MÉTODO DE APRENDIZAGEM DE MÁQUINA VISANDO PREVER A DIREÇÃO DE RETORNOS DE EXCHANGE TRADED FUNDS (ETFS) COM UTILIZAÇÃO DE MODELOS DE CLASSIFICAÇÃO E REGRESSÃO[1]

Raphael Paulo Beal Piovezan [2]

RESUMO

Este artigo tem como objetivo propor e aplicar um método de aprendizagem de máquinas visando analisar a direção de retornos de Fundos Negociados em Bolsa (*Exchange Traded Funds* - ETFs) utilizando os dados históricos de retorno de suas componentes auxiliando na tomada de decisões de estratégias de investimento por meio de algoritmo de negociações. Em termos metodológicos foram aplicados modelos de regressão e classificação, utilizando conjuntos de dados padrão de mercados brasileiro e americano, além de métricas de erro algorítmicas. Em termos de resultados de pesquisa, foram analisados e comparados aos da previsão Naïve e aos retornos obtidos pela técnica de *buy & hold* no mesmo período de tempo. Em termos de risco e retorno, os modelos apresentaram desempenho superior às métricas de controle em sua maioria, com destaque ao modelo de regressão linear e aos modelos de classificação por regressão logística, máquina de vetores de suporte (utilizando o modelo LinearSVC), *Gaussian Naive Bayes* e K-ésimos Vizinhos Mais Próximos, onde em determinados conjuntos de dados os retornos superaram em duas vezes e o índice de Sharpe em até quatro vezes os do modelo de controle do *buy & hold.*

Palavras-chave: Análise de Ativos. Mercado Financeiro. Inteligência Artificial.

ABSTRACT

This article aims to propose and apply a machine learning method to analyze the direction of returns from Exchange Traded Funds (ETFs) using the historical return data of its components, helping to make investment strategy decisions through a trading algorithm. In methodological terms, regression and classification models were applied, using standard datasets from Brazilian and American markets, in addition to algorithmic error metrics. In terms of research results, they were analyzed and compared to those of the Naïve forecast and the returns obtained by the buy & hold technique in the same period of time. In terms of risk and return, the models mostly performed better than the control metrics, with emphasis on the linear regression model and the classification models by logistic regression, support vector machine (using the LinearSVC model), Gaussian Naive Bayes and K-Nearest Neighbours, where in certain datasets the returns exceeded by two times and the Sharpe ratio by up to four times those of the buy & hold control model.

Keywords: Securities Analysis. Financial Market. Artificial Intelligence.

1. INTRODUÇÃO

Nos anos 1990 o mercado financeiro nacional possuía participação majoritária do público de alta renda. Não obstante com o passar dos anos o assunto passou a adentrar o

[1] Trabalho de Conclusão de Curso apresentado como requisito parcial para obtenção do grau de bacharel no Curso de Ciência e Tecnologia, Centro Tecnológico de Joinville (CTJ), Universidade Federal de Santa Catarina (UFSC), sob orientação do Dr. Pedro Paulo de Andrade Junior.

[2] Graduando como Bacharel em Ciência e Tecnologia - raphaelpbp@hotmail.com

cotidiano das demais classes. Com o processo de globalização, democratização da bolsa de valores, criação de instituições financeiras e grande queda de juros nacional, barreiras para pequenos investidores adentrarem no mercado foram quebradas, permitindo um crescimento do público participante (COMISSÃO DE VALORES MOBILIÁRIOS, 2021; B3 BOLSA BRASIL BALCÃO, 2021). Neste sentido, os rendimentos da renda fixa passaram a ser insuficientes para muitos investidores, os quais buscaram alocação de investimentos em áreas diversificadas, incentivando a criação de novos produtos, métodos e regulamentações. O crescimento de investidores na bolsa de valores é notório, passando de aproximadamente 500 mil investidores ao final de 2011, para mais de 3,2 milhões de investidores ao final do primeiro semestre de 2021 (B3 BOLSA BRASIL BALCÃO, 2021), impulsionando inúmeras discussões e estudos sobre o tema por despertar maior interesse perante a população.

O mercado de capitais brasileiro é um componente importante do Sistema Financeiro Nacional (SFN), onde são negociados títulos e valores mobiliários emitidos por empresas que buscam se capitalizar e regulados pela Comissão de Valores Mobiliários (CVM) (COMISSÃO DE VALORES MOBILIÁRIOS, 2021).

Em virtude disso, a análise de ativos e a modelagem da precificação de ativos nos mercados financeiros é um dos assuntos mais desafiadores nos dias atuais. Envolvendo a Inteligência Artificial (IA), sendo essa uma área de pesquisa ainda em seus passos iniciais, faz-se necessária a realização de experimentos e testes para avaliar a confiabilidade de tais métodos computacionais para a precificação e previsão desses ativos financeiros. Esses tipos de modelos, quando implementados, testados e comprovados, podem resultar em uma ferramenta para a gestão de ativos, podendo ser aplicada tanto por pessoas comuns quanto por grandes gestoras de capitais visando gerar mais renda e também proteção de portfólio (DAVENPORT; BEAN, 2021; NATARAJAN, 2021).

Estima-se que entre 60-73% das negociações em bolsas de valores americanas já sejam automatizadas (MORDOR INTELLIGENCE, 2021). Tais automatizações fazem uso de parâmetros pré estabelecidos em código programado para identificar padrões baseados em diversos elementos de análise, como: preços históricos, médias móveis, indicadores de tendência, padrões de candles, volatilidade, volume e para algoritmos mais avançados: notícias e análises de sentimento do ambiente relacionado a redes sociais (BHARATHI; GEETHA, 2017).

Estratégias de investimento são compostas em duas categorias, as técnicas e as fundamentalistas. As análises técnicas visam seguir indicadores gráficos e encontrar padrões dentro do ambiente de negociação, fazendo uso de gráficos, indicadores como RSI (Relative Strength Index), médias móveis, linhas de tendência, entre outras técnicas (PRING, 2014).

No que se refere a análise fundamentalista visa utilizar indicadores da saúde financeira das empresas em busca de empresas que possuam potencial de crescimento e/ou estejam descontadas em seu real valor de mercado (GRAHAM; DODD, 2008).

Observa-se que análises recentes indicam que estratégias aleatórias de investimentos possuem melhor desempenho no longo prazo do que estratégias propagadas por analistas tanto técnicos quanto fundamentalistas, além de apresentarem menor custo para o investidor, trazem menor risco atrelado às operações. Enquanto no curto prazo possam desemprenhar melhor que estratégias aleatórias, as estratégias técnicas e fundamentalistas possuem maior risco e custo atrelado às operações (BIONDO; PLUCHINO; RAPISARDA; HELBING, 2013).

Nota-se que o mercado de capitais é flutuante e não-linear, prever as flutuações com exatidão é uma tarefa de difícil execução. Os recursos de previsão utilizam métodos que estão em constante atualização e melhoramento, baseando-se em sistemas iterativos, engenharia financeira, análise de dados atuais e históricos, assim como tentativa e erro. Considerando isso, pesquisou-se a aplicação de algoritmos de negociação de ativos com o uso da Inteligência

Artificial, focado em Aprendizagem Supervisionada (*Supervised Learning*) que levam em consideração os dados históricos dos ativos.

O objetivo deste artigo é utilizar os preços históricos de ativos financeiros que compõem *Exchange Traded Funds* (ETFs) para prever a direção dos retornos no dia seguinte, aplicando ordens de compra ou venda de acordo com a previsão feita por cada modelo aplicado. Ao comparar o desempenho computacional dos modelos propostos tendo como base um modelo bem estabelecido baseado na Hipótese do Passeio Aleatório é permitido avaliar o desempenho computacional e preditivo de cada método. Entre si os modelos serão avaliados utilizando métricas de erro algorítmico e cálculo do risco/retorno. O retorno financeiro de cada modelo é comparado ao retorno da técnica de comprar e segurar (*buy & hold*) os mesmos ativos durante o mesmo período de tempo, visando selecionar os mais consistentes métodos de negociação algorítmica com uso de inteligência artificial, baseando-se em desempenho computacional, de retorno e de risco.

2. FUNDAMENTAÇÃO TEÓRICA

Os métodos a serem utilizados fazem uso de preços históricos de selecionados ativos, fazendo uso dos conceitos de retornos, que são muito aplicados para diversos cálculos e análises financeiras por serem mais estacionários que valores de fechamento. Os valores obtidos quando convertidos para retornos são transformados em base logarítmica garantindo um conjunto de dados mais normalizado dentro do padrão estocástico que são os preços de ativos (TSAY, 2005).

O aprendizado de máquina é uma aplicação de inteligência artificial que simplificadamente se resume em problemas de geometria. Os modelos aplicados são programados para interpretar dados de um conjunto de dados, aprender e se melhorar de acordo com sua experiência dentro do que lhe foi fornecido. Os dados fornecidos ao sistema são cruciais para o pleno funcionamento dos modelos, fazendo-se necessário entender de questões de ciências da computação, estatística, séries temporais, geometria e álgebra para que se entenda o tipo de dados de entrada que será disposto à interpretação do algoritmo. Sem os dados corretos, ou pré-processados para o tipo correto de problema a ser resolvido pelo algoritmo, o mesmo retornará dados matematicamente pouco interpretáveis e sem conotação (MÜLLER; GUIDO, 2017).

Aprendizado de máquina, sendo uma subdivisão de estudo de inteligência artificial (IA), também pode ser dividido em três subclasses: aprendizagem reforçada, aprendizagem supervisionada e aprendizagem não supervisionada.

Nos algoritmos de aprendizagem supervisionada, que são o escopo deste projeto, o modelo busca saídas para os dados de entrada, para auxiliá-lo são utilizados pares de dados entrada/saída já conhecidos para que o algoritmo seja treinado. Então, para testar os padrões aprendidos feitos no treinamento são feitos testes que avaliam a efetividade do modelo a partir de dados de entrada ainda não vistos pelo algoritmo de maneira que é possível avaliar e classificar estatisticamente o desempenho de cada modelo perante cada conjunto de dados aplicado. Dentro da aprendizagem supervisionada os modelos podem ser classificados como de regressão ou classificação, onde os modelos regressivos buscam encontrar valores reais, como o retorno ou preço de um ativo, e os modelos de classificação buscam apenas distingui-los em uma classificação binária, como se o retorno de um ativo será positivo ou negativo (MÜLLER; GUIDO, 2017).

O índice de Sharpe é um dos instrumentos utilizados para avaliar os métodos em análise, indicando o retorno sobre investimento (ROI) comparado ao seu risco (CAMPBELL; LO; MACKINLAY, 1996; PARDO, 2008). Outros métodos importantes de avaliação de cada modelo proposto será sua comparação de retorno ao método de *buy & hold*, assim como a

robustez de cada algoritmo, medida pela sua reprodutibilidade e erros obtidos na computação dos resultados do mesmo set de dados.

As métricas que serão utilizadas para avaliar cada método são Erro Quadrático Médio (*Mean Squared Error* - MSE), Raiz Quadrada do Erro Médio ou Desvio Padrão dos Residuais (*Root Mean Square Error* - RMSE) e Erro Médio Absoluto (*Mean Absolute Error* - MAE) para os modelos de regressão, e as pontuações de acurácia, precisão, média harmônica da precisão e lembrança (F1), Característica de Operação do Receptor (*Receiver Operating Characteristic* - ROC) e Área Abaixo da Curva (*Area Under the Curve* - AUC), para os modelos de classificação, buscando garantir um meio estatístico comparativo para classificar a acurácia dos modelos. A pontuação roc_auc_score refere-se à área abaixo da curva de ROC, onde o valor varia de 0 a 1, avaliando a capacidade preditiva do modelo (MASÍS, 2021).

O modelo de previsão ingênua (Naïve Forecast) é o método mais básico de previsão em séries temporais, e uma métrica utilizada para avaliação de modelos testados. O modelo se baseia na Hipótese do Passeio Aleatório, a qual propõe que os preços de ativos seguem um padrão completamente aleatório, tornando-os imprevisíveis por serem despadronizados e não relacionados a quaisquer acontecimentos passados. É um modelo relevante, levado em consideração para a avaliação de modelos de previsão, sendo adotado para fundamentar o modelo de previsão ingênua a qual considera que a melhor previsão para o preço de um ativo amanhã é o preço em que o ativo possui hoje (MALKIEL, 2019).

Desta maneira a previsão ingênua permite o estabelecimento de uma linha de base para comparação com outros modelos. Assim, copiando o valor de um retorno do dia anterior, o algoritmo permite a comparação com algoritmos que buscam esses valores com modelos mais sofisticados. A comparação com os modelos sofisticados é medida através das pontuações de treino, teste e cálculo dos erros obtidos por cada modelo, com ênfase no resultado dos testes, pois é onde o algoritmo testará seu aprendizado verdadeiramente. Se a previsão ingênua possuir menos erros e mais pontuação de acerto que os modelos sofisticados o modelo testado pode ser considerado ruim, pois usa mais capacidade computacional, possui mais erros e erra mais as previsões do que um modelo simples.

