Universidade Federal de Santa Catarina
Departamento de Informática e Estatística


Felipe Longarai Trisotto

# Modelo Analítico para Gerenciamento de Portfólio

Florianópolis
2024

**Felipe Longarai Trisotto**

# Modelo Analítico para Gerenciamento de Portfólio

Trabalho de Conclusão de Curso apresentado à Universidade Federal de Santa Catarina como parte dos requisitos necessários para a obtenção do Grau de Bacharel em Ciências da Computação.
Orientador: Prof. Dr. Elder Rizzon Santos

Universidade Federal de Santa Catarina
Departamento de Informática e Estatística

Florianópolis
2024

**Felipe Longarai Trisotto**

# Modelo Analítico para Gerenciamento de Portfólio

Trabalho de Conclusão de Curso apresentado à Universidade Federal de Santa Catarina como requisito parcial para a obtenção do grau de "Bacharel em Ciências da Computação".

**Comissão Examinadora**

---

Prof. Dr. Elder Rizzon Santos
Universidade Federal de Santa Catarina
Orientador

---

Profa. Dra. Andréa Cristina Konrath
Universidade Federal de Santa Catarina

---

Prof. Me. José Eduardo De Lucca
Universidade Federal de Santa Catarina

Florianópolis, 9 de julho de 2024

Dedico este trabalho a todos aqueles que, de alguma forma,
auxiliaram para a concretização desta etapa.

# Agradecimentos

A conclusão deste TCC do curso de Ciências da Computação representa um marco significativo em minha vida, e não poderia deixar de expressar minha gratidão àqueles que estiveram ao meu lado durante toda essa jornada.

Primeiramente, agradeço aos meus pais, Charles e Elisângela, pelo apoio emocional incondicional e pela confiança em minha capacidade, especialmente nos momentos de tropeços e incertezas. Sua fé em mim foi crucial para que eu pudesse superar os desafios e seguir em frente, mesmo nas horas mais difíceis.

Ao meu colega veterano João Paulo T.I.Z, minha imensa gratidão por ter virado inúmeras noites na sala do PET e BU, me auxiliando nos primeiros projetos quando eu ainda era calouro. Sua paciência e disposição para ajudar foram fundamentais para o meu desenvolvimento acadêmico.

Ao Gustavo Olegário, que mesmo após ter se formado, esteve sempre disponível para me salvar nas horas de maior desespero técnico. Sua generosidade e dedicação nunca serão esquecidas.

À Salomão R. Jacinto e Nikolas Martins, que sempre me incentivaram a continuar a jornada, independente dos momentos difíceis. Suas palavras de encorajamento foram uma fonte constante de motivação.

À Victor Goulart, cuja amizade proporcionou momentos inesquecíveis de risadas e diversão. Além disso, sua prontidão em pedir despensa do trabalho para ajudar um amigo enfermo mostrou o verdadeiro valor da amizade.

À George Goulart, sou grato por abrir as portas para o mercado de trabalho, confiando no meu potencial e me ensinando os primeiros passos na minha carreira. Sua orientação foi essencial para o meu crescimento profissional.

Agradeço também a todos os professores que acreditaram em mim, mesmo quando eu não via o meu próprio potencial. Em especial, agradeço a Rafael de Santiago, Mauro Roisenberg, Leandro José Komosinski (in memoriam), Jean Everson Martina, Maicon Rafael Zatelli, Márcio Bastos Castro, Patrícia Vilain e Renato Fileto. Suas aulas e orientações foram determinantes para a minha formação.

À banca avaliadora, composta por Andréa Cristina Konrath e José Eduardo De Lucca, agradeço pelas correções precisas que possibilitaram extrair o melhor deste trabalho.

Por fim, ao meu orientador Elder Rizzon Santos, agradeço por estar sempre presente, disponibilizando inúmeras horas de sua atenção para me orientar da melhor forma possível e se desdobrando para encontrar a melhor forma de abordar as dificuldades encontradas no caminho.

A todos vocês, meu mais sincero muito obrigado. Este trabalho é, também, fruto do apoio e dedicação de cada um de vocês.

*"Sometimes it is the people no one can imagine anything of who do the things no one can imagine."*
(Alan Turing)

# Resumo

No âmbito de finanças e investimentos sabe-se que o mercado de capitais é o meio mais inclusivo para se investir em negócios de sucesso, construir e preservar patrimônio ao longo do tempo, principalmente com foco previdenciário. Na situação atual, aonde grande parte da população possui pouca educação financeira para gerir seus investimentos de forma autônoma, um modelo capaz de gerar recomendações automatizadas a partir da análise do perfil do investidor e dados financeiros dos ativos disponíveis no mercado pode ser vista como uma forma de facilitar o processo de gerenciar investimentos de maneira que o usuário foque em seu desenvolvimento pessoal e profissional. Neste trabalho, foram desenvolvidos e avaliados modelos preditivos para gerar um portfólio de investimento previdenciário alinhado ao perfil do usuário, buscando o melhor retorno possível dentro dos níveis de volatilidade permitidos. Após uma análise abrangente do estado da arte em ciência de dados e séries temporais e a coleta e preparação de dados fundamentalistas históricos de ações da B3, foram testados diversos modelos preditivos utilizando as bibliotecas PyCaret e Prophet. O modelo AdaBoost com Cond. Deseasonalize & Detrending, desenvolvido com PyCaret, foi identificado como o de melhor desempenho, superando significativamente o modelo Prophet em todas as métricas avaliadas (MAE, RMSE e MAPE). Embora o Prophet tenha demonstrado leve superioridade em termos de precisão prática, suas previsões não foram consistentes. Concluiu-se que o modelo PyCaret é mais adequado para previsão de P/L da ação ABEV3, destacando a necessidade de escolher modelos adequados e realizar avaliações rigorosas para alcançar previsões confiáveis no mercado financeiro. Sugere-se, para trabalhos futuros, explorar outros modelos de aprendizado de máquina, otimização de hiperparâmetros, inclusão de mais variáveis fundamentalistas e a realização de estudos comparativos com diferentes horizontes de previsão.

**Palavras-Chave:** 1. Investimentos. 2. Modelo Analítico. 3. Aprendizado estatístico. 4. Aprendizado de máquina. 5. Inteligência artificial.

# Abstract

In the field of finance and investments, it is known that the capital market is the most inclusive way to invest in successful businesses, build and preserve assets over time, especially with a social security focus. In the current situation where a large part of the population has little financial education to manage their investments autonomously, a model capable of generating automated recommendations from the analysis of the investor profile and financial data of the assets available in the market can be seen as a way to facilitate the process of managing investments so that the user can focus on their personal and professional development. In this work, predictive models were developed and evaluated to generate a retirement investment portfolio aligned with the user's profile, seeking the best possible return within the permitted volatility levels. After a comprehensive analysis of the state of the art in data science and time series, and the collection and preparation of fundamental historical data of B3 stocks, various predictive models were tested using the PyCaret and Prophet libraries. The AdaBoost model with Cond. Deseasonalize & Detrending, developed with PyCaret, was identified as the best performing, significantly outperforming the Prophet model in all evaluated metrics (MAE, RMSE, and MAPE). Although Prophet showed slight superiority in practical precision, its predictions were not consistent. It was concluded that the PyCaret model is more suitable for predicting the P/E of ABEV3 stock, highlighting the need to choose appropriate models and conduct rigorous evaluations to achieve reliable forecasts in the financial market. It is suggested, for future work, to explore other machine learning models, hyperparameter optimization, inclusion of more fundamental variables, and conducting comparative studies with different forecasting horizons.

**Keywords:** 1. Investments. 2. Analytical Model. 3. Statistical learning. 4. Machine learning. 5. Artificial intelligence.

# Lista de figuras

# Lista de tabelas

# Lista de Siglas e Abreviaturas

| | |
|---|---|
| UFSC | *Universidade Federal de Santa Catarina* |

# Sumário

# 1 Introdução

Desde o início do século XVII, com as primeiras emissões de títulos, dívidas e ações para custear as grandes navegações de companhias como a Companhia Holandesa das Índias Orientais[1], essa se mantém uma das principais formas de investimento de capital mundial inclusive no Brasil, que teve início após a criação da Bolsa de Valores Bahia Sergipe Alagoas (BOVESBA), que se encontra desativada hoje[2].

