcoin](https://github.com/MarquesRodrigues/TCC_Bitcoin)

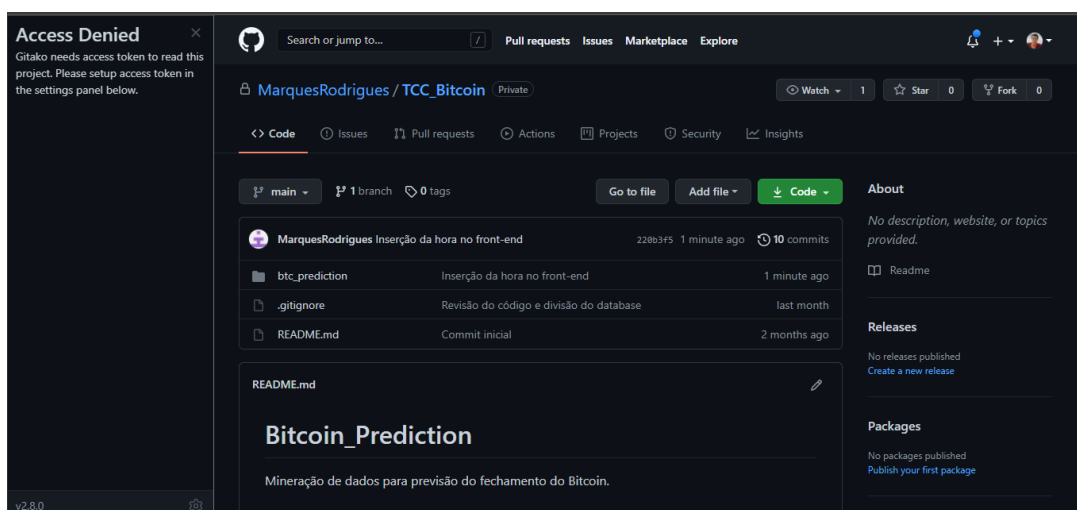


Figura 10 - Tela principal do projeto no Github

Fonte: Os autores (2021)

## 5. BITCOIN PREDICTION

Abordaremos neste capítulo todas as etapas envolvidas no desenvolvimento do software bitcoin *prediction*, desde a origem da base dados e a geração do modelo até a apresentação dos dados ao usuário final com a linguagem de programação *Python*.

### 5.1 Definição da base de dados

Para criação do modelo de regressão linear, precisamos de uma grande fonte de dados onde seremos capazes de treinar o algoritmo de maneira mais precisa. Buscamos as maiores *exchanges* para dar embasamento ao projeto, porém há uma grande limitação na quantidade de registros retornados. Por conta desse empecilho e através do site CryptoDataDownload, conseguimos acesso a um forte *database* na extensão csv, que tem a possibilidade de escolher entre os intervalos de minuto, de horário e diário.

Com essa opção, optamos por escolher a base de dados da corretora Gemini, que entre os artigos classificatórios das *exchanges* a colocou entre as 10 melhores casas de câmbio do mundo.

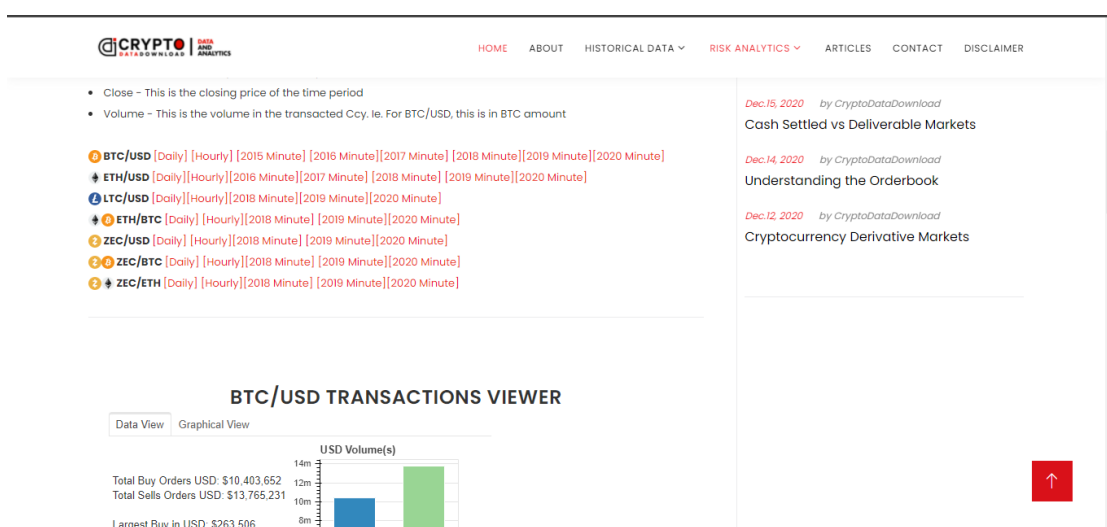


Figura 11 - Seleções dos intervalos no site CryptoDataDownload

Fonte: Os autores (2021)



A fonte de dados em csv possui os seguintes campos:

- **Unix *Timestamp*:** É a marcação da data e da hora unix ou também conhecido como "hora da época". Utilizado para converter em seu fuso horário local.
- ***Date*:** A marcação da data e da hora baseado no UTC.
- ***Symbol*:** O símbolo ao qual os dados da base se referem, como o símbolo que utilizaremos neste projeto BTCUSD.
- ***Open*:** Este é o preço de abertura do período, ou seja, é o preço da última oferta de venda declarado no livro de ordens do último intervalo.
- ***High*:** O preço mais alto do período.
- ***Low*:** O preço mais baixo do período.
- ***Close*:** A cotação final do período.
- ***Volume*:** O volume transacionado no símbolo escolhido. Ou seja, para BTC-USD, é apresentado o volume em bitcoins.

A apresentação e a ordem dos campos seguem esse padrão quando tratamos da base de dados do sítio CryptoDataDownload.

## 5.2 Manipulação da base de dados

Após entendermos a base de dados e quais são os campos que a compõe, precisamos definir qual período tem maior aproveitamento estatístico, para isso utilizaremos os três intervalos disponíveis na ferramenta Weka.

Foi escolhida a base de dados de horas pelos seguintes motivos:

- A base de minutos, apesar de ter o maior volume de dados, do ponto de vista prático não é tão interessante para um *trader*, visto que ele possui pouco tempo hábil para realizar suas operações;
- A base diária, não possui muitos dados para o treinamento do algoritmo.

Assim, a base de dados de horas foi selecionada para criarmos o modelo de previsão, contendo mais de 49 mil linhas, considerando o período entre outubro de 2015 até os dias de hoje.