Dentre os métodos utilizados, para essa pesquisa foram selecionados 12 modelos de algoritmos para a previsão das direções ativos financeiros com diferentes métodos de análise e de processamento de informações, que são eles: Regressão linear (*Linear Regression*); Regressão de crista (*Ridge Regression*); XGBoost (*Extreme Gradient Boosting* para regressão e classificação); LGBM (*Light Gradient Boosting Machine* para regressão e classificação); Regressão logística (*Logistic Regression*); *Linear Support Vector Machines* (LinearSVC e SVC); Floresta Aleatória (*Random Forest*). K-ésimos Vizinhos Mais Próximos (*K-Nearest Neighbors*); e *Gaussian Naive Bayes* (KUMAR et al, 2018; NABIPOUR et al, 2020; YANG et al, 2021; ASHFAQ; NAWAZ; ILYAS, 2021).

2.1. Modelos de Regressão

2.1.1. Regressão Linear (*Linear Regression*)

Os modelos de regressão linear utilizam a equação (1) como modelo genérico de regressão.

$$y = \beta_0 * x_0 + \beta_1 * x_1 + \cdots + \beta_n * x_n = \sum_{i=0}^{n} \beta_i x_i \qquad (1)$$

Onde β_0 representa o valor em que o eixo y intercepta a linha $x_0 = 1$, e os valores de x_1 a x_n representam as características de dados mensuráveis (variáveis independentes) presentes em um único ponto de dados. Ex.: Preço médio, de abertura, fechamento, volume, e retornos de diferentes ativos, que é o caso deste estudo. Já β são parâmetros aprendidos pelo modelo e y é a previsão feita pelo modelo (RASCHKA; MIRJALILI, 2017).

A regressão linear é o mais simples e clássico dos modelos de regressão, o qual encontra os parâmetros para β que minimizam o erro quadrático médio entre as previsões. Onde o erro quadrático médio é a soma das diferenças ao quadrado entre a previsão e os valores reais, e por fim encontra os valores para y em seu treinamento. Este método, por natureza, impede que o modelo descubra relações não lineares das variáveis independentes a serem relacionadas (MÜLLER; GUIDO, 2017).

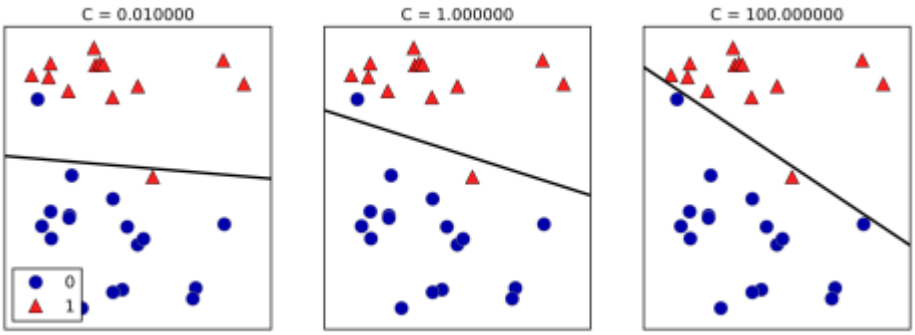
2.1.2. Regressão de Crista (*Ridge Regression*)

A regressão de crista é uma variação da regressão linear, porém, os coeficientes β são selecionados para prever e ainda para se adequar às restrições compostas pelo modelo linear. Com um processo chamado regularização L2 (ALBON, 2018), o modelo atribui ao coeficiente β à menor magnitude possível, próximas à zero, com a finalidade de cada coeficiente interferir o mínimo possível na previsão enquanto o modelo ainda prevê melhor que o da regressão linear. A regularização L2 garante ao processo que os dados fornecidos não gerem *overfitting*, ou seja, que os dados se adequem ao modelo nos treinos, resultando em um baixo viés e uma grande variação nas previsões (MASÍS, 2021).

2.2. Modelos de Classificação

Assim como o regressor de crista, alguns dos modelos de classificação utilizam o regressor L2 (ALBON, 2018), visando minimizar o *overfitting* dos dados, buscam encontrar os menores valores possíveis para os coeficientes β, e utilizam um parâmetro C implementado em código para controlar a intensidade da regularização L2, quanto maior o valor de C, menores as regularizações do modelo (HARRELL, 2015). A efetividade do parâmetro C na regularização é demonstrada pela Figura 1, onde é evidenciado como atuam os diferentes graus de regularização L2 na classificação dos dados de maneira binária. Os círculos azuis representam os valores reais 0 e os triângulos vermelhos os valores reais 1. A linha representa o plano de fronteira de decisão que faz a classificação preditiva entre 0 e 1 de acordo com o grau de L2 definido por C.

Figura 1 – Mudanças na fronteira de decisão pelas regularizações L2 controladas pelo parâmetro C



Fonte: MÜLLER; GUIDO (2017, p. 58).

No caso de classificação, os modelos lineares utilizam a equação (2), similarmente à dos modelos de regressão linear.

$$y = \beta_0 * x_0 + \beta_1 * x_1 + \cdots + \beta_n * x_n = \sum_{i=0}^{n} \beta_i x_i > 0 \quad (2)$$

Porém ao invés de retornar a soma dos valores, o valor previsto é limitado pelo zero. Se o valor for inferior ou igual à 0 é atribuído 0 (zero), se superior, é atribuído $+1$. Em outras palavras, ao invés de retornar uma linha, plano ou hiperplano (em dimensões superiores) como no modelo linear, o método de classificação retorna valores binários separados em duas classes por uma linha, plano ou hiperplano (ALBON, 2018; AUFFARTH, 2021; MASÍS, 2021).

2.2.1. Regressão Logística (*Logistic Regression*)

Apesar do nome, a regressão logística é um método de classificação binária, que adota um modelo linear ($\beta_0 * x_0 + \beta_1 * x_1$) dentro de uma função logística, também chamada de função sigmoide $\frac{1}{1+e^{-z}}$, tal como a equação (3) (ALBON, 2018).

$$P(y_i = 1|X) = \frac{1}{1 + e^{-(\beta_0 * x_0 + \beta_1 * x_1)}} \quad (3)$$

Onde $P(y_i = 1|X)$ é a probabilidade de observação do i-ésimo valor y_i sendo classe 1, X os dados treinados, β_0 e $\beta_1$os parâmetros a serem aprendidos, e e o número de Euler.

Sendo um classificador binário, o modelo de regressão logística não consegue suportar vetores com mais de duas classes, portanto, para solucionar esse problema o modelo possui uma extensão que faz uso da equação (4) (ALBON, 2018).

$$P(y_i = k|X) = \frac{e^{\beta_k * x_i}}{\sum_{j=1}^{K} e^{\beta_j * x_i}} \quad (4)$$

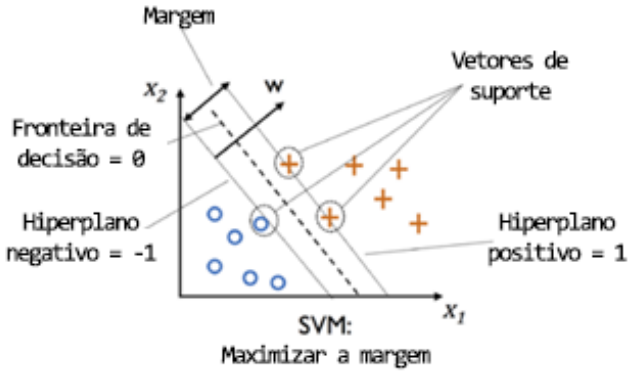
Onde $P(y_i = k|X)$ é a probabilidade de observação do i-ésimo valor y_i sendo classe k, e K o número total de classes. Podendo também ser parametrizado para obter valores binários variando entre -1 e 1 (ALBON, 2018).

2.2.2. Máquinas de Vetores de Suporte (*Support Vector Machines*)

O método de classificação por máquina de vetor de suporte linear assim como o regressor logístico visa classificar os dados de maneira binária $(0,1)$, e é implementado pelos modelos SVC (*Support Vector Classifier*) e LinearSVC (*Linear Support Vector Classifier*) o qual utiliza os valores dos dados de treino mais próximos da fronteira de decisão (chamados de vetores de suporte) buscando maximizar as margens, definidas pela distância entre os valores de treinos e o hiperplano de separação (fronteira de decisão) (RASCHKA; MIRJALILI, 2017).

A Figura 2 é uma representação de como o método faz a divisão dos dados, onde a margem é definida pela distância que separa os hiperplanos (fronteira de decisão) e as amostras de treino, que são os valores mais próximos desses hiperplanos, chamados de vetores de suporte (RASCHKA; MIRJALILI, 2017).

Figura 2 – Ilustração dos hiperplanos, vetores de suporte, margem e a fronteira de decisão



Fonte: RASCHKA; MIRJALILI (2017, p. 77).

Visando maximizar as margens, o problema de classificação pode ser expresso pela equação (5).

$$y = \beta^T \varphi(x) + b \tag{5}$$

Onde para calcular a função $\varphi(x)$ é utilizado engenharia de características (*feature engineering*), fazendo o uso de kernels para transformar a característica dos dados de entrada de lineares para não lineares, assim tornando $\varphi(x) = [x^1, x^2, \dots x^n]$, resultando em um modelo de saída não linear. O kernel mais utilizado para essa transformação é o RBF Kernel, utilizado pelo modelo SVM, também chamado de Kernel Gaussiano, que faz uso da equação (6) para classificar o ponto de análise baseado em sua distância dos vetores de suporte. Enquanto o modelo SVC utiliza o Kernel Gaussiano, o modelo LinearSVC faz uso de uma função linear como base para o Kernel, sendo menos parametrizável que o anterior (ZHENG; CASARI, 2018).

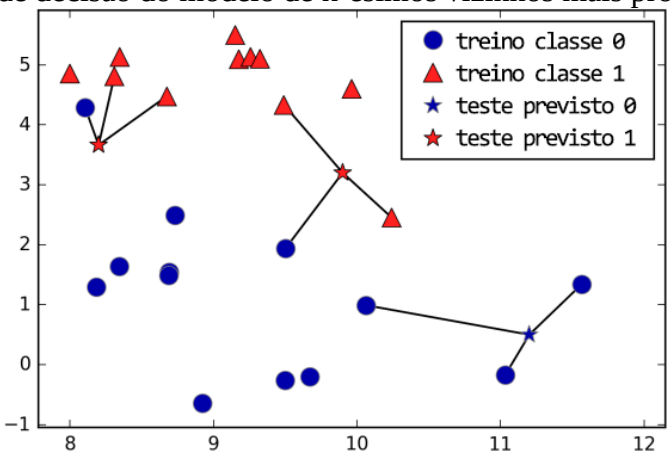
$$k_{rbf}(x_1, x_2) = \exp(\gamma \|x_1 - x_2\|^2) \tag{6}$$

Onde x_1 e x_2 são pontos de dados, $\|x_1 - x_2\|$ denota a distância Euclidiana e γ (gama) é o parâmetro que controla a largura do kernel gaussiano (MÜLLER; GUIDO, 2017; RASCHKA; MIRJALILI, 2017).

2.2.3. K-ésimos Vizinhos Mais Próximos (*K-Nearest Neighbors*)

O modelo do vizinho mais próximo consiste em utilizar as observações feitas no treino para encontrar o valor de entrada x mais próximo para encontrar o valor de saída Y (HASTIE; TIBSHIRANI; FRIEDMAN, 2009). Ao invés de considerar o único vizinho mais próximo no conjunto de dados, o modelo do k-ésimos vizinhos mais próximos utiliza um número arbitrário k, de vizinhos, decidindo o valor de saída por meio de votação. Isso significa que para cada ponto de teste, são contados quantos vizinhos são de classificação 0 e quantos são de classificação 1, sendo decidido pelo valor binário de maior frequência, como demonstrado na Figura 3 para a escolha de $k = 3$. (MÜLLER; GUIDO, 2017).

Figura 3 – Tomada de decisão do modelo de k-ésimos vizinhos mais próximos em que $k = 3$



Fonte: MÜLLER; GUIDO (2017, p. 36).

Em forma de equação, o modelo de k-ésimos vizinhos mais próximos, ou KNN, pode ser expresso pela equação (7) em busca de Y.

$$Y(x) = \frac{1}{k}\sum_{x_i \, \epsilon \, N_k(x)} y_i \tag{7}$$

Onde $N_k(x)$ é a vizinhança de x definido pelos k números próximos x_i na amostragem de treino. A proximidade dos valores implica na utilização de uma métrica, assumida pela distância euclidiana. Em palavras, são encontradas as k observações com o valor x_i mais próximas ao dado de entrada x e tomadas a média dos valores de resposta (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

2.2.4. *Gaussian Naive Bayes*

O modelo *Naive Bayes* utiliza o cálculo de probabilidade condicional do teorema de Bayes (8) para auxiliar na classificação de dados. A inclusão do nome Gaussian dá-se pelo fato do modelo presumir que os valores contínuos possuem uma distribuição normal (Gaussiana) (MASÍS, 2021).

$$P\left(y\middle|x_1, \cdots, x_j\right) = \frac{P(x_1, \cdots, x_j|y)\, P(y)}{P(x_1, \cdots, x_j)} \tag{8}$$

Onde, $P\left(y\middle|x_1, \cdots, x_j\right)$ é a probabilidade que uma observação seja de classe y dados os valores das observações para os j recursos $x_1, \cdots, x_j$; $P(x_1, \cdots, x_j|y)$ é a probabilidade dos valores $x_1, \cdots, x_j$ acontecerem dada a classe y; $P(y)$ é a probabilidade da classe y e $P(x_1, \cdots, x_j)$ é a probabilidade marginal.

É presumido que a probabilidade dos valores de recursos x, dada a observação é de classe y, seguindo a distribuição normal na equação (9).

$$p(x_j|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}}\, e^{-\frac{(x_j-\mu_y)^2}{2\sigma_y^2}} \tag{9}$$

Onde σ_y^2 e μ_y são a variância e os valores médios dos recursos x_j para a classe y (ALBON, 2018).