Com a evolução dos mecanismos de integração, como home brokers e internet banking, que facilitaram o acesso de muitos investidores a essa categoria de ativo e a constante queda da taxa básica de juros (Taxa Selic) que alcançou patamares de juros reais negativos, criou uma migração de muitos pequenos investidores para produtos de renda variável a procura por maiores rentabilidades como alternativa a produtos de renda fixa que não mais representavam um ganho adequado com anteriormente.

Entretanto, hoje temos um pouco mais de 91,5 milhões (IBGE 2010) de pessoas economicamente ativas no Brasil e apesar desse número apenas 3 milhões, pouco mais de 3%, possuem investimentos registrados na B3 (Brasil, Bolsa, Balcão), única bolsa de valores em funcionamento no Brasil, sendo os dois dos principais fatores para isso a complexidade para investir e administrar recursos em ativos de risco e a ausência de educação financeira em todos os âmbitos desde ensino fundamental até superior.

Nos últimos anos, a ciência de dados surgiu como uma disciplina nova e importante sendo uma mescla de disciplinas, como estatística, mineração de dados, bancos de dados e sistemas distribuídos[3]. Ciência de dados é o estudo da extração de conhecimento a partir de dados, um requisito básico para avaliar se o novo conhecimento é útil para a tomada de decisão e seu poder preditivo, não apenas sua habilidade de explicar o passado[4].

Dentro do âmbito do mercado de capitais, temos a análise fundamentalista um método para medir o valor intrínseco de títulos por meio do estudo de indicadores macro e microeconômicos e posteriormente comparar com preço de mercado do mesmo[5]. Durante essas análises, indicadores fundamentalistas são amplamente usados por analistas, são índices de desempenho calculados a partir de dados administrativos e contábeis fornecidos publicamente que variam entre indicadores de lucratividade, liquidez, alavancagem e avaliativos[6].

Tendo esse cenário em perspectiva, podemos unir essa necessidade à possibilidade que estudos de máquina e modelos analíticos no campo da ciência de dados, que tem apresentado rápido avanço por meio novas bibliotecas em linguagens como Python e R, para captar, modelar e analisar a ampla quantidade de dados fundamentalistas disponíveis ao público para apoiar o gerenciamento de investimentos de forma simples e prática, possibilitando assim o investidor iniciante a dar seus primeiros passos e escapar de vieses com-

portamentais comuns a investidores iniciantes, como viés de confirmação e ancoragem[7].

Sendo assim, as próximas etapas dessa pesquisa serão, primeiramente, apresentar o tema de investimento focado para pessoas físicas e como modelos analíticos podem ser usados nessa área. Em seguida, será realizado um estudo de estratégias para análise de perfil e correlação entre retorno e volatilidade para algumas categorias de investimento, além da captura dos indicadores fundamentalistas para os mesmos. E por fim, será realizado o desenvolvimento do modelo capaz de recomendar um portfólio adequado ao perfil do investidor.

## 1.1 Objetivos

### 1.1.1 Objetivo Geral

Desenvolver um modelo analítico capaz de gerar um portfólio de investimento previdenciário que melhor se adéqua ao perfil do usuário e apresente o melhor retorno possível nos níveis de volatilidade permitidos pelo perfil.

### 1.1.2 Objetivos Específicos

1. Analisar o estado da arte em ciência de dados e series temporais.
2. Realizar coleta e tratamento dos dados para análise.
3. Realizar experimentos para testar o modelo.
4. Analisar os resultados obtidos com os experimentos.

## 1.2 Metodologia

Para o melhor desenvolvimento deste trabalho, com objetivo de produzir um modelo analítico capaz de cumprir os requisitos propostos inicialmente, será utilizado o método de pesquisa *Design Science Research*. Seguindo as seguintes etapas:

1. Avaliar técnicas de aprendizado de máquina e representação de dados quanto ao potencial representativo em análises fundamentalista.
2. a) Utilizar fontes de dados confiáveis e acessíveis.
   b) Desenvolver script utilizando APIs e frameworks de web-crawling para coletar e padronizar os dados coletados.
3. a) Analisar os dados coletados manualmente e desenvolver um script para extrair as informações mais relevantes para o modelo.

b) Desenvolver um script que gere um modelo analítico capaz de analisar os dados e gerar um retorno desejado.

c) Executar estudos de máquina para aperfeiçoar o retorno.

4. a) Realizar execuções do modelo com amostras de validação.

   b) Validar a qualidade dos retornos obtidos.

5. a) Analisar os resultados obtidos com os experimentos.

   b) Plotar os resultados de forma didática para o usuário final.

O modelo será desenvolvido pelo autor utilizando seu desktop, serviços de armazenamento e computação em nuvem e outros softwares como o GitHub, RStudio, Jupyter Notebook. Por fim, será utilizado ferramentas como PowerBI e Ploty, mediante a necessidade de melhores recursos para apresentação dos resultados obtidos pelos experimentos.

### 1.2.1 Problema e Motivação

A complexidade do ambiente de investimento e a diversidade de regimes de pensões e produtos disponíveis dificultam determinar a carteira de investimento ideal. A falta de aconselhamento personalizado geralmente leva a decisões de investimento abaixo do ideal, resultando em retornos insatisfatórios e exposição excessiva ao risco. Portanto, é necessário um modelo analítico que possa alavancar informações específicas do usuário para criar portfólios de planos de investimento que correspondam a metas financeiras, tolerâncias de risco e horizontes de tempo. A motivação por trás deste projeto decorre da crescente importância do planejamento de aposentadoria e do desejo de fornecer aos indivíduos ferramentas eficazes para tomar decisões de investimento informadas. Modelos analíticos robustos, capazes de criar portfólios de planejamento de aposentadoria otimizados, auxiliam os usuários a maximizar o acúmulo de riqueza a longo prazo, gerenciando o risco adequadamente. Ao proporcionar o maior retorno possível do investimento em um nível aceitável de volatilidade, o modelo aumenta a segurança financeira e melhora a aposentadoria de indivíduos de diversos perfis e origens.

### 1.2.2 Objetivos da solução

Avaliar diversos modelos analíticos e técnicas para identificar a abordagem mais adequada para gerar uma carteira de investimentos para um plano de previdência alinhada ao perfil do usuário.

1. Utilizar bases de pesquisa abrangentes, como bancos de dados financeiros, periódicos acadêmicos e relatórios do setor, para reunir informações e pesquisas relevantes sobre otimização de portfólio de investimentos.

2. Analisar dados específicos do usuário, incluindo metas financeiras, tolerância a riscos, horizonte de tempo e requisitos de renda durante a aposentadoria, para extrair as informações mais relevantes para a geração de portfólio.

3. Desenvolva uma tabela de dados organizada que incorpore dados históricos do mercado, desempenho da classe de ativos e condições de mercado predominantes para avaliar o impacto na otimização do portfólio.

4. a) Utilizar ferramentas e software de biblioteca que permitem a aplicação consistente e personalizável de modelos analíticos selecionados.

   b) Identificar e extrair variáveis-chave para cada modelo analítico selecionado para otimizar o processo de geração de portfólio de investimentos.

   c) Implementar métodos para representação e validação de integridade para garantir a aplicação precisa de cada modelo analítico.

5. Apresentar resultados quantitativos e criar visualizações para demonstrar a diferença no desempenho do portfólio e nos perfis de risco entre vários modelos analíticos.

Ao atingir os objetivos da solução, o projeto visa analisar e desenvolver um modelo analítico que efetivamente gere carteiras de investimentos para planos de previdência. O uso de bases de pesquisa abrangentes, análise de dados específicos do usuário e implementação de métodos de avaliação robustos contribuirão para o desenvolvimento de um portfólio de investimentos personalizado e otimizado que se alinhe com as metas financeiras do usuário, tolerância ao risco e retorno do investimento desejado.

# 2 Fundamentação Teórica

Neste capítulo, serão abordados os temas fundamentais que sustentam a análise e comparação de técnicas de análise de dados no contexto da geração de carteiras de investimentos para planos de previdência. Dada a natureza interdisciplinar deste projeto, abrangendo disciplinas como finanças, estatística e inteligência artificial, serão discutidos os conceitos fundamentais subjacentes a essas aplicações. O capítulo começa com uma visão geral de conceitos gerais sobre perfil do investidor brasileiro, seguida de uma discussão sobre as configurações relacionadas aos métodos de aprendizado estatístico que funcionam como classificadores no processo de geração de portfólio. Além disso, serão apresentadas as principais características a serem avaliadas após a aplicação desses métodos, garantindo uma compreensão abrangente de sua eficácia no alcance dos objetivos do projeto.