Para utilizarmos a fonte de dados obtida pelo Weka, temos que eliminar a primeira linha para facilitar o reconhecimento pela ferramenta, visto que a informação ali presente faz com que o algoritmo do software se perca e a base de dados não seja carregada. Também iremos dividir a base em duas, sendo uma parte utilizada para gerar o modelo e a outra para validação deste treinamento:

```

gemin1_BTCUSD_1hr.csv - Bloco de Notas
Arquivo  Editar  Formatar  Exibir  Ajuda
created on 05-21-2021,https://www.CryptoDataDownload.com
Unix Timestamp,Date,Symbol,Open,High,Low,Close,Volume
1621555200000,2021-05-21 00:00:00,BTCUSD,40571.72,41257.14,40500.0,40941.99,8.4273316206
1621551600000,2021-05-20 23:00:00,BTCUSD,41533.9,41768.31,40383.4,40571.72,66.6135067347
1621548000000,2021-05-20 22:00:00,BTCUSD,40750.0,41999.99,39523.28,41533.9,183.6838715835
1621544400000,2021-05-20 21:00:00,BTCUSD,40075.01,40912.29,39798.34,40750.0,85.6046985259
1621540800000,2021-05-20 20:00:00,BTCUSD,40248.93,40701.32,39729.64,40075.01,196.16744612
1621537200000,2021-05-20 19:00:00,BTCUSD,39920.72,40504.2,39669.65,40248.93,236.3439742543
1621533600000,2021-05-20 18:00:00,BTCUSD,39893.33,40100.0,39341.94,39920.72,307.6314857393
1621530000000,2021-05-20 17:00:00,BTCUSD,39383.56,40252.27,38869.11,39893.33,383.0061951335
1621526400000,2021-05-20 16:00:00,BTCUSD,41526.04,41569.19,38200.0,39383.56,771.0572916457
1621522800000,2021-05-20 15:00:00,BTCUSD,41547.04,42151.87,41110.41,41526.04,258.2970275424
1621519200000,2021-05-20 14:00:00,BTCUSD,41835.51,42319.92,41351.64,41547.04,249.8137425285
1621515600000,2021-05-20 13:00:00,BTCUSD,41810.08,42625.43,41402.74,41835.51,387.536948506
1621512000000,2021-05-20 12:00:00,BTCUSD,40565.04,42058.12,40198.89,41810.08,430.786561761
1621508400000,2021-05-20 11:00:00,BTCUSD,40462.91,40768.52,39809.59,40565.04,98.148945026
1621504800000,2021-05-20 10:00:00,BTCUSD,39976.43,40557.4,39558.64,40462.91,138.850258246
1621501200000,2021-05-20 09:00:00,BTCUSD,39877.12,40243.76,38829.13,39976.43,123.2051742651
1621497600000,2021-05-20 08:00:00,BTCUSD,40072.05,40327.55,39408.61,39877.12,81.5417353391
1621494000000,2021-05-20 07:00:00,BTCUSD,39806.13,40837.6,39721.17,40072.05,172.2645225351
1621490400000,2021-05-20 06:00:00,BTCUSD,40000.0,40276.56,39310.35,39806.13,102.8895587338
1621486800000,2021-05-20 05:00:00,BTCUSD,39531.3,40045.96,39245.01,40000.0,150.6932300075
1621483200000,2021-05-20 04:00:00,BTCUSD,38411.14,39842.81,38292.4,39531.3,188.6632626882
1621479600000,2021-05-20 03:00:00,BTCUSD,38138.39,38693.69,37858.99,38411.14,181.9420967604
1621476000000,2021-05-20 02:00:00,BTCUSD,37047.59,38365.26,36918.11,38138.39,259.0086198246
1621472400000,2021-05-20 01:00:00,BTCUSD,35863.9,37556.6,35576.03,37047.59,992.4064263756
1621468800000,2021-05-20 00:00:00,BTCUSD,36757.42,38261.63,34989.0,35863.9,743.8213595062
1621465200000,2021-05-19 23:00:00,BTCUSD,39163.44,39474.26,36700.0,36757.42,697.0104532716
1621461600000,2021-05-19 22:00:00,BTCUSD,38847.62,39664.78,38361.75,39163.44,285.8238653392
1621458000000,2021-05-19 21:00:00,BTCUSD,38338.34,39000.0,37102.8,38847.62,290.7263877056
1621454400000,2021-05-19 20:00:00,BTCUSD,39396.53,40250.0,38105.46,38338.34,456.785730212
1621450800000,2021-05-19 19:00:00,BTCUSD,38152.92,39637.52,37825.0,39396.53,540.8725471696
1621447200000,2021-05-19 18:00:00,BTCUSD,39469.08,39739.28,37714.81,38152.92,727.4906380225
1621443600000,2021-05-19 17:00:00,BTCUSD,39854.5,40740.0,38772.31,39469.08,780.9530956525
1621440000000,2021-05-19 16:00:00,BTCUSD,37459.01,40373.03,36218.98,39854.5,937.8668312312
1621436400000,2021-05-19 15:00:00,BTCUSD,37755.47,38187.67,36600.0,37459.01,886.7605412609
1621432800000,2021-05-19 14:00:00,BTCUSD,36101.33,37790.47,34000.0,37755.47,934.4522707866
Ln 1, Col 1    100%    Windows (CRLF)    UTF-8

```

Figura 12 - Eliminação da primeira linha do csv

Fonte: Os autores (2021)

Com a finalidade de entendermos quais campos iremos utilizar, vamos avaliar quais deles possuem uma correlação linear considerável para gerar o modelo. Para isso, faremos um teste com um algoritmo de regressão linear com o Weka.

Excluiremos os campos **Date** e **Symbol**, pois o Weka somente aceita valores numéricos, sendo **Symbol** uma constante que não afeta o modelo:

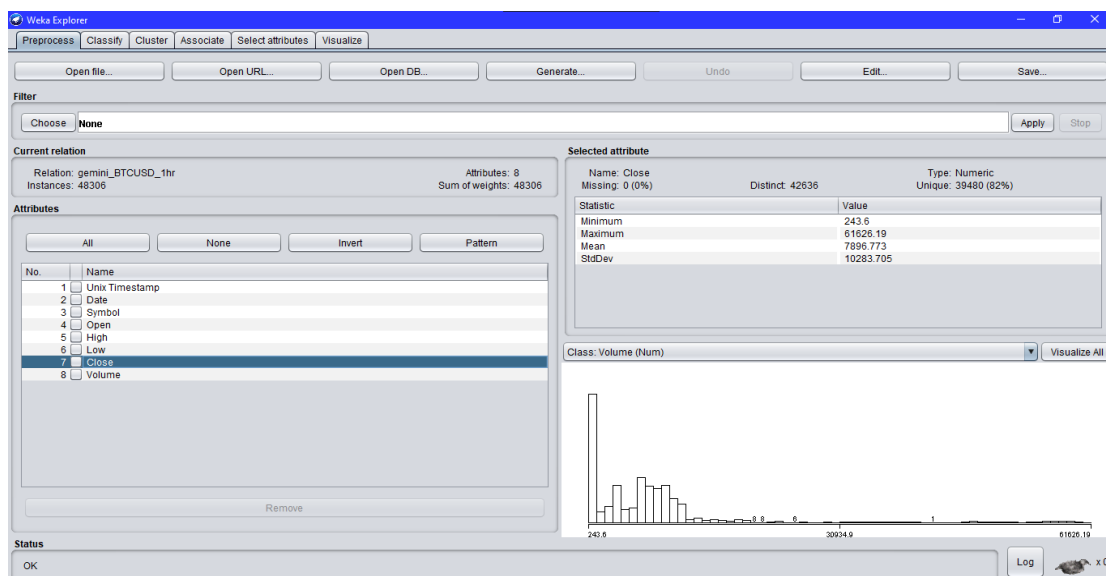


Figura 13 - Campos reconhecidos no Weka

Fonte: Os autores (2021)

Organizaremos a base de maneira decrescente em relação à data, pois no arquivo csv os registros mais recentes ficam embaixo, como uma pilha, assim ao realizarmos o treinamento poderemos realizar os testes baseados nas datas mais atuais:

No.	1: Unix Timestamp	2: Date	3: Symbol	4: Open	5: High	6: Low	7: Close	8: Volume
1	1.4443092E9	2015-10-08 13:00:00	BTCUSD	0.0	245.0	0.0	245.0	0.60665438
2	1.4443128E9	2015-10-08 14:00:00	BTCUSD	245.0	245.0	244.5	245.0	4.453648931
3	1.4443164E9	2015-10-08 15:00:00	BTCUSD	245.0	245.0	244.92	244.92	3.016925828
4	1.44432E9	2015-10-08 16:00:00	BTCUSD	244.92	244.92	244.25	244.25	3.89525246
5	1.4443236E9	2015-10-08 17:00:00	BTCUSD	244.25	244.99	244.02	244.99	3.920632003
6	1.4443272E9	2015-10-08 18:00:00	BTCUSD	244.99	244.99	244.0	244.0	3.69047209
7	1.4443308E9	2015-10-08 19:00:00	BTCUSD	244.0	244.0	244.0	244.0	1.531455579
8	1.4443344E9	2015-10-08 20:00:00	BTCUSD	244.0	244.0	243.6	244.0	0.17704918
9	1.444338E9	2015-10-08 21:00:00	BTCUSD	244.0	244.0	243.95	243.95	0.04132231
10	1.4443416E9	2015-10-08 22:00:00	BTCUSD	243.95	244.0	243.8	243.95	4.71303031
11	1.4443452E9	2015-10-08 23:00:00	BTCUSD	243.95	243.95	243.72	243.72	0.0
12	1.4443488E9	2015-10-09 00:00:00	BTCUSD	243.95	243.95	243.6	243.6	0.0
13	1.4443524E9	2015-10-09 01:00:00	BTCUSD	243.6	243.6	243.6	243.6	0.0
14	1.444356E9	2015-10-09 02:00:00	BTCUSD	243.6	243.6	243.6	243.6	0.0
15	1.4443596E9	2015-10-09 03:00:00	BTCUSD	243.6	243.6	243.6	243.6	0.0
16	1.4443632E9	2015-10-09 04:00:00	BTCUSD	243.6	243.6	243.6	243.6	0.0
17	1.4443668E9	2015-10-09 05:00:00	BTCUSD	243.6	243.6	243.6	243.6	0.0
18	1.4443704E9	2015-10-09 06:00:00	BTCUSD	243.6	243.6	243.6	243.6	0.0
19	1.444374E9	2015-10-09 07:00:00	BTCUSD	243.6	243.6	243.6	243.6	0.0
20	1.4443776E9	2015-10-09 08:00:00	BTCUSD	243.6	243.6	243.6	243.6	0.0
21	1.4443812E9	2015-10-09 09:00:00	BTCUSD	243.6	243.6	243.6	243.6	0.0
22	1.4443848E9	2015-10-09 10:00:00	BTCUSD	243.6	243.6	243.6	243.6	0.0
23	1.4443884E9	2015-10-09 11:00:00	BTCUSD	243.6	243.75	243.6	243.63	1.0
24	1.444392E9	2015-10-09 12:00:00	BTCUSD	243.63	243.63	243.63	243.63	0.0
25	1.4443956E9	2015-10-09 13:00:00	BTCUSD	243.63	244.0	243.63	244.0	3.973254282
26	1.4443992E9	2015-10-09 14:00:00	BTCUSD	244.0	244.0	244.0	244.0	4.17182852
27	1.4444028E9	2015-10-09 15:00:00	BTCUSD	244.0	244.0	244.0	244.0	0.0
28	1.4444064E9	2015-10-09 16:00:00	BTCUSD	244.0	244.0	244.0	244.0	0.0
29	1.44441E9	2015-10-09 17:00:00	BTCUSD	244.0	244.0	243.7	243.7	0.229184