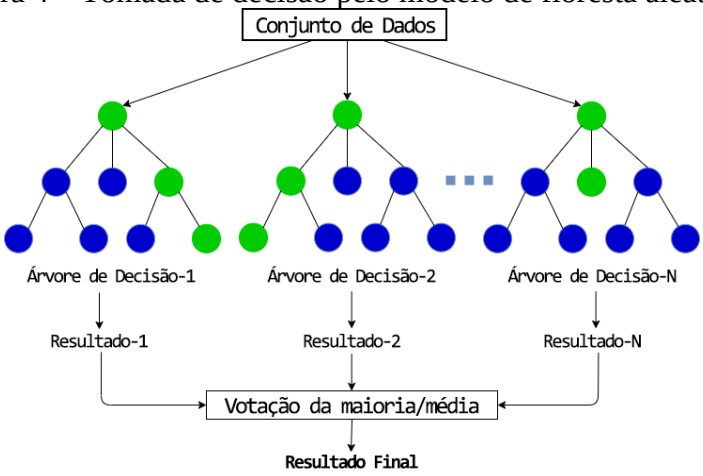
2.2.5. Floresta Aleatória (*Random Forest*)

Os modelos que utilizam árvores de decisões são um agrupamento de questões if/then/else em cadeia que visam selecionar o melhor caminho para uma tomada de decisão com base em seu treinamento. Para a construção da decisão, o algoritmo busca todos os caminhos possíveis a serem tomados pelas questões (chamadas de testes) e encontra o caminho que leva ao resultado mais relevante mediante ao resultado alvo apresentado ao algoritmo (MÜLLER; GUIDO, 2017). Modelos que utilizam apenas uma árvore de decisão são mais suscetíveis à um *overfitting* de dados ocasionando uma alta variância, para evitar isso, com o objetivo de formar um modelo mais robusto e com uma melhor performance é introduzido o uso de modelos com múltiplas árvores de decisão, como as presentes nos modelos de floresta aleatória, LightGBM e XGBoost (RASCHKA; MIRJALILI, 2017).

O método da floresta aleatória é uma composição de múltiplas árvores de decisão que utilizam seções do conjunto de dados para serem treinadas, deste modo, diminuindo a quantidade de dados de entrada para cada árvore, evita-se a tomada de decisão baseada em um todo do conjunto de dados, ao combinar os resultados de múltiplas árvores (ALBON, 2018). Intuitivamente, cada árvore faz uma previsão perfeita de cada seção de dados que lhe foi apresentada, resultando em previsões diferentes que então são tiradas a média, um processo chamado de votação, onde a maioria das decisões semelhantes vencerá. A função retornada por um modelo de floresta aleatória é muito mais suave que a gerada por apenas uma árvore de decisão, suavizando quaisquer variações ocorridas por grandes variações no conjunto de dados de entrada por motivo da divisão (MÜLLER; GUIDO, 2017).

A Figura 4 apresenta um fluxograma simplificado de como são compostas as árvores de decisão e o processo de decisão dentro do modelo de floresta aleatória.

Figura 4 – Tomada de decisão pelo modelo de floresta aleatória.



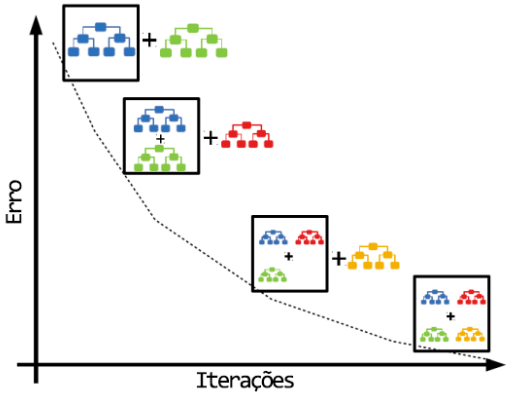
Fonte: SHARMA (2020).

2.2.6. *Gradient Boosting*

Modelos de *gradient boosting*, também chamados de *gradient boosted regression trees* são composições de múltiplas árvores de decisão que buscam incorporá-las para formar um modelo mais eficiente. Os modelos de *gradient boosting* podem ser utilizados para tarefas tanto de regressão quanto de classificação. Em contraste com a floresta aleatória, esses modelos buscam construir árvores de maneira sequencial, visando corrigir os erros das árvores anteriores (MÜLLER; GUIDO, 2017). Os modelos mais conhecidos de implementação de *gradient boosting* são *Light Gradient Boosting Machine* (LightGBM, pela Microsoft) e XGBoost

(AUFFARTH, 2021). A implementação do modelo de *gradient boosting* generalizado e sua redução de erros com a evolução das iterações pode ser demonstrado pela Figura 5.

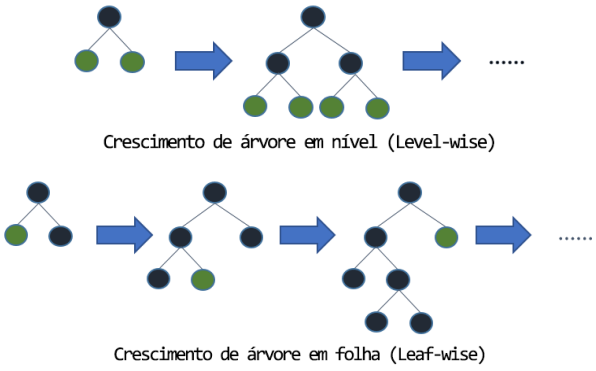
Figura 5 – Redução do erro pela adição de novas árvores de decisão pela iteração nos modelos de *gradient boosting*



Fonte: SAHA (2020).

Dentre os dois modelos citados, o LightGBM apresenta mais rapidez em sua execução por utilizar duas técnicas, o *Gradient-Based One-Side Sampling* (GOSS) para amostragem e *Exclusive Feature Bundling* (EFB) para o agrupamento de recursos esparsos, enquanto o XGBoost utiliza as técnicas mais rigorosas de Quantile Sketch e Sparsity-aware Split Finding. A principal diferença entre os modelos é de como as árvores são construídas, enquanto o XGBoost constrói suas árvores de maneira horizontal focado em níveis (*level-wise* ou *depth-first*), o LightGBM as constrói de maneira vertical (*leaf-wise* ou *best-first*) (MASÍS, 2021). A diferença de construção das árvores de cada modelo é demonstrada na Figura 6.

Figura 6 – Diferença entre construções de árvores de decisões *level-wise* e *leaf-wise*



Fonte: KE (2016).

Na engenharia financeira, faz-se necessária a compreensão dos conceitos de covariância e correlação, que são muito utilizados no mercado financeiro a fim de evidenciar as relações entre duas variáveis. Ambas as métricas relacionam o movimento entre dois ativos.

A covariância mede o grau de dependência entre duas variáveis. A função de correlação entre as duas variáveis é expressa na equação (10) (FRANKE; HÄRDLE; HAFNER, 2008; BENNINGA, 2014).

$$cov(x,y) = \frac{1}{N}\sum_{t=1}^{N}(r_{x(t)} - \bar{r}_x)(r_{y(t)} - \bar{r}_y) \qquad (10)$$

Onde os ativos são representados por x, y , $r_{x(t)}$ e $r_{y(t)}$ representam os retornos dos ativos no período t e $\bar{r}_x, \bar{r}_y$ são as médias dos retornos dos ativos em todo o período N.

A correlação mede o grau de associação linear entre as duas variáveis, com valores de -1 a 1, onde em 1, as variáveis se relacionam de maneira totalmente simétrica, enquanto em -1, de maneira totalmente assimétrica. Essa métrica é utilizada para descrever a relação dos movimentos entre dois ativos (BROOKS, 2008). Utilizando o exemplo de retornos de ativos, quando a correlação entre os ativos x e y for igual a 1, $corr(x, y) = 1$, o retorno do ativo x será idêntico ao do ativo y, ou seja, se um tem retorno de 1%, o outro terá também retorno de 1%. No caso oposto, quando $corr(x, y) = -1$, o retorno do ativo x será oposto ao do ativo y, em outras palavras, se um tiver retorno de 1%, o outro terá de -1%. O cálculo da correlação, como demonstrado na equação (11), faz uso da covariância conforme mostrado na equação (10) e permite que as relações entre as variáveis sejam melhor visualizadas (CAPINSKI; ZASTAWNIAK, 2003).

$$corr(x, y) = \frac{cov(x,y)}{\sigma_x \sigma_y} \tag{11}$$

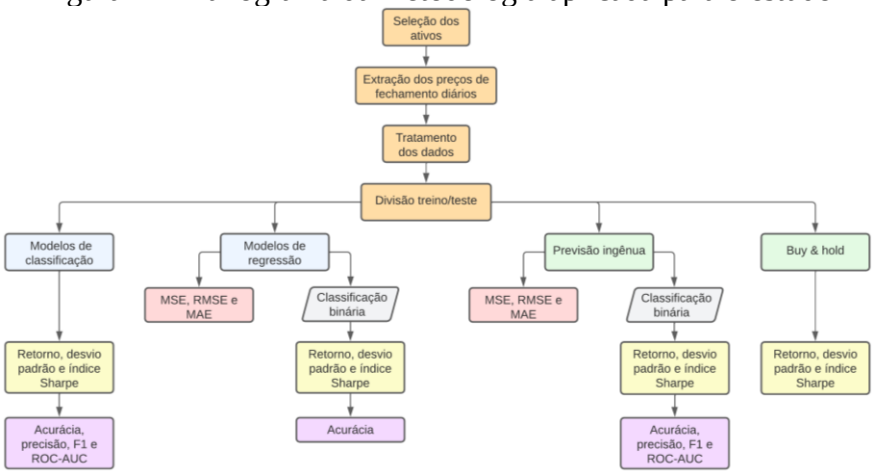
Onde σ_x e σ_y são os desvios padrão dos ativos x e y.

A utilização de todos os conceitos apresentados possui o objetivo de encontrar uma aplicabilidade dos dados do mercado financeiro fazendo uso da engenharia financeira para compreendê-los e torná-los utilizáveis em modelos de machine learning dentro do ambiente de programação. Os modelos propostos possuem uma grande capacidade de gerar resultados proveitosos em diversas áreas. A composição de múltiplos modelos agrega na tomada de decisão no ramo de negociação de ativos por possuírem técnicas e abordagens diferentes aos dados fornecidos, podendo assim, a composição dos modelos ser uma ferramenta auxiliar nas análises quantitativas quando implementadas adequadamente. A composição também permite que os modelos tenham seu desempenho comparado entre si, permitindo que sejam percebidos os comportamentos de cada abordagem ao conjunto de dados fornecido.

3. PROCEDIMENTO METODOLÓGICO

Para representar o desenvolvimento da pesquisa e a metodologia aplicada, foi elaborado o fluxograma presente na Figura 7, visando evidenciar os passos tomados e como os dados foram comparados.

Figura 7 – Fluxograma da metodologia aplicada para o estudo



Fonte: Autor.

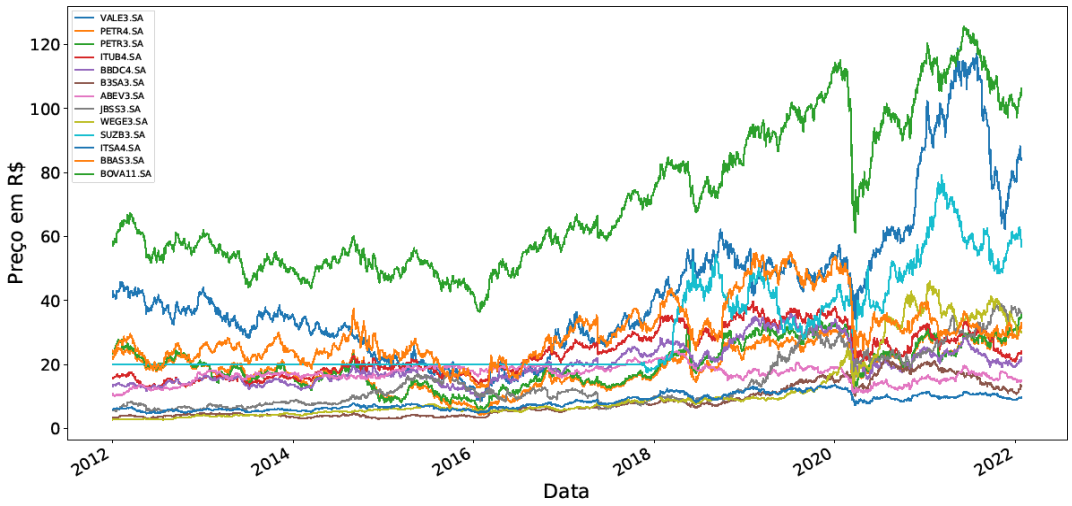
Foram selecionados dois índices base para o estudo de cada um dos métodos utilizados. O índice brasileiro Ibovespa e o índice americano S&P500 que são representados respectivamente pelos ETFs BOVA11 e SPY. Ambos os ETFs buscam replicar os índices selecionados como benchmark, compondo suas carteiras com as maiores ações listadas em suas respectivas bolsas. Para garantir a aplicabilidade das análises, foram selecionadas as 12 maiores empresas de cada índice, as quais continham valores negociados em bolsa a partir da data de 01/01/2012, com o objetivo de prever o retorno e direção para o cada um dos índices utilizando suas 12 componentes selecionadas.