## 2.1 O Perfil do Investidor Brasileiro

De forma geral, o perfil do brasileiro como investidor é conservador, com a escolha de aplicações financeira mais simples e populares. Ou seja, a caderneta de poupança ainda é a aplicação preferida e mais popular entre os brasileiros. De acordo com Lampenius e Zickar (2005)[8] grande parte dos estudos sobre a análise do perfil do investidor considera fatores que podem ser influenciais. Segundo Lampenius e Zickar (2005)[8], o foco recai em relação às características como gênero, idade, estado civil, profissão, renda, escolaridade e conhecimento sobre a parte financeira. Portando, com isso justifica-se a escolha do tema.

Longe de querer criticar as práticas que divergem do modus operandi de uma orientação para investimento de cunho profissional, reitera-se que dentro desse tipo de orientação, é indispensável o conhecimento de causa, a familiaridade com o comportamento do mercado e o reconhecimento da impossibilidade de se fazer mágica quando o assunto em pauta são as finanças de uma pessoa ou empresa.

Nesse ponto, cabe dizer que a orientação profissional ou mesmo a autoaprendizagem para ação neste segmento é uma ciência, um modelo de gestão que como todo processo de gerenciamento, está passível de equívocos, caso o comprometimento profissional ocorra de forma superficial ou leviana. A gestão nesse sentido, deixa implícita a necessidade de entendimento teórico sobre o tema finanças, abarcando toda a multiplicidade que esse tema possui.

Junto a isso, é indispensável também a compreensão quanto ao fato de que, uma orientação para realização de investimentos, ocorre fundamentada na familiaridade e domínio de técnicas que vão além dos macetes que habitam o senso comum. Logo, as máximas que indicam um padrão de atuação arrojado e destemido, a constante exposição a riscos

e a crença e validação de uma ação completamente intuitiva que não considera pontos essenciais de um mercado com grande oscilação, representam tão somente a possibilidade maximizada de erros na realização de um investimento financeiro.

Desta feita, é saudável reiterar que a consideração quanto à eficácia de assimilação de conhecimento para a realização de investimentos, considera o conhecimento histórico e entendimento macro sobre a atual condição financeira de quem busca por esse tipo de serviço. Assim, é importante dizer que o diagnóstico e prognóstico sobre esse tipo de situação, precisa ocorrer de forma singularizada, dando ênfase nessa análise, ao perfil de cada um.

Com isso, chega-se ao entendimento de que um dos princípios de uma ação investidora realmente coesa, habita na validação de unicidade dos casos que se tem à frente.

Desse modo, fica mais fácil compreender que, dizer que mais dinheiro é a solução para todos os problemas, é um pecado mortal quando se trata de responsabilidade financeira, de crescimento real e também da aproximação de uma pessoa rumo a sua meta estipulada, isso porque é importante considerar que há casos e casos, assim, a injeção de valores visando sanar uma falha monetária representa tão somente um comportamento leigo, que segue desconhecendo a raiz de um problema maior que voltará à tona em um momento posterior, possivelmente, com maior grau de criticidade.

Coloca-se em ênfase que a frieza da análise do mercado e evidenciando a aplicabilidade de tal conceito como fator consideravelmente útil para a redução de riscos para os investidores. Os autores explanam tal situação expondo que, a análise de previsibilidade como ratificador de uma postura confiável, funciona como um viés utilitário para ambos os lados, uma vez que para os investidores, há uma probabilidade de significativa diminuição de obtenção de prejuízos; à quem requer crédito, em caso de negativa, resta a oportunidade de ajuste comportamental como forma de recuperação da confiança econômica[8].

Nesse sentido, o ato de pensar a respeito de uma orientação financeira exige de quem se propõe a refletir sobre essa temática, o entendimento de que a busca por um serviço que auxilie na recuperação e/ou organização financeira de um indivíduo, representa antes de tudo o reconhecimento da necessidade de se ter ao alcance, um profissional com conhecimento de causa e com disposição para enfrentamento de uma determinada situação.

Nesse sentido, é importante dizer que quando se quer realizar um investimento, analisa-se primeiramente que os ensinamentos que se busca são de aplicabilidade prática, e que requerem disciplina e entendimento específicos para que se consiga ter êxito em um modelo econômico que requer do investidor uma ação totalmente atenta a todos os pormenores que circundam essa esfera.

Fica então subtendido que o conjunto de comportamentos e decisões éticas de uma entidade ou nação, seguem sendo fatores relevantes para a consolidação da confiança econômica. Mais que isso, mesmo com o conceito de previsibilidade, a confiança não perde a sua importância, pelo contrário, ela é o instrumento que ratifica o selo de bom

pagador de um país e que por sua vez assinala a probabilidade de êxito em uma negociação.

Pensar na consolidação do mercado nacional nesse sentido, requer a observação de que em um cenário econômico onde a política de concorrência acontece de forma bem estruturada, é relevante e saudável para a economia loca, tendo em vista que é esse tipo de ação, que pode fomentar e incentivar as ações, voltadas para o reconhecimento de que as legislações que se encontram vigentes no cenário brasileiro, ratificam o comportamento já vigente em grandes economias, fazendo com que a diversidade de ações mercadológicas, ajudem a aquecer a economia local, fazendo com que diretamente, todos possa interagir economicamente, de forma saudável e com grande potencial de edificação da cultura econômica local.

## 2.2 Técnicas de Inteligência Artificial Aplicadas ao Mercado Financeiro

Nos últimos anos, o setor financeiro testemunhou uma onda transformadora com a integração de técnicas de Inteligência Artificial (IA) e Aprendizado de Máquina (ML). Estas tecnologias tornaram-se fundamentais na análise de vastos conjuntos de dados, na identificação de padrões e na realização de previsões com uma precisão sem precedentes. No contexto do mercado financeiro, IA e ML oferecem ferramentas poderosas para otimização de portfólio, gestão de riscos, detecção de fraudes e modelagem preditiva.

IA refere-se à simulação da inteligência humana em máquinas programadas para pensar, aprender e resolver problemas. Já ML é um subconjunto de IA que se concentra no desenvolvimento de algoritmos que permitem que os sistemas aprendam e melhorem com a experiência. Em vez de serem programados explicitamente, os modelos de ML usam dados para melhorar iterativamente seu desempenho em uma tarefa específica. No setor financeiro, a IA abrange um amplo espectro de aplicações, que vão desde sistemas baseados em regras até redes neurais avançadas.

No setor financeiro, as aplicações de IA e ML são diversas e impactantes. A negociação algorítmica se destaca como um caso de uso proeminente, onde modelos de aprendizado de máquina analisam dados históricos do mercado para identificar padrões e executar negociações em momentos ideais, capturando ineficiências do mercado e respondendo a condições dinâmicas. O gerenciamento de portfólio é outro caso que se beneficia da IA/ML ao otimizar a alocação de ativos, considerando o desempenho histórico, a tolerância ao risco e as tendências do mercado para construir portfólios que maximizam os retornos e minimizam o risco. Além disso, a IA/ML desempenha um papel vital na detecção de fraudes, analisando padrões de transações e comportamento do usuário para evitar atividades fraudulentas.

No domínio da análise preditiva, várias técnicas são empregadas, incluindo análise fun-

damentalista, técnica e de sentimento. A análise fundamentalista envolve a avaliação do valor intrínseco de um título, examinando demonstrações financeiras, indicadores econômicos e desempenho da empresa. A análise técnica concentra-se em padrões históricos de preços e volumes, utilizando gráficos e ferramentas estatísticas para prever movimentos futuros de preços. A análise de sentimento avalia o sentimento do mercado analisando notícias, mídias sociais e outras fontes para avaliar o humor geral e prever tendências do mercado.[9]

Estas aplicações contribuem coletivamente para um ecossistema financeiro eficiente e orientado por dados, revolucionando as práticas tradicionais e moldando o futuro dos mercados financeiros.