Figura 14 - Ordenação decrescente do arquivo csv

Fonte: Os autores (2021)

### 5.3 Mineração da base de dados

Iremos utilizar o algoritmo de regressão linear do Weka através da opção “use training set” para gerar nosso modelo matemático, sendo o alvo da previsão a variável *close* pois ao estarmos dentro do intervalo de hora, o preço final do período é o valor desejado.

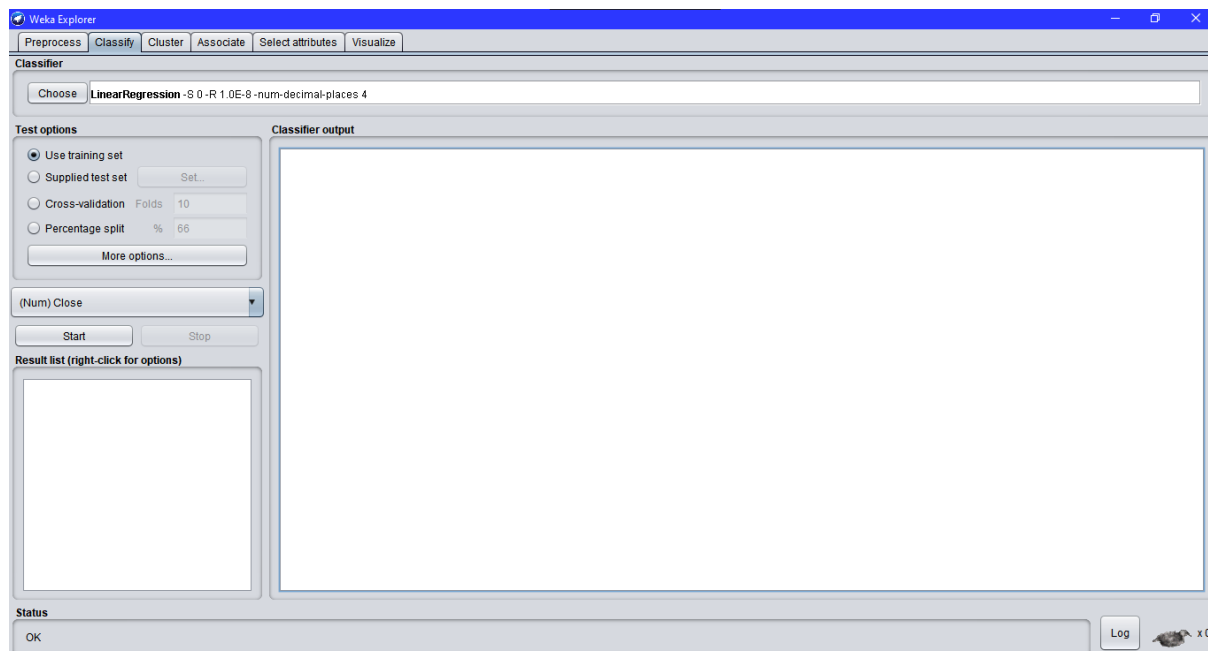


Figura 15 - Configuração do Weka para criação do modelo matemático

Fonte: Os autores (2021)

Executando o algoritmo de regressão linear, obtivemos os seguintes resultados:

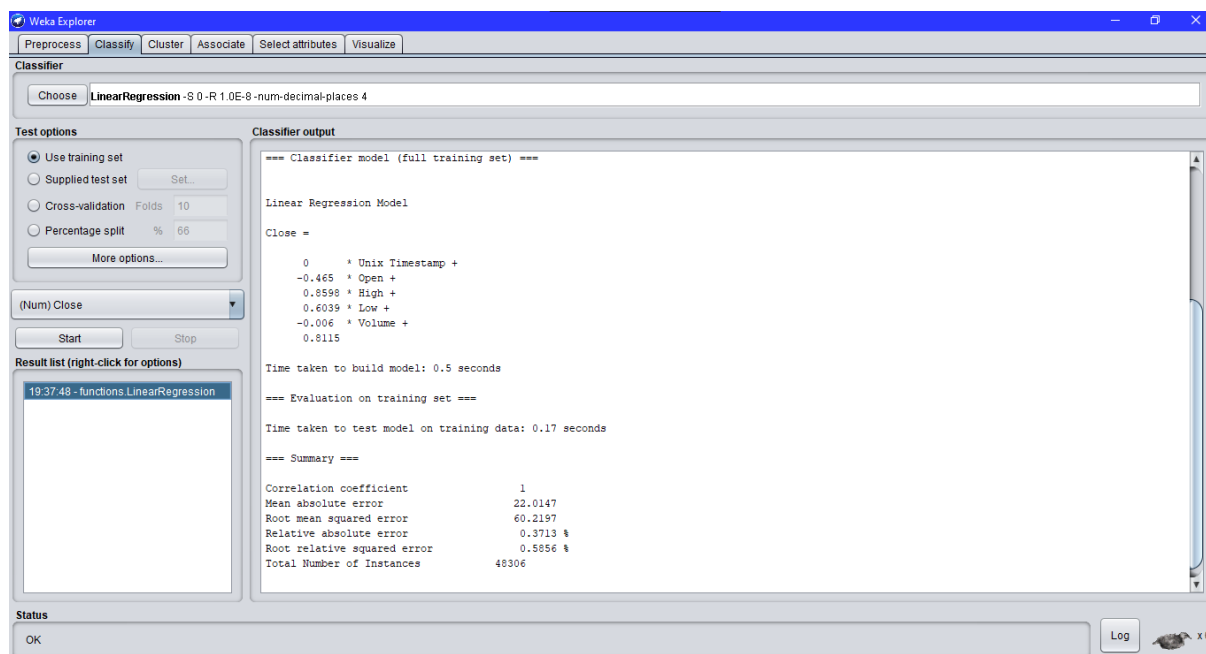


Figura 16 - Resultado do treinamento do modelo

Fonte: Os autores (2021)

Com a finalização da execução do algoritmo, temos os seguintes resultados na mineração de dados:

- *Correlation coefficient: 1*

O coeficiente de correlação nos mostra a relação entre os valores verdadeiros e os valores estimados em uma classificação, assim esse valor se encontra estritamente entre  $-1$  e  $1$ . Onde  $0$  representa nenhuma relação,  $1$  uma relação muito forte e  $-1$  uma relação muito forte inversa. Como a figura mostra (Martins, 2016):

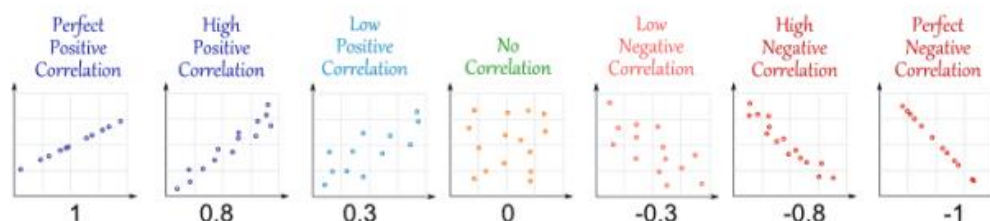


Figura 17 - Gráfico da métrica de coeficiente de correlação

Fonte: Math is fun (2018)

- *Mean absolute error: 22.0147*

A média do erro absoluto não leva em conta se foi superestimado ou subestimado, é uma média dos erros cometidos pelo modelo de previsão durante uma série de execuções. Seu cálculo é feito subtraindo o valor da previsão ao valor do verdadeiro em cada período de execução. O resultado é sempre em módulo, somando e dividindo pelo número de valores que é usado para obter a soma, representado pela equação (Martins, 2016):

$$MAE = \frac{1}{N} \sum_{i=1}^N |\hat{\theta}_i - \theta_i|$$

O valor de  $\theta$  é denotado como o valor verdadeiro de interesse, e  $\theta^\wedge$  é denotado como valor estimado (Martins, 2016).