O ambiente de programação foi o ambiente de desenvolvimento integrado (IDE) Anaconda, utilizando o programa de edição de código fonte Visual Studio Code, e a linguagem de programação Python 3.6.5. Importantes interfaces de programação de aplicações (APIs) foram utilizados para auxiliar na manipulação, visualização e avaliação de cada método, scikit-learn, lightgbm, xgboost, yfinance, pandas, numpy e matplotlib.

Os dados extraídos através da API do Yahoo finance de cada ação foram os preços de fechamento diários entre o período de 01/01/2012 a 25/01/2022 aqui expostas no formato dd/mm/aaaa, contendo 2496 dias de negociação no conjunto de dados brasileiro e 2533 dias de negociação no conjunto de dados americano.

As ações selecionadas para o índice Ibovespa foram: Vale (VALE3), Petróleo Brasileiro - Petrobras (PETR4 e PETR3), Itaú Unibanco (ITUB4), Banco Bradesco (BBDC4), B3 – Brasil Bolsa Balcão (B3SA3), Ambev (ABEV3), JBS (JBSS3), WEG (WEGE3), Suzano (SUZB3), Itaúsa (ITSA4) e Banco do Brasil (BBAS3). As ações selecionadas representam em torno de 50% da composição da carteira do ETF BOVA11. A evolução dos preços dos ativos brasileiros selecionados no período estudado é demonstrada na Figura 8.

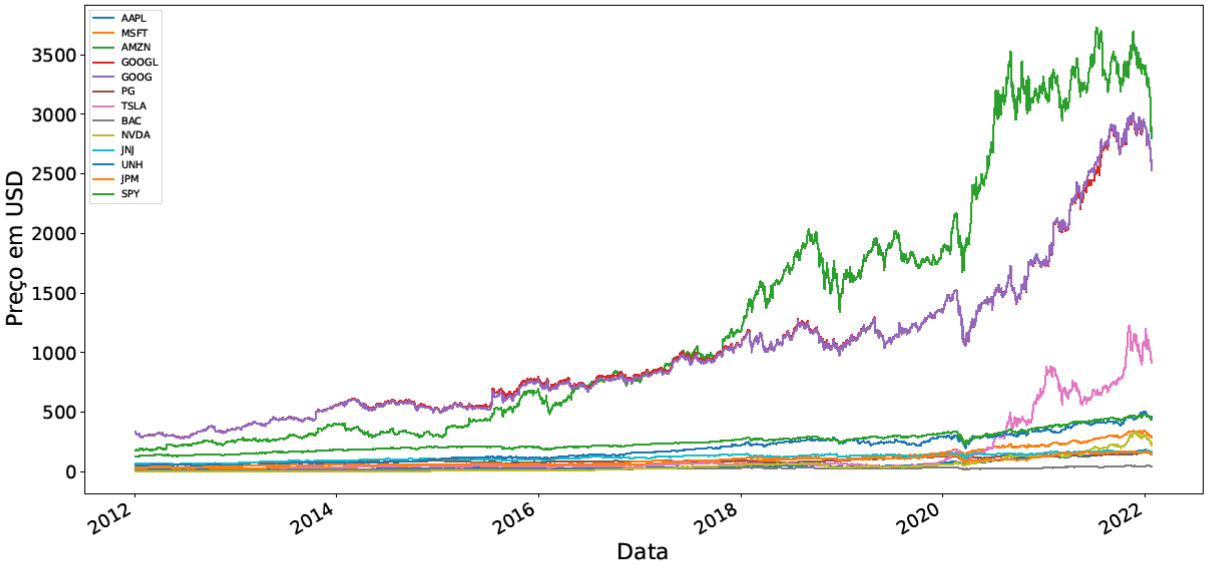
Figura 8 – Evolução do preço das ações brasileiras selecionadas e do ETF BOVA11



Fonte: Autor.

As ações selecionadas para o índice S&P 500 foram: Apple (AAPL), Microsoft (MSFT), Amazon (AMZN), Alphabet* (FORMER GOOGLE) (GOOGL e GOOG), Procter & Gamble (PG), Tesla (TSLA), Bank of America (BAC), Nvidia (NVDA), Johnson & Johnson (JNJ), UnitedHealth Group (UNH) e JPMorgan Chase (JPM). As ações selecionadas representam em torno de 50% da composição da carteira do ETF SPY. A evolução dos preços dos ativos americanos selecionados no período estudado é demonstrada na Figura 9.

Figura 9 – Evolução do preço das ações americanas selecionadas e do ETF SPY



Fonte: Autor.

Para que seja visualizada e compreendida a relação entre os ativos selecionados e seus respectivos ETFs, são demonstradas nas Tabelas 1 e 2 as matrizes de correlação.

Tabela 1 - Correlação dos ativos brasileiros

CORR()	VALE3	PETR4	PETR3	ITUB4	BBDC4	B3SA3	ABEV3	JBSS3	WEGE3	SUZB3	ITSA4	BBAS3	BOVA11
VALE3	1.0000	0.4633	0.4937	0.3437	0.3788	0.3319	0.2393	0.1971	0.2086	0.1514	0.3368	0.3493	0.5785
PETR4	0.4633	1.0000	0.9594	0.5779	0.6118	0.5334	0.3573	0.3001	0.3066	0.1029	0.5834	0.6312	0.7664
PETR3	0.4937	0.9594	1.0000	0.5720	0.6092	0.5203	0.3636	0.2999	0.3089	0.1119	0.5711	0.6022	0.7612
ITUB4	0.3437	0.5779	0.5720	1.0000	0.8488	0.5873	0.4504	0.2778	0.3307	0.0305	0.9099	0.7336	0.7759
BBDC4	0.3788	0.6118	0.6092	0.8488	1.0000	0.6064	0.4587	0.3005	0.3406	0.0426	0.8304	0.7650	0.7953
B3SA3	0.3319	0.5334	0.5203	0.5873	0.6064	1.0000	0.3979	0.3171	0.3755	0.0609	0.6108	0.5834	0.7222
ABEV3	0.2393	0.3573	0.3636	0.4504	0.4587	0.3979	1.0000	0.2461	0.3593	0.1112	0.4607	0.3906	0.5434
JBSS3	0.1971	0.3001	0.2999	0.2778	0.3005	0.3171	0.2461	1.0000	0.2144	0.1480	0.3088	0.3102	0.4394
WEGE3	0.2086	0.3066	0.3089	0.3307	0.3406	0.3755	0.3593	0.2144	1.0000	0.1566	0.3517	0.3096	0.4717
SUZB3	0.1514	0.1029	0.1119	0.0305	0.0426	0.0609	0.1112	0.1480	0.1566	1.0000	0.0344	0.0147	0.1466
ITSA4	0.3368	0.5834	0.5711	0.9099	0.8304	0.6108	0.4607	0.3088	0.3517	0.0344	1.0000	0.7464	0.7873
BBAS3	0.3493	0.6312	0.6022	0.7336	0.7650	0.5834	0.3906	0.3102	0.3096	0.0147	0.7464	1.0000	0.7597
BOVA11	0.5785	0.7664	0.7612	0.7759	0.7953	0.7222	0.5434	0.4394	0.4717	0.1466	0.7873	0.7597	1.0000

Fonte: Autor.

Tabela 2 - Correlação dos ativos americanos

CORR()	AAPL	MSFT	AMZN	GOOGL	GOOG	PG	TSLA	BAC	NVDA	JNJ	UNH	JPM	SPY
AAPL	1.0000	0.5635	0.4472	0.5191	0.5190	0.3326	0.3260	0.3730	0.4779	0.3386	0.3986	0.3825	0.6622
MSFT	0.5635	1.0000	0.5350	0.6342	0.6347	0.4198	0.3419	0.4254	0.5541	0.4339	0.4659	0.4530	0.7470
AMZN	0.4472	0.5350	1.0000	0.5829	0.5839	0.2267	0.3253	0.2811	0.4385	0.2826	0.3036	0.2738	0.5535
GOOGL	0.5191	0.6342	0.5829	1.0000	0.9931	0.3472	0.3264	0.4132	0.4961	0.3872	0.4233	0.4265	0.6949
GOOG	0.5190	0.6347	0.5839	0.9931	1.0000	0.3463	0.3253	0.4161	0.4918	0.3830	0.4216	0.4285	0.6923
PG	0.3326	0.4198	0.2267	0.3472	0.3463	1.0000	0.1331	0.3130	0.2448	0.5439	0.3785	0.3462	0.5630
TSLA	0.3260	0.3419	0.3253	0.3264	0.3253	0.1331	1.0000	0.2402	0.3358	0.1492	0.2104	0.2385	0.4033
BAC	0.3730	0.4254	0.2811	0.4132	0.4161	0.3130	0.2402	1.0000	0.3539	0.3769	0.4385	0.8690	0.7109
NVDA	0.4779	0.5541	0.4385	0.4961	0.4918	0.2448	0.3358	0.3539	1.0000	0.2627	0.3441	0.3421	0.5944
JNJ	0.3386	0.4339	0.2826	0.3872	0.3830	0.5439	0.1492	0.3769	0.2627	1.0000	0.4658	0.4277	0.6195
UNH	0.3986	0.4659	0.3036	0.4233	0.4216	0.3785	0.2104	0.4385	0.3441	0.4658	1.0000	0.4700	0.6379
JPM	0.3825	0.4530	0.2738	0.4265	0.4285	0.3462	0.2385	0.8690	0.3421	0.4277	0.4700	1.0000	0.7438
SPY	0.6622	0.7470	0.5535	0.6949	0.6923	0.5630	0.4033	0.7109	0.5944	0.6195	0.6379	0.7438	1.0000

Fonte: Autor.

Para facilitar as aplicações algorítmicas, garantindo mais robustez aos modelos e facilidade de processamento dos dados, foram utilizados os preços de fechamento para calcular os retornos logarítmicos diários de cada ativo, pelo motivo de algoritmos de machine learning não serem bons em extrapolação por sua natureza, sendo retornos dados mais estacionários, são uma boa opção como dados de entrada. O retorno logarítmico r_t pode ser expresso pela diferença entre os preços logarítmicos p_t pela equação (12) (TSAY, 2005).

$$r_t = p_t - p_{t-1} \tag{12}$$

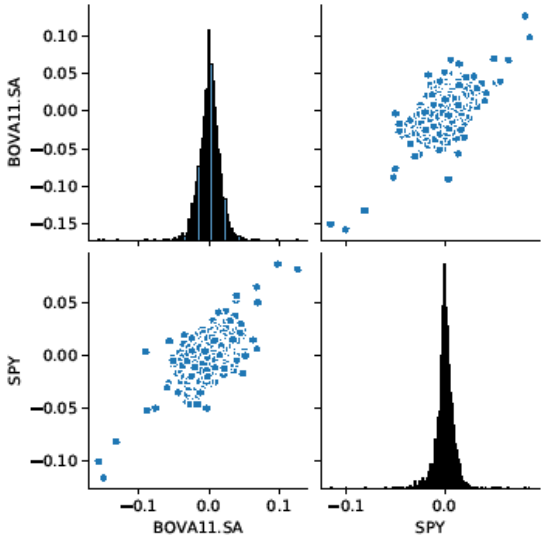
Onde o retorno R_t é expresso pela equação (13) (TSAY, 2005).

$$R_t = \frac{P_t}{P_{t-1}} - 1 \tag{13}$$

Os dados de fechamento são utilizados para calcular os retornos logarítmicos utilizando as funções log() e diff() do numpy, sendo adicionados uma nova coluna ao conjunto de dados chamada LogReturn.

Para fins de visualização, foi plotado a matriz de dispersão (scatter plot) dos retornos dos ETFs BOVA11 e SPY na Figura 10, para que seja verificada a correlação dos retornos de cada índice dos conjuntos de dados propostos. Os histogramas representam a distribuição dos retornos de cada ETF, enquanto os gráficos de dispersão relacionam os retornos de cada ETF em ambos os eixos, onde uma linha reta representaria uma correlação perfeita entre os ativos (TSAY, 2005; BENNINGA, 2014; BROOKS, 2008).