### 2.2.1 Previsões do Mercado de Ações

O panorama da análise e previsão do mercado financeiro apresenta desafios intrigantes, mesmo com a crescente acessibilidade aos dados. Apesar dos avanços, a aquisição e o processamento de dados para extrair informações valiosas, especialmente no que diz respeito ao seu impacto nos preços das ações, continua a ser uma tarefa formidável. A extração de características de dados financeiros introduz complexidade, exigindo uma consideração cuidadosa de diversas variáveis cruciais para a previsão. Os conjuntos de dados financeiros, muitas vezes caracterizados por ruído, influenciam significativamente as previsões do mercado de ações.

A aplicação em tempo real de metodologias propostas anteriormente enfrenta desafios em testes ao vivo devido a fatores como variações de preços, ruído e eventos imprevistos, como exemplificado pela Tragédia da Knight Capital, resultando em uma perda substancial de US$ 440 milhões para a empresa[10]. A volatilidade, impulsionada pela incerteza e pela inflação, acrescenta outra camada de complexidade à previsão dos preços das ações, com eventos como o flash crash que destruiu 860 mil milhões de dólares dos mercados de ações dos EUA em 30 minutos. A volatilidade do mercado intensifica-se durante as vendas de pânico desencadeadas por factores como especulação, questões políticas e instabilidade econômica.

O fluxo contínuo de novos algoritmos nos mercados, muitas vezes mantidos em sigilo para manter a eficácia, acrescenta uma camada de complexidade na avaliação da sua precisão e eficácia. A ascensão da negociação algorítmica e o seu impacto no comportamento do mercado, especialmente durante eventos como vendas de pânico, colocam desafios aos pesquisadores. Além disso, o fluxo de dados das plataformas de redes sociais, influenciados tanto por humanos como por bots, introduz complexidades na análise de sentimentos para previsão de ações. A detecção de bots sociais torna-se crucial para previsões precisas, exemplificadas por eventos como a invasão da conta do Twitter da Associated Press pelo Exército Eletrônico Sírio, causando uma quebra imediata do mercado.

A natureza multifacetada da análise de sentimentos baseada em dados de redes sociais, influenciada por fatores como notícias falsas e conteúdo gerado por bots, sublinha os desafios na identificação de dados de qualidade. Para equilibrar esses desafios, os relatórios trimestrais ou anuais das empresas surgem como ativos valiosos para a previsão de ações, fornecendo insights sobre o status de uma organização quando decodificados com precisão. Assim, enfrentar estes desafios requer uma abordagem diferenciada, combinando algoritmos sofisticados, considerações éticas e uma compreensão das complexidades dos mercados financeiros.

A integração de técnicas de IA/ML no mercado financeiro revolucionou a negociação de ações, proporcionando aos investidores a flexibilidade de participar em negociações online a partir de qualquer dispositivo ligado à internet. Esta mudança tecnológica não só mudou a forma como as ações são compradas e vendidas, mas também transformou os mercados financeiros num mercado globalmente interligado. Como resultado, os indivíduos podem agora testemunhar o crescimento dos seus investimentos com a evolução das tecnologias financeiras.

Neste cenário em evolução, as previsões do mercado de ações fizeram a transição de estruturas convencionais para metodologias avançadas, aproveitando as técnicas de AI/ML. Estas tecnologias melhoram os processos de tomada de decisão, tornando as previsões mais otimizadas e eficientes. No entanto, esta evolução também introduz vulnerabilidades, com os mercados suscetíveis a sentimentos das redes sociais e a ataques cibernéticos. Os pesquisadores desempenham um papel crucial no desenvolvimento de tecnologias para um comércio seguro e melhorado nestas condições dinâmicas.

## 2.3 Series Temporais

Uma série temporal é uma coleção de variáveis aleatórias indexadas de acordo com a ordem em que foram obtidas no tempo. Normalmente, assumimos que essas variáveis aleatórias são medições ou observações feitas em pontos igualmente espaçados no tempo.[11]

De acordo com Brockwell, os componentes de uma série temporal são uma tendência, um componente sazonal e flutuações irregulares. A tendência representa um movimento de longo prazo, a componente sazonal capta padrões regulares e as flutuações irregulares são variações inexplicáveis nos dados.[11] Juntamente com a ciclos externos irregulares que podem afetar a série temporal da mesma forma, principalmente em análises econômicas e comerciais. A Figura 1 apresenta uma representação gráfica das componentes de uma série temporal:

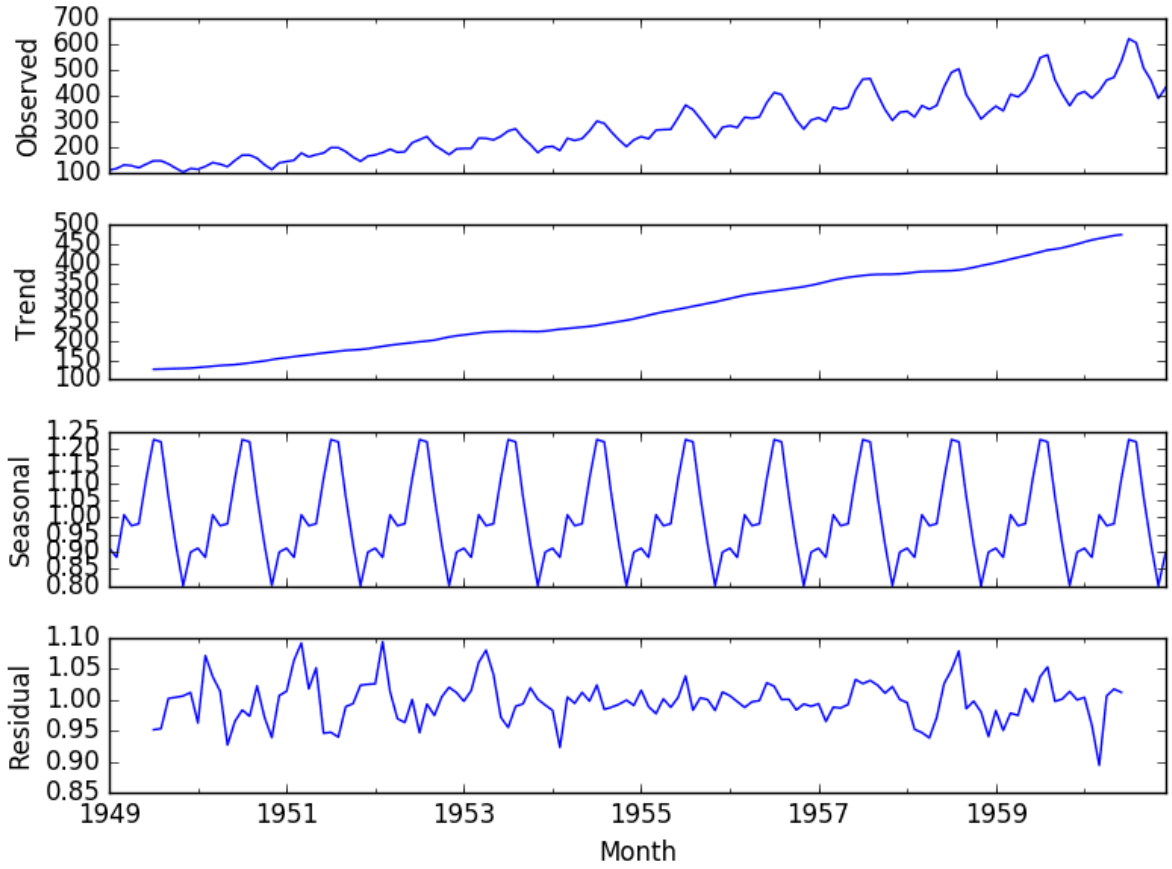



Figura 1 – Exemplo dos componentes em uma série temporal

- **Tendência**: Representa o movimento de longo prazo na série temporal. É a direção-geral na qual os dados estão se movendo ao longo de um período prolongado, podendo ser crescente, decrescente ou constante. Uma tendência crescente indica um aumento contínuo nos valores ao longo do tempo, como o crescimento populacional ou a inflação, enquanto uma tendência decrescente mostra uma diminuição contínua nos valores, como o declínio nas vendas de um produto obsoleto. A tendência pode ser identificada plotando a série temporal e observando o comportamento geral dos dados, utilizando métodos estatísticos como suavização ou ajuste de linha de regressão para isolá-la.

