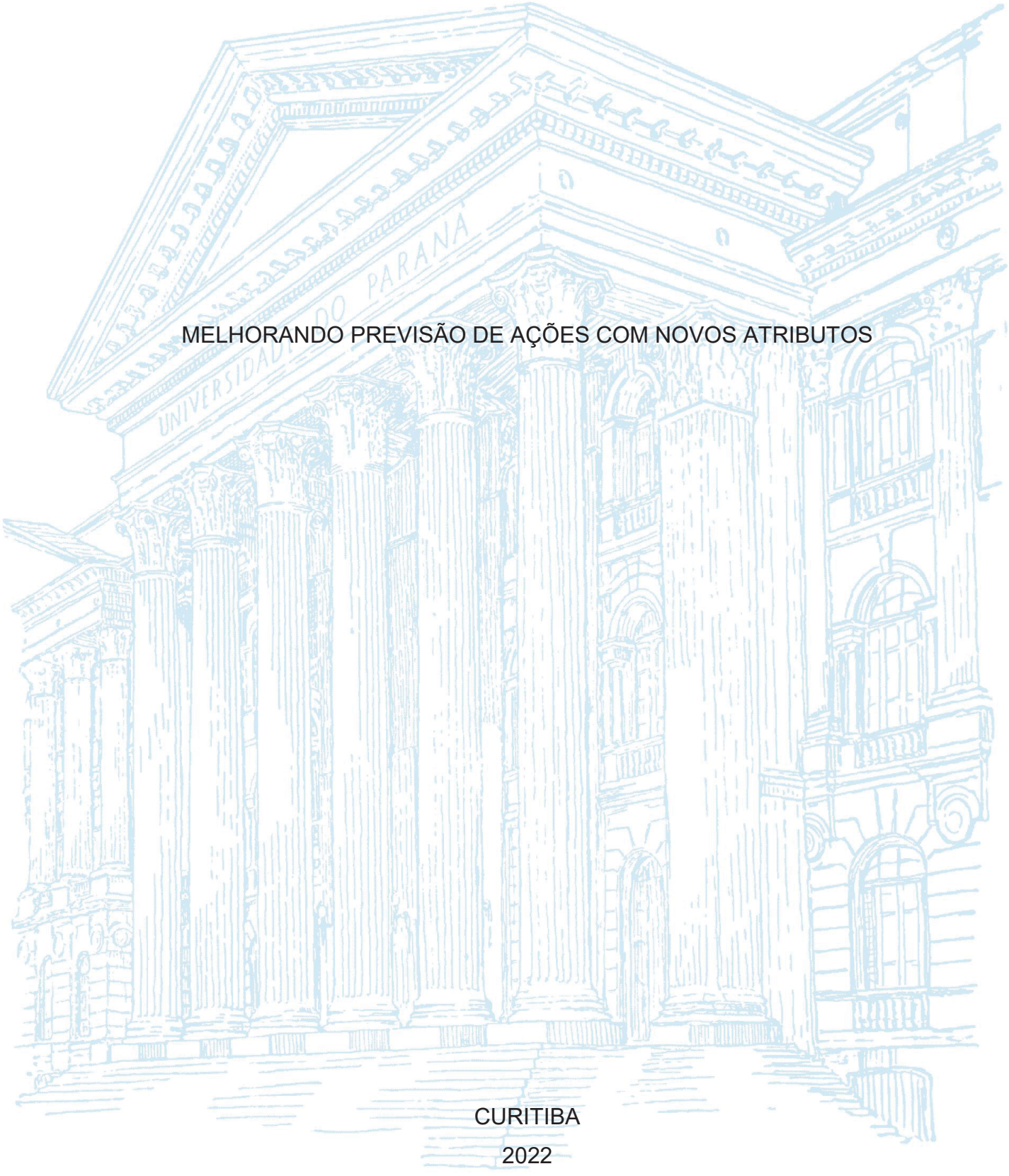
UNIVERSIDADE FEDERAL DO PARANÁ

BRUNNO CUNHA MOUSQUER DE OLIVEIRA



MELHORANDO PREVISÃO DE AÇÕES COM NOVOS ATRIBUTOS

CURITIBA

2022

BRUNNO CUNHA MOUSQUER DE OLIVEIRA

MELHORANDO PREVISÃO DE AÇÕES COM NOVOS ATRIBUTOS

TCC apresentado ao curso de Pós-Graduação em Especialização em Inteligência Artificial Aplicada, Setor de Educação Profissional e Tecnológica, Universidade Federal do Paraná, como requisito parcial à obtenção do título de Especialista em Inteligência Artificial.

Orientador: Prof. Dr. Razer Anthom Nizer Rojas Montaño

CURITIBA

2022

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação INTELIGÊNCIA ARTIFICIAL APLICADA da Universidade Federal do Paraná foram convocados para realizar a arguição da Monografia de Especialização de **BRUNNO CUNHA MOUSQUER DE OLIVEIRA** intitulada: **MELHORANDO PREVISÃO DE AÇÕES COM NOVOS ATRIBUTOS**, que após terem inquirido o aluno e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.
A outorga do título de especialista está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

Curitiba, 13 de Setembro de 2022.

RAZER ANTHOM NIZER ROJAS MONTAÑO
Presidente da Banca Examinadora

JAIME WOJCIECHOWSKI
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Melhorando previsão de ações com novos atributos

Brunno Cunha Mousquer de Oliveira
Especialização em Inteligência Artificial Aplicada
Universidade Federal do Paraná (UFPR)
Curitiba, Brasil
brunnokick@gmail.com

Razer Anthom Nizer Rojas Montaño
Especialização em Inteligência Artificial Aplicada
Universidade Federal do Paraná (UFPR)
Curitiba, Brasil
razer@ufpr.br

***Resumo*—Ocorreu um forte crescimento no mercado de ações no Brasil, muitas pessoas físicas trouxeram seus investimentos para o mercado de renda variável devido a baixa da SELIC e corretoras de investimento sem taxa. Esse estudo tem como objetivo comparar modelos de aprendizado de máquina com atributos diferentes para identificar se é possível melhorar a predição com informações da própria companhia. Foram avaliados quatro algoritmos: árvore de decisão, floresta aleatória, gradiente *boosting* e *long short-term memory*. E quatro datas para previsão: 1, 7, 14 e 28 dias. Os experimentos com as novas informações, como fluxo de caixa e rendimento, tiveram melhoria na previsão de até 33%.**

***palavras chave*—bolsa de valores, aprendizado de máquina**

***Abstract*—There has been a strong growth in Brazil stock market, many people change their investiments to stocks due to a lower SELIC tax and no-fee investiment brokers. This article aims to compare machine learning models with different atributes to identify whether it is possible to improve the prediction with information from the company itself. Four algorithms were evaluated: decision tree, random forest, gradient boosting and long short-term memory. And four dates for prediction: 1, 7, 14 and 28 days. The experiments with the new information, such as cash flow and income had improved forecasting by up to 33%.**

***palavras chave*—stock options, machine learning**

I. Introdução

Conforme Fortuna (2015) [17], o sistema financeiro faz parte de qualquer sociedade moderna e seus serviços estão intrinsecamente ligados ao bem-estar socioeconômico. Produtos de processamento em tempo real como Transferência Eletrônicas Disponível (TED) e boleto eletrônico de cobrança são exemplos dessas melhorias. A tecnologia bancária do Brasil, em geral, é bastante desenvolvida. Como exemplo temos o PIX, criado em novembro de 2020 é um meio de pagamento eletrônico onde as transações são concluídas em poucos segundos.

O sistema financeiro também possibilitou um fluxo de recursos (dinheiro) entre as partes, de quem está com dinheiro disponível no mercado e quem está precisando de dinheiro.

Investimento é a troca de um valor atual, por uma promessa futura. Por exemplo, emprestar R$100,00 hoje para daqui a 12 meses receber de volta R$105,00, obtendo 5% de lucro. Investimentos sempre estão ligados à risco e retorno, e geralmente quanto menor o risco menor o retorno e maior o risco maior o retorno.

Os tipos de investimentos podem ser separados em três categorias. Títulos de renda fixa, prometem um fluxo fixo com base em uma fórmula, taxa de juros por exemplo. Esses títulos possuem um risco muito baixo, consequentemente possuem um baixo retorno. Ao contrário de títulos de renda fixa, ações ou participações acionárias de uma companhia não prometem nenhum retorno. Porém, qualquer dividendo distribuído pela companhia o acionista terá direito de forma proporcional à quantidade de ações em sua posse. Isso significa que o desempenho dos investimentos em ações está diretamente associado à performance da companhia. E por último existe o mercado derivativo, como o nome sugere, esse investimento depende de outro título. A principal função desse ativo é fornecer proteção contra riscos [6].

Em 2018 a B3 (a bolsa de valores brasileira) contava com 814 mil investidores pessoas físicas em renda variável (ações, FIIs, BDRs, ETFs, etc) [8]. Em dezembro de 2021, o número de investidores atingiu a marca de 3,2 milhões, um aumento de 300% em três anos. Dois grandes fatores que ajudaram a contribuir com isso. A baixa histórica da SELIC diminuindo os ganhos da poupança para abaixo da inflação IPCA [25], a democratização do investimento com o movimento de corretoras taxa zero, isso é, não existe cobrança para fazer uma operação de compra ou venda de ações [56].

Na década de 70 Malkiel propôs teoria do passeio aleatório em *Wall Street*, bolsa de valores dos Estados Unidos. Conforme Codling *et al* (2008) [9], o passeio aleatório é um processo que ocorre dentro de um espaço matemático e consiste em uma sucessão de passos aleatórios. Dado uma posição em um espaço unidimensional definido pela função $P(x,t)$ onde x é a posição no espaço e t o tempo. Iniciando o passeio na posição $P(0,0)$ o primeiro passo tem 50% de chance de ser para direita e 50% de chance de ser para esquerda, $P(-1,1)$ ou $P(1,1)$. No segundo passo a chance é de 25% para cada uma das posições abaixo:

$$P(-2,2), P(0,2), P(1,2), P(2,2)$$

Após uma quantidade significativa de passos podemos gerar uma distribuição gaussiana, com a média 0 e variância a quantidade de passos dados. Aplicando a teoria no mercado de ações significa que as variações da bolsa são independente do histórico [33].

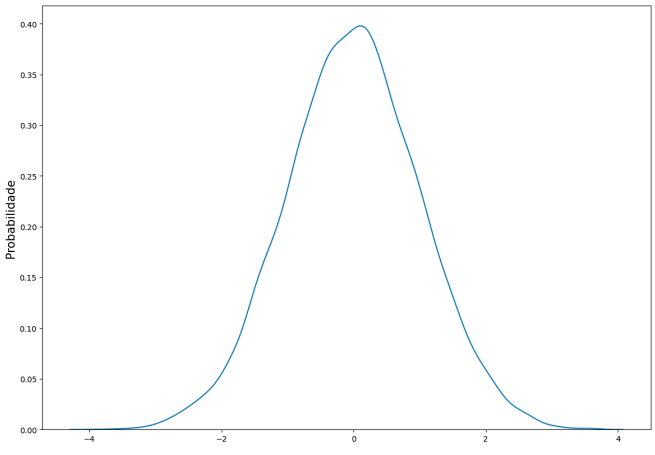


Figura 1. Distribuição Gaussiana. Fonte: Elaborado pelo Autor.

Segundo Kobori (2019) [31], dentre as diversas técnicas de análises praticadas pelo mercado, duas se destacam. A análise técnica busca tendências por meio de interpretações de gráficos, enquanto a análise fundamentalista, por meio de relatórios sobre a gestão da empresa.

Segundo Lima (2016) [32], a análise técnica busca indicadores gráficos para identificar uma tendência nos preços dos ativos. Para isso é analisado o comportamento histórico do ativo para mostrar um comportamento semelhante no futuro. O termo "técnica" se refere ao estudo da ação do próprio mercado.

De acordo com Edward e Magee (2007) [15], a análise técnica argumenta que é inútil atribuir um valor intrínseco a uma ação, pois o valor real em qualquer tempo é determinado pela oferta e demanda, e isso se reflete na ação do ativo.

Ao contrário da análise técnica, a análise fundamentalista acredita que exista um valor intrínseco na ação, ao analisar variáveis externas de desempenho econômico, avaliações e comparações setoriais e variáveis internas como demonstrativo, e ao adotar indicadores como: Lucro por Ação, Indicadores de Dividendos, Índice Preço/Lucro e EBITDA [49].

Segundo Penteado (2003) [39], o objetivo da técnica é determinar o real valor da ação e com isso compará-la com o preço de mercado. Ações que estão com os preços acima do valor de mercado estão sobreavaliadas, enquanto ações com o preço abaixo do valor de mercado estão subavaliadas.