- *Root mean squared error*: 60.2197

A raiz quadrada do quadrado do erro médio também é uma métrica de erro de previsão. É a soma dos erros de previsão ao quadrado e dividido pelo número de erros usado no cálculo. A raiz quadrada do quadrado do erro médio pode ser expressa pela equação (Martins, 2016):

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{\theta}_i - \theta_i)^2}$$

A diferença média entre a média do erro absoluto e a raiz quadrada do quadrado do erro médio e pode ser comparada em relação ao valor da variável (Martins, 2016).

- *Relative absolute error*: 0.3713 %

O erro relativo absoluto é apenas o erro absoluto total, tendo o mesmo tipo de normalização da raiz quadrática do erro médio. Os erros são normalizados pelos erros do preditor simples que prevê os valores médios, como a seguinte equação mostra (Martins, 2016):

$$RAE = \frac{\sum_{i=1}^N |\hat{\theta}_i - \theta_i|}{\sum_{i=1}^N |\bar{\theta} - \theta_i|}, \text{ sendo } \bar{\theta} \text{ a média dos valores de } \theta$$

- *Root relative squared error: 0.5856 %*

Na raiz quadrada do erro relativo o cálculo do erro é realizado em relação ao que um preditor simples utilizado. O preditor simples utilizado é apenas uma média dos valores reais dos dados, denotados por 1. Assim, a raiz quadrada do erro relativo faz uma normalização, dividindo-se pelo erro quadrado total do indicador padrão, como na equação (Martins, 2016):

$$RRSE = \sqrt{\frac{\sum_{i=1}^N (\hat{\theta}_i - \theta_i)^2}{\sum_{i=1}^N (\bar{\theta} - \theta_i)^2}}, \text{ sendo } \bar{\theta} \text{ a média dos valores de } \theta$$

- *Total number of instances: 48306*

É o número total de instâncias utilizado para gerar o modelo com o intervalo de horário.

Podemos ver através da visualização dos gráficos do Weka que as variáveis *unix timestamp* e *volume* também podem ser descartadas, para fins de otimização do modelo

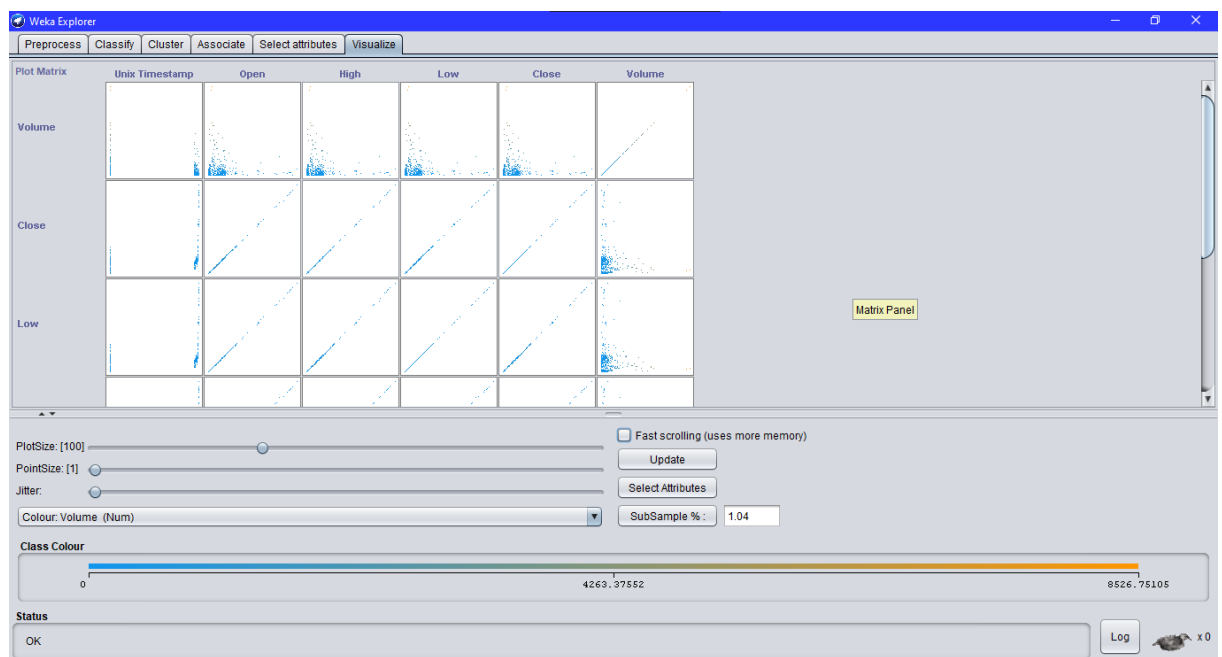




Figura 18 - Visualização gráfica do modelo

Fonte: Os autores (2021)

Assim ganhamos performance e aumentamos insignificamente a taxa de erros para fins de validação do modelo:

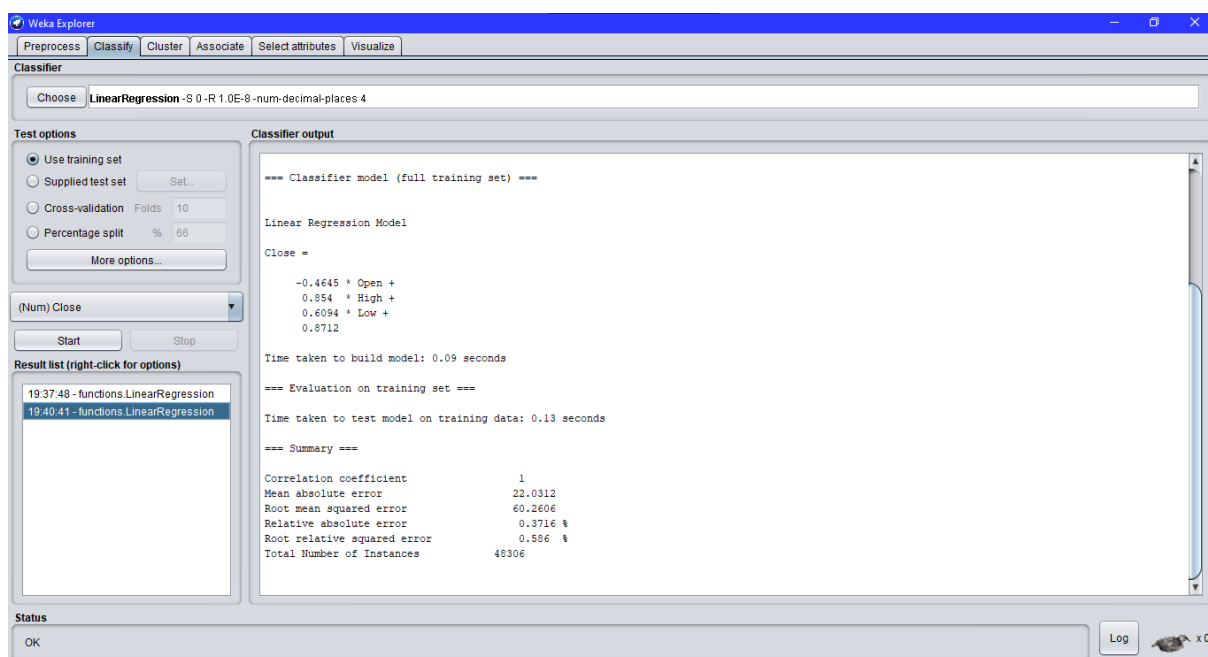


Figura 19 - Resultado pós remoção de variáveis

Fonte: Os autores (2021)

## 5.4 Validação da base de dados

Agora iremos utilizar a base de teste, separada conforme informado no capítulo 4.2 para validar o modelo:

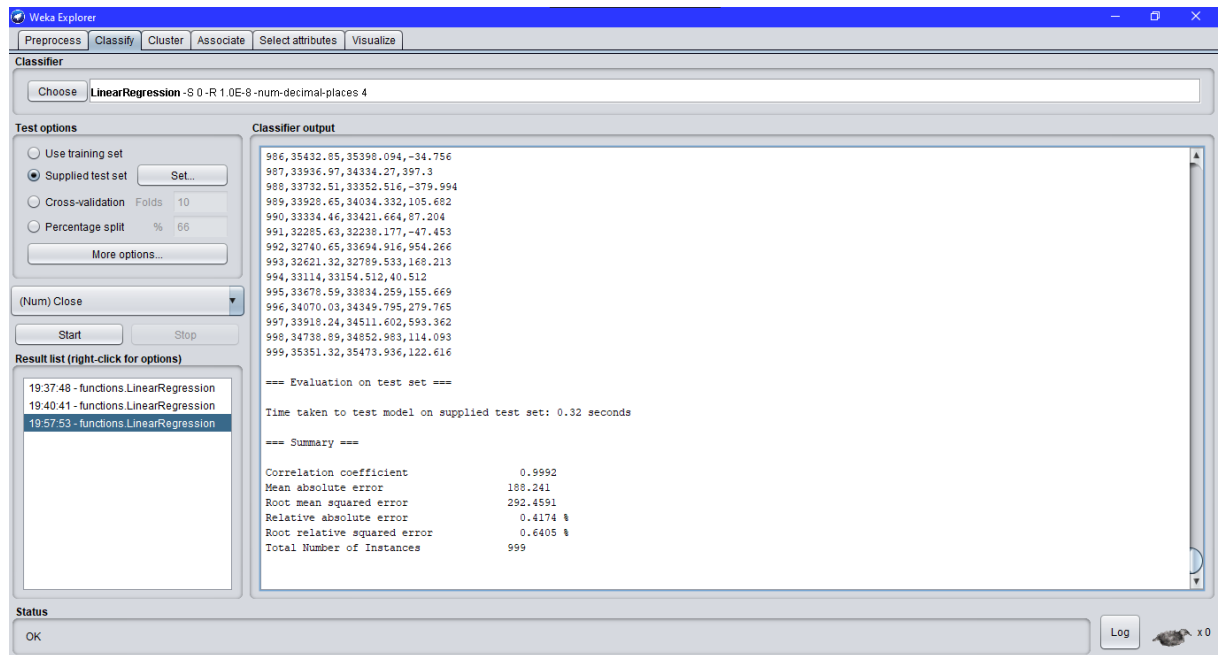


Figura 20 - Resultado da execução com a base de testes

Fonte: Os autores (2021)

- Coeficiente de correlação: 0,9992 – Indicando uma alta correlação entre as variáveis.
- Erro médio absoluto: 188,241
- Raiz quadrada do quadrado do erro médio: 292,4591
- Erro relativo absoluto: 0,4174%
- Raiz quadrada do erro relativo: 0,6405%
- Número total de instâncias: 999

A saída gerada pelo Weka em formato csv, apresentando as primeiras e últimas 10 linhas:

```

=== Run information ===

Scheme:      weka.classifiers.functions.LinearRegression -S 0 -R 1.0E-8 -num-decimal-places 4
Relation:    gemini_BTICUSD_lhr-weka.filters.unsupervised.attribute.Remove-RI-3,8
Instances:   48306
Attributes:  4
              Open
              High
              Low
              Close

Test mode:   user supplied test set:  size unknown (reading incrementally)

=== Classifier model (full training set) ===

Linear Regression Model

Close =

-0.4645 * Open +
0.854 * High +
0.6094 * Low +
0.8712

Time taken to build model: 0.08 seconds

=== Predictions on test set ===

inst#    actual    predicted    error
1 59772.74 59813.486 40.746
2 60226.07 60092.682 -133.388
3 60425.93 60423.814 -2.116
4 60338.11 60427.031 88.921
5 59877 60038.432 161.432
6 60013.08 59895.162 -117.918
7 59946.29 59841.437 -104.853
8 59715.58 59733.846 18.266
9 59947.74 59854.358 -93.382
10 60037.53 60034.016 -3.514
11 59976.7 60005.074 28.374
12 60006.31 59874.939 -131.371
13 60049.99 59877.912 -172.078
14 59863.12 59897.142 34.022
15 60438.38 60264.235 -174.145
16 60619.6 60390.14 -229.46
17 60814.05 60777.748 -36.302
18 60415.91 60719.441 303.531
19 60540.42 60491.192 -49.228
20 60661.74 60620.659 -41.081
21 60725.42 61032.446 307.026
22 61051.16 60963.699 -87.461
23 62648.54 62420.061 -228.479
24 62896.55 62878.68 -17.87
25 62759.06 62720.906 -38.154
26 62960.12 62850.035 -110.085
986 35432.85 35398.094 -34.756
987 33936.97 34334.27 397.3
988 33732.51 33352.516 -379.994
989 33928.65 34034.332 105.682
990 33334.46 33421.664 87.204
991 32285.63 32238.177 -47.453
992 32740.65 33694.916 954.266
993 32621.32 32789.533 168.213
994 33114 33154.512 40.512
995 33678.59 33834.259 155.669
996 34070.03 34349.795 279.765
997 33918.24 34511.602 593.362
998 34738.89 34852.983 114.093
999 35351.32 35473.936 122.616

=== Evaluation on test set ===

Time taken to test model on supplied test set: 0.58 seconds

=== Summary ===

Correlation coefficient      0.9992
Mean absolute error         188.241
Root mean squared error     292.4591
Relative absolute error      0.4174 %
Root relative squared error  0.6405 %
Total Number of Instances   999

```

Figura 21 - Resultado da previsão da base de teste

Fonte: Os autores (2021)

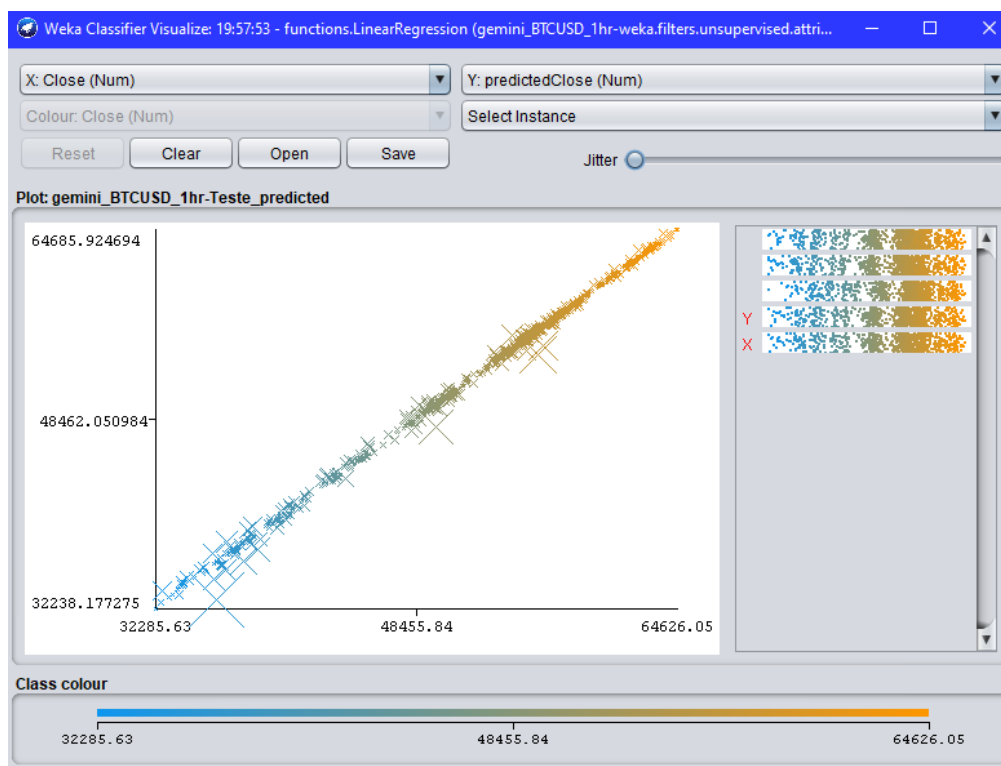


Figura 22 - Visualização da classificação no Weka

Fonte: Os autores (2021)

## 5.5 Importação da base de dados

Tendo definido o intervalo da base de dados após as diversas verificações e validações no Weka, iremos importá-la em nosso *software* através da biblioteca *pandas* do *Python*. Utilizaremos o recurso de configuração através de *ini* com a biblioteca *configparser*, para ocultar e isolar o arquivo que controla os caminhos a serem utilizados no programa a fim de melhorar a legibilidade do código fonte

```
1 [DEFAULT]
2 API_URL = https://api.gemini.com/v2/ticker/btcusd
3 API_DATABASE_1H = https://www.cryptodatadownload.com/cdd/gemini_BTCUSD_1hr.csv
4
```