- **Sazonalidade**: Refere-se a padrões ou flutuações regulares e repetitivas em um período fixo, como um ano, um mês ou um dia, causados por fatores sazonais ou periódicos. Esses padrões são visíveis em eventos como aumentos de vendas de sorvete durante o verão ou compras de presentes durante as festas de fim de ano. A sazonalidade pode ser identificada mediante gráficos de séries temporais onde padrões repetitivos são evidentes. Análise de Fourier ou decomposição de séries temporais são úteis para isolar essa componente.

- **Ciclicidade**: Refere-se a movimentos ascendentes e descendentes em uma série

temporal que não ocorrem em intervalos regulares, mas são influenciados por ciclos econômicos, industriais ou outros tipos de ciclos. Esses movimentos ocorrem em intervalos irregulares e estão frequentemente associados a fases econômicas como expansão, recessão, depressão e recuperação. A ciclicidade pode ser identificada por meio da análise de longo prazo de séries temporais, onde os padrões de subida e descida não seguem um intervalo regular.

- **Erro aleatório ou ruído**: É a variação residual na série temporal que não pode ser explicada pela tendência, sazonalidade ou ciclicidade. Representa a variabilidade aleatória que ocorre em cada ponto do tempo, não seguindo nenhum padrão ou estrutura definida. Geralmente, é de menor amplitude em comparação com os outros componentes, mas pode impactar a precisão das previsões. O ruído pode ser identificado após a remoção dos componentes de tendência, sazonalidade e ciclos da série temporal, utilizando análises estatísticas como a Análise de Componentes Principais (PCA) para entender melhor a natureza do ruído.

Os principais objetivos da análise de séries temporais são reconhecer a natureza do fenômeno representado pela sequência de observações e ajustar um modelo que represente a estrutura a ser usada para previsão ou estimativa de quantidades relacionadas.[12] Esses objetivos podem ser divididos em:

- **Descrição**: A etapa inicial na análise de séries temporais envolve traçar os dados e derivar medidas descritivas simples para compreender suas principais propriedades, como tendências e efeitos sazonais. Para algumas séries, modelos básicos são suficientes para descrever essas variações, enquanto outras requerem modelos mais complexos, como processos estocásticos.

- **Explicação**: Quando variações de múltiplas variáveis são observadas ao longo do tempo, é possível usar uma série para explicar as variações de outra, levando a uma compreensão mais profunda do mecanismo subjacente. Modelos de regressão múltipla e análise de sistemas lineares são úteis neste contexto.

- **Previsão**: Prever valores futuros de uma série temporal é crucial em vários campos, como previsão de vendas e análise econômica. Os termos "predição"e "previsão"são frequentemente usados de forma intercambiável, embora alguns autores os distingam com base em métodos subjetivos versus objetivos.

- **Controle**: Em cenários de controle de qualidade, a análise de série temporal auxilia no controle dos processos. Os procedimentos de controle variam desde controle estatístico de qualidade com cartas de controle até estratégias mais sofisticadas envolvendo modelos estocásticos para previsão e ajuste de variáveis de entrada para manter a qualidade do processo.

Também é possível observar que algumas séries temporais não oscilam no mesmo nível e não possuem padrões sazonais. Desse modo, a análise de séries temporais pode também ser classificada de outras duas formas com base no modelo usado para a análise, sendo elas:

- **Determinística**: Uma série temporal determinística é aquela que pode ser descrita por uma função matemática sem qualquer componente aleatório. Em outras palavras, os valores futuros da série podem ser previstos com exatidão a partir de uma fórmula específica e tendo características como previsibilidade e estrutura fixa, como projeções de retornos fixos em investimentos com taxas de juros garantidas. Uma série temporal determinística pode também ser identifica por sua inflexibilidade e sensibilidade a mudanças que são suas limitações.

- **Estocástica ou Não-Determinística**: Uma série temporal estocástica ou não-determinística inclui um componente aleatório e, portanto, não pode ser prevista com precisão absoluta. Os valores futuros da série são influenciados por fatores aleatórios e incertezas e possui características como imprevisibilidade, havendo sempre uma margem de erro, e estrutura variável, como preços das ações no mercado financeiro. Uma série temporal determinística estocástica possue limitações devido a sua complexidade para desenvolvimento de modelos e difícil interpretação.

Compreender as diferenças e aplicações de cada abordagem facilita previsões mais precisas e tomadas de decisão informadas. Enquanto os modelos determinísticos são úteis para séries com padrões claros e previsíveis, os modelos estocásticos são essenciais para lidar com a incerteza e variabilidade presentes na maioria das séries temporais reais. A escolha entre modelos determinísticos e estocásticos depende das características dos dados e dos objetivos da análise.

# 3 Trabalhos Relacionados

## 3.1 Uma revisão sistemática da análise fundamentalista e técnica das previsões do mercado de ações

Esta revisão abrangente investiga as complexidades da previsão do mercado de ações, navegando por conceitos fundamentais e metodologias modernas. A discussão começa explorando a imprevisibilidade do mercado de ações, introduzindo a Hipótese do Passeio Aleatório (RWH) e a Hipótese do Mercado Eficiente (EMH). Embora estas hipóteses afirmem que o mercado é inerentemente estocástico e, portanto, não previsível, o artigo desafia esta noção examinando o papel do conhecimento fundamentalista e técnico.[13]

A análise reconhece que, apesar da posição da EMH contra a previsibilidade baseada em dados históricos, os mercados emergentes podem oferecer oportunidades para previsões mais precisas. A economia comportamental e as perspectivas socioeconômicas contribuem ainda mais para o argumento de que existe alguma previsibilidade, desafiando a eficiência absoluta da HME.[13]

O artigo faz a transição para o domínio do aprendizado de máquina, um ramo da inteligência artificial, como ferramenta de previsão do mercado de ações. Vários algoritmos de aprendizado de máquina são explorados no contexto de modelos preditivos, com dados fundamentalistas e técnicos servindo como conjuntos de dados de entrada cruciais. A revisão enfatiza a importância do pré-processamento de dados para melhorar o desempenho e a precisão do modelo.[13]

Ao discutir os dados de entrada, é destacada a distinção entre dados quantitativos (estruturados) e qualitativos (não estruturados). Os preços históricos das ações, as notícias financeiras na web, os dados de sentimento das redes sociais e as variáveis macroeconômicas são identificados como componentes-chave que influenciam as previsões. A etapa de pré-processamento envolve remoção de ruído, verificações de consistência de dados, seleção de recursos, transformação de dados e normalização.[13]

O artigo apresenta uma série de métricas de desempenho para avaliar modelos de previsão, incluindo coeficiente de correlação, raiz do erro quadrático médio (RMSE), erro percentual médio absoluto (MAPE), erro absoluto médio (MAE) e muito mais. Essas métricas fornecem uma avaliação abrangente da exatidão, precisão, recall e F-score do modelo.[13]

São reconhecidas revisões sistemáticas sobre previsão do mercado de ações, enfatizando a necessidade de uma análise comparativa considerando tipos de dados de entrada, fontes de dados, técnicas e porcentagens de conjuntos de dados de treinamento/teste. A revisão da literatura abrange uma ampla gama de estudos, incluindo análise de sentimento, redes

neurais artificiais (RNA) e metodologias de mineração de texto.[13]

A seção de desenho de pesquisa descreve o processo de seleção de 122 ensaios relevantes entre 2007 e 2018. Os artigos são categorizados com base nos tipos de dados de entrada (textuais, históricos ou combinados) e posteriormente analisados quanto à precisão, prazo e pacotes de software usados para modelagem. O quadro de investigação prepara o terreno para um exame detalhado destas categorias.[13]

Na seção de resultados e discussões é apresentada a distribuição da literatura, mostrando a prevalência da análise técnica (66%) sobre a análise fundamental (28%) e abordagens combinadas (13%). O artigo aprofunda então as descobertas específicas de cada categoria, detalhando as tendências, fontes de dados e modelos preditivos empregados.[13]

A configuração empírica envolve uma experiência prática utilizando dados do mercado de ações disponíveis publicamente na Bolsa de Valores do Gana. Três algoritmos proeminentes de aprendizado de máquina (árvores de decisão, máquinas de vetores de suporte e redes neurais artificiais) são comparados com base em métricas de desempenho. Os resultados confirmam a alta previsibilidade dos movimentos do mercado de ações, com o modelo de Rede Neural Artificial apresentando precisão superior em comparação com Árvores de Decisão e Máquinas de Vetores de Suporte.[13]