A. *Objetivos*

O objetivo desse artigo é avaliar se é possível melhorar modelos de inteligência artificial de previsão de ações com atributos contábeis ou financeiros.

As análises serão feitas comparando técnicas de aprendizado de máquina com e sem esses atributos e comparando os modelos entre si. Para entender quais modelos se beneficiaram com os atributos e qual modelo obteve o melhor resultado.

Serão testados quatro algoritmos com dois conjuntos de atributos para 4 datas de previsão. Totalizando 32 experimentos, conforme a Tabela I.

Tabela I
EXPERIMENTOS

Dias	Modelo	Atributos
1	Árvore de Decisão	Com e Sem atributos financeiros
1	Floresta Aleatória	Com e Sem atributos financeiros
1	Gradient Boosting	Com e Sem atributos financeiros
1	Long Short-Term Memory	Com e Sem atributos financeiros
7	Árvore de Decisão	Com e Sem atributos financeiros
7	Floresta Aleatória	Com e Sem atributos financeiros
7	Gradient Boosting	Com e Sem atributos financeiros
7	Long Short-Term Memory	Com e Sem atributos financeiros
14	Árvore de Decisão	Com e Sem atributos financeiros
14	Floresta Aleatória	Com e Sem atributos financeiros
14	Gradient Boosting	Com e Sem atributos financeiros
14	Long Short-Term Memory	Com e Sem atributos financeiros
28	Árvore de Decisão	Com e Sem atributos financeiros
28	Floresta Aleatória	Com e Sem atributos financeiros
28	Gradient Boosting	Com e Sem atributos financeiros
28	Long Short-Term Memory	Com e Sem atributos financeiros

B. *Justificativas*

O artigo trata previsão da ações na bolsa de valores adicionando atributos de históricos da companhia, unificando a análise técnica com a fundamentalista.

C. *Organização do Artigo*

O artigo é composto por 5 capítulos. Começando por uma introdução do feita no Capítulo I e que aprofundada no Capítulo II com a fundamentação teórica. O Capítulo III irá detalhar como os dados e os modelos foram tratados e desenvolvidos. O Capítulo IV será apresentado os resultados e análises dos experimentos enquanto o Capítulo V é contêm as considerações finais.

II. FUNDAMENTAÇÃO TEÓRICA

A. *Sistema Financeiro*

Segundo Assaf (2018) [3], o sistema financeiro brasileiro é composto pelas instituições do Sistema Financeiro Nacional (SFN). Essas instituições são:

- Conselho Monetário Nacional (CMN);
- Conselho Nacional de Seguros Privados (CNSP);
- Conselho Nacional de Previdência Complementar (CNPC).

Fundado em 31 de dezembro de 1964, o Conselho Monetário Nacional é formado pelo ministro da fazenda, ministro do planejamento e presidente do banco central. O CMN tem como objetivo o desenvolvimento econômico e social do país. Para isso, a instituição age como regulador do valor da moeda interna e externamente. Além de orientar a aplicação dos recursos das instituições financeiras públicas e privadas.

Existem duas entidades que estão sob supervisão do CMN: o Banco Central do Brasil (Bacen) e a Comissão de Valores Mobiliários (CVM). Também conhecido como banco dos bancos, o Bacen regula a quantidade de moeda em circulação para garantir a estabilidade financeira e dos preços.

Conforme a resolução número 24, de cinco de março de 2021 [13], a CVM foi criada em 1976 e é uma entidade autárquica, independente, vinculada ao Ministério da Economia.

Conforme o segundo artigo da resolução, a CVM possui oito finalidades:

- Estimular a formação de poupança e a sua aplicação em valores mobiliários;
- Promover a expansão e o funcionamento eficiente e regular do mercado de ações, estimular as aplicações permanentes em ações do capital social de companhias abertas sob controle de capitais privados nacionais;
- Assegurar o funcionamento eficiente e regular dos mercados da bolsa e do balcão;
- Proteger os titulares de valores mobiliários e os investidores do mercado contra:
  - Emissões irregulares de valores mobiliários;
  - Atos ilegais de administradores e acionistas controladores das companhias abertas, ou de administradores de carteira de valores mobiliários;
  - Uso de informação relevante não divulgada no mercado de valores mobiliários;
- Evitar ou coibir modalidades de fraude ou manipulação destinadas a criar condições artificiais de demanda, oferta ou preço dos valores mobiliários negociados no mercado;
- Assegurar o acesso do público a informações sobre os valores mobiliários negociados e as companhias que os tenham emitido;
- Assegurar a observância de práticas comerciais equitativas no mercado de valores mobiliários;
- Assegurar a observância, no mercado, das condições de utilização de crédito fixadas pelo Conselho Monetário Nacional.

B. Mercado Financeiro

Conforme Pesente (2019) [40], o Mercado Financeiro possibilitou que os agentes superavitários (poupadores) emprestem seus recursos diretamente aos agentes deficitários (tomadores). Esses recursos podem ser títulos, por exemplo. Essas operações ocorrem através de uma prestadora de serviço chamada instituição financeira. De acordo com Silvério (2008) [52], as instituições financeiras são responsáveis por realizar diversas operações de crédito. O crédito é uma troca de algum bem por uma promessa de pagamento no futuro. Corretoras da bolsa de valores como a Clear Corretora estruturam as operações, assessoram na formação de preços buscam clientes, entre outros.

Figura 2. Mercado de capitais. Fonte: CVM.

Segundo Pesente (2019) [40], o Mercado Financeiro é segmentado em quarto formas: Mercado de Monetário, Mercado de Capitais, Mercado de Crédito e Mercado de Câmbio.

No Mercado Monetário as transações são realizadas entre as próprias instituições financeiras, ou entre as instituições e o Banco Central. Essas transferências ocorrem em um curto prazo, geralmente um dia, também conhecidas como *overnight*. Esse mercado tem forte atuação do Banco Central, que intervém para controlar a liquidez da economia. Por exemplo, se o volume de dinheiro estiver menor na economia do que o esperado, o Banco Central irá comprar títulos para injetar dinheiro na economia.

O Mercado de Capitais diferente do Monetário possui um prazo maior e cria as condições para que empresas privadas captem recursos diretamente dos investidores. Para o investidor é uma forma de investir diferente das aplicações tradicionais oferecidas pelos banco e governos, é esperado um rendimento maior do que as aplicações tradicionais porém um risco maior também.

O Mercado de Crédito é utilizado para que instituições financeiras emprestem dinheiro a famílias ou empresas. A diferença entre o custo da captação e o que é cobrado dos tomadores é chamado de *spread*.

Mercado de Câmbio ocorre operações de troca entre a moeda nacional e moedas estrangeiras. Assim como no Mercado Monetário, o Banco Central participa para poder aplicar a sua política cambial, comprando ou vendendo moeda para controlar o volume de reservas internacionais.

C. Bolsa de Valores

Bolsas de valores são locais que disponibilizam as condições para compra e venda de títulos mobiliários. Atualmente as negociações são realizadas de forma eletrônica através de um software disponibilizado pela Bolsa.

A bolsa de valores atua no Mercado de Capitais que por sua vez possui dois segmentos: o mercado primário, quando o ativo é negociado pela primeira vez, e o mercado secundário, para compra e venda de títulos já lançados pelo mercado primário [3].

Como o exemplo a companhia Nubank [36] que durante o desenvolvimento desse artigo estava na abertura do IPO (Oferta Pública Inicial). IPO é quando uma empresa entra na bolsa de valores e disponibiliza suas ações ao público. Os valores das ações primárias vão direto para o caixa da companhia, pois são compradas diretamente da companhia. Quando essas ações forem negociadas entre investidores serão chamadas de ações secundárias, que nesse caso os valores não vão para a companhia, e sim para quem negociou a ação [14].

A história da bolsa de valores no Brasil tem mais de 100 anos. As primeiras bolsas surgiram em 1851 no Rio de Janeiro - RJ e Salvador - BA. Porém, somente em 1934 abriu a primeira Bolsa Oficial de São Paulo, com corretores oficiais de fundos públicos nomeados pelo governo. Em 1967, essa bolsa passou a se chamar Bovespa (Bolsa de Valores de São Paulo), e os corretores oficiais se tornaram sociedades corretoras. Devido à evolução do mercado e às inovações

financeiras, novos títulos e valores mobiliários foram surgindo. Para atender essas evoluções foram criadas duas novas bolsas: a CETIP (Central de Custódia e Liquidação Financeira de Títulos Privados) em 1984, para lidar com títulos de renda fixa no Brasil, e a BM&F (Bolsa de Mercadorias e Futuros) em 1986, para negócios de contrato de mercadorias e derivados. Buscando as melhores práticas do mercado internacional de qualidade e governança, ocorreu em 2008 uma fusão entre a BOVESPA e BM&F, formando a BM&FBOVESPA. E em 2017, a BM&FBOVESPA e a CETIP se fundiram para formar a B3 - Brasil, Bolsa, Balcão.

Atualmente a B3 é uma das maiores bolsas de valores do mundo, proporcionando um ambiente transparente para negociação de títulos mobiliários [14].

De acordo com Aguiar, (2015) [2], o mercado de ações é influenciado pela cultura dos povos. Por exemplo, a Alemanha e o Japão são países que possuem maior conservadorismo contábil, atribuindo menor importância para resultados de curto prazo e priorizando a continuidade da empresa. Em contrapartida, empresas britânicas historicamente dependeram mais do mercado acionário, adotando maior ênfase em resultados a curto prazo.

Para Leonardo Faccini (2015, p. 106) [16]:

> Ação é um título que confere a propriedade da menor parcela em que se divide o capital social de uma sociedade anônima. O acionista é um dos proprietários da empresa e, nessa condição, têm direito a participar dos seus resultados, na proporção do número de ações que detenha em relação ao total emitido.

Conforme a CVM, (2019) [14], o acionista, detentor da ação é um sócio da companhia. Em caso de sucesso, isso significa valorização do preço da ação e também a o recebimento da distribuição de dividendos e juros sobre capital próprio. Os pagamentos aos acionistas dependem de uma série de fatores, como o caixa disponível e a necessidade de investimento.

Existem duas classes de ações, as ordinárias e as preferenciais. As ações ordinárias dão direito ao voto nas assembleias, sendo um voto para cada ação. O mínimo de ações ordinárias obrigatórias é de 50%, conforme à lei nº 10.303/2001. Isso significa que para uma pessoa controlar uma empresa ela precisa de 25% de ações ordinárias mais uma ação para ter maioria em votações. As ações preferenciais possuem prioridades em distribuição de dividendos e reembolso de capital com ou sem prêmio [16].

D. Aprendizado de Máquina

Segundo Rashka e Mirjalili (2019) [44], aprendizado de máquina ou *machine learning* é a aplicação e ciência que traz sentido aos dados. Os algoritmos de aprendizado de máquina já estão presentes em todos os lugares: transações de cartão de crédito são validadas em tempo real para detectar fraude; algoritmos que buscam o melhor resultado; câmeras digitais que identificam rosto de pessoas, além de reconhecimento de voz utilizado em *smartphones* [51].

Os modelos de aprendizado de máquina podem ser classificados em três tipos, supervisionado, não supervisionado e de reforço.

Conforme Harrington (2012) [26], os modelos supervisionados possuem dois tipos de variáveis, as independentes e as dependentes também chamadas de atributo alvo. O objetivo do modelo supervisionado é compreender as regras das variáveis independentes para explicar o atributo alvo. Por exemplo, dado um conjunto de e-mails classificados como *spam* e não *spam*, o título e o corpo do e-mail são as variáveis independentes e a classificação de *spam* é o atributo alvo. O modelo deve então aprender os padrões das variáveis independentes que possam explicar o atributo alvo, essa etapa é chamada de treinamento do modelo. Com o modelo treinado é possível apresentar e-mails que não foram classificados para que o modelo possa inferir uma classificação com base no que ele aprendeu.

A classificação é uma categoria de modelo supervisionado que além dos valores binários ele também pode ser usado em multi classe, como categorizar se um e-mail é propaganda, rede social ou compras. A outra categoria de modelo supervisionado é a regressão, que trata valores numéricos contínuos do atributo alvo, por exemplo ações na bolsa de valores.

Ao contrário dos modelos supervisionados, os modelos não supervisionados não possuem um atributo alvo e eles são utilizados para agrupar informações [51]. Por exemplo, regras de associação de um carrinho de compras podem ser utilizadas para sugerir itens geralmente comprados em conjunto, se o cliente adicionar queijo e leite ao carrinho ele tem 70% de chance comprar pão [4].

Conforme Rashka e Mirjalili (2019) [44], os modelos de reforço são sistemas que interagem com o ambiente para escolher a melhor ação com base em uma função de recompensa. Por exemplo, um modelo para jogar xadrez conforme a Figura 3. Com base na disposição das peças (ambiente) o sistema irá fazer um novo de movimento de peça (ação) que irá gerar uma nova recompensa, que nesse exemplo é ganhar ou perder o jogo.