Figura 23 - Conteúdo do arquivo de configurações

Fonte: Os autores (2021)

```

8  config = configparser.ConfigParser()
9
10 config.read("btc_prediction/config.ini")
11
12 ✓ database = pandas.read_csv(config.get(
13     "DEFAULT", "API_DATABASE_1H"), header=1)
14

```

Figura 24 - Importação do csv obtido do sítio *CryptoDataDownload*

Fonte: Os autores (2021)

## 5.6 Criação do modelo

Com o carregamento em memória do .csv obtido através do sítio *CryptoDataDownload* é necessário a manipulação da base de dados a fim de extrair e determinar o eixo X e Y a ser composto pela função de regressão linear da biblioteca *sklearn*. Através dessa separação é efetuado cálculo do intercepto através da função *fit*.

```

15 X = database.loc[:, ['Open', 'High', 'Low']]
16 Y = database.loc[:, ['Close']]
17
18 model = linear_model.LinearRegression().fit(X, Y)
19
20

```

Figura 25 - Alocação dos eixos e criação do modelo matemático

Fonte: Os autores (2021)

## 5.7 Obtenção dos dados atuais

Tendo a aplicação calculado os valores de intercepto, são necessários os dados atuais de mercado para que seja efetuada a previsão do próximo fechamento do período, com isso, utilizamos a API da Gemini para nos retornar os valores mais atualizados disponíveis. O método *get* da interface com a *exchange* está contido no arquivo ini (figura 21, linha de código 2), nele é informado que desejamos obter as informações recentes de *trading* para a moeda que determinamos (parâmetro *ticker*)

e informamos no par de moedas, que no caso é do Bitcoin em relação ao dólar americano.

Com a url indicando o retorno desejado, utilizamos o método *loads* da biblioteca json, com ela salvamos o resultado do retorno da API em uma variável para posterior tratamento.

```
23     api_response = json.loads(requests.get(  
24         |         config.get("DEFAULT", "API_URL")).text)  
25
```

Figura 26 - Retorno da API com dados atuais

Fonte: Os autores (2021)

## 5.8 Previsão do fechamento

Através dos dados obtidos pela API e do modelo gerado através da base de dados em csv da Gemini, o programa gera a previsão do encerramento do intervalo trabalhado e calcula a variação da moeda.

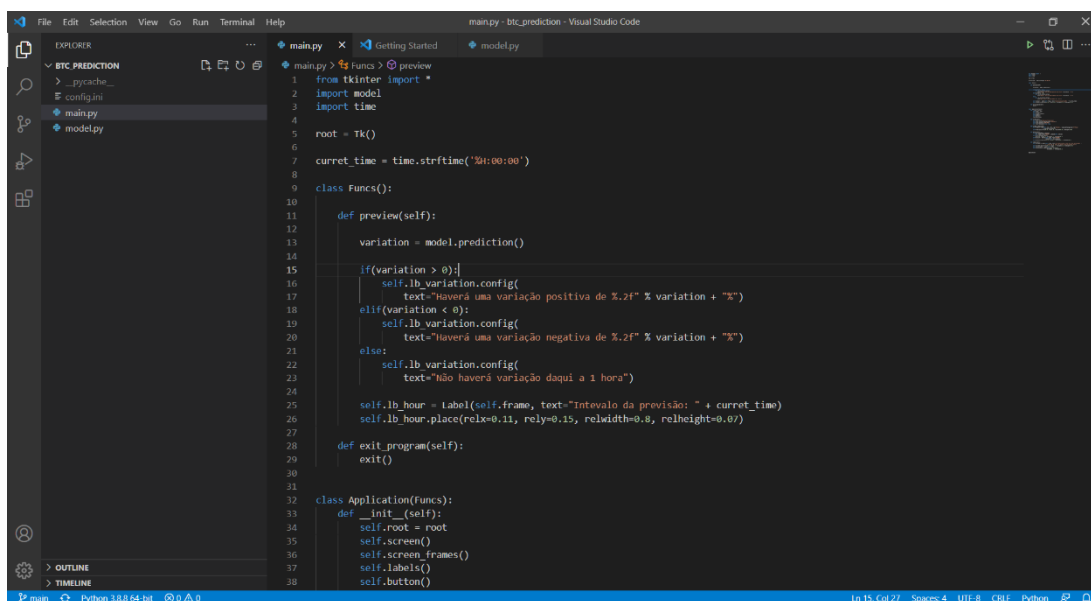
```
25     close_predict = model.predict(  
26         |         [[api_response['open']] + [api_response['high']] + [api_response['low']]])  
27  
28     actual_open = float(api_response['open'])  
29  
30     variation = ((close_predict[0] - actual_open) / actual_open) * 100  
31  
32     return [variation]  
33
```

Figura 27 - Previsão do fechamento e cálculo de variação

Fonte: Os autores (2021)

## 5.9 Apresentação do resultado

Utilizando a biblioteca *tkinter* é gerado os objetos a serem apresentados ao usuário, o *front-end* faz uma chamada na função *prediction* e informa através de desvio condicional o resultado das operações.



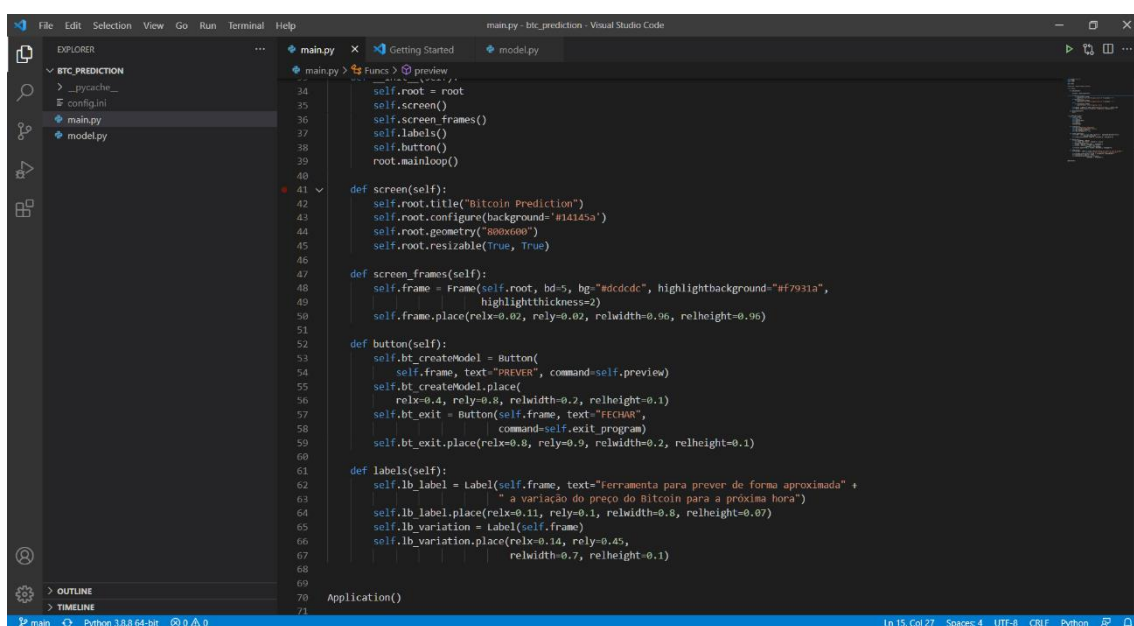
```

1 from tkinter import *
2 import model
3 import time
4
5 root = Tk()
6
7 current_time = time.strftime('%H:%M:%S')
8
9 class Funcs():
10
11     def preview(self):
12
13         variation = model.prediction()
14
15         if(variation > 0):
16             self.lb_variation.config(
17                 text="Haverá uma variação positiva de %.2f" % variation + "%"
18             )
19         elif(variation < 0):
20             self.lb_variation.config(
21                 text="Haverá uma variação negativa de %.2f" % variation + "%"
22             )
23         else:
24             self.lb_variation.config(
25                 text="Não haverá variação daqui a 1 hora"
26             )
27
28         self.lb_hour = Label(self.frame, text="Intervalo da previsão: " + current_time)
29         self.lb_hour.place(relx=0.11, rely=0.15, relwidth=0.8, relheight=0.07)
30
31     def exit_program(self):
32         exit()
33
34 class Application(Funcs):
35     def __init__(self):
36         self.root = root
37         self.screen()
38         self.screen_frames()
39         self.labels()
40         self.button()