O resumo termina com insights sobre o futuro da pesquisa de previsão do mercado de ações. Enfatiza a necessidade de estudos que incorporem variáveis de entrada comportamentais e fundamentalistas, comparando técnicas de conjuntos em diferentes continentes e abordando restrições do mundo real, como derrapagens e custos de negociação. A revisão posiciona-se como um recurso valioso para investigadores e profissionais que pretendem navegar no cenário dinâmico da previsão do mercado de ações.[13]

## 3.2 Modelos de redes neurais para seleção de ações com base em análise fundamentalista

O artigo investiga o domínio da predição financeira, concentrando-se especificamente na aplicação de arquiteturas de redes neurais, incluindo redes neurais feed-forward (FNN) e sistemas de inferência neural fuzzy adaptativos (ANFIS). O objetivo principal da pesquisa é avaliar a eficácia dessas arquiteturas na previsão de movimentos de ações com base em índices financeiros fundamentalistas. O contexto da negociação de ações é apresentado como um processo complexo influenciado por inúmeros fatores, tornando a previsão precisa uma tarefa desafiadora. O estudo posiciona-se contra a hipótese do mercado eficiente (HME), que afirma que os preços das ações refletem toda a informação disponível e são, portanto, imprevisíveis.[14]

A introdução prepara o terreno, destacando a natureza complexa da previsão do mercado de ações e o debate em curso sobre a eficiência dos mercados. A HME é apresentada

como pano de fundo, destacando a dificuldade de prever os preços das ações dada a afirmação da hipótese de que todas as informações relevantes já estão incorporadas aos preços correntes. O estudo desafia esta noção ao introduzir a aplicação de técnicas de aprendizado de máquina, especificamente redes neurais, no domínio da análise fundamentalista. Salienta a importância dos rácios financeiros fundamentalista em oposição aos preços históricos ou aos indicadores técnicos, alinhando-se com o objectivo mais amplo de imitar o processo de tomada de decisão dos especialistas em investimento.[14]

Passando para o cerne do estudo, desdobra-se uma análise comparativa, avaliando o desempenho do FNN e do ANFIS na previsão de ações. O critério de avaliação é o retorno relativo das carteiras selecionadas em relação a um índice de ações de referência. Os resultados revelam que tanto o FNN como o ANFIS apresentam a capacidade de distinguir vencedores e perdedores dentro do universo de ações, com as carteiras selecionadas superando consistentemente o benchmark. No entanto, o estudo afirma que a FNN demonstra desempenho superior ao ANFIS neste contexto. Embora o resumo não forneça métricas quantitativas específicas, a ênfase está na capacidade prática dos modelos de superar o índice de mercado, desafiando assim o EMH.[14]

O artigo contribui para o panorama mais amplo da pesquisa de previsão financeira, concentrando-se no uso de aprendizado de máquina, especificamente redes neurais, para seleção de ações com base na análise fundamentalista. A relevância desta investigação reside no seu afastamento das abordagens tradicionais que muitas vezes se baseiam em preços históricos ou indicadores técnicos. Em vez disso, defende um processo de tomada de decisão mais realista, incorporando rácios financeiros fundamentalistas. A conclusão do estudo não só destaca o desempenho superior da FNN, mas também sublinha as implicações mais amplas do emprego da aprendizagem automática no domínio da previsão do mercado de ações. À medida que o cenário financeiro continua a evoluir, esta investigação fornece informações valiosas sobre o potencial dos modelos de redes neurais na melhoria da tomada de decisões e na previsão de movimentos de ações com base na análise fundamentalista.[14]

## 3.3 Comparando indicadores técnicos e fundamentalista na previsão do preço das ações

O estudo avalia a eficácia da análise fundamentalista e técnica, e sua combinação, na previsão de preços de ações usando modelos de aprendizado de máquina. Com base em testes realizados em 140 empresas do S&P 500, a pesquisa conclui que os modelos que utilizam indicadores de análise fundamentalista geralmente superam aqueles que utilizam indicadores técnicos, com graus variados de desempenho superior entre os setores. Notavelmente, a combinação de ambas as abordagens resulta num erro quadrático médio

(RMSE) inferior em mais de 95% dos casos, em comparação com a utilização isolada de indicadores fundamentalistas ou técnicos.[15]

A Hipótese do Mercado Eficiente (EMH) afirma que os preços das ações seguem um passeio aleatório, tornando impossível o lucro sustentado da previsão. Em contraste, a Hipótese Adaptativa do Mercado (AMH) sugere previsibilidade nos preços das ações. Este artigo explora a concorrência e a combinação de análises técnicas e fundamentais na previsão de preços de ações, abordando uma lacuna onde a pesquisa em aprendizado de máquina geralmente se concentra em indicadores técnicos, negligenciando os fundamentalistas.[15]

A análise técnica baseia-se nos preços históricos das ações e no volume de negociações, enquanto a análise fundamentalista avalia os impulsionadores de negócios subjacentes de uma empresa. Apesar da sua oposição histórica, os investigadores exploraram a combinação de ambas as abordagens, com evidências que sugerem a eficácia de tal integração. No entanto, a discrepância entre as práticas dos profissionais de finanças e dos pesquisadores de aprendizagem automática persiste.[15]

O estudo procura responder a duas questões principais: Que indicadores, Fundamentalistas ou Técnicos, são mais relevantes para a previsão dos preços das ações durante seis meses ou um ano utilizando métodos de aprendizagem automática? O desempenho das previsões melhora quando os dois tipos de indicadores são usados em conjunto?[15]

Os experimentos envolvem prever a variação percentual no preço das ações de uma empresa ao longo de 126 e 252 dias de negociação. O estudo emprega modelos de Rede Neural Artificial (RNA) e Regressão de Vetores de Suporte (SVR) com indicadores Técnicos, fundamentalistas e Combinados. A pesquisa concentra-se em 140 empresas do S&P 500, incorporando períodos de mercado turbulentos e estáveis, de janeiro de 1996 a dezembro de 2015.[15]

Os indicadores técnicos incluem Average True Range, Moving Average Convergence Divergence, Money Flow Index, Stochastic Oscillator e outros. Os indicadores fundamentalistas cobrem o desempenho da empresa, concorrentes, indústria e fatores macroeconômicos. Os dados são coletados diariamente e mesclados em conjuntos de dados combinados para análise.[15]

Os dados são divididos em conjuntos de treinamento e teste com uma divisão de 80-20. Os modelos de aprendizado de máquina são implementados usando RNA e SVR, com parâmetros determinados por meio de testes rigorosos. O método Random Walk serve como cenário base para comparação.[15]

No primeiro experimento, ele compara o desempenho de previsão (RMSE) dos modelos NN e SVR usando conjuntos de dados técnicos e fundamentalistas. Os modelos baseados em análise fundamentalista superam consistentemente os modelos baseados em análise técnica em todos os setores, com lacunas de desempenho mais estreitas em setores como Finanças e Energia e lacunas mais amplas nos Cuidados de Saúde. Independentemente

do horizonte de previsão (126 ou 252 dias), os modelos baseados em indicadores fundamentalistas geralmente se destacam, com desempenho superior notável em Cuidados de Saúde. Os modelos NN e SVR superam o método Random Walk.[15]

No segundo experimento, o estudo explora a eficácia da combinação de indicadores Técnicos e Fundamentalistas. Os modelos que utilizam o conjunto combinado superam consistentemente aqueles que utilizam apenas indicadores técnicos ou fundamentalistas, conforme evidenciado pelos valores mais baixos de RMSE. Os testes estatísticos confirmam a importância deste desempenho superior, indicando que a utilização de um conjunto combinado de indicadores produz melhores resultados de previsão em mais de 95% dos casos para modelos NN e mais de 98% para modelos SVR.[15]

Para compreender a melhoria alcançada através da utilização de um conjunto de indicadores combinados, foram aplicadas árvores aleatórias condicionais para identificar a importância relativa das características. Dez indicadores consistentemente classificados como mais importantes em todas as empresas, incluindo elementos de conjuntos de informações técnicas e fundamentalistas. O estudo conclui que a relação sinérgica entre os indicadores Técnicos e Fundamentalistas melhora o desempenho das previsões em comparação com a sua utilização separadamente.[15]