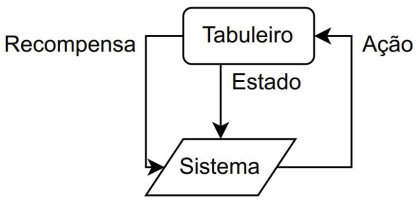


Figura 3. Modelo de reforço. Fonte: Elaborado pelo Autor.

Conforme (Shalev-Shwayds e Ben-David (2014) [51], um modelo de *machine learning* pode seguir a seguinte estrutura:

- *Domain set*: Também conhecido como X, representa o conjunto de atributos em um determinado espaço;
- *Label set*: Também conhecido como y, representa os valores a serem inferidos;
- Dados de treinamento: Uma sequência de pares de dados de X e y que o modelo irá aprender;

- Dados de validação: Uma sequência de pares de dados de X e y que o modelo irá inferir o y e comparar com o y real;
- Dados de teste: Uma sequência de dados de X, que o modelo irá inferir o y.

A Tabela II é uma amostra de informações sobre uma ação qualquer com dados diários para demonstrar a estrutura de um modelo de *machine learning*.

Por exemplo, todas as 10 linhas e as colunas "Índice", "Data" e "Variação do dia anterior" da representam *Domain set*. A coluna "Valor" representa o *Label set*, que é coluna que com base no *Domain set* será aprendida. Para esse exemplo iremos usar uma separação dos dados de 60% para treinamento, 20% para validação e 20% para teste. Com isso as linhas do "Índice" de 1 a 6 serão utilizados como dados de treinamento, enquanto as linhas 7 e 8 serão os dados para validação. Os dados para validação são utilizados para gerar a métrica utilizada no modelo, direcionando se o modelo está bem ou não. Uma vez que o modelo alcance um resultado satisfatório com os dados de validação, o modelo será treinado novamente com os dados de treinamento (linhas 1 a 6) e com os dados de validação (linhas 7 e 8) para que possa prever os dados de teste.

Tabela II
EXEMPLO DE DADOS TABULADOS

Índice	Data	Variação do dia anterior	Valor
1	2021-02-01	0,02	2,90
2	2021-02-02	0,01	2,92
3	2021-02-03	0,02	2,91
4	2021-02-04	-0,01	2,95
5	2021-02-05	0,04	2,97
6	2021-02-08	0,02	2,93
7	2021-02-09	-0,04	2,92
8	2021-02-10	-0,02	2,90
9	2021-02-11	0,02	2,92
10	2021-02-12	0,02	2,89

Na literatura existem vários tipos de algoritmos de regressão que podem ser utilizados para previsão no mercado de ações. A seguir serão descritos quais foram utilizados.

1) Árvore de decisão: as árvores de decisão são uma forma de gerar uma série de regras de classificação com base em algoritmos do tipo *TDIDT* (*Top-Down Induction of Decision Trees*), como ID3 e C4.5.

A árvore de decisão é criada a partir da divisão no valor dos atributos. Um atributo pode ser categórico, finito e relativamente pequeno, como tempo chuvoso ou ensolarado, curso de especialização ou estado federativo. Ou mesmo numérico, como idade, temperatura ou umidade.

O processo de divisão do valor dos atributos ocorre testando o valor do atributo e criando um galho com no mínimo um e no máximo dois possíveis resultados. Em casos de valores numéricos contínuos são criadas regras como 'menor ou igual que' ou 'maior que'. O processo é repetido até que todos os valores sejam classificados [4].

De acordo com Rokach e Maimon (2005) [45], uma árvore de decisão possui um nó chamado raiz que é o único ponto de entrada da árvore. A partir da raiz a árvore se divide em nós e esses nós se dividem em mais nós, formando uma árvore enraizada. Quando o nó não tem saída, ele é chamado de folha ou nó de decisão. A Figura 5 representa um árvore de decisão que possui os nós: "PAGA DIVIDENDOS", "VALOR >= 100" e "VALOR < 500". E os nós de decisão: "NÃO COMPRAR"e "COMPRAR"

Utilizando como exemplo os dados da Tabela III podemos treinar um modelo de árvore de decisão para comprar ou não determinada ação. Os atributos "Paga Dividendos" e "Banco" são do tipo categórico binário, podendo variar somente em "SIM" e "NÃO". O atributo "Ação" é somente uma identificação e não será usado para treinar o modelo. Com base nas colunas "Paga Dividendos" e "Banco" o algoritmo deve ser treinado para identificar "Comprar", um problema de classificação.

Tabela III
EXEMPLO DE DADOS TABULADOS PARA ÁRVORE DE DECISÃO (1)

Ação	Paga Dividendos	Banco	Comprar
AAA	SIM	SIM	SIM
BBB	NÃO	NÃO	NÃO
CCC	NÃO	SIM	SIM
DDD	SIM	SIM	NÃO
EEE	SIM	NÃO	SIM
FFF	NÃO	SIM	NÃO

Conforme Mitchel (1997) [21], o algoritmo de classificação da árvore de decisão utiliza uma técnica da teoria da informação chamada entropia. A entropia da informação pode ser definida pela quantidade de impureza de uma determinada coleção de dados. Dado a Figura 4 onde o eixo *y* é a entropia e o eixo *x* a probabilidade de uma classificação binária, a maior entropia possível (1) ocorre quando a probabilidade é de 50%. Isso significa que os dados estão divididos igualmente, caso a entropia seja diferente de 50% os dados não estão balanceados e uma das variáveis binárias possui mais ou menos registros.

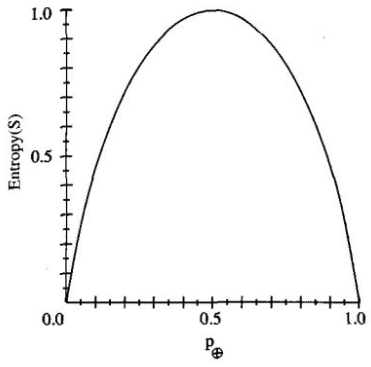


Figura 4. A função da entropia. Fonte: Elaborado pelo Autor.

Abaixo será aplicado o cálculo da entropia para identificar o grau de impureza da coluna "Comprar" da Tabela III.

$$Entropia = -\sum_{i=1}^{K} p_i \log_2 p_i$$

$$E = -(\frac{3}{6}.\log_2 \frac{3}{6} + \frac{3}{6}.\log_2 \frac{3}{6})$$

$$E = -(\frac{1}{2}.\log_2 \frac{1}{2} + \frac{1}{2}.\log_2 \frac{1}{2})$$

$$E = -(\frac{1}{2}.(-1,0) + \frac{1}{2}.(-1,0)$$

$$E = -(0,5.(-1,0) + 0,5.(-1,0)$$

$$E = 1$$

O lado esquerdo da primeira função corresponde ao cálculo do "SIM" que possui $\frac{1}{2}$ da probabilidade, enquanto a segunda parte é em relação ao "NÃO" com $\frac{1}{2}$ da probabilidade. Com a entropia da coluna "Comprar" calculada ($E = 1$), aplica-se a cálculo do ganho de informação com a fórmula abaixo:

$$Info = Entropia_{pai} - \sum Peso_{filho}.Entropia_{filho}$$

Sendo a $Entropia_{pai}$ o valor calculado anteriormente e $Peso_{filho}$ e $Entropia_{filho}$ são valores calculados para cada coluna de atributo. Abaixo o calculo de entropia para o atributo "Paga Dividendos":

$$Entropia = -\sum_{i=1}^{K} p_i \log_2 p_i$$

$$E = -(\frac{2}{3}.\log_2 \frac{2}{3} + \frac{1}{3}.\log_2 \frac{1}{3})$$

$$E = -(0,66.(-0,58) + 0,33.(-1,58)$$

$$E = 0,90$$

O lado esquerdo da primeira função corresponde a probabilidade de pagar dividendos e comprar a ação serem verdadeiros, conforme a Tabela IV. Enquanto o lado direito o corresponde a pagar dividendo e não comprar a ação, conforme a Tabela V.

Tabela IV
PAGA DIVIDENDOS EXPLICANDO COMPRAR

Ação	Paga Dividendos	Comprar
AAA	SIM	SIM
CCC	NÃO	SIM
EEE	SIM	SIM

Tabela V
PAGA DIVIDENDOS EXPLICANDO NÃO COMPRAR

Ação	Paga Dividendos	Comprar
BBB	NÃO	NÃO
DDD	SIM	NÃO
FFF	NÃO	NÃO

O peso de cada atributo é calculado pela formula abaixo, sendo S_v a quantidade de amostras do nó filho para um atributo específico e S a quantidade de amostras do nó pai:

$$Peso = \frac{|S_v|}{|S|}$$

Aplicando a formula para no atributo "Paga Dividendos":

$$Peso_{sim} = \frac{3}{6} = \frac{1}{2}$$

$$Peso_{nao} = \frac{3}{6} = \frac{1}{2}$$

Aplicando a formula de ganho da informação no atributo "Paga Dividendos":

$$Info = Entropia_{pai} - \sum Peso_{filho}.Entropia_{filho}$$

$$Info = 1 - (\frac{1}{2}.0,90 + \frac{1}{2}.0,90)$$

$$Info = 0.09$$

O ganho de informação aplicado para o atributo "Banco" é zero, isso significa que o atributo não serve para explicar a variável "Comprar". Somente com os atributos "Paga Dividendos" e "Banco" não foi possível explicar quando comprar ou não determinada ação, para melhorar o algoritmo será adicionado a coluna "Valor" disponível na Tabela VI.

Tabela VI
EXEMPLO DE DADOS TABULADOS PARA ÁRVORE DE DECISÃO (2)

Ação	Paga Dividendos	Valor	Comprar
AAA	SIM	80	SIM
BBB	NÃO	200	NÃO
CCC	NÃO	500	SIM
DDD	SIM	300	NÃO
EEE	SIM	90	SIM
FFF	NÃO	50	NÃO

O modelo pode ser descrito pela Figura 5 que inicia pelo nó raiz "Paga Dividendos" que possui dois galhos de decisão. No caso de "NÃO" pagar dividendos é levado para um novo galho com duas possibilidades, "VALOR" menor que 500 e "VALOR" maior igual a 500. O "VALOR" maior igual a 500 será uma folha de decisão para "COMPRAR", enquanto menor igual a 500 será uma folha de decisão para "NÃO COMPRAR". Caso a ação pague dividendos, é necessário entrar em um novo galho para dividir o atributo "VALOR". Caso o "VALOR" seja maior ou igual a 100 a ação não deve ser comprada, caso contrário deve comprar a ação.

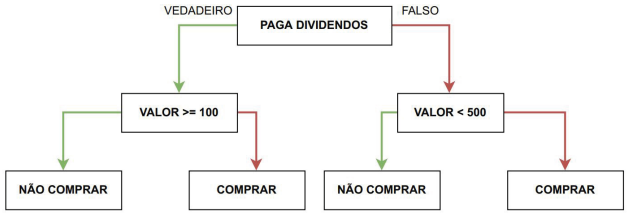


Figura 5. Árvore de decisão. Fonte: Elaborado pelo Autor.

A simplicidade do modelo de árvore de decisão é uma vantagem devido à fácil demonstração de como o algoritmo se comporta. Por exemplo, ao selecionar a ação "AAA" que paga dividendo e possui o valor de 100, ao seguir o algoritmo da árvore de decisão irá resultar na folha de decisão "Não Comprar" na esquerda da imagem.

2) Floresta aleatória: conforme Aggarwal (2015) [1], classificadores diferentes podem ter predições diferentes devido às características de cada classificador. Métodos de conjunto (*Ensemble Methods*) são uma forma de melhorar a precisão de uma predição combinando o resultado de múltiplos modelos, ou gerando diferentes dados de treino para um mesmo modelo. Essa técnica se chama *bootstrap aggregation* ou *bagging* e a partir de uma fonte de dados, *N* amostragens são selecionadas. As amostras selecionadas possuem substituição, cada unidade amostral tem chance de ser selecionada mais de uma vez. Com as amostras selecionadas, são criados novos modelos ou treinado um único modelo com essas várias amostras. Os resultados então são combinados em um único modelo final [4].

Podemos exemplificar a técnica de *bagging* utilizando a Tabela VII. A tabela possui dez observações únicas, ao aplicar a técnica na Tabela VIII os registros 3, 5 e 9 são repedidos repetidos. O processo de gerar novas tabelas reduz o viés e a variância ao comparado com os modelos individuais. O Viés é um erro de generalização do modelo, por exemplo, gerou uma hipótese errada com os dados de treino e aplicou essa hipótese no conjunto de teste [22]. A variância se deve a uma sensibilidade excessiva do modelo ao dados de treino. Por exemplo, treinar o modelo com diferentes conjuntos de dados irá gerar resultados inconsistentes para um mesmo conjunto de teste [1].