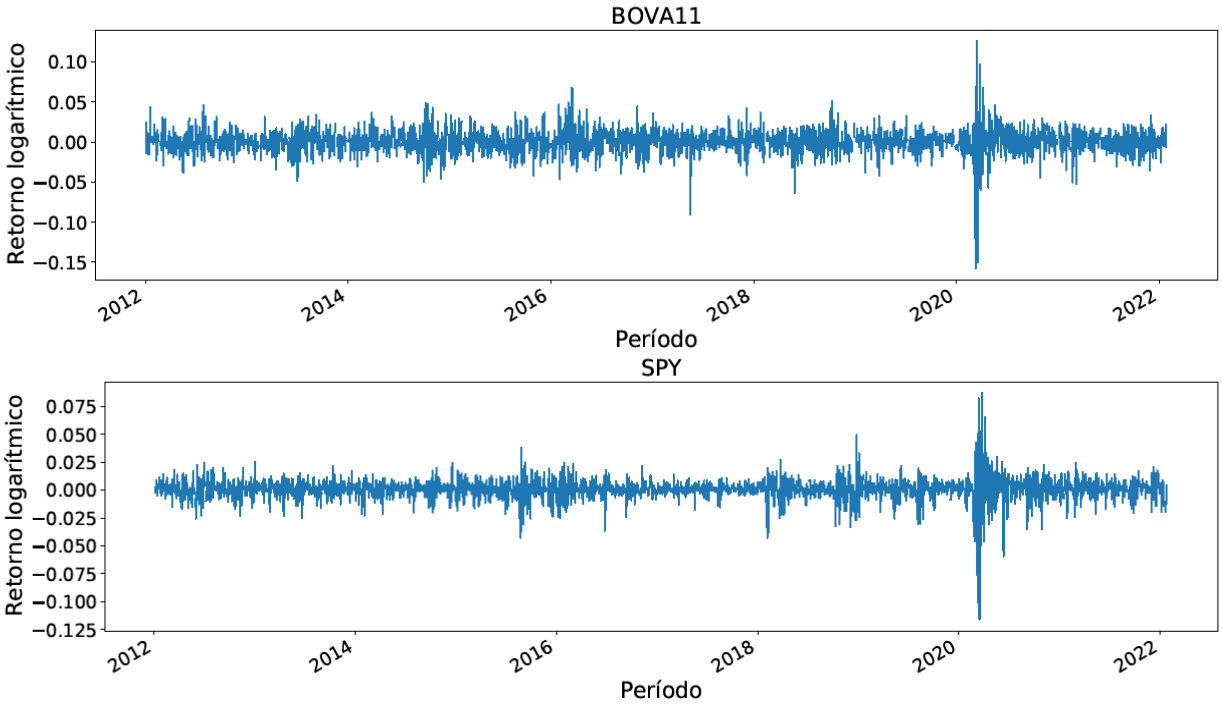
Figura 10 – Matriz de dispersão e histogramas dos retornos dos ETFs selecionados



Fonte: Autor.

Os retornos de cada ETF foram plotados, sendo apresentados na Figura 11, de maneira a evidenciar a distribuição de seus retornos ao longo do período total selecionado, assim como o agrupamento de volatilidade (volatility clustering), fenômeno financeiro que ocorre demonstrando que momentos de alta volatilidade são seguidos por momentos de alta volatilidade e momentos de baixa volatilidade são seguidos por momentos de baixa volatilidade (TSAY, 2005; BROOKS, 2008).

Figura 11 – Gráfico dos retornos - Agrupamento de volatilidade dos ETFs



Fonte: Autor.

Os conjuntos de dados então são divididos em duas partes, sendo considerados os últimos mil (1000) dias de negociação, os dados utilizados para testes em cada algoritmo, restando 1496 e 1533 dias de negociação para os treinos dos conjuntos de dados brasileiro e americano respectivamente, garantindo uma divisão em torno de 60/40% para os períodos de treino e teste. Os algoritmos consideram um investimento de 100% do capital no caso de compra, assim como um zeramento total de posição quando indicada a venda, a fim de utilizar o valor 0 como retorno livre de risco no cálculo do índice de Sharpe.

Os modelos de regressão e classificação são utilizados para analisar todos os dados disponíveis até o passo de tempo (t) buscando prever a direção do retorno para o passo (t+1). Os dados de entrada são os retornos logarítmicos das 12 ações brasileiras selecionadas para o ETF brasileiro e das 12 ações americanas selecionadas para o ETF americano, como exemplificado pelos dados de entrada (x) para os de saída (y), sendo esses os retornos do ETF BOVA11.

Aplicando os modelos de regressão e classificação providos pelos APIs scikit-learn, lightgbm e xgboost, os modelos analisam os mesmos dados de entrada, ou seja, o mesmo conjunto de dados é fornecido para cada modelo, permitindo analisar o comportamento de previsão de cada modelo sob um mesmo conjunto de dados padrão.

Para treinar o algoritmo, os retornos do ETF de cada dia seguinte, obtidos nas séries históricas, são atribuídos aos retornos do dia anterior das 12 ações componentes, permitindo que o modelo estabeleça uma relação entre os retornos das componentes no passo de tempo (t), e o retorno do ETF obtido no passo de tempo (t+1). Para que sejam atribuídos os valores dos retornos reais do ETF em (t+1) aos retornos das componentes em (t), a tabela do conjunto de dados é manipulada, com o objetivo de que os dados de entrada estejam na mesma linha do resultado esperado de saída como demonstrado na Tabela 3. Fazendo essa associação, os algoritmos fazem a aprendizagem no período de treino, permitindo que estabeleçam relações obtendo ambos os valores de entrada (x) e saída (y). No período de testes, obtendo apenas os valores de entrada (x), os modelos utilizam a aprendizagem feita para entregar os valores de saída (y), sendo essas as previsões.

Tabela 3 - Associação dos retornos das ações no período t com o retorno obtido do ETF em t+1

Período	Entrada (x) Ação 1	Ação 2	…	Ação 12	Saída (y) ETF
t	Retorno(t)	Retorno(t)	…	Retorno(t)	Retorno(t+1)
t+1	Retorno(t+1)	Retorno(t+1)	…	Retorno(t+1)	Retorno(t+2)
t+2	Retorno(t+2)	Retorno(t+2)	…	Retorno(t+2)	Retorno(t+3)
…	…	…	…	…	…
t+n	Retorno(t+n)	Retorno(t+n)	…	Retorno(t+n)	Retorno(t+n+1)

Fonte: Autor.

Os modelos então, de acordo com cada uma das técnicas, classificam os dados de maneira binária (0,1) para quando a previsão do retorno do ETF no dia seguinte (t+1) no Período (t) é menor ou maior que zero, respectivamente, ou seja, de acordo com sua direção.

As classificações, sendo binárias, são atribuídas a uma nova coluna, chamada Posição. A posição, além de ser a direção prevista do retorno do ETF para o próximo dia, indica a compra ou venda do ativo para o final do dia presente, podendo então ser utilizada para computar o retorno obtido pelo algoritmo no dia seguinte, visto que são relacionados na mesma linha da tabela, conforme a Tabela 4, que demonstra o funcionamento do algoritmo.

Tabela 4 - Criação da coluna posição e computação do retorno do algoritmo baseada na previsão do modelo

Período	Entrada (x) Retorno das Ações Ação 1	…	Ação 12	Saída (y) Retorno do ETF previsto no dia seguinte	Posição	Retorno real do ETF no dia seguinte	Retorno do algoritmo
t	Ret(t)	…	Ret(t)	Ret_Prev(t+1)	(0,1)	Ret_Real(t+1)	Posição*Ret_Real(t+1)
t+1	Ret(t+1)	…	Ret(t+1)	Ret_Prev(t+2)	(0,1)	Ret_Real(t+2)	Posição*Ret_Real(t+2)
t+2	Ret(t+2)	…	Ret(t+2)	Ret_Prev(t+3)	(0,1)	Ret_Real(t+3)	Posição*Ret_Real(t+3)
…	…	…	…	…	…	…	…
t+n	Ret(t+n)	…	Ret(t+n)	Ret_Prev(t+n+1)	(0,1)	Ret_Real(t+n+1)	Posição*Ret_Real(t+n+1)

Fonte: Autor.

Observando a Tabela 4 pode-se relacionar que no período de teste, dados os valores de entrada (x), sendo esses os retornos das ações no dia presente (t), o algoritmo faz a previsão para o retorno do ETF (y) no dia seguinte (t+1). Obtendo a direção da previsão, armazenando-a na coluna Posição, pode-se relacionar a decisão de investimento ou não ao final do período em questão. Se por exemplo no período (t), dados os valores (x), o algoritmo previu que (y > 0), então a coluna Posição recebe valor = 1, sendo esse valor multiplicado pelo retorno real do ETF, que por estar uma linha adiantado é referente ao do dia seguinte, desta maneira permitindo computar o retorno obtido pelo algoritmo no próximo dia de negociação. O retorno total de cada modelo refere-se à soma de todos os valores da coluna Retorno do algoritmo na Tabela 4.

3.1. Modelos de Regressão

Os modelos de regressão utilizando os valores de entrada (x) do período (t), buscam as previsões para o valor real do retorno do ETF (y) em (t+1) e após isso então classificá-los de maneira binária (0,1). Importando os dados de retornos do ETF e de suas componentes, os

métodos são implementados chamando os modelos de aprendizagem LinearRegression, Ridge, XGBRegressor e LGBMRegressor correspondentes aos modelos de regressão linear, regressão de crista, *extreme gradient boosting* e *light gradient boosting machine* respectivamente, por meio dos APIs scikit-learn, xgboost e lightgbm.

3.2. Modelos de Classificação

Os modelos de classificação utilizando os valores de entrada (x) buscam prever a direção do retorno do ETF (y) em (t+1) classificando-os de maneira binária $(0,1)$ de acordo com seu sinal. Os retornos reais são convertidos para booleanos visando medir a acurácia da classificação após a implementação dos modelos. Importando os dados de retornos do ETF e de suas componentes, os métodos são implementados chamando os modelos de aprendizagem LogisticRegression, SVC, LinearSVC, RandomForestClassifier, XGBClassifier, LGBMClassifier, KNeighborsClassifier e GaussianNB, correspondentes aos modelos de classificação logística, máquinas de vetores de suporte (SVC e LinearSVC), floresta aleatória, *extreme gradient boosting*, *light gradient boosting machine*, K-ésimos vizinhos mais próximos e *Gaussian Naive Bayes*, respectivamente, por meio dos APIs scikit-learn, xgboost e lightgbm.

Diferente dos métodos de regressão, alguns dos métodos de classificação possuem parâmetros de regularização (C) ou de estado aleatório (random_state) que precisam ser especificados para garantir uma otimização da classificação.

Os valores atribuídos aos parâmetros são generalizados para os modelos que utilizam o parâmetro de regularização C em C=10, definido por experimentação, e o valor para estado inicial aleatório em random_state=0, para o classificador por floresta aleatória. O motivo de especificar o parâmetro random_state evita com que o modelo encontre resultados diferentes para cada vez em que o código for rodado, visto que seus processos de decisão são aleatórios, ele permite estabilidade nas decisões com o mesmo conjunto de dados. Vale mencionar que o parâmetro pode ser alterado, podendo resultar em melhores classificações pelo modelo utilizando estados diferentes que se adequem proveitosamente a cada conjunto de dados, porém visando a análise comparativa, o estado é mantido o mesmo em todos os conjuntos de dados. O modelo KNN não sofreu parametrização, utilizando como default o número de vizinhos $k = 5$.

3.3. Implementação Das Métricas Para Comparação

Os retornos da técnica de *buy & hold* dos ETFs são computados fazendo a soma dos retornos logarítmicos durante todo o período de análise, ou seja, considerando a compra no início do período e a venda dos ativos no fim do período, desta forma permitindo estabelecer uma comparação entre os retornos obtidos pelas múltiplas negociações realizadas pelos modelos implementados.

3.3.1. Naïve Forecast

O modelo de previsão ingênuo é implementado, ao considerar o valor do dia anterior como melhor previsão para o dia presente. Os retornos do dia anterior (t-1) do ETF em análise são considerados como os retornos previstos para o ETF no dia presente (t) modificando as colunas do conjunto de dados conforme a Tabela 5.

Tabela 5 - Representação do modelo de previsão ingênua

Período	ETF	Previsão
t	Retorno(t)	Retorno(t-1)
t+1	Retorno(t+1)	Retorno(t)
t+2	Retorno(t+2)	Retorno(t+1)
…	…	…
t+n	Retorno(t+n)	Retorno(t+n-1)

Fonte: Autor.

Onde a coluna naive_prediction é criada para armazenar os dados de retorno do ETF com atraso de um dia, resultando em um valor faltante para o retorno do ETF para os retornos das componentes no primeiro passo de tempo, sendo então não considerado o primeiro passo do conjunto de dados. O modelo serve como meio comparativo para os erros, em relação aos retornos obtidos pelos modelos, serão comparados à técnica de *buy & hold*.

3.3.2. Métricas Estatísticas

Foi calculado o índice de Sharpe para cada método, considerando a taxa livre de risco como 0 pelo motivo de toda negociação computada ser considerada como investimento de 100% do capital envolvido, tornando assim, 100% do capital é investido, ou, com 100% do capital em caixa, sendo esta definição usada para facilitar a compreensão da obtenção de retornos e redução dos riscos, podendo considerar como 0 o valor do ativo livre de risco r_f no índice de Sharpe, apresentado na equação (14).

$$\mathrm{IS} = \frac{R_A - r_f}{\sigma_A} \tag{14}$$

Os índices de Sharpe IS são calculados utilizando o retorno acumulado obtido pelo modelo R_A e o desvio padrão dos valores do ativo σ_A nos dias em que o algoritmo o teve comprado, não sendo considerados os valores para os dias não negociados, permitindo visualizar o comportamento do risco de acordo com a decisão de negociação de cada algoritmo. Os testes foram pontuados quanto a sua capacidade de prever de acordo com os dados reais e quanto à suas capacidades de classificar os dados.

Os modelos regressivos também foram avaliados quanto a capacidade de prever os valores reais para os retornos, o que está fora do escopo da pesquisa, visto que são buscadas as capacidades preditivas das direções de cada retorno e as capacidades de maximização de retorno e minimização de riscos dos modelos.

Utilizando a seção de métricas do API scikit-learn são importadas as funções e implementadas utilizando os valores obtidos nos treinos e testes de cada modelo para o cálculo dos erros.

A aplicação dos modelos considera que os ativos são negociados em operações de compra ou venda ativamente, tendo como objetivo gerar mais retorno ao negociador ao visar dias de baixa dos ativos e somente negociar nos dias com o retorno previsto positivo. No capítulo seguinte é permitida a visualização do desempenho de cada método nos conjuntos de dados ao observar os retornos obtidos, assim como a noção de risco atrelada ao retorno de cada método, podendo classifica-los como mais ou menos rentáveis e seguros do que à técnica de *buy & hold*. Quanto aos erros algorítmicos, os erros dos modelos poderão ser comparados aos erros da previsão ingênua, permitindo verificar a eficiência computacional individualmente para cada modelo.