```

Figura 28 – Desenvolvimento do *front-end*

Fonte: Os autores (2021)



```

34 self.root = root
35 self.screen()
36 self.screen_frames()
37 self.labels()
38 self.button()
39 root.mainloop()
40
41 def screen(self):
42     self.root.title("Bitcoin Prediction")
43     self.root.configure(background="#14145a")
44     self.root.geometry("800x600")
45     self.root.resizable(True, True)
46
47 def screen_frames(self):
48     self.frame = Frame(self.root, bd=5, bg="#dcdcdc", highlightbackground="#f7931a",
49                         highlightthickness=2)
50     self.frame.place(relx=0.02, rely=0.02, relwidth=0.96, relheight=0.96)
51
52 def button(self):
53     self.bt_createModel = Button(
54         self.frame, text="PREVER", command=self.preview)
55     self.bt_createModel.place(
56         relx=0.4, rely=0.8, relwidth=0.2, relheight=0.1)
57     self.bt_exit = Button(self.frame, text="FECHAR",
58                           command=self.exit_program)
59     self.bt_exit.place(relx=0.8, rely=0.9, relwidth=0.2, relheight=0.1)
60
61 def labels(self):
62     self.lb_label = Label(self.frame, text="ferramenta para prever de forma aproximada" +
63                             " a variação do preço do Bitcoin para a próxima hora")
64     self.lb_label.place(relx=0.11, rely=0.1, relwidth=0.8, relheight=0.07)
65     self.lb_variation = Label(self.frame)
66     self.lb_variation.place(relx=0.14, rely=0.45,
67                             relwidth=0.7, relheight=0.1)
68
69 Application()
70
71

```

Figura 29 – Continuação do desenvolvimento do *front-end*

Fonte: Os autores (2021)

Com a execução desses códigos, é gerado a interface gráfica onde o usuário irá interagir através de um único botão.

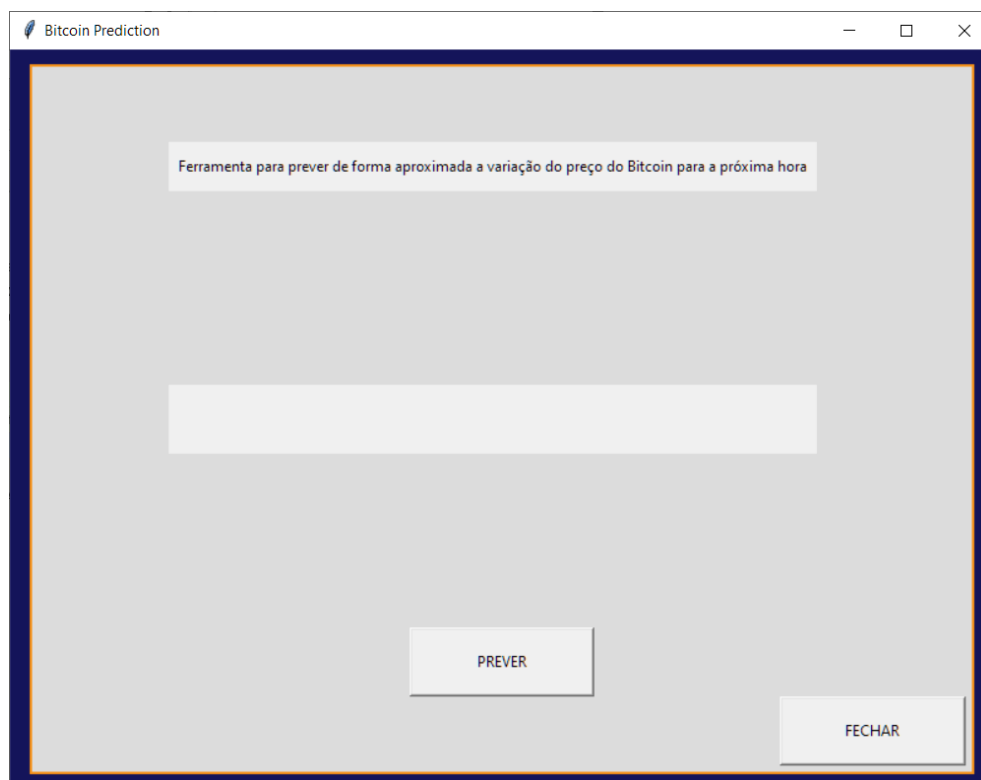


Figura 30 - Execução da aplicação

Fonte: Os autores (2021)

Com o *click* no botão PREVER é efetuada a chamada da função *prediction* e apresentado o resultado na tela, assim como o período de hora ao qual a previsão está baseada.



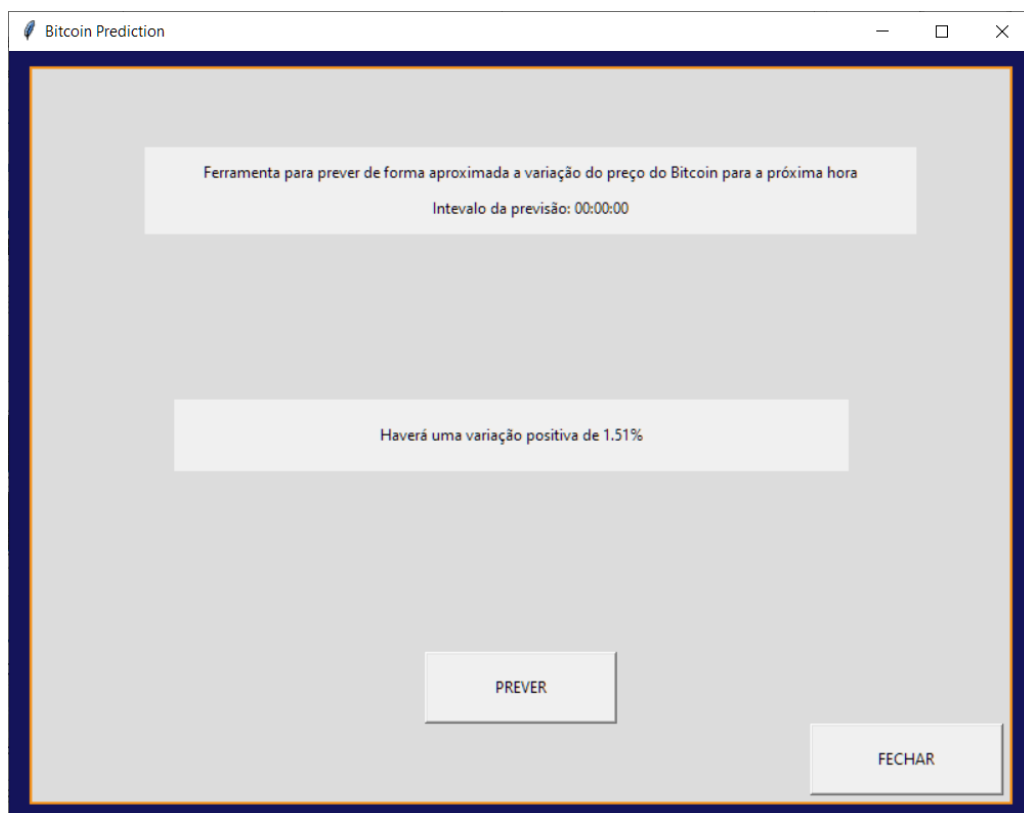


Figura 31 - Apresentação da previsão

Fonte: Os autores (2021)

## 6. RESULTADOS

Neste trabalho abordamos os conceitos que rondam as criptomoedas, o novo dinheiro digital, com uma grande capacidade de revolucionar o modo como vivemos. Além disso, diversos conceitos que tornam esse dinheiro digital ainda mais interessantes, como a *blockchain*, uma tecnologia que como vimos, pode ser aplicada em diversas áreas desde segurança da informação até sistemas de contratos inteligentes, devido a sua grande segurança contra fraudes.

Após as diversas análises através do *software* Weka foi constatado que o algoritmo de regressão linear atende os requisitos do projeto. Tendo resultados satisfatórios, com taxas de erro relativo absoluto menores que 1% e uma alta correlação linear entre as variáveis utilizadas para gerar o modelo.

Com isso, contamos entre os mil casos de validação com aproximadamente apenas 188 dólares de erro médio absoluto, o que representa que a soma da diferença da média da previsão variou 0,41%. Considerando o cenário preditivo, onde aborda números de 5 dígitos e 2 casas decimais é um valor considerável.

Desta forma, conseguimos passar confiança na ferramenta aos investidores, trazendo um resultado com uma ferramenta de desenvolvimento universitário estrangeiro, com grande relevância no nível universitário.

## 7. CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Apresentamos de maneira breve neste trabalho o surgimento do Bitcoin, uma moeda cibernética revolucionária do ponto de vista tecnológico que teve como inovação o uso da *blockchain*, que armazena todas as transações realizadas da moeda de maneira transparente onde qualquer um da rede consegue verificar, porém sem conhecer os reais participantes devido a criptografia dos dados.