Tabela VII
TABELA COM OBSERVAÇÕES ÚNICAS

Índice	Código Ação	Data	Valor
1	AAA	2021-03-01	10,53
2	AAA	2021-03-02	11,03
3	AAA	2021-03-03	10,75
4	BBB	2021-03-01	20,26
5	BBB	2021-03-03	21,5
6	BBB	2021-03-06	20,75
7	BBB	2021-03-07	20,89
8	CCC	2021-03-02	7,53
9	CCC	2021-03-03	7,42
10	CCC	2021-03-04	7,44

Tabela VIII
TABELA APÓS APLICAR A TÉCNICA BAGGING

Índice	Código Ação	Data	Valor
1	AAA	2021-03-01	10,53
2	AAA	2021-03-02	11,03
3	AAA	2021-03-03	10,75
3	AAA	2021-03-03	10,75
5	BBB	2021-03-03	21,5
5	BBB	2021-03-03	21,5
5	BBB	2021-03-03	21,5
8	CCC	2021-03-02	7,53
9	CCC	2021-03-03	7,42
9	CCC	2021-03-03	7,42

Breiman (2001) [5] define floresta aleatória como uma coleção de algoritmos de árvore em que cada árvore possuí um voto de decisão na predição final. Conforme Hastie *et al* (2009) [27], as florestas aleatórias utilizam a técnica de *bagging* em conjunto com árvore de decisão. Quando está trabalhando com regressão o mesmo modelo é treinado com amostras diferentes e é feita uma média os resultados preditos para obter um único resultado. Para classificação é realizado um processo de votação para a classe com mais probabilidade.

A Figura 6 é um exemplo de floresta aleatória para regressão. Partindo de uma única fonte de dados são gerados N árvores aleatórias. A aleatoriedade do algoritmo é inserida na criação das árvores, diferente da árvore de decisão que sempre busca o melhor caminho. Ao predizer um resultado através desse modelo, é feito uma média de todas as árvores para chegar no resultado final, no caso da regressão.

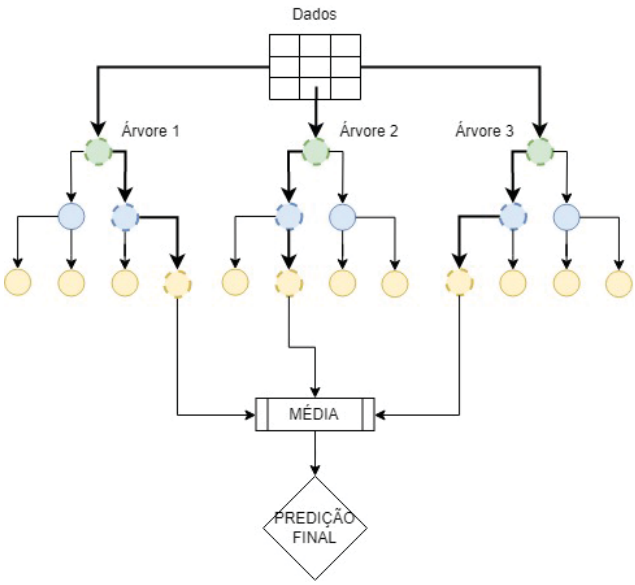


Figura 6. Árvore Aleatória. Fonte: Elaborado pelo Autor.

3) Gradient Boosting: proposto em 1999 por Jerome H. Friedman [21] e atualizado em 2001 [20], é um modelo utilizado para regressão e classificação. Conforme Natekin e Knoll (2013) [35], o objetivo do modelo é gerar diversos novos modelos correlacionados com o gradiente da função

de perda. Para isso, são utilizados algoritmos como árvore de decisão, porém diferente da técnica *bagging* utilizada em florestas aleatórias, os novos modelos são criados de forma sequencial, melhorando em cada nova etapa.

AdaBoost é um algoritmo que foi introduzido pela primeira vez em 1996 por Freund e Schapire [18] utiliza a técnica de aprendizado fraco, definida pelo próprio Schapire em (1990, pág. 201) [47] como:

> O aprendizado fraco é quando o modelo consegue performar um pouco melhor do que a aleatoriedade.

O algoritmo inicia aplicando um peso para cada observação $D = 1/n$, onde n é quantidade de observações disponíveis dos dados de treino e D o peso. Nesse primeiro momento todas as observações possuem a mesma importância. Em sequência é realizado uma série de rodadas sequenciais $t = 1, ..., T$, sendo T uma quantidade de modelos fracos. Ao final de cada rodada é calculado a importância do modelo com base em uma função de perda e os pesos das observações são atualizadas, aumentado o peso das observações erradas e reduzindo das observações corretas. Funções de perda como método dos mínimos quadrados

$$\sum_{i=1}^{n}(y_{verdadeiro} - y_{predito})^2$$

são utilizadas para calcular o erro do modelo com base nos resíduos, valores verdadeiros e preditos de *y*.

Na rodada seguinte o peso das obervações gerados pela rodada anterior é levado em consideração para que o novo modelo treine com as observações erradas. Diferente da técnica utilizada pela floresta aleatória onde todos os modelos possuem o mesmo peso na votação, *boosting* que assim como *bagging* são técnicas de *ensemble*, dá um peso maior aos modelos que obtiveram menor erro [19].

Conforme Vargas [55], gradiente descendente é um técnica para buscar de forma interativa os valores dos parâmetros que minimizam uma função. Dada a função abaixo:

$$\beta^{(k+1)} = \beta^{(k)} - \alpha k \lambda J(\beta^{(k)}), k = 0, 1, ...$$

A primeira parte da função $\beta^{(k+1)}$ diz que parâmetro β na iteração $k + 1$ é igual ao β na posição k menos a taxa de aprendizado λ na iteração k multiplicada pela função J aplicada no $\beta^{(k)}$. Essa iteração é realizada até que a convergência de β seja alcançada.

O *gradient boosting* utiliza a técnica *boosting* para gerar os novos modelos com o gradiente descente para corrigir o erro em cada iteração.

4) Redes Neurais Artificiais: as Redes Neurais Artificiais (RNA) são inspiradas no sistema nervoso humano, que é composto por células chamadas de neurônio. Os neurônios são conectados entre si e essa conexão se chama sinapse. O aprendizado é realizado alterando a força das conexões (sinapses) entre os neurônios. Para caso da RNA, os neurônios são unidades computacionais que recebem dados (*inputs*), realizam cálculos e geram uma saída (*output*). Os cálculos são funções definidas por pesos semelhante a sinapse, alterando o valor desse peso altera a saída [4].

Conforme Haykin (2008) [28], em 1958 o psicólogo Frank Rosenblatt cria o algoritmo *Perceptron*, que é a forma mais simples de uma rede neural. Conforme a Figura 7, o *Perceptron* possui uma ou N entrada de dados e um peso para cada entrada, um neurônio, uma função de ativação e uma saída. O neurônio calcula um peso para cada uma das entradas de dados e faz um somatório, o resultado da soma é enviado para uma função de ativação que irá decidir se o neurônio irá ativar ou não. A equação do neurônio é descrita pela formula abaixo onde o "b" corresponde ao *bias*, que é um valor aplicado após a somatória dos pesos e assim como os pesos é um valor atualizado durante o aprendizado.

$$Z = \sum_{i=1}^{m} w_i * x_i + b$$

Conforme Rashka e Mirjalili (2016) [44], a forma de aprendizado do *perceptron* de Frank Rosenblatt pode ser descrito pelos seguintes passos:

1) Inicializar os pesos e o bias com zeros ou valores próximos de zero
2) Para cada linha do conjunto de dados:
   a) Calcular o valor predito
   b) Aplicar a regra de aprendizado
   c) Atualizar os pesos

O cálculo de atualização pesos é descrito pela fórmula abaixo:

$$w_j := w_j + \Delta w_t$$

O valor Δw_t é calculado pela regra de aprendizado do *perceptron* pela fórmula:

$$\Delta w_t = \eta(y^{(i)} - \hat{y}^{(i)})x_j^{(i)}$$

Onde w_t é o peso, $y^{(i)}$ é o valor real, $\hat{y}^{(i)}$ o valor predito pelo algoritmo e η é a taxa de aprendizado. A taxa de aprendizado é um valor constante que pode variar entre 0.0 e 1.0.

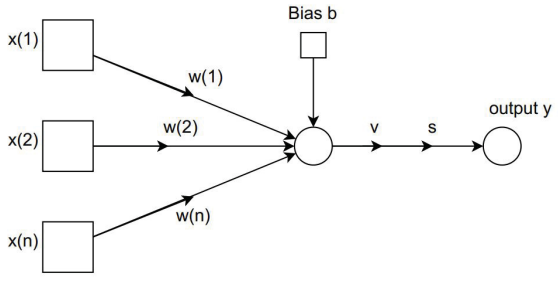


Figura 7. Arquitetura de um perceptron. Fonte: Elaborado pelo Autor.

Conforme Géron (2019) [22], a rede *perceptron* de multicamada (MLP) possui uma camada de entrada, uma ou mais

camadas ocultas e uma camada de saída. Quando uma rede possui duas ou mais camadas ocultas ela é denominada de *rede neural profunda*. O termo "oculto" da camada é devido que a rede neural não observa diretamente essas camadas, o objetivo delas são extrair as informações entre a entrada e a saída.

A *Feedforward Network* (FNN) é uma rede neural com uma ou mais camadas ocultas, a saída de uma camada pode ser a entrada da camada seguinte, conforme a Figura 8 onde a saída da Camada 1 é a entrada da Camada 2 [28].

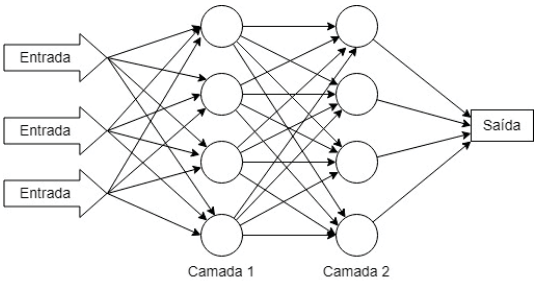


Figura 8. Exemplo de rede neural FNN. Fonte: Elaborado pelo Autor.

Conforme Haykin (2008) [60], as rede neurais recorrentes (RNN) se diferem das redes FNN por possuir pelo menos um *feedback loop*. A Figura 9 representa uma RNN desdobrada no tempo, é possível observar que a predição no tempo $t(0)$ recuperou informação do tempo $t(-1)$ e gerou informação para o tempo $t(+1)$. Essa propriedade da RNN guarda informação de um momento da predição para utilizar na sequência, enquanto aas redes FNN cada predição é independente da predição anterior.

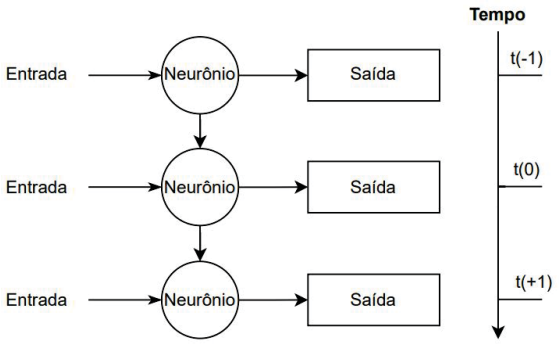


Figura 9. RNN desdobrada no tempo. Fonte: Elaborado pelo Autor.

Proposta em 1997 por Hochreiter e Schmidhuber [29], e atualizada em 2014 por Chung *et al* [12] a Long-Time Short Memory (LSTM) é arquitetura de RNN específica para lidar com dados de longo prazo. A rede contêm uma unidade especial de memória no neurônio recorrente para manter os dados de longo prazo. Cada unidade de memória contêm os *input gate* e *output gate* propostos em 1997, e o *forget gate* em 2014. Conforme a Figura 10.

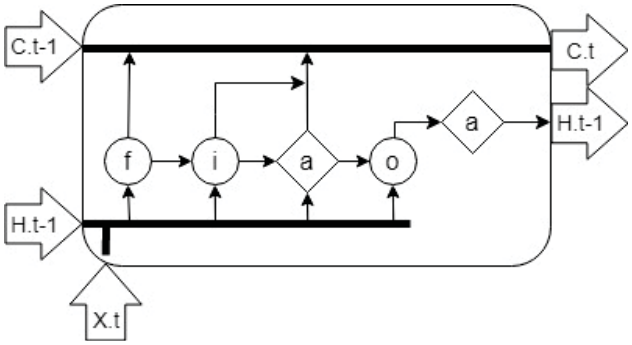


Figura 10. Neurônio de uma LSTM. Fonte: Elaborado pelo Autor.