4. RESULTADOS DA PESQUISA

As Tabelas 6 e 7 apresentam os resultados obtidos para os modelos de referência para fins de comparação com os resultados obtidos nos métodos implementados.

Tabela 6 - Resultados do Buy & Hold para cada ETF

	Buy & Hold	
	BOVA11	SPY
Retorno log. do treino	0.2581	0.7455
Retorno log. do teste	0.3411	0.4789
Desvio Padrão do treino	0.0146	0.0077
Desvio Padrão do teste	0.0181	0.0132
Índice Sharpe do treino	0.0118	0.0633
Índice Sharpe do teste	0.0188	0.0362

Fonte: Autor

Tabela 7 - Métricas no modelo de previsão ingênua

	Previsão Ingênua	
Erros	BOVA11	SPY
MSE treino	5.71E-04	2.40E-04
MSE teste	7.81E-04	4.28E-04
RMSE treino	2.39E-02	1.55E-02
RMSE teste	2.80E-02	2.07E-02
MAE treino	1.68E-02	9.73E-03
MAE teste	1.80E-02	1.23E-02
Acurácia do treino	0.4906	0.4768
Acurácia do teste	0.4690	0.5000
Precisão do treino	0.4900	0.5226
Precisão do teste	0.4933	0.5560
Pontuação F1 do trein	0.4903	0.5229
Pontuação F1 do teste	0.4928	0.5560
ROC-AUC do treino	0.4906	0.4719
ROC-AUC do teste	0.4678	0.4919

Fonte: Autor

Os retornos logarítmicos dos períodos de treino e teste (Train e Test Log Return) obtidos pela técnica de *buy & hold* (Tabela 6) são avaliados quanto a sua grandeza, sendo um número maior, a representação de mais rentabilidade. Os dados obtidos são adequados para ambos os conjuntos de dados dentro do período de tempo, notando-se assim que a técnica permite uma rentabilidade atraente, principalmente em relação aos retornos do ETF americano, que são maiores em ambos os períodos, permitindo visualizar que há uma maior linearidade de crescimento nas ações de sua composição, enquanto as ações que compõem o ETF brasileiro possuem menor retorno, apesar de estarem atreladas a um maior risco pelas características do mercado nacional brasileiro. Ao observar os conjuntos de dados na Tabela 6, os índices de Sharpe evidenciam este risco nacional, sendo 5 vezes menor que o americano no período de treino. Os desvios padrão também evidenciam a grande variação dos preços dos ativos brasileiros, caracterizando-os como até duas vezes mais voláteis que os dos ativos americanos.

As informações mais importantes fornecidas pelas Tabelas 6 e 7 são os valores obtidos para os períodos de testes, pois são onde os valores são comparados à execução dos modelos

estudados para a validação em ação. As principais informações obtidas dos resultados nestas tabelas são os retornos logarítmicos e o índice de Sharpe do período de testes, sendo respectivamente 34,11% e 0,0188 para o ETF BOVA11, e 47,89% e 0,0633 para o ETF SPY. Os dados de erros são consideráveis, porém por serem complexos de serem compreendidos e visualizados independentemente, precisam dos erros obtidos pelos modelos propostos para que se tornem uma métrica relevante de comparação. As Tabelas 8, 9 e 10 apresentam os resultados obtidos para os modelos implementados para fins de comparação com os resultados obtidos nos métodos de controle, nas Tabelas 6 e 7, e também para comparação entre modelos, possibilitando também a análise de adequação de cada modelo para cada conjunto de dados.

Tabela 8 - Resultados dos modelos de regressão

	Regressão Linear		Regressão de Crista		Regressão XGBoost		Regressão LGBM	
	BOVA11	SPY	BOVA11	SPY	BOVA11	SPY	BOVA11	SPY
Retorno log. do treino	0.8667	0.9841	0.5116	0.7377	8.1935	4.5489	7.7680	4.3051
Retorno log. do teste	0.7984	0.7590	0.4543	0.6359	0.2675	0.1404	-0.2062	0.0702
Desvio Padrão do treino	0.0111	0.0069	0.0118	0.0077	0.0089	0.0046	0.0091	0.0047
Desvio Padrão do teste	0.0128	0.0107	0.0148	0.0123	0.0134	0.6483	0.0152	0.0098
Índice Sharpe do treino	0.0525	0.0927	0.0289	0.0626	0.6176	0.0102	0.5725	0.5998
Índice Sharpe do teste	0.0626	0.0712	0.0307	0.0516	0.0200	0.0138	-0.0136	0.0071
ERROS								
MSE treino	2.11E-04	5.88E-05	2.12E-04	5.90E-05	5.22E-06	9.80E-07	4.29E-05	1.17E-05
MSE teste	3.27E-04	1.73E-04	3.28E-04	1.74E-04	4.39E-04	2.15E-04	3.67E-04	1.97E-04
RMSE treino	1.45E-02	7.67E-03	1.46E-02	7.68E-03	2.28E-03	9.90E-04	6.55E-03	3.42E-03
RMSE teste	1.81E-02	1.32E-02	1.81E-02	1.32E-02	2.10E-02	1.47E-02	1.91E-02	1.40E-02
MAE treino	1.09E-02	5.45E-03	1.09E-02	5.45E-03	1.53E-03	7.02E-04	4.90E-03	2.47E-03
MAE teste	1.20E-02	8.20E-03	1.20E-02	8.18E-03	1.41E-02	9.17E-03	1.28E-02	8.80E-03
Acurácia do treino	0.0086	0.0063	0.0046	0.0018	0.9755	0.9834	0.7984	0.8018
Acurácia do teste	0.0022	0.0063	0.0006	0.0058	-0.3398	-0.2295	-0.1183	-0.1231
Ac. de classificação do treino	0.5043	0.5529	0.4910	0.5477	0.9351	0.9556	0.8562	0.8721
Ac. de classificação do teste	0.5095	0.5485	0.5115	0.5646	0.4975	0.4955	0.4825	0.4935

Fonte: Autor

Apresentados os valores dos erros obtidos pelos modelos de regressão, é permitido avaliá-los quantitativamente ao comparar os valores de cada modelo na Tabela 8 com os valores da previsão ingênua Tabela 7. Ao observar os erros MSE, RMSE e MAE, é permitido notar a eficiência computacional dos modelos de regressão, sendo todos os valores inferiores aos do modelo de controle, com a válida menção dos modelos de regressão linear e de crista, que obtiveram resultados muito inferiores aos da previsão ingênua no conjunto de dados americano.

Tabela 9 - Resultados dos modelos de classificação (Parte 1)

	Regressão Logística		SVM (SVC)		SVM (LinearSVC)		Floresta Aleatória	
	BOVA11	SPY	BOVA11	SPY	BOVA11	SPY	BOVA11	SPY
Retorno log. do treino	0.6767	0.7220	4.1192	2.6827	0.7745	0.8690	8.2498	4.5750
Retorno log. do teste	0.8020	0.6847	0.5165	0.2816	0.8262	0.7063	0.5037	0.2012
Desvio Padrão do treino	0.0103	0.0075	0.0098	0.0059	0.0103	0.0073	0.0089	0.0046
Desvio Padrão do teste	0.0136	0.0117	0.0145	0.0116	0.0139	0.0114	0.0132	0.0096
Índice Sharpe do treino	0.0439	0.0627	0.2806	0.2991	0.0504	0.0781	0.6233	0.6537
Índice Sharpe do teste	0.0592	0.0584	0.0358	0.0243	0.0596	0.0618	0.0381	0.0209
PONTUAÇÕES								
Acurácia do treino	0.5284	0.5470	0.7398	0.7604	0.5378	0.5620	0.9980	1.0000
Acurácia do teste	0.5315	0.5566	0.5175	0.5135	0.5335	0.5475	0.5265	0.5075
Precisão do treino	0.5287	0.5496	0.7652	0.7237	0.5378	0.5613	0.9960	1.0000
Precisão do teste	0.5553	0.5646	0.5412	0.5571	0.5562	0.5631	0.5532	0.5578
Pontuação F1 do treino	0.5175	0.6998	0.7259	0.8065	0.5309	0.6976	0.9980	1.0000
Pontuação F1 do teste	0.5456	0.7013	0.5338	0.6042	0.5519	0.6848	0.5284	0.5788
ROC-AUC do treino	0.5284	0.5025	0.7397	0.7444	0.5378	0.5236	0.9980	1.0000
ROC-AUC do teste	0.5313	0.5038	0.5170	0.4925	0.5328	0.5009	0.5276	0.4941

Fonte: Autor

Tabela 10 - Resultados dos modelos de classificação (Parte 2)

	Classificação XGBoost		Classificação LGBM		KNN		Gaussian NB	
	BOVA11	SPY	BOVA11	SPY	BOVA11	SPY	BOVA11	SPY
Retorno log. do treino	8.2498	4.5750	8.2498	4.5488	3.4260	1.8636	0.5476	0.8385
Retorno log. do teste	0.3578	0.3604	0.2296	0.4813	0.8851	0.2467	0.5222	0.7350
Desvio Padrão do treino	0.0089	0.0046	0.0089	0.0046	0.0099	0.0056	0.0095	0.0064
Desvio Padrão do teste	0.0134	0.0104	0.0128	0.0097	0.0117	0.0094	0.0384	0.0099
Índice Sharpe do treino	0.6233	0.6537	0.6233	0.6521	0.2309	0.2184	0.0059	0.0850
Índice Sharpe do teste	0.0268	0.0345	0.0180	0.0497	0.0758	0.0262	0.0885	0.0745
PONTUAÇÕES								
Acurácia do treino	0.9980	1.0000	0.9980	0.9980	0.6997	0.6906	0.5338	0.5490
Acurácia do teste	0.5115	0.5135	0.5115	0.5395	0.5445	0.5165	0.4945	0.5285
Precisão do treino	0.9960	1.0000	0.9960	0.9976	0.7089	0.7002	0.5463	0.5685
Precisão do teste	0.5354	0.5653	0.5385	0.5842	0.5726	0.5707	0.6118	0.5668
Pontuação F1 do treino	0.9980	1.0000	0.9980	0.9982	0.6918	0.7298	0.4533	0.6414
Pontuação F1 do teste	0.5271	0.5752	0.5081	0.6062	0.5445	0.5691	0.1708	0.6211
ROC-AUC do treino	0.9980	1.0000	0.9980	0.9980	0.6996	0.6830	0.5335	0.5290
ROC-AUC do teste	0.5111	0.5032	0.5131	0.5266	0.5459	0.5092	0.5149	0.5059

Fonte: Autor

No período de treino, os modelos utilizaram os dados históricos para aprender como se comportar no período de testes, sendo este período então, onde os modelos foram validados. Alguns modelos fazem previsões com alta acurácia no período de treinos que podem ser visualizados pela Acurácia de treino, onde os modelos XGBoost, LGBM para ambas as tarefas de regressão e classificação e o modelo de classificação por Floresta Aleatória atingiram pontuação de 1 ou próximas a 1, garantindo retornos próximos a casa de 824% dentro de 1496 dias para o conjunto de dados brasileiro, e de 457% dentro de 1533 dias para o conjunto de dados americano.

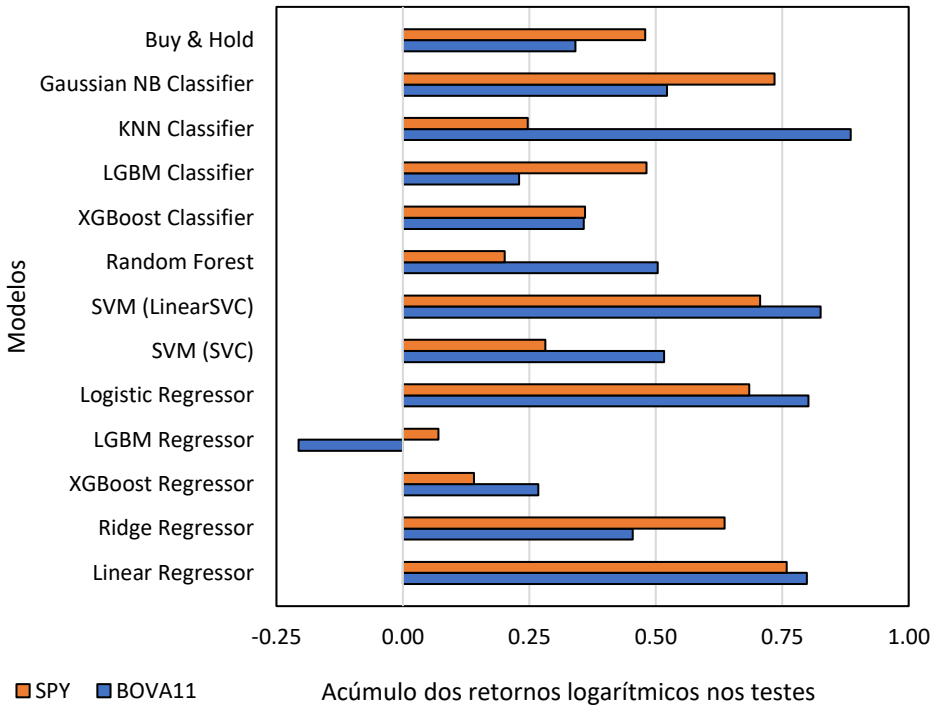
Com os erros e pontuações de todos os modelos apresentados, os modelos podem ser comparados, podendo notar a semelhança entre as abordagens de alguns métodos. Nos modelos de regressão é percebido um melhor desempenho quanto aos valores de erros quando comparados aos obtidos pela previsão ingênua, permanecendo inferiores em todos os parâmetros correspondendo aos respectivos conjuntos de dados. Os valores das pontuações dos modelos de classificação apresentam resultados superiores em todas as categorias quando comparados à previsão ingênua, com a exceção do modelo *Gaussian Naive Bayes* que apresentou resultado inferior na pontuação F1, no conjunto de dados brasileiro. Os erros dos

modelos de regressão sendo todos inferiores aos erros da previsão ingênua, e as pontuações dos modelos de classificação sendo todas superiores às da previsão ingênua (com exceção do *Gaussian Naive Bayes* em uma pontuação), denotam uma eficiência numérica e algorítmica dos modelos em relação à capacidade computacional, permitindo notar que ocorrem menos erros e mais acertos nos modelos testados do que no modelo de controle.