Com a confiança garantida pelo Weka, a aplicação foi desenvolvida em *Python*, uma linguagem de programação que conta com diversas bibliotecas voltadas para ciência e análise de dados, como a biblioteca *sklearn* que foi utilizada para gerar o modelo com uma função de regressão linear.

E ainda com a linguagem *Python*, a biblioteca *tkinter* nos permitiu de forma prática e eficiente construir uma interface mais amigável para o usuário. Assim, finalizando o projeto que foi exposto neste trabalho.

Este trabalho teve como objetivo auxiliar os novos investidores na tomada de decisão quanto às suas ordens de compra e venda, agregando com uma solução tecnológica que se embasa através de dados históricos das moedas escolhidas.

A partir deste projeto, há alguns pontos possíveis a serem trabalhados futuramente, como:

- Aprimorar a apresentação dos dados no *front-end* através de gráficos e estatísticas obtidas pelo algoritmo utilizado.
- Desenvolver a possibilidade de trocar a moeda trabalhada e o intervalo de tempo.
- Utilizar uma base de dados diferente de um arquivo csv para melhorar o desempenho da previsão.

## 8. REFERÊNCIAS BIBLIOGRÁFICAS

BITCOIN TRADE. **O que é exchange e como funcionam?**. Disponível em: <https://blog.bitcointrade.com.br/o-que-e-exchange/>. Acesso em: 18 mai. 2021.

BITCOIN TRADE. **O que são altcoins e qual a diferença para o Bitcoin?**. Disponível em: <https://blog.bitcointrade.com.br/o-que-sao-altcoins-e-qual-a-diferenca-para-o-bitcoin-saiba-aqui/>. Acesso em: 21 mai. 2021.

BRITO, Marcelo. Aspectos teóricos da mineração de dados e aplicação das regras de classificação para apoiar o comércio: Veja neste artigo uma apresentação e discussão a respeito da mineração de dados, bem como sua importância no apoio às atividades do comércio. Serão apresentados exemplos práticos de algoritmos de análise de dados estatísticos de determinada empresa.. **DEV MEDIA**, BOM DESPACHO - MG, 2012. Disponível em: <https://www.devmedia.com.br/aspectos-teoricos-da-mineracao-de-dados-e-aplicacao-das-regras-de-classificacao-para-apoiar-o-comercio/25429>. Acesso em: 27 mai. 2012.

BUY BITCOIN WORLDWIDE. **How Many People Own Bitcoin?**. Disponível em: <https://www.buybitcoinworldwide.com/how-many-bitcoin-users/>. Acesso em: 20 mai. 2021.

CH & CO. **How many active crypto traders are there across the globe?**. Disponível em: [https://www.chappuishalder.com/wp-content/uploads/2019/06/Publication\\_Crypto-traders-06-2019.pdf](https://www.chappuishalder.com/wp-content/uploads/2019/06/Publication_Crypto-traders-06-2019.pdf). Acesso em: 20 mai. 2021.

CHAGUE, Fernando; GIOVANNETTI, Bruno. **É possível viver de day-trading? É possível viver de day-trading?**, São Paulo, v. 23, n. 1, p. 4, abr./2019. Disponível em: <https://cointimes.com.br/wp-content/uploads/2019/03/Viver-de-day-trading-1.pdf>. Acesso em: 20 mai. 2021.

CHAPPUISHALDER. **How many active crypto traders are there across the globe**. Disponível em: [https://www.chappuishalder.com/wp-content/uploads/2019/06/Publication\\_Crypto-traders-06-2019.pdf](https://www.chappuishalder.com/wp-content/uploads/2019/06/Publication_Crypto-traders-06-2019.pdf). Acesso em: 23 mai. 2021.

CIÊNCIA E DADOS. **Por que Cientistas de Dados escolhem Python?**. Disponível em: <https://www.cienciaedados.com/por-que-cientistas-de-dados-escolhem-python/>. Acesso em: 21 mai. 2021.

COINEXT. **O que é Ethereum (ETH)**. Disponível em: <https://coinext.com.br/o-que-e-ethereum>. Acesso em: 22 mai. 2021.

CRIPTOFACIL. **O que é mecanismo de consenso?**. Disponível em: <https://www.criptofacil.com/o-que-e-mecanismo-de-consenso/>. Acesso em: 20 mai. 2021.

FAYYAD, U. M.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From Data Mining to Knowledge Discovery in Databases. Artificial Intelligence Magazine, v. 17, n. 3, p. 37-54, 1996a. Acesso em: 28 mai. 2021.

FINANCEONE. **Trade de Bitcoin: o que é e como funciona?**. Disponível em: <https://financeone.com.br/trade-de-bitcoin>. Acesso em: 18 mai. 2021.

HEWLETT PACKARD ENTERPRISE. **O QUE É INTELIGÊNCIA ARTIFICIAL?**. Disponível em: <https://www.hpe.com/br/pt/what-is/artificial-intelligence.html>. Acesso em: 20 mai. 2021.

HOSTINGER. **O Que é GitHub e Para Que é Usado?**. Disponível em: <https://www.hostinger.com.br/tutoriais/o-que-github>. Acesso em: 23 mai. 2021.

INFOMONEY. **Criptomoedas: Um guia para dar os primeiros passos com as moedas digitais**. Disponível em: <https://www.infomoney.com.br/guias/criptomoedas/>. Acesso em: 15 mai. 2021.

INFOMONEY. **O que faz um trader?**. Disponível em: <https://www.infomoney.com.br/guias/trader/>. Acesso em: 17 mai. 2021.

INFOMONEY. **Sobre Bitcoin**. Disponível em: <https://www.infomoney.com.br/cotacoes/bitcoin-btc/>. Acesso em: 17 mai. 2021.

INFONOVA. **Mineração de dados: o que é, para que serve e como fazer?**. Disponível em: <https://www.infonova.com.br/negocios/mineracao-de-dados-para-que-serve-como-fazer/>. Acesso em: 22 mai. 2021.

MARTINS, R. D. S. O. Trabalho de Conclusão de Curso: Aplicação da Mineração de Dados para a recomendação de parâmetros para o COIN-OR Branch and Cut. **Universidade Federal de Ouro Preto**, João Monlevade - MG, ago./2016. Disponível em:

[https://www.monografias.ufop.br/bitstream/35400000/225/1/MONOGRAFIA\\_Aplica%C3%A7%C3%A3oMinera%C3%A7%C3%A3oDados.pdf](https://www.monografias.ufop.br/bitstream/35400000/225/1/MONOGRAFIA_Aplica%C3%A7%C3%A3oMinera%C3%A7%C3%A3oDados.pdf). Acesso em: 24 mai. 2021.

MIT NEWS. **Artificial intelligence model detects asymptomatic Covid-19 infections through cellphone-recorded coughs**. Disponível em: <https://news.mit.edu/2020/covid-19-cough-cellphone-detection-1029>. Acesso em: 21 mai. 2021.

OTÁVIO, João. Tkinter: Interfaces gráficas em Python: Neste artigo você verá uma introdução à criação de interfaces gráficas em Python de maneira simples usando a

biblioteca nativa Tkinter.. **DEVMEDIA**, Fortaleza - CE, 2016. Disponível em: <https://www.devmedia.com.br/tkinter-interfaces-graficas-em-python/33956>. Acesso em: 28 mai. 2021.

SHARDIX. **Jan Lansky's Six Cryptocurrency Requirements**. Disponível em: <https://medium.com/@Shardix/jan-lanskys-six-cryptocurrency-requirements-adaed5058f7f>. Acesso em: 20 mai. 2021.

SILVA, D. C. A. D. INTELIGÊNCIA ARTIFICIAL: TEMA 2: APRENDIZADO DE MÁQUINA. **Unicarioca**, Rio de Janeiro - RJ mai./2021. Disponível em: [Unicarioca.com.br](http://Unicarioca.com.br). Acesso em: 25 mai. 2021.

TECHTUDO. **O que é Hash?**. Disponível em: <https://www.techtudo.com.br/artigos/noticia/2012/07/o-que-e-hash.html>. Acesso em: 20 mai. 2021.

TECNOBLOG. **O que é Python?**. Disponível em: <https://tecnoblog.net/405640/o-que-e-python-guia-para-iniciantes/>. Acesso em: 21 mai. 2021.

THE CAPITAL ADVISOR. **O Que é Bitcoin e Como Funciona a Moeda Digital**. Disponível em: <https://comoinvestir.thecap.com.br/o-que-e-bitcoin-e-como-funciona-a-moeda-digital/>. Acesso em: 15 mai. 2021.

WAIKATO. **Weka**. Disponível em: <https://www.cs.waikato.ac.nz/ml/weka/>. Acesso em: 21 mai. 2021.