- *i*: *Input gate*, controla quais estados devem ser removido do estado de longo prazo;
- *o*: *Output gate*, controla quais estados devem ser adicionados ao estado de longo prazo;
- *f*: *Forget gate*, controla quais estados devem ser apagados;
- *a*: Função de ativação, controla se a célula de memória e o neurônio irão ativar;
- *C.t-1*: Estado anterior da célula de memória;
- *C.t*: Estado atual da célula de memória;
- *H.t-1*: Estado oculto anterior;
- *H.t*: Estado oculto atual.

Conforme Chung *et al* (2014) [12], com a introdução dos *gates* a rede é capaz de detectar informações importantes da sequencia de dados e decidir o que manter na memória de longo prazo.

A função de ativação é uma propriedade que existem em todos os neurônios e é utilizado para limitar o resultado do somatório para um determinado alcance. A função de ativação *Rectified Linear Unit* (ReLU) é uma função contínua não diferenciável, isso é, resultados negativados serão sempre zero [28].

$$ReLU(z) = max(0, z)$$

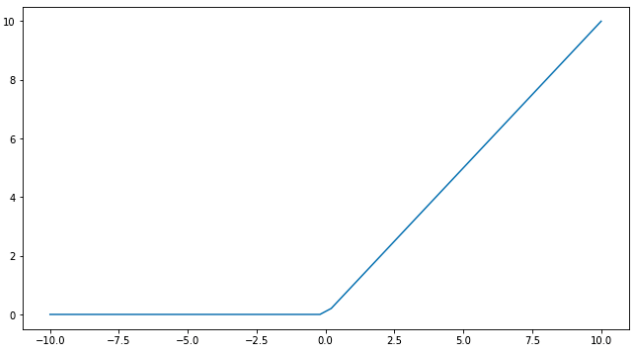


Figura 11. Função de Ativação ReLU. Fonte: Elaborado pelo Autor.

E. Feature Engineering

De acordo com Zheng e Casari (2018) [60], *feature engineering* é uma forma de extrair dados brutos e transformar em

um formato mais adequado ao modelo de *machine learning*.

Conforme Rashka e Mirjalili (2016) [44] *feature scaling* é uma técnica utilizada na etapa de pré-processamento que tem como objetivo transformar os dados de atributos para a mesma escala. Isso é necessário pois dados podem estar em escalas diferentes, por exemplo, a idade de uma pessoa pode ter alcance de um a cem, enquanto o salário anual pode trazer dados na casa de milhares. Essa discrepância pode impactar funções de otimização, que o algoritmo acaba ignorando dados em escalar menor. Existem duas técnica principais, normalização e estandardização. Normalização se limita a escalar os dados entre zero e um, enquanto a estandardização escala os dados para ficarem ao redor de zero, gerando dados positivos e negativos. A técnica de normalização mínimo e máximo que pode ser descrito pela seguinte formula.

$$x_{norm}^{(i)} = \frac{x^{(i)} - x_{min}}{x_{max} - x_{min}}$$

Onde o $x_{norm}^{(i)}$ representa um valor qualquer observado, o x_{min} o menor valor encontrado nas colunas e o x_{max} o maior valor.

F. Hiperparâmetros

Conforme Yang e Shami (2020) [59], modelos de aprendizado de máquina possuem dois tipos de parâmetros. O primeiro é um parâmetro não configurável que é inicializado e atualizado durante o aprendizado do modelo, na fase de treinamento, como por exemplo os pesos das entradas de dados de uma rede neural. O segundo tipo de parâmetro são configurações aplicadas antes de iniciar o treinamento que impactam na arquitetura do algoritmo.

Algoritmos diferentes podem possuir parâmetros diferentes e encontrar essa configuração ideal é o processo de *tuning parameters* [43]. Os hiperparâmetros impactam diretamente nos resultados do modelo e sua configuração deve ser feita cuidadosamente para que não ocorra *overfitting*, quando os parâmetros do modelo se encaixam perfeitamente nos dados de treino mas o modelo não alcança os mesmos resultados em dados não vistos anteriormente pelo modelo [51].

G. Métricas

A análise de regressão é um assunto bastante profundo e enfatizado na ciência de dados. Utilizado de diversas maneiras como previsão de casos de COVID-19 [10] e até mesmo interpretando toda a estatística como uma regressão [11]. Devido a essas diversas maneira de trabalhar com regressão, a comunidade não patronizou uma única forma de medir a performance de um modelo e várias métricas foram criadas. A raiz do erro quadrático médio, ou RMSE pode ser utilizada para detecção de *outliers* e quanto mais próximo de zero melhor o resultado. Dada a fórmula abaixo onde o Y_i representa o valore real e o $\bar{Y}_i$ o valore predito pelo modelo.

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(Y_i - \bar{Y}_i)^2}$$

Utilizando a Tabela IX como exemplo, ao aplicar o RMSE nos valores de "Fechamento" e "Predito"o resultado seria 0,3722.

Tabela IX
EXEMPLO COM 14 DIAS FECHAMENTO E PREDITO

Código Ação	Data	Fechamento	Predito
INEP3	2021-06-29	3,13	2,97
INEP3	2021-06-30	3,04	2,97
INEP3	2021-07-01	3,12	3,05
INEP3	2021-07-02	3,05	3,05
INEP3	2021-07-05	2,99	3,12
INEP3	2021-07-06	2,93	3,12
INEP3	2021-07-07	3,0	3,12
INEP3	2021-07-08	3,04	3,05
INEP3	2021-07-12	2,9	3,62
INEP3	2021-07-13	2,96	3,62
INEP3	2021-07-14	2,94	3,62
INEP3	2021-07-15	2,92	3,29
INEP3	2021-07-16	3,01	2,57
INEP3	2021-07-19	3,04	2,74

III. MATERIAIS E MÉTODOS

Essa sessão irá apresentar as etapas de coleta e transformação dos dados brutos, explicando cada fonte de dados e como foi transformada para se enquadrar no modelo supervisionado de aprendizado de máquina, como foi feito a separação entre as bases de treino, validação e teste. E também como os modelos foram criados e com quais hiperparâmetros.

A. Coleta dos dados

Para o estudo foram coletados dados de 333 diferentes ações com início em 18/05/2019 e término em 06/08/2021, fechando mais de dois anos de informações de cada ação. A escolha da data final foi feita com base na divulgação das datas dos resultados do primeiro trimestre de 2021. Conforme o relatório da Suno [53], a Rede D'or São Luiz S.A. (RDOR3) como a última companhia a divulgar os resultados no dia 17/05/2021. Por isso os experimentos de previsão iniciaram a partir do dia 19/05/2021, pois os dados de treinamento não seriam vazados para a predição.

Os dados coletados podem ser separados em três categorias diferentes: histórico da ação, setor e financeiro da companhia.

1) Histórico: foi utilizado a biblioteca yfinance (em python) [58] para extrair o histórico por dia das ações. Cada registro (linha) possui oito colunas: data, valor da abertura, valor da máxima, valor da mínima, valor de fechamento, valor de fechamento ajustado, volume total de ações negociadas e nome da ação. Todos os valores correspondem à ação e à data

O código abaixo é um exemplo de utilização para buscar os dados da ação AALR3 (Centro de Imagem Diagnósticos SA - Alliar) entre os períodos de 20/05/2019 e 24/05/2019. O resultado pode ser visto na Figura 12.

```
1. import yfinance as yf
2. import pandas as pd
3. acao = 'AALR3'
4. df_historico =  yf.download(
5.    ticker=acao,
```

```
6.     start_date='2019-05-20',
7.     end_date='2019-05-24')
8. print(df_historico)
```

Das 333 ações iniciais, 81 ações foram removidas no total, sendo 67 ações que não tinham uma quantidade de pelo menos 254 dias (um ano de dias úteis) de histórico, pois o experimento consiste em analisar ações que tenham no mínimo essa quantidade de datas de histórico, e 14 ações que tiveram algum problema na busca de dados. A remoção dessas ações pode ocorrer devido a problemas na API utilizada para extrair os dados, ou ações que entraram recentemente na B3 e não possuem o histórico necessário para participar do experimento. Os dados foram extraídos e salvos em disco no formato parquet, um arquivo para cada ação.

Date	Open	High	Low	Close	Adj Close	Volume	ticker
2019-05-20	14.7	14.89	14.54	14.77	14.65	96300	AALR3
2019-05-21	14.8	14.9	14.68	14.7	14.58	106700	AALR3
2019-05-22	14.6	14.95	14.2	14.2	14.08	177300	AALR3
2019-05-23	14.2	14.33	14.1	14.14	14.02	120500	AALR3
2019-05-24	14.18	14.32	13.95	13.99	13.88	165400	AALR3

Figura 12. Histórico de ação AALR3. Fonte: Elaborado pelo Autor.

2) Setor: são informações sobre o setor econômico, o subsetor e o segmento de companhias que estão listadas na B3. Os dados foram obtidos em formato XLSX disponibilizados pela própria B3 [7], conforme o exemplo na Figura 13.

SETOR ECONÔMICO	SUBSETOR	SEGMENTO	LISTAGEM	
			CÓDIGO	SEGMENTO
Petróleo, Gás e Biocombustíveis	Petróleo, Gás e Biocombustíveis	Exploração, Refino e Distribuição		
		3R PETROLEUM	RRRP	NM
		COSAN	CSAN	NM
		DOMMO	DMMO	
		ENAUTA PART	ENAT	NM
		PET MANGUINH	RPMG	
		PETROBRAS	PETR	N2
		PETROBRAS BR	BRDT	NM
		PETRORECSA	RECV	NM
		PETRORIO	PRIO	NM
		ULTRAPAR	UGPA	NM
		Equipamentos e Serviços		
		LUPATECH	LUPA	NM
		OCEANPACT	OPCT	NM
		OSX BRASIL	OSXB	NM

Figura 13. Classificação setorial para bens industriais. Fonte: B3.

3) Financeiro: para os dados do financeiro foi utilizada a biblioteca yahooquery (em python) [24]. Cada registro representa um trimestre de uma companhia e possui 160 colunas ao total. Com o código abaixo é possível buscar os dados trimestrais da ação AALR3. Uma prévia dos dados pode ser vista na Figura 14.

```
1. from yahooquery import Ticker
2. import pandas as pd
3. acao = 'AALR3.SA'
4. df_finance =  Ticker(
5.   acao).all_financial_data(
6.              frequency="q")
7. print(df_finance)
```

symbol	as Of Date	period Type	currency Code	Accounts Payable	Accounts Receivable	Accumulated Depreciation	Available For Sale Securities	Basic Average Shares
AALR3.SA	2020-06-30	3M	BRL	46997000	48364000	-522312000	74979000	117882000
AALR3.SA	2020-09-30	3M	BRL	72892000	73746000	-544842000	73597000	117882000
AALR3.SA	2020-12-31	3M	BRL	80425000	79679000	-570160000	71766000	
AALR3.SA	2021-03-31	3M	BRL	75151000	90239000	-593818000	69990000	118025000
AALR3.SA	2021-06-30	3M	BRL	75843000	88004000	-740754000	67819000	118025000

Figura 14. Exemplo de dados financeiros para a ação AALR3 (Centro de Imagem Diagnosticos SA - Alliar). Fonte: Elaborado pelo Autor.

Os dados foram extraídos e salvos em disco no formato parquet, um arquivo para cada companhia.

B. Preparação dos dados

Cada categoria de informação (histórico, setor e financeiro) passou por um processo de transformação para adequar os dados ao processo de modelagem.

1) Histórico: a primeira etapa do processo de transformação foi carregar todos os arquivos e os consolida, gerando uma tabela com 129.910 linhas e sete colunas. As colunas "abertura", "fechamento ajustado", "menor baixa", "maior alta" e "volume" são removidas, pois essas informações não estarão disponíveis na previsão de um dia não conhecido. Em sequencia foi aplicado a técnica de *dummy variable* para as ações, transformando cada símbolo de ação em variáveis binárias.

O próximo passo foi criar uma nova coluna que contenha o código da ação sem a sua categoria, ou seja, removendo a informação de ordinária ou preferência. Essa coluna é necessária para cruzar os dados com as fontes de setor e financeiro, que são a nível de companhia, diferente do histórico que é a nível de ação. Por último foram removidos registros que tenham a coluna "fechamento" em branco (somente nove registros tiveram esse problema) e arredondado os valores da coluna para duas casas decimais. A tabela final possui 129.901 registros e 262 colunas, das quais 252 colunas são binárias de códigos de ação e as demais correspondem à data e ao fechamento, conforme o exemplo da Figura 15.

ticker	symbol	close	date	ticker.AALR3	ticker.AAPL34	ticker.ABEV3
AALR3	AALR	14.77000046	2019-05-20	1	0	0
AALR3	AALR	14.69999981	2019-05-21	1	0	0
AALR3	AALR	14.19999981	2019-05-22	1	0	0
AALR3	AALR	14.14000034	2019-05-23	1	0	0
AALR3	AALR	13.98999977	2019-05-24	1	0	0

Figura 15. Dados de histórico após transformação. Fonte: Elaborado pelo Autor.