A pontuação de acurácia dos modelos mede a diferença entre os valores reais e previstos, tornando impraticável a comparação com os modelos de classificação, para isso, os valores previstos nos modelos regressivos foram convertidos à valores booleanos e então mensurados quanto à classificação binária da direção dos retornos previstos, sendo apresentados como Acurácia de classificação na Tabela 8, permitindo a comparação com os valores de Acurácia nas Tabelas 9 e 10 dos modelos de classificação. Então, considerando a acurácia de previsão para todos os modelos, os modelos de regressão apresentaram resultados semelhantes aos modelos de classificação, sendo evidenciada a ineficácia dos modelos XGBoost e LightGBM para as tarefas de regressão, os quais apresentaram resultados substancialmente inferiores aos da previsão ingênua especialmente no conjunto de dados americano. O restante dos modelos garantiu acurácia superior aos da previsão ingênua em todos os conjuntos de dados.

Para que os dados fornecidos pelas tabelas sejam visualizados e compreendidos de maneira mais didática, os gráficos plotados em barras presentes nas Figuras 12, 13, 14 e 15 apresentam os dados mais relevantes para as análises, onde as Figuras 12 e 13 apresentam respectivamente os retornos e os índices de Sharpe obtidos para cada modelo, a Figura 14 apresenta os resultados para os erros médios absolutos nos modelos de regressão, e a Figura 15 apresenta um comparativo entre os modelos quanto a acurácia na classificação da direção dos retornos.

Figura 12 – Retorno dos modelos e do *buy & hold* no período de testes



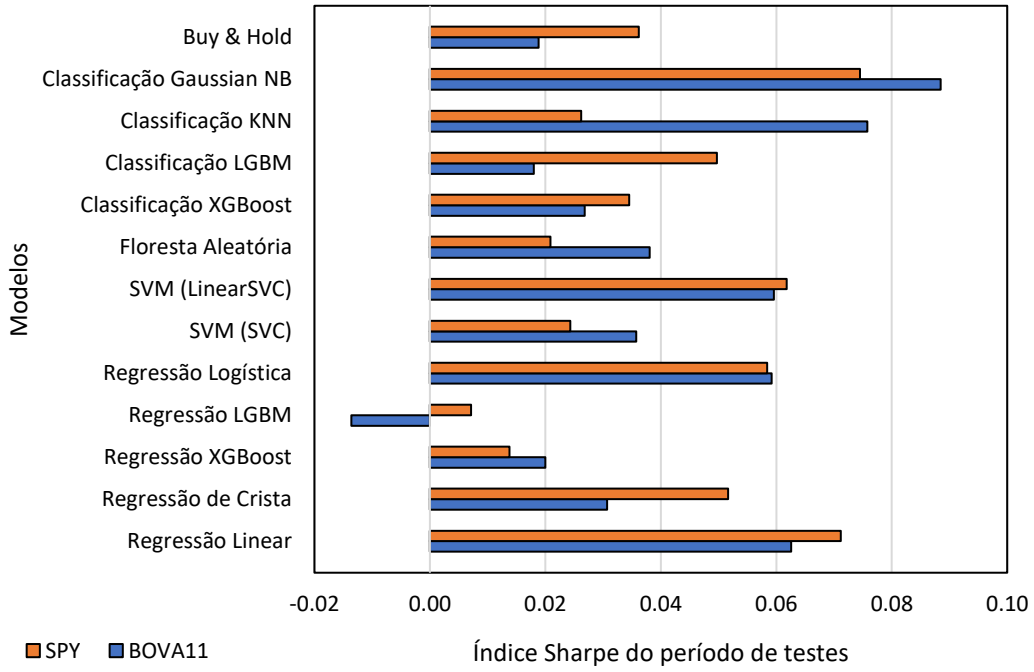
Fonte: Autor

Os retornos logarítmicos plotados na Figura 12 apresentam o desempenho de cada modelo dentro do período de testes, evidenciando a eficácia de cada modelo ao prever a direção

dos retornos futuros com os dados fornecidos. A eficácia dos modelos no conjunto de dados americano ao prever a direção dos retornos do ETF SPY, onde os modelos de regressão linear (LinearRegression) e de crista (RidgeRegression) e os modelos de classificação logística (LogisticRegression), *linear support vector classifier* (LinearSVC) e *Gaussian Naive Bayes* (GaussianNB) apresentaram retornos substancialmente superiores aos da técnica de *buy & hold.*

Já para a eficácia dos modelos no conjunto de dados brasileiro quanto à previsão da direção dos retornos do ETF BOVA11, apenas três dos doze modelos apresentaram retornos menores que a técnica de *buy & hold*, modelos os quais todos fazem uso de árvores de decisão, que são eles, XGBoost para regressão, LightGBM para regressão (único modelo a apresentar retorno negativo) e LightGBM para classificação. Obtendo retornos de mais que o dobro do *buy & hold* na negociação do ETF BOVA11, ficam evidentes os modelos de regressão linear (LinearRegression) e os modelos de classificação logística (LogisticRegression), *linear support vector classifier* (LinearSVC) e k-ésimos vizinhos mais próximos (KNN).

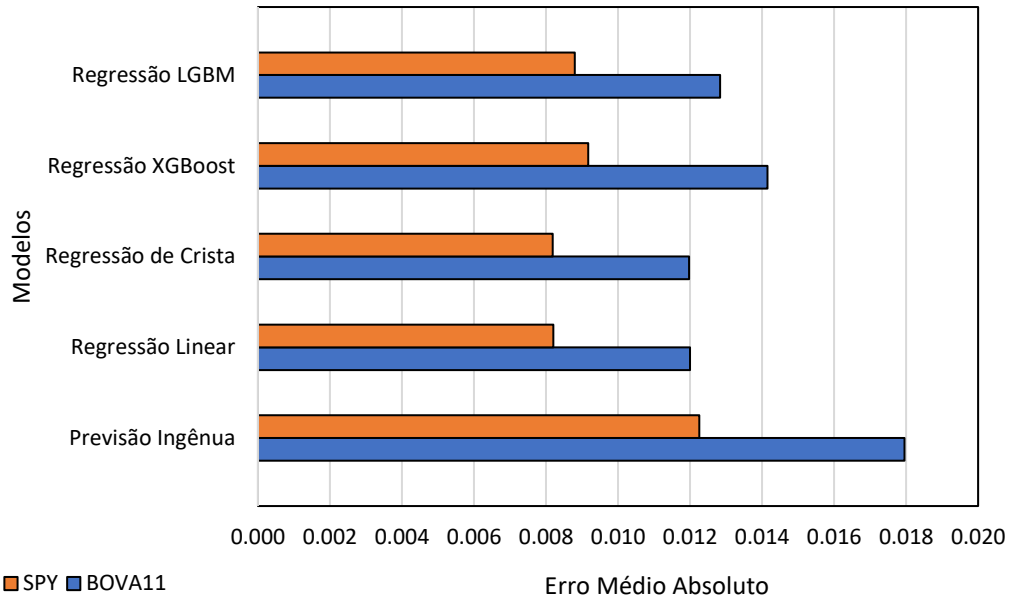
Figura 13 – Índice de Sharpe de cada modelo e do *buy & hold* no período de testes



Fonte: Autor

Os índices de Sharpe medem o excesso de rendimento por unidade de risco, neste caso, quanto maior seu valor, melhor, indicando que há menos risco atrelado ao período onde o ativo estava em carteira. Os modelos que obtiveram retornos acima do *buy & hold* fizeram o mesmo para o índice, com evidência ao modelo *Gaussian Naive Bayes* que apresentou valor quatro vezes superior ao *buy & hold*. Esse índice indica que além de obter retornos superiores, alguns modelos também permitem uma maior segurança nos investimentos ao reduzir o risco.

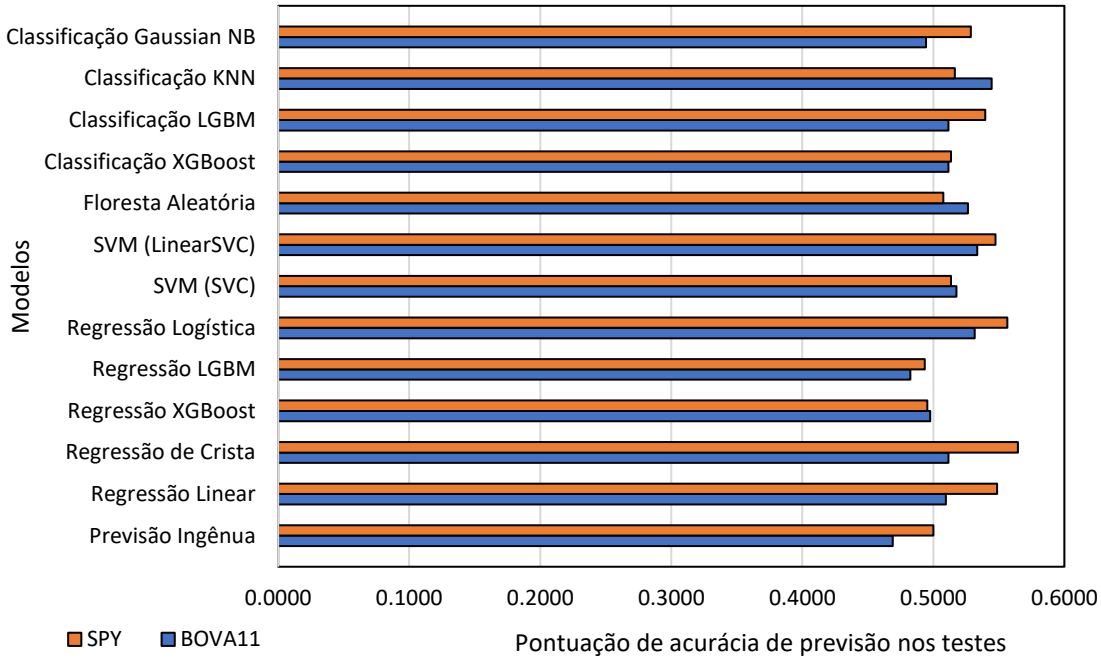
Figura 14 – Erros médios absolutos de cada modelo regressivo e para a previsão ingênua no período de testes.



Fonte: Autor

Os erros médios absolutos (MAE) apresentados na Figura 14 medem os desvios entre os valores previstos e os reais de cada modelo e conjunto de dados. Nota-se que modelos de regressão possuem erro menor que o modelo de controle da previsão ingênua, denotando maior eficiência numérica, pois os valores previstos possuem maior proximidade que os reais.

Figura 15 – Pontuação da acurácia de classificação dos modelos



Fonte: Autor

Para a real avaliação de todos os modelos quanto a classificação leva-se em consideração a pontuação dada para a acurácia de classificação dos modelos, sendo apresentada nas Tabelas 7, 9 e 10 como Acurácia do teste e na Tabela 8 como Acurácia de classificação do

teste. A pontuação de 0 a 1 avalia a habilidade do modelo em classificar os dados fornecidos de maneira binária, que é o escopo deste projeto. A pontuação de acurácia dos modelos é compreendida de maneira que 1 corresponde a todas as previsões feitas sendo corretas, 0 sendo todas as previsões incorretas. A Figura 15 apresenta as pontuações de classificação para cada modelo. Os modelos de regressão por árvores de decisões (XGBoost e LightGBM) apresentaram pontuações abaixo da previsão ingênua no conjunto de dados americano.

Notavelmente os modelos apresentam desempenho levemente superior a 50% quanto a acurácia de classificação, sendo um resultado que pode intuir ao descarte dos métodos, porém ao relacionar a acurácia dos modelos (Figura 14) aos retornos (Figura 12) e índices de Sharpe (Figura 13) obtidos pelos algoritmos, pode-se relacionar quanto ao momento de tomada de decisão de cada algoritmo permitindo uma análise quantitativa da acurácia, onde os modelos quando erraram ao ordenar a compra, obtiveram baixas perdas de retornos e quando erraram ao ordenar a venda, apenas deixaram de ganhar.

Permitida a visualização dos modelos de maneira comparativa dos retornos, riscos, erros e eficácia na classificação dos dados, é possível que sejam tomadas conclusões quanto aos modelos individualmente e a pesquisa realizada como um todo, visando serem utilizados como ferramentas de auxílio para a tomada de decisão no ramo de negociação de ativos ou até mesmo de maneira automatizada dentro do ambiente de negociação.