A pesquisa ressalta que os indicadores fundamentalistas geralmente superam os indicadores técnicos na previsão dos preços das ações. As empresas dos setores de saúde e tecnologia da informação beneficiam particularmente dos indicadores fundamentalistas. A combinação de indicadores técnicos e fundamentalistas melhora significativamente o desempenho das previsões em mais de 95% dos casos. O estudo recomenda o uso combinado de ambos os tipos de indicadores para melhorar a previsão dos preços das ações usando modelos baseados em aprendizado de máquina.[15]

## 3.4 ARIMA: Um modelo aplicado de previsão de séries temporais para o índice de ações Bovespa

O artigo explora a aplicação do modelo ARIMA na previsão do Índice de Ações Bovespa, uma tarefa desafiadora devido às incertezas que afetam o comportamento do mercado financeiro. O estudo segue o método Box-Jenkins e utiliza o Erro Percentual Médio Absoluto (MAPE) como métrica de avaliação.[16]

A introdução destaca a importância de melhorar os modelos para medir e prever riscos nos mercados financeiros, sendo os investimentos em ações uma alternativa significativa. Prever o comportamento do Índice Bovespa envolve abordar incertezas relacionadas a diversas variáveis que impactam os preços futuros.[16]

O artigo discute métodos de previsão, distinguindo entre abordagens qualitativas e quantitativas. Centra-se em métodos quantitativos, particularmente análise de séries

temporais, mencionando métodos clássicos como Média Móvel, Ajuste Exponencial, Tendência Linear e Tendência Não Linear. Os modelos ARIMA, especificamente AR e ARMA, são considerados adequados para prever séries estacionárias.[16]

A seção de metodologia descreve a abordagem da pesquisa como aplicada e descritiva, enfatizando a modelagem matemática. O método Box-Jenkins é empregado, envolvendo etapas como identificação do modelo, especificação, estimativa de parâmetros e verificação do modelo.[16]

A seção de análise dos resultados apresenta a aplicação do ARIMA aos dados do Índice Bovespa. A necessidade de uma transformação logarítmica é identificada para resolver a não estacionariedade. Os testes de autocorrelação e autocorrelação parcial sugerem um modelo ARIMA, e a análise residual confirma sua adequação. O artigo inclui gráficos e tabelas que ilustram a análise.[16]

A conclusão destaca a eficácia do modelo ARIMA na previsão do Índice Bovespa, com MAPE de 0,052%, indicando desempenho superior em relação aos demais modelos. O estudo centra-se nas previsões de curto prazo (com um mês de antecedência), alinhando-se com a natureza dinâmica dos mercados financeiros onde podem ser necessárias decisões imediatas. O modelo ARIMA é considerado adequado para auxiliar mecanismos de tomada de decisão relacionados ao Índice Bovespa.[16]

No geral, o artigo sublinha a importância da utilização de modelos sofisticados de previsão de séries temporais, como o ARIMA, na navegação pelas complexidades dos mercados financeiros, oferecendo informações que podem ser valiosas para decisões de investimento.[16]

## 3.5 Outros Trabalhos

### 3.5.1 Propriedades de série temporal de um mercado de ações artificial

No estudo, os autores apresentam um modelo de mercado de ações artificial para explorar a dinâmica da aprendizagem baseada em agentes e da adaptação de regras nos mercados financeiros. O mercado artificial envolve agentes que negociam com base nas suas regras de previsão em evolução, combinando informações técnicas e fundamentalistas.[17]

Os pesquisadores investigam o impacto de um parâmetro-chave, a frequência de aprendizagem, no comportamento do mercado. No caso de aprendizagem lenta, onde os agentes atualizam as suas regras com menos frequência, o mercado converge para um equilíbrio linear de expectativas racionais (REE), demonstrando comportamentos consistentes com os modelos econômicos tradicionais.[17] Contudo, no caso de aprendizagem rápida, em que os agentes atualizam as suas regras com mais frequência, o mercado apresenta características

normalmente observadas nos mercados financeiros reais, tais como fraca previsibilidade, persistência da volatilidade e retornos esperados mais elevados.[17]

O estudo enfatiza a importância do parâmetro frequência de aprendizagem, mostrando como um ajuste aparentemente pequeno pode levar a mudanças substanciais no comportamento do mercado. As conclusões sugerem que a velocidade de aprendizagem e o horizonte temporal ao longo do qual os agentes adaptam as suas regras desempenham um papel crucial na definição da dinâmica do mercado, ilustrando a complexa interação entre a aprendizagem adaptativa e os resultados do mercado. Esta pesquisa contribui para a compreensão da dinâmica artificial do mercado e destaca a importância de considerar mecanismos de aprendizagem no estudo do comportamento do mercado financeiro.[17]

### 3.5.2 Processamento de dados de série temporal financeira para aprendizado de máquina

O artigo investiga os desafios únicos colocados pelos dados de séries temporais financeiras e apresenta estratégias abrangentes para o processamento eficaz de dados para aplicações em aprendizado de máquina. O estudo explora vários métodos de escalonamento, enfatizando seu impacto na estacionalidade e na preservação de informações relevantes para previsão de tendências.[18]

Ao realizar testes empíricos e propor diferentes abordagens de rotulagem para tarefas de classificação e regressão, o artigo busca orientar os pesquisadores na tomada de decisões informadas sobre o processamento de dados de séries temporais financeiras. Ele ressalta a importância de dimensionar, fatiar e testar adequadamente a estacionalidade dos conjuntos de dados, reconhecendo a natureza distinta dos dados de séries temporais financeiras e seus processos estocásticos. O artigo conclui enfatizando a aplicabilidade dos métodos descritos a diversos instrumentos financeiros e escalas de tempo, destacando a importância da seleção, dimensionamento e rotulagem de recursos no processo geral de pesquisa de aprendizado de máquina.[18]

A pesquisa ressalta o papel crítico das técnicas adequadas de divisão de dados na mitigação do risco de overfitting durante o treinamento e as possíveis armadilhas associadas aos métodos tradicionais de embaralhamento na análise de séries temporais. Além disso, aborda o escalonamento de recursos, considerando indicadores limitados e ilimitados, e fornece insights sobre o impacto do escalonamento no desempenho do modelo. O artigo também explora vários métodos de rotulagem, incluindo a criação de novas métricas como %Q, projetadas para focar em modelos negociáveis na análise de séries temporais financeiras. Ao oferecer orientação prática sobre técnicas de processamento de dados, a pesquisa contribui para uma compreensão mais ampla de como lidar com as complexidades dos dados de séries temporais financeiras no contexto da pesquisa de aprendizado de máquina.[18]

## 3.6 Comparação entre trabalhos

Os seis artigos formam coletivamente um rico conjunto de contribuições de pesquisa para o domínio da previsão do mercado de ações, oferecendo diversas perspectivas, metodologias e insights. Um tema recorrente nos artigos é a exploração de técnicas de aprendizado de máquina para superar os desafios inerentes à previsão dos movimentos do mercado de ações. A revisão sistemática estabelece uma compreensão fundamental ao examinar análises fundamentalistas e técnicas, desafiando a hipótese do mercado eficiente e defendendo a aplicação diferenciada da aprendizagem automática na previsão dos preços das ações. O artigo de Beyaz, sobre a comparação de indicadores técnicos e fundamentalistas avança ainda mais este discurso ao fornecer uma análise abrangente da eficácia destes indicadores, defendendo a sua utilização combinada para alcançar um desempenho de previsão superior.[15] Juntos, estes artigos sublinham a mudança contínua das teorias financeiras tradicionais para abordagens mais baseadas em dados e orientadas para a aprendizagem automática no domínio da previsão do mercado de ações.

Enquanto as pesquisas de Nti[13], Beyaz[15] and LeBaron[17] se concentram na comparação de indicadores e metodologias, outros se aprofundam em modelos específicos e suas aplicações, como Huang[14], Junior[16] e Daniel[18]. O estudo sobre modelos ARIMA para o Índice de Ações Bovespa exemplifica isso ao aplicar um modelo de previsão de séries temporais para navegar pelas complexidades de um mercado de ações do mundo real. O artigo sobre o mercado de ações artificial introduz uma dimensão única ao incorporar a aprendizagem baseada em agentes, esclarecendo o papel da frequência de aprendizagem na formação da dinâmica do mercado. A pesquisa sobre processamento de dados de séries temporais financeiras fornece uma ponte crucial entre dados brutos e modelos de aprendizado de máquina, enfatizando a importância de técnicas adequadas de pré-processamento, seleção de recursos e métodos de escalonamento. A exploração de modelos de redes neurais para seleção de ações com base na análise fundamental avança ainda mais a compreensão da interação entre arquiteturas sofisticadas de aprendizado de máquina e índices financeiros fundamentalistas.