2) Setor: o primeiro passo foi fazer uma limpeza manual no arquivo. Como o formato era de arquivo Excel, e não dados tabulados, foi preciso remover as linhas desnecessárias do arquivo original. Após esse ajuste, foi possível processar o arquivo, agora convertido para o formato CSV (valores separados por vírgula).

SETOR	SUBSETOR	SEGMENTO	CODIGO	LISTAGEM SEGMENTO
Petróleo, Gás e Biocombustív	Petróleo, Gás e Biocombustíveis	Exploração, Refino e Distribuiç	RRRP	NM
Petróleo, Gás e Biocombustív	Petróleo, Gás e Biocombustíveis	Exploração, Refino e Distribuiç	CSAN	NM
Petróleo, Gás e Biocombustív	Petróleo, Gás e Biocombustíveis	Exploração, Refino e Distribuiç	DMMO	
Petróleo, Gás e Biocombustív	Petróleo, Gás e Biocombustíveis	Exploração, Refino e Distribuiç	ENAT	NM
Petróleo, Gás e Biocombustív	Petróleo, Gás e Biocombustíveis	Exploração, Refino e Distribuiç	RPMG	
Petróleo, Gás e Biocombustív	Petróleo, Gás e Biocombustíveis	Exploração, Refino e Distribuiç	PETR	N2
Petróleo, Gás e Biocombustív	Petróleo, Gás e Biocombustíveis	Exploração, Refino e Distribuiç	BRDT	NM
Petróleo, Gás e Biocombustív	Petróleo, Gás e Biocombustíveis	Exploração, Refino e Distribuiç	PRIO	NM
Petróleo, Gás e Biocombustív	Petróleo, Gás e Biocombustíveis	Exploração, Refino e Distribuiç	UGPA	NM
Petróleo, Gás e Biocombustív	Petróleo, Gás e Biocombustíveis	Equipamentos e Serviços	LUPA	NM
Petróleo, Gás e Biocombustív	Petróleo, Gás e Biocombustíveis	Equipamentos e Serviços	OSXB	NM

Figura 16. Dados tabulados de setor, subsetor e segmento. Fonte: Elaborado pelo Autor.

Com o arquivo tabulado, conforme a Figura 16, foi iniciado o processo de limpeza dos valores. A limpeza textual foi aplicada nas colunas "setor", "subsetor" e "segmento". Removendo caractere especial, vírgulas, e substituindo cada espaço por *underline* (_). Por exemplo, o setor "Petróleo, Gás e Biocombustíveis" virou "petroleo_gas_e_biocombustiveis", o subsetor "Serviços Financeiros Diversos" virou "servicos_financeiros_diversos", e por último, o segmento "Artefatos de Ferro e Aço" virou "artefatos_de_ferro_e_aco". Esse processo foi feito para facilitar o acesso programático às colunas e aos valores quando necessário.

Com as colunas padronizadas, foi aplicada a técnica de *dummy variable* no setor, subsetor e segmento. O setor se tornou 10 colunas, o subsetor, 42 colunas, e o segmento, 81 colunas. A tabela final tem 435 linhas e 134 colunas, sendo 133 colunas de atributos e uma coluna de companhia.

3) Financeiro: assim como no pré-processamento dos dados de histórico, a primeira etapa foi consolidar todos os arquivos gerados na extração dos dados. 314 arquivos resultaram em uma tabela de 1.479 registros e 328 colunas totais e 324 atributos. Porém, essa tabela contêm registros duplicados por companhia. Isso ocorre porque uma companhia pode ter mais de uma ação. Por exemplo, a Bardella S.A. Indústrias Mecânicas possui na bolsa as ações BDLL3 e BDLL4. Para corrigir esses dados duplicados foi criada uma coluna na qual a classe (ordinárias ou preferenciais) da ação foi removida. Com a classe removida, a nova coluna para as ações BDLL3 e BDLL4 transformou em BDLL somente e foi possível remover os dados duplicados. Em seguida, são removidas as ações que tiveram algum problema na etapa de extração do histórico, seja por falha na extração ou por poucos registros. Na sequência foi excluída qualquer informação com data posterior à 17/05/2021, a fim de não vazar informações futuras para o modelo. Com os dados preparados as colunas com mais de 5% de valores faltando foram removidas, sobrando apenas 46 colunas dos 324 atributos iniciais. A Tabela X possui a lista de atributos que foram selecionados e a seguir algumas descrições de o que cada atributo significa. Com as colunas necessárias selecionadas foi aplicada a técnica de *feature scaling* em todos os atributos.

- InvestingCashFlow: Fluxo de Caixa de Investimento;
- TotalRevenue: Rendimento Total;
- CommonStock: Ações Ordinárias;

Tabela X
ATRIBUTOS SELECIONADOS DA CATEGORIA DE FINANCEIRO

Colunas
NetIncomeFromContinuingOperations
ReconciledDepreciation
ChangeInCashSupplementalAsReported
ChangeInWorkingCapital
InvestingCashFlow
BeginningCashPosition
FinancingCashFlow
EndCashPosition
OperatingCashFlow
LongTermDebtAndCapitalLeaseObligation
ChangesInCash
FreeCashFlow
SellingGeneralAndAdministration
TotalDebt
TaxProvision
NetPPE
Payables
NetInterestIncome
CommonStock
CapitalStock
CashAndCashEquivalents
InvestedCapital
TotalCapitalization
NetIncomeFromContinuingAndDiscontinuedOperation
NetIncome
NetIncomeCommonStockholders
TaxRateForCalcs
TaxEffectOfUnusualItems
TotalRevenue
NetIncomeContinuousOperations
PretaxIncome
OrdinarySharesNumber
OperatingRevenue
NetIncomeFromContinuingOperationNetMinorityInterest
NetIncomeIncludingNoncontrollingInterests
NormalizedIncome
DilutedNIAvailtoComStockholders
ShareIssued
NetTangibleAssets
TotalEquityGrossMinorityInterest
TotalAssets
TangibleBookValue
CommonStockEquity
TotalLiabilitiesNetMinorityInterest
StockholdersEquity

- NetIncomeFromContinuingOperations: Lucro Líquido de Operações Continuadas;
- TotalDebt: Dívida Total.

4) Consolidação: as três tabelas foram carregadas, e a união começa pelo histórico e setor. Os dados históricos possuem 129.901 linhas e sete colunas, enquanto os dados de setor somente 435 linhas e 134 colunas. A junção das tabelas é realizada na coluna *symbol*, resultando uma nova tabela com 105.396 linhas e 396 colunas. Essa redução de linhas ocorreu por que enquanto os dados de histórico possuíam 205 companhias, os dados de setor não contêm as informações de todas companhias, somente de 153. Ao analisar essas companhias é possível identificar que são empresas internacionais, como Apple e Nike. Em sequência foi feita uma nova junção de tabela com os dados financeiros, novamente com as colunas que representam a companhia e não a ação. A base final possui 46.991 registros e 575 colunas, 90 ações de 68 companhias. Os

registros removidos foram referentes as ações que possuíam poucos dados de histórico.

C. Configuração

A etapas de extração e processamento foram desenvolvidas na linguagem Python (v3.7.9) [42] devido a ampla opções de bibliotecas para raspagem e manipulação de dados. As principais bibliotecas utilizada foram:

- yahooquery (v2.2.15) [38]: Extração de dados da bolsa de valores;
- pandas (v1.2.2) [38]: Manipulação de dados tabulados;
- numpy (v1.19.4) [37]: Manipulação de dados tabulados e criação de arrays;
- pyarrow (v5.0.0) [41]: Exportação de dados um formato parquet;
- seaborn (v0.1.11) [50]: Criação de gráficos.

O projeto foi versionado através do Github e está disponível no repositório através do *link https://github.com/brunno-oliveira/b3-invest*. Foi desenvolvido em uma máquina local com CPU Intel Core i7-4770K com 16GB de memória Ram e sistema operacional Windows 10.

D. Conjunto de treino, validação e teste

A separação dos dados entre treino e teste foi feito com base nas datas, dividindo em duas etapas. A primeira etapa de validação e busca por hiperparâmetros, utilizou os dados de 18/05/2019 a 18/05/2021 para o treino e 19/05/2021 a 28/06/2021 para validação. A etapa de teste utilizou os dados de 18/05/2021 a 28/06/2021 para o treino, ou seja, buscou todos os dados da etapa de validação. E para os teste foram utilizados as datas de 29/06/2021 até 06/08/2021. Ambas etapas possuem 28 dias para validação e teste. A separação pode ser observada na Figura 17.

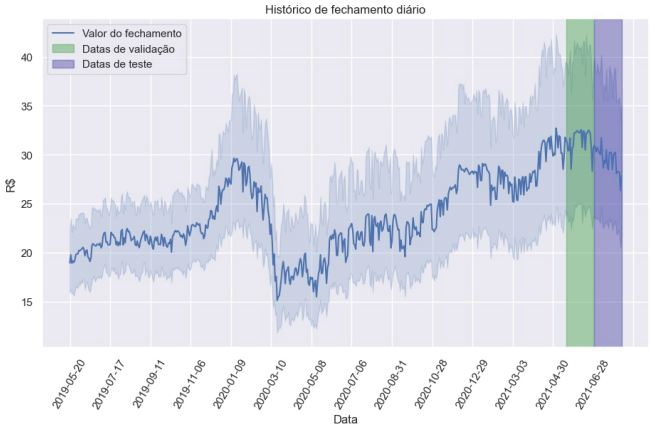


Figura 17. Dados de validação e teste. Fonte: Elaborado pelo Autor.

E. Treinamento dos modelos

As três etapas de execução dos modelos foram realizados no ambiente de nuvem Google Cloud [23], busca por hiperparâmetros, treinamento e predição. Foi utilizado dois ambientes diferentes para isso, os modelos de árvore de decisão, floresta aleatória e gradient boosting foram treinados em uma *Compute Engine* com oito *vCPU* e 32 GB de memória RAM. Enquanto o modelo LSTM foi treinado em uma máquina em uma *Compute Engine* com quatro *vCPU* e 16 GB de memória RAM e uma placa de vídeo *NVIDIA Tesla K80*.

Assim como na etapa anterior de extração e transformação, os modelos foram desenvolvidos na linguagem Python. A principal motivação para isso foi a facilidade para encontrar bibliotecas de *machine-learning*, das quais foram utilizadas:

- scikit-learn (v0.24.2) [48]: Modelos de árvore de decisão e floresta aleatória; Normalização mínimo e máximo e métricas;
- xgboost (v1.4.2) [57]: Modelo de *gradient boosting*;
- keras (v2.6.0) [30]: Modelo de LSTM;
- tensorflow (v2.6.0) [54]: *Framework* para executar o keras.

F. Hiperparâmetros

Para uma melhor avaliação dos modelos foi adotada a técnica de busca por hiperpârametros, com exceção da rede LSTM que foi desenvolvida por experimentações.

1) Árvore de decisão: para a árvore de decisão foram selecionados três parâmetros para otimização:

- *splitter*: A estratégia para separara cada nó da árvore.
- min_samples_split: A quantidade mínima de dados para quebrar em um novo nó.
- min_samples_leaf: A quantidade mínima de dados para um nó de decisão.

A Tabela XI representa todas as possibilidades testadas para ambos os modelos. A diferença dos hiperpârametros entre os modelos ficou por conta do *splitter*, que no modelo sem os atributos ficou configurado como *best* conforme a Tabela XII, enquanto no modelo com os novos atributos ficou com o *splitter* configurado como *random* conforme a Tabela XIII.

Tabela XI
PARÂMETROS PERCORRIDOS PARA A ÁRVORE DE DECISÃO

Parâmetro	Opções Testadas
splitter	best, random
min_samples_split	2, 4, 6, 8
min_samples_leaf	1, 2, 3, 4, 5, 6, 7, 8, 9, 10

Tabela XII
PARÂMETROS DEFINIDOS PARA A ÁRVORE DE DECISÃO SEM OS NOVOS ATRIBUTOS

Parâmetro	Opção Selecionada
splitter	best
min_samples_split	2
min_samples_leaf	1

Tabela XIII
PARÂMETROS DEFINIDOS PARA A ÁRVORE DE DECISÃO COM OS NOVOS ATRIBUTOS

Parâmetro	Opção Selecionada
splitter	random
min_samples_split	2
min_samples_leaf	1

2) Floresta aleatória: assim como a árvore de decisão a floresta aleatória também contou com três parâmetros para otimização. Além dos parâmetros *min_samples_split* e *min_samples_leaf* já descritos, a floresta aleatória conta com o *n_estimators* que corresponde ao número de árvores na floresta.