5. CONCLUSÃO

Os resultados obtidos com a proposta de aprendizagem de máquina analisaram a direção de retornos de *Exchange Traded Funds* (ETFs) e permitiram a visualização da ação dos algoritmos perante os dados fornecidos, auxiliando na tomada de decisões em estratégias de negociação ativa de ativos por meio de algoritmos.

Em um primeiro momento, nos resultados das pontuações dos modelos, que são um comparativo entre os valores previstos e os valores reais, o modelo ingênuo, de referência, permite notar a grande eficácia de alguns modelos na classificação da direção dos retornos. Durante o período de treinos fica visível a efetividade dos modelos que fazem uso de árvores de decisões (Floresta Aleatória, XGBoost e LightGBM), denotando um possível *overfitting* dos dados. *Overfitting* que não se justificou ao observar as pontuações de acurácia dos modelos usados para classificação no período de treinos, permanecendo estáveis como o restante dos modelos de classificação, levemente acima de 0,5. O *overfitting* é justificado para os dados obtidos nos modelos XGBoost e LightGBM quando utilizados para regressão, pois apresentam grande pontuação de acurácia no período de treinos e desempenho mediano ou ruim de ambos os modelos quanto à obtenção de retornos.

A pontuação generalizada de 0,5 na acurácia dos modelos, pode ser compreendida de maneira diferente observando as informações dispostas nas Figuras 12 e 13. Permitindo concluir que apesar de todos os modelos ficarem próximos à acurácia de classificação de 50%, os modelos que obtiveram retornos e índices de Sharpe superiores aos do *buy & hold* foram capazes de evitar grandes perdas de retornos. Isso significa que os modelos quando erraram em ordenar a compra do ativo, obtiveram retornos negativos mínimos, e quando erraram ao ordenar a venda, apenas deixaram de ganhar, evidenciando o acerto do momento de negociações dos modelos que apresentaram retornos consideráveis (regressão linear, logística e LinearSVC).

Os erros numéricos avaliados pelas métricas de erros MSE, RMSE e MAE, com ênfase nos modelos de regressão (Tabela 8), apresentaram desempenho positivo, sendo inferiores aos do modelo de referência da previsão ingênua (Tabela 7), assim como os dados obtidos para os modelos de classificação (Tabelas 9 e 10), que atingiram valores superiores aos de referência nas pontuações de acurácia, precisão, F1 e de ROC e AUC, demonstrando a eficiência numérica

dos modelos. Dado o exposto, os modelos apresentaram capacidade numérica superior ao modelo de controle no geral.

Ao avaliar os retornos obtidos e relacionando-os aos índices de Sharpe, foi permitido selecionar alguns modelos que performaram de maneira superior em cada conjunto de dados. Em relação aos retornos obtidos no conjunto de dados brasileiro, sobressaem-se oito dos doze modelos selecionados, os quais obtiveram resultado melhor tanto na avaliação de riscos quanto em retornos obtidos em relação à técnica de controle do *buy & hold*. Faz-se necessária a menção dos modelos LinearSVC, *Logistic Regressor* e *Linear Regressor*, que obtiveram os maiores retornos de ambos os conjuntos de dados, superando em mais de duas vezes os retornos do *buy & hold* no conjunto de dados brasileiro, assim como apresentando um índice de Sharpe até três vezes maior. Já para o conjunto de dados americano, 50% dos modelos apresentaram retornos e índices de Sharpe superiores ao do *buy & hold*, com retornos reais (fora da base logarítmica) ultrapassando os 113% no modelo de regressão linear, porém os retornos são mais modestos comparados aos do conjunto de dados brasileiro. Isso se dá também pelo fato da técnica de *buy & hold* performar melhor no mercado americano em termos de retornos, mostrando uma maior linearidade e crescimento das empresas componentes do ETF SPY. Demonstrada a dificuldade dos modelos performarem melhor no mercado americano, pode-se também justificar pela Tabela 6 e Figuras 10 e 11, que demonstram a menor volatilidade (variação nos retornos) e uma maior linearidade de crescimento do que o mercado brasileiro.

Os principais resultados evidenciaram que a aprendizagem de máquina é uma ferramenta válida para a previsão de dados financeiros, podendo auxiliar na tomada de decisões quanto à compra e venda de ativos no mercado financeiro visando maximizar o retorno e minimizar o risco. A previsão da direção dos retornos garantiu à maioria dos modelos um retorno superior ao do *buy & hold*, também demonstrando menores erros algorítmicos, além de maior segurança no investimento perante os dados de controle. Os algoritmos foram capazes de identificar padrões de comportamento com grau de confiança acima de 50%, visto que esse grau pode ser ainda ampliado utilizando mais dados de entrada para que o algoritmo leve em consideração variáveis de maior relevância, além de uma melhor parametrização individual de cada modelo e um tratamento de dados adequado a cada modelo, sendo necessárias avaliações empíricas para que se perceba a atuação do algoritmo quanto aos parâmetros e dados testados.

Os modelos que obtiveram retornos e acurácias inferiores aos dados de controle (XGBoost e LightGBM) podem ser melhorados com um tratamento dos dados adequado aos modelos, além de uma melhor parametrização visando encontrar um valor ótimo para que se tornem ferramentas que também obtenham retornos superiores as de controle assim como os modelos linear, logístico e linearSVC. O baixo desempenho desses modelos neste estudo não os classificam como ferramentas inaptas, mas ferramentas que necessitam de uma melhor implementação, onde seu desempenho fica refém dos parâmetros e dados fornecidos.

Para estudos futuros, recomenda-se a inclusão de dados de indicadores financeiros, taxa de juros, dados de análises técnicas dos ativos, volumes de negociações, dados de mercados estrangeiros, commodities, inclusão de análise de redes sociais quanto à ativos em empresas, podendo atrelar todas essas variáveis em uma Rede Neural Artificial (Artificial Neural Network - ANN) para que se verifique a possibilidade de aumentar a acurácia das previsões.

REFERÊNCIAS

ALBON, Chris. **Machine Learning with Python Cookbook**: practical solutions from preprocessing to deep learning. Sebastopol: O'reilly Media, Inc., 2018. 366 p.

ASHFAQ, Nazish; NAWAZ, Zubair; ILYAS, Muhammad. **A comparative study of Different Machine Learning Regressors for Stock Market Prediction.** arXiv preprint arXiv:2104.07469, 2021.

AUFFARTH, Ben. **Machine Learning for Time Series with Python**: forecast, predict, and detect anomalies with state-of-the-art machine learning methods. Birmingham: Packt Publishing Ltd., 2021. 502 p.

BENNINGA, S. **Financial modeling**: with a section on visual basic for applications by benjamin czaczkes. 4. ed. rev. Cambridge, Massachussetts: The MIT Press, 2014. 1143 p.

BHARATHI, Shri; GEETHA, Angelina. **Sentiment analysis for effective stock market prediction**. International Journal of Intelligent Engineering and Systems, v. 10, n. 3, p. 146-154, 2017.

BIONDO, Alessio Emanuele; PLUCHINO, Alessandro; RAPISARDA, Andrea; HELBING, Dirk. **Are Random Trading Strategies More Successful than Technical Ones?** 2013. 13 f. Monografia (Doutorado) - Curso de Economia, Departamento de Economia e Negócios, Universitá di Catania, Catania, 2013.

BROOKS, C. **Introductory econometrics for finance.** 2. ed. Cambridge, Massachussetts: Cambridge University Press, 2008. 674 p.

B3 BOLSA BRASIL BALCÃO. **Investidor Pessoa Física** 2021. Disponível em: https://www.b3.com.br/pt_br/noticias/porcentagem-de-investidores-pessoa-fisica-cresce-na-b3.htm. Acesso em: 18 mar. 2022.

CAMPBELL, J. Y.; LO, A. W.; MACKINLAY, A. C. **The econometrics of financial markets**. Princeton: Princeton University Press, 1996. 611 p.

CAPINSKI, M.; ZASTAWNIAK, T. **Mathematics for finance**: an introduction to financial engineering. Londres: Springer, 2003. 321 p.

COMISSÃO DE VALORES MOBILIÁRIOS, Portal do Investidor. **Estrutura do Sistema Financeiro Nacional - SFN**. 2021. Disponível em: https://www.gov.br/cvm/pt-br. Acesso em: 05 jan. 2022.

COMISSÃO DE VALORES MOBILIÁRIOS, Portal do Investidor. **Investidores no mercado de capitais brasileiro**: Uma análise dos critérios regulatórios para investimento em valores mobiliários 2021. Disponível em: https://www.gov.br/cvm/pt-br/centrais-de-conteudo/publicacoes/estudos/air_investidores-no-mercado-de-capitais-brasileiro_2021-07-19.pdf. Acesso em: 18 mar. 2022.

DAVENPORT, Thomas H.; BEAN, Randy. **The Pursuit of AI-Driven Wealth Management**. 2021. Disponível em: https://sloanreview.mit.edu/article/the-pursuit-of-ai-driven-wealth-management/. Acesso em: 04 jan. 2022.

FRANKE, J.; HÄRDLE, W. K.; HAFNER, C. M. **Statistics of financial markets**: an introduction. 2. ed. Berlim: Springer, 2008. 508 p.

GRAHAM, Benjamin; DODD, David L. **Security Analysis**. 6. ed. New York: McGraw Hill, 2008. 818 p.

HARRELL, Frank E. **Regression Modeling Strategies**: with applications to linear models, logistic and ordinal regression, and survival analysis. 2. ed. Nashville: Springer, 2015. 598 p.

HASTIE, T; TIBSHIRANI, R; FRIEDMAN, J. **The elements of statistical learning: data mining, inference, and prediction**. New York: springer, 2009. 764p.

KE, Guolin. Microsoft Research (org.). **LightGBM's documentation**. 2016. Originalmente criada por Guolin Ke. Disponível em: https://lightgbm.readthedocs.io/en/latest/Features.html. Acesso em: 27 fev. 2022.

KUMAR, Indu et al. **A comparative study of supervised machine learning algorithms for stock market trend prediction.** In: 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT). IEEE, 2018. p. 1003-1007.

MALKIEL, Burton G. **A Random Walk Down Wall Street**: the time-tested strategy for successful investing. 12. ed. New York: W. W. Norton & Company, 2019. 423 p.

MASÍS, Serg. **Interpretable Machine Learning with Python**: learn to build interpretable high-performance models with hands-on real-world examples. 2. ed. Birmingham: Packt Publishing Ltd., 2021. 737 p.

MORDOR INTELLIGENCE. **Algorithmic Trading Market**: growth, trends, covid-19 impact, and forecasts (2022 - 2027). Growth, Trends, Covid-19 Impact, And Forecasts (2022 - 2027). 2021. Disponível em: https://www.mordorintelligence.com/industry-reports/algorithmic-trading-market. Acesso em: 06 jan. 2022.

MÜLLER, Andreas C.; GUIDO, Sarah. **Introduction to Machine Learning with Python**: a guide for data scientists. 2. ed. Sebastopol: O'reilly Media, Inc., 2017. 392 p.

NABIPOUR, Mojtaba et al. **Predicting stock market trends using machine learning and deep learning algorithms via continuous and binary data; a comparative analysis.** IEEE Access, v. 8, p. 150199-150212, 2020.

NATARAJAN, Gopinath. **AI in investing is about human empowerment, not displacement**. 2021. Disponível em: https://economictimes.indiatimes.com/markets/stocks/news/ai-in-investing-is-about-human-empowerment-not-displacement/articleshow/83685863.cms?from=mdr. Acesso em: 04 jan. 2022.

PARDO, Robert. **The Evaluation and Optimization of Trading Strategies**. 2. ed. Hoboken: John Wiley & Sons, Inc, 2008. 367 p.

PRING, Martin J. **Technical Analysis Explained**: the successful investor's guide to spotting investment trends and turning point. 5. ed. New York: McGraw Hill, 2014. 814 p.

RASCHKA, Sebastian; MIRJALILI, Vahid. **Python Machine Learning**: machine learning and deep learning with python, scikit-learn, and tensorflow. 2. ed. Birmingham: Packt Publishing Ltd., 2017. 622 p.

SAHA, Sumit. **XGBoost vs LightGBM**: how are they different. How Are They Different. 2022. Disponível em: https://neptune.ai/blog/xgboost-vs-lightgbm. Acesso em: 27 fev. 2022.

SHARMA, Abhishek. **Decision Tree vs. Random Forest**: which algorithm should you use?. Which Algorithm Should You Use?. 2020. Disponível em: https://www.analyticsvidhya.com/blog/2020/05/decision-tree-vs-random-forest-algorithm/. Acesso em: 20 fev. 2022.

TSAY, R. S. **Analysis of financial time series**. 2. ed. Chicago: John Wiley & Sons, Inc., 2005. 638 p.

YANG, Yue et al. **Stock Price Prediction Based on XGBoost and LightGBM.** In: E3S Web of Conferences. EDP Sciences, 2021. p. 01040.

ZHENG, Alice; CASARI, Amanda. **Feature Engineering for Machine Learning**: principles and techniques for data scientists. Sebastopol: O'reilly Media, Inc., 2018. 217 p.

APÊNDICE

APÊNDICE A – Código fonte da extração dos dados, implementação dos modelos testados e de controle.

O código-fonte do projeto encontra-se publicado no repositório GITHUB e conta com acesso público. A URL do repositório é <https://github.com/RaphaelPiovezan/ETF-direction-prediction/>.