A Tabela XIX representa todas as possibilidades testadas para ambos os modelos. A diferença de otimização ficou no parâmetro *n_estimators*, que para o modelo sem os novos atributos ficou em 125, conforme a Tabela XV, e para o modelo com os novos atributos em 1000, conforme a Tabela XVI.

Tabela XIV
PARÂMETROS PERCORRIDOS PARA A FLORESTA ALEATÓRIA

Parâmetro	Opções Testadas
n_estimators	50, 100, 125, 150, 500, 1000, 1500
min_samples_split	2, 4, 6, 8
min_samples_leaf	1, 2, 3, 4

Tabela XV
PARÂMETROS DEFINIDOS PARA A FLORESTA ALEATÓRIA SEM OS NOVOS ATRIBUTOS

Parâmetro	Opção Selecionada
n_estimators	125
min_samples_split	2
min_samples_leaf	1

Tabela XVI
PARÂMETROS DEFINIDOS PARA A FLORESTA ALEATÓRIA COM OS NOVOS ATRIBUTOS

Parâmetro	Opção Selecionada
n_estimators	1000
min_samples_split	2
min_samples_leaf	1

3) LSTM: para o modelo de LSTM foi realizado experimentos manuais para chegar no melhor resultado. As duas arquiteturas com e sem os novos atributos ficaram iguais, conforme a Figura 18. A rede possui duas camadas escondidas de LSTM e duas camadas de *dropout*, além da entrada e saída.

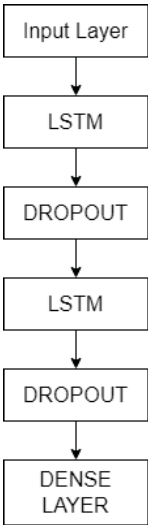


Figura 18. Arquitetura das redes LSTM. Fonte: Elaborado pelo Autor.

4) Gradient Boosting: para o gradient boosting foram selecionados três parâmetros para otimização:

- n_estimator: Semelhante ao parâmetro da floresta aleatória, mas aqui também é equivalente a quantidade de rodadas do boosting;
- learning_rate: A taxa de aprendizado do boosting;
- booster: Algoritmo do utilizado pelo booster.

A Tabela XVII representa todas as possibilidades testadas para ambos os modelos. A diferença dos hiperpârametros entre os modelos ficou por conta do *booster*, que no modelo sem os novos atributos ficou configurado como *dart* conforme a Tabela XVIII, enquanto no modelo com os novos atributos ficou com o *booster* configurado como *gbtree* conforme a Tabela XIX.

Tabela XVII
PARÂMETROS PERCORRIDOS PARA O GRADIENT BOOSTING

Parâmetro	Opções Testadas
n_estimators	50, 100, 125, 150, 500, 1000, 1500, 2000, 2500, 5000, 10000, 15000
learning_rate	0,01, 0,2, 0,5, 1
booster	gbtree, gblinear, dart

Tabela XVIII
PARÂMETROS DEFINIDOS PARA O GRADIENT BOOSTING SEM OS NOVOS ATRIBUTOS

Parâmetro	Opção Selecionada
n_estimators	10000
learning_rate	1
booster	gbtree

Tabela XIX
PARÂMETROS DEFINIDOS PARA O GRADIENT BOOSTING COM OS NOVOS ATRIBUTOS

Parâmetro	Opção Selecionada
n_estimators	10000
learning_rate	1
booster	gbtree

IV. RESULTADOS E DISCUSSÕES

Nessa sessão são discutidos os resultados obtidos através das metodologias utilizadas. Foram utilizados 4 experimentos com quantidade de dias diferentes e com modelos diferentes para verificar se ocorreu uma melhoria ou não do modelo.

Os experimentos foram realizados para 1, 7, 14 e 28 dias com os modelos de árvore de decisão, floresta aleatória, LSTM e *gradient boosting*. Foi utilizada a métrica RMSE para comparar os modelos com e sem atributos financeiros, a métrica indica que quanto menor, melhor o modelo se comportou. As tabelas de resultados desse capítulo como a Tabela XX, possuem 4 colunas:

- Modelo: Identifica o modelo;
- RMSE Sem Atributos Financeiros: A métrica RMSE para o modelo sem atributos financeiros;
- RMSE Com Atributos Financeiros: A métrica RMSE para o modelo com atributos financeiros;
- Desempenho Com Atributos Financeiros: A razão entre a métrica com atributos financeiros e sem atributos financeiros. O resultado positivo indica que o modelo com atributos financeiros foi melhor e quanto foi melhor, o resultado negativo o modelo sem atributos financeiros foi melhor e quanto foi melhor.

Conforme a Tabela XX que contêm os experimentos para 1 dia de previsão, os experimentos com os novos atributos tiveram melhor desempenho nos modelo de floresta aleatória, com uma melhoria de **1,19%** e LSTM, com um aumento de **12,58%** em relação ao modelo sem os atributos. Os de modelos árvore de decisão e *gradient boosting* tiverem pior desempenho com atributos, especialmente a árvore de decisão que o modelo ficou **15,53%** pior em relação ao sem atributos.

Tabela XX
RESULTADOS COM UM DIA DE PREDIÇÃO (VALORES EM RMSE)

Modelo	RMSE **Sem** Atributos Financeiros	RMSE **Com** Atributos Financeiros	Desempenho Com Atributos Financeiros
Árvore de Decisão	1,0546	1,2198	-15,53%
Floresta Aleatória	1,1947	1,1981	1,19%
LSTM	22,3005	19,4946	12,58%
Gradient Boosting	1,0906	1,0992	-0,78%

Ao analisar os resultados buscando pelo melhor e pior desempenho individual, um empate com quatro ações com a previsão exata, sendo duas delas para a ação EUCA3 (Eucatex) com o modelo de árvore de decisão observado na Tabela XXI. O pior resultado, sendo a ação CEBR3 (Companhia Energética de Brasília) utilizando a rede LSTM e sem os novos atributos na Tabela XXII.

Tabela XXI
MELHOR RESULTADO INDIVIDUAL COM UM DIA DE PREDIÇÃO

Modelo	Atributos	Ação	Data da predição	Real	Predito	RMSE
Árvore de Decisão	Com Atributos	EUCA3	2021-06-29	21,50	21,50	0,00
Árvore de Decisão	Sem Atributos	EUCA3	2021-06-29	21,50	21,50	0,00
Floresta Aleatória	Sem Atributos	ALUP3	2021-06-29	8,71	8,71	0,00
LSTM	Sem Atributos	CAML3	2021-06-29	9,66	9,66	0,00

Tabela XXII
PIOR RESULTADO INDIVIDUAL COM UM DIA DE PREDIÇÃO

Modelo	Atributos	Ação	Data da predição	Real	Predito	RMSE
LSTM	Sem Atributos	CEBR3	2021-06-29	185,98	78,02	107,96

O resultado dos experimentos para sete dias da Tabela XXIII demonstram que todos os modelos foram melhor com os novos atributos. O maior ganho ocorreu na floresta aleatória, uma melhoria de **33,59%** e o menor ganho ficou no LSTM com **9,4%**, ressaltando que todos os modelos tiveram mais de **9,0%** de melhoria.

Tabela XXIII
RESULTADOS COM SETE DIAS DE PREDIÇÃO (VALORES EM RMSE)

Modelo	RMSE **Sem** Atributos Financeiros	RMSE **Com** Atributos Financeiros	Desempenho Com Atributos Financeiros
Árvore de Decisão	3,3194	2,8271	14,86%
Floresta Aleatória	3,4578	2,5884	33,59%
LSTM	21,5112	19,4895	9,40%
Gradient Boosting	3,9658	2,9816	33,01%

A melhor previsão para 7 dias ficou com a ação DMMO3 (Dommo Energia SA) para o modelo de árvore de decisão com os novos atributos. O pior resultado ficou novamente para a rede LSTM sem os novos atributos para ação CEBR3. O melhor resultado possui o RMSE de **0,0239**, enquanto o pior resultado obteve **101,5869** de RMSE. As Figuras 17 e 18 demonstram a linha do tempo de previsão para as respectivas ações.

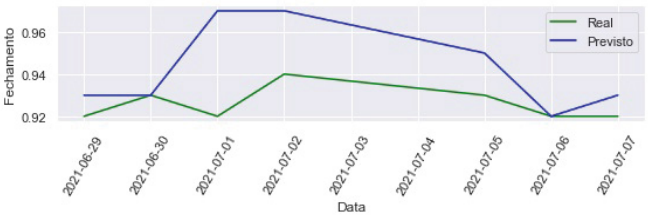


Figura 19. Histórico de fechamento diário com 7 dias para a ação DMMO3. Fonte: Elaborado pelo Autor.

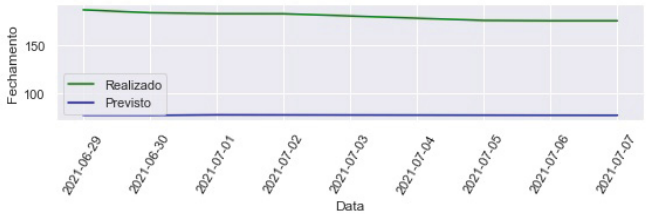


Figura 20. Histórico de fechamento diário com 7 dias a para a ação CEBR3. Fonte: Elaborado pelo Autor.

Assim como no experimento anterior, para 14 dias de previsão todos os modelos com os novos atributos tiveram melhor desempenho conforme a Tabela XXIV. O maior foi novamente com a floresta aleatória, chegando a **26,58%** de melhoria. A árvore de decisão também teve uma melhoria significativa, chegando a **16,98%**. O menor ganho ficou com LSTM novamente, com **6,18%** de melhoria em relação ao modelo sem os novos atributos.

Tabela XXIV
RESULTADOS COM 14 DIAS DE PREDIÇÃO (VALORES EM RMSE)

Modelo	RMSE **Sem** Atributos Financeiros	RMSE **Com** Atributos Financeiros	Desempenho Com Atributos Financeiros
Árvore de Decisão	3,8373	3,1858	16,98%
Floresta Aleatória	3,8802	3,0654	26,58%
LSTM	20,7924	19,5068	6,18%
Gradient Boosting	4,2866	3,5483	20,81%

b A melhor melhoria foi com a ação DMMO3, a pior com CEBR3. Conforme as respectivas Figuras 19 e 20.

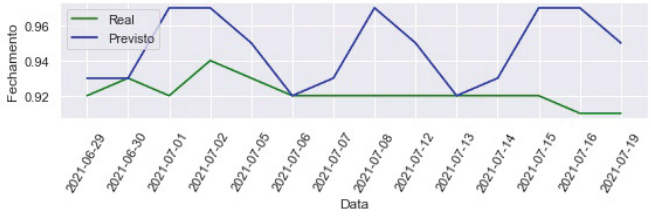


Figura 21. Histórico de fechamento diário com 14 dias para a ação DMMO3. Fonte: Elaborado pelo Autor.

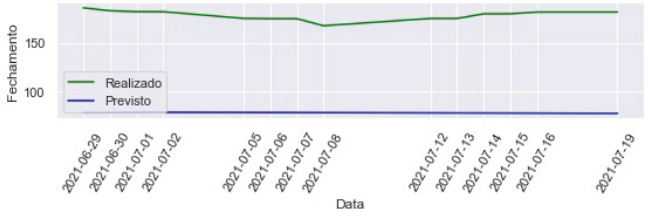


Figura 22. Histórico de fechamento diário com 14 dias a para a ação CEBR3. Fonte: Elaborado pelo Autor.

Conforme a Tabela XXV, o experimento com a maior quantidade de dias não ficou muito diferente dos anteriores 7 e 14 dias. Todos os modelos obtiveram melhores resultados com os novos atributos e a floresta aleatória ainda permanece com o melhor ganho com **21,48%**, enquanto LSTM fica na pior posição com somente **2,2%**

Tabela XXV
RESULTADOS COM 28 DIAS DE PREDIÇÃO (VALORES EM RMSE)

Modelo	RMSE **Sem** Atributos Financeiros	RMSE **Com** Atributos Financeiros	Desempenho Com Atributos Financeiros
Árvore de Decisão	4,0392	3,5072	13,17%
Floresta Aleatória	4,0958	3,3716	21,48%
LSTM	20,1301	19,6881	2,20%
Gradient Boosting	4,5315	4,2883	5,57%

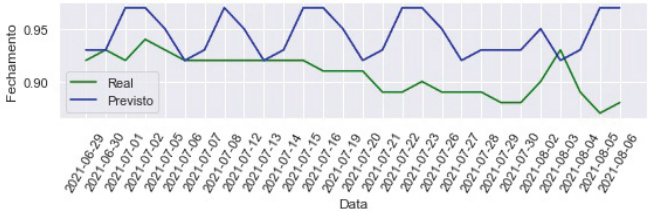


Figura 23. Histórico de fechamento diário com 28 dias para a ação DMMO3. Fonte: Elaborado pelo Autor.

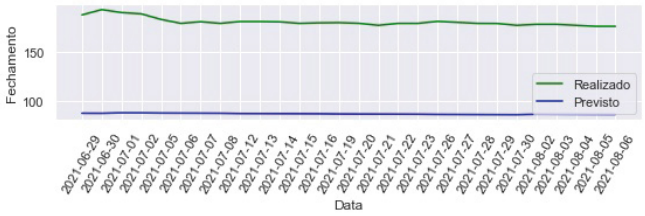


Figura 24. Histórico de fechamento diário com 28 dias a para a ação CEBR3. Fonte: Elaborado pelo Autor.

É possível identificar que a qualidade do modelo degrada conforme mais dias são inseridos na previsão, conforme a Tabela XXVI. Selecionando o melhor modelo de cada experimento, de 1 dia para 7 dias a queda do modelo é de **145%**, de 1 dia para 14 dias **190%** e de 1 dia para 28 dias é **219%**.

Tabela XXVI
DEGRADAÇÃO DOS MELHORES MODELOS DE CADA EXPERIMENTO

Experimento	RMSE	Degradação
1 dia	1,0546	0%
7 dias	2,5884	145%
14 dias	3,0654	190%
28 dias	3,3716	219%

V. CONSIDERAÇÕES FINAIS

Conforme os resultados apresentados no capítulo anterior foi confirmado que é possível melhorar modelos de previsão de bolsa de valores com atributos da própria companhia. Para trabalhos futuros pode ser criado uma nova forma de

avaliação, ao invés de utilizar uma métrica estatística simular uma carteira de investimentos. Existem oportunidades com *feature engineering* para serem desenvolvidas, por exemplo:

- Informações macroeconômicas como o desempenho do setor de atuação da companhia;
- Índices econômicos como IPCA e IGP-M;
- Cotações de moedas como o dólar;
- Variáveis de *lag* como a cotação da semana anterior.

Outro ponto a ser testado é se um único modelo para todas as ações é a melhor abordagem. Podendo testar um modelo para cada ação ou mesmo aplicar um algoritmo de clusterização (aprendizado não-supervisionado) e criar modelos por *cluster*.

Uma outra forma de validação do modelo que seria mais próxima da realidade é através de performance de carteira. Por exemplo, dado X reais na ação Y no dia Z, após 28 dias qual modelo previu corretamente o valor da carteira no último dia? Com essa abordagem é possível identificar também quais são as melhores oportunidades de negócio.

REFERÊNCIAS

[1] AGGARWAL, C. C. *Data Mining - The Textobook*. Londres: Springer, 2015.

[2] AGUIAR, João. *O Capital Financeiro Internacional e o Mercado de Ações: breve histórico*. Research Gate, 2010.

[3] ASSAF, A. N. *Mercado Financeiro*. 2. ed. São Paulo: Grupo GEN, 2018.

[4] BRAMER, M. *Principles of Data Mining*. Londres: Springer, 2016.

[5] BREIMAN, L. *Random Forests*. *Kluwer Academic Publishers*, 2001.

[6] BODIE, Z.; KANE, A.; MARCUS, A. *Investimentos*. Porto Alegre: Bookman, 2015.

[7] B3 - CLASSIFICAÇÃO SETORIAL. Disponível em: <http://www.b3.com.br/pt_br/produtos-e-servicos/negociacao/renda-variavel/acoes/consultas/classificacao-setorial/>. Acesso em: 24 Ago 2021.

[8] B3 - Perfil Pessoa física. Disponível em:«http://www.b3.com.br/pt_br/market-data-e-indices/servicos-de-dados/market-data/consultas/mercado-a-vista/perfil-pessoas-fisicas/perfil-pessoa-fisica/>. Acesso em: 28 Ago. 2021.

[9] CODLING, E. A.; PLANK, M. J.; BENHAMOU, S. *Random walk models in biology*. Journal of The Royal Society, 2008.

[10] CHAN S.; CHU J.; ZHANG Y.L NADARAJAH S. *Count regression models for COVID-19. Physica A: Statistical Mechanics and its Applications*. PeerJ, 2021.

[11] CHICCO, D.; WARRENS, M. J.; JURMAN, G.*The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation*. PeerJ, 2021.

[12] CHUNG, J.; GULCEHRE, C.; CHO, K.; BENGIO, Y.*Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling*. arXiv.org, 2014.

[13] CVM. Resolução CVM Nº 24, de 5 de março de 2021, com as alterações introduzidas pela resolução CVM nº 40/21.

[14] CVM, Comissão de Valores Mobiliários. *TOP - Mercado de Valores Mobiliários Brasileiro*. Rio de Janeiro: Comissão de Valores Mobiliários, 2019.

[15] EDWARDS. R. D.; MAGEE, J. *Technical Analysis of Stock Trends*. Boca Raton: CRC Press, 2007.

[16] FACINI, L. *Série Provas & Concursos - Mercado de Valores Mobiliários*. São Paulo: Grupo GEN, 2015.

[17] FORTUNA, E. *Mercado Financeiro*. Rio de Janeiro: Quality Mark Ltda., 2015.

[18] FREUND, Y.; SCHAPIRE, R. E. *A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting*. ScienceDirect, 1997.

[19] FREUND, Y.; SCHAPIRE, R. E. *A Short Introduction to Boosting*. Journal of Japanese Society for Artificial Intelligence, 1999.

[20] FRIEDMAN, J. *Greedy Funcion Approximation: A Gradient Boosting Machine*. Stanford, 2001.

[21] FRIEDMAN, J. Stochastic Gradient Boosting. North Ryde, 1999.

[22] GÉRON, A. *Mãos a Obra: Aprendizado de Máquina com Scikit-Learn & Tensoflow*. Rio de Janeiro: Alta Books, 2019.

[23] GOOGLE CLOUD. *Serviços de computação em nuvem* Disponível em: «https://cloud.google.com». Acesso em: 19 Out 2021.

[24] GUTHRIE, D. *Yahooquery*. Disponível em: <https://github.com/dpguthrie/yahooquery». Acesso em: 24 Ago. 2021.

[25] G1. *Poupança perde para a inflação em 2020 e tem pior rentabilidade em 18 anos*. Disponível em: <https://g1.globo.com/economia/noticia/2021/01/12/poupanca-perde-para-a-inflacao-em-2020-tem-pior-rentabilidade-em-18-anos.ghtml>. Acesso em: 28 Ago 2021.

[26] HARRINGTON, P.*Machine Learning in Action*. Shelter Island: Manning, 2012.

[27] HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. *The Elements of Statistical Learning - Data Mining, Inference, and Prediction*. Stanford: Springer, 2009.

[28] HAYKIN, S. *Neural Networks and Learning Machines*. New Jersey: Pearson Education, 2008.

[29] HOCHREITER, S.; SCHMIDHUBER, J. *Long Short-term Memory. Neural computation*. Research Gate, 1997.

[30] KERAS. *Keras: the Python deep learning API* Disponível em: <https://keras.io/>. Acesso em: 19 Out 2021.

[31] KOBORI, J. *Análise Fundamentalista - Como obter uma performance superior e consistente no mercado de ações*. Rio de Janeiro: Alta Books, 2019.

[32] LIMA, M. L. *Um Modelo para Predição de Bolsa de Valores Baseado em Mineração de Opinião*. Dissertação de Mestrado (Programa de Pós-Graduação em Engenharia de Eletricidade) — Universidade Federal do Maranhão, São Luís, 2016.

[33] MALKIEL B. G. *A Random Walk Down Wall Street* 11. ed. Princeton: W. W. Norton & Company, 2014.

[34] MITCHEL, T. M. *Machine Learning*. Portland: From Book News, Inc, 1997.

[35] NATEKIN, A,; KNOLL, A. Gradient boosting machines, a tutorial. Munique, 2013.

[36] NUBANK. *Comunicado ao mercado de modificação da oferta, do cronograma e abertura de prazo para desistência no âmbito da oferta pública de distribuição primária de ações ordinárias classe a, inclusive sob a forma de certificados de depósito de ações, de emissão da NU HOLDING LTD*. Disponível em: <https://blog.nubank.com.br/wp-content/uploads/2021/11/Projeto-Meteor-Comunicado-ao-Mercado-Modificac%CC%A7a%CC%83o-da-Oferta-30.11.2....pdf>. Acesso em: 02 Dez 2021.

[37] NUMPY. *NumPy* Disponível em: <https://numpy.org/>. Acesso em: 19 Out 2021.

[38] PANDAS. *pandas - Python Data Analysis Library* Disponível em: <https://pandas.pydata.org/>. Acesso em: 19 Out 2021.

[39] PENTEADO, M. A. B. *Uma avaliação estatística da análise gráfica no mercado de ações brasileiro à luz da teoria dos mercados eficientes e das finanças comportamentais*. Dissertação de Mestrado (Programa de Pós-Graduação em Engenharia de Eletricidade) — Universidade de São Paulo, São Paulo, 2003.

[40] PESENTE, R. *Mercados Financeiros*. Salvador, 2019.

[41] PYARROW. *Installing PyArrow* Disponível em: <https://arrow.apache.org/docs/python/install.html>. Acesso em: 19 Out 2021.

[42] PYTHON. *Welcome to Python.org* Disponível em: <https://www.python.org/>. Acesso em: 19 Out 2021.

[43] PROBST, P.; WRIGHT, M.; BOULESTEIX, A. L. Hyperparameters and Tuning Strategies for Random Forest. *Department of Electrical and Computer Engineering*, 2019.

[44] RASCHKA, S.; MIRJALILI, V. *Python Machine Learning*. Birmingham: Packt, 2016.

[45] ROKACH, L.; MAIMON, O;. *Data Mining and Knowledge Discovery Handbook*. Berlim: Springer, 2005.

[46] SANCHEZ, C. L. *Mercado Financeiro Brasileiro*. São Paulo: Grupo GEN, 2015.

[47] SCHAPIRE, R. E. *The Strength of Weak Learnability*. Boston: Springer, 1990.

[48] SCIKIT-LEARN. *scikit-learn: machine learning in Python* Disponível em: <https://scikit-learn.org/>. Acesso em: 19 Out 2021.

[49] SELAN, B. *Mercado Financeiro*. Rio de Janeiro: Estácio, 2015.

[50] SEABORN. *seaborn: statistical data visualization* Disponível em: <https://seaborn.pydata.org>. Acesso em: 19 Out 2021.
[51] SHALEV-SHWARDS, S.; BEN-DAVID, S. *Understanding Machine Learning - From Theory to Algorithms*. Nova Iorque: Cambridge University Press, 2014.
[52] SILVÉRIO, B. *O mercado financeiro brasileiro: foco nos financiamentos a exportação das linhas BNDS-EXIM*. Monografia de graduação (Curso de Comércio Exterior do Centro de Ciências Sociais Aplicadas) — Universidade do Vale de Itajaí, Itajaí, 2008. 015.
[53] SUNO. *Agenda de resultados do 1T 2021 (1T21) das empresas da B*. Disponível em: <https://www.suno.com.br/artigos/agenda-de-resultados-1t-2021-1t21>. Acesso em: 24 Aug. 2021.
[54] TENSORFLOW. *Tensorflow* Disponível em: <https://www.tensorflow.org>. Acesso em: 19 Out 2021.
[55] VARGAS, E. F. *Machine Learning para Cientista de Dados*. Disponível em: <http://cursos.leg.ufpr.br/ML4all/>. Acesso em: 3 Fev 2022.
[56] VOGLINO, E. *Como as Corretoras com Taxa Zero Ganham Dinheiro?* Disponível em: <https://comoinvestir.thecap.com.br/como-as-corretoras-com-taxa-zero-ganham-dinheiro/>. Acesso em: 28 Ago 2021.
[57] XGBOOST. *XGBoost Documentation* Disponível em: <https://xgboost.readthedocs.io/>. Acesso em: 19 Out 2021.
[58] YFINANCE. Disponível em: <https://github.com/ranaroussi/yfinance>. Acesso em: 24 Ago 2021.
[59] YANG, L.; SHAMI, A. On Hyperparameter Optimization of Machine Learning Algorithms. *Department of Electrical and Computer Engineering*, 2020.
[60] ZHENG, A.; CASARI, A. *Feature Engineering For Machine Learning*. Sebastopol: O'Reilly Media, 2018.