Coletivamente, estes artigos contribuem para uma compreensão abrangente dos desafios e oportunidades na previsão do mercado de ações, mostrando a natureza interdisciplinar da investigação neste campo.

Tabela 1 – Tabela de Comparação

| **Trabalhos** | **Demonstra predição** | **Compara algoritmos** | **Dados fundamentalistas** | **Dados técnicos** | **Métricas utilizadas** | **Conjunto de dados** | **Melhor Resultado*** |
|---|---|---|---|---|---|---|---|
| Nti et al., 2020 | Sim | Sim | Sim | Sim | RMSE, MAE, MSE | Público | 0.093, 0.009, 0.00086 |
| Huang et al., 2019 | Sim | Sim | Sim | Não | N/A | Público | N/A |
| Beyaz et al., 2018 | Sim | Sim | Sim | Sim | RMSE | Público | 0.1464 |
| Junior et al., 2014 | Sim | Sim | Não | Sim | MAPE | Público | 0.052% |
| LeBaron et al., 1998 | Não | Não | N/A | N/A | N/A | Privado | N/A |
| Daniel F., 2019 | Não | Não | N/A | N/A | N/A | Privado | N/A |

** O resultado da coluna "Resultado" utiliza as métricas listadas na coluna "Métricas utilizadas".*

# 4 Desenvolvimento

Para atingir os objetivos traçados neste projeto, o fluxo de trabalho desdobra-se em três fases principais:

1. **Coleta e Preparação de Dados:**
    - A fase inicial envolve a coleta e preparação de dados de séries temporais.
2. **Comparação e Avaliação de Modelos:**
    - A segunda fase concentra-se em testar e avaliar vários modelos preditivos disponíveis na biblioteca PyCaret. Visando identificar o modelo mais eficaz com base em métricas de avaliação pré-definidas, fornecendo insights sobre os pontos fortes e fracos de cada modelo.
3. **Análise dos Resultados com Novos Dados:**
    - Com base nos conhecimentos obtidos na comparação de modelos, a terceira fase concentra-se numa análise detalhada dos resultados do modelo preditivo com melhor desempenho usando novos dados.

## 4.1 Arquitetura da Solução

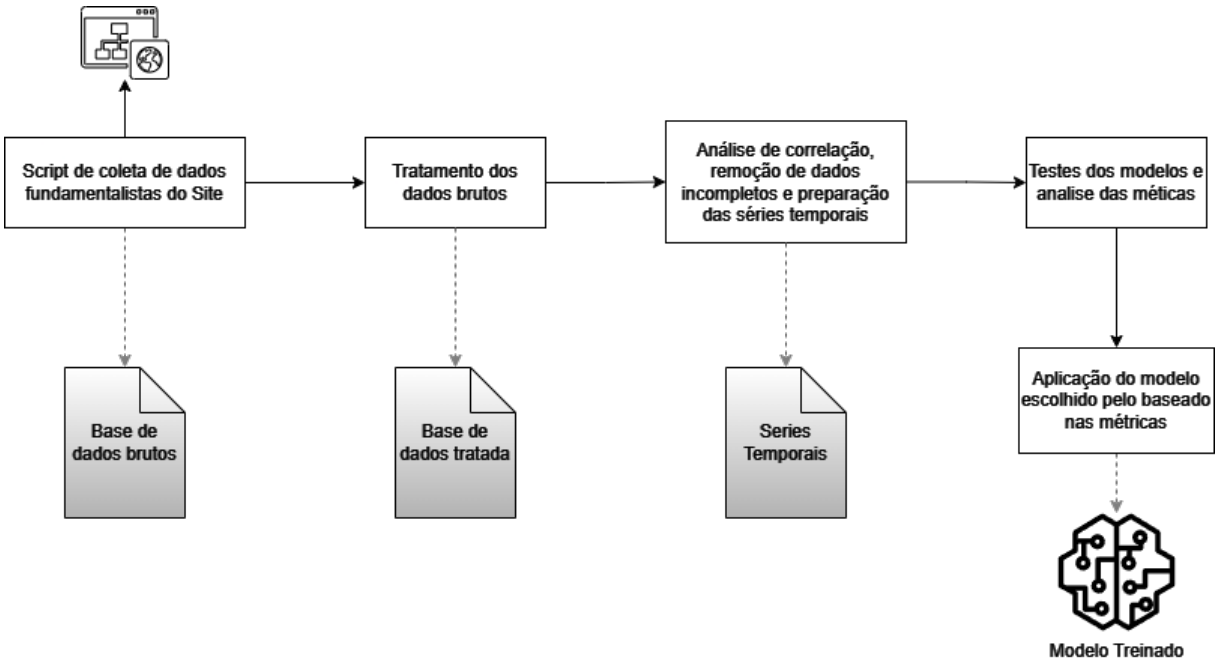



Figura 2 – Fluxograma da Arquitetura da Solução

Na primeira etapa da arquitetura realizamos a coleta dos dados, o objetivo é coletar dados fundamentalistas históricos brutos de uma página web específica. Esse processo é executado por meio de um script, desenvolvido para navegar e recuperar informações da web. Algumas funções essenciais são implementadas para aumentar a eficiência e a confidencialidade deste processo de recolha de dados. O script tem como alvo páginas específicas relacionadas às ações da B3, extraindo URLs relevantes associadas à análise fundamental. Iterando por meio dessas URLs, o script faz solicitações e captura dados para cada ativo financeiro. O conjunto de dados brutos adquiridos é então armazenado em um formato padronizado, estabelecendo as bases para os estágios subsequentes de refinamento e análise de dados.

Após a coleta de dados, o foco subsequente está no processamento do conjunto de dados brutos obtido na etapa anterior. O código então inicia esta etapa incorporando bibliotecas de manipulação e analise necessárias, assim como ferramentas para visualizar as alterações. O script lê sistematicamente todos os arquivos da pasta do conjunto de dados brutos, padroniza os nomes das colunas e os reúne em um conjunto de dados consolidado. O conjunto de dados passa por procedimentos de limpeza de dados, abordando aspectos como substituição de valores faltantes, conversão de colunas para diversos formatos numéricos e gerenciamento de valores percentuais. Visualizações, incluindo mapas de calor de correlação e gráficos de distribuição, são elaboradas para oferecer insights sobre as nuances estruturais do conjunto de dados, facilitando assim a análise. O conjunto de dados refinado é preservado como um arquivo CSV em uma pasta designada, estabelecendo uma base para as fases subsequentes de análise de dados e criação de séries temporais. Além disso, o script incorpora a biblioteca ferramentas avançadas para gerar um relatório de perfil abrangente, melhorando a exploração e compreensão dos dados.

Com o conjunto de dados limpo e padronizado, a próxima etapa envolve carregar os dados e prepará-los para análise de série temporal. O código começa com a instalação da biblioteca de ML low-code, uma ferramenta poderosa para automatizar o fluxo de trabalho de aprendizado de máquina. O script então importa as bibliotecas necessárias e carrega o conjunto de dados consolidado. Posteriormente, concentra-se em uma ação específica (ABEV3 neste exemplo) filtrando o conjunto de dados com base no Ticker. As colunas relacionadas ao tempo são transformadas em um formato de data e hora, e o conjunto de dados é configurado para ter a coluna 'Ano' como índice, criando uma estrutura de série temporal. A configuração da série temporal é iniciada, especificando a variável alvo ('P/L'), o horizonte de previsão e o período sazonal anual. O script então compara vários modelos de série temporal usando a funções presentes da biblioteca gerando as métricas para comparação.

Com a geração da tabela de comparação, geradas pela comparação entre os modelos preditivos disponibilizados pelo treinamento realizado com a biblioteca e a série temporal, obtemos todas as principais métricas utilizadas para verificar a eficácia de modelos de

aprendizado de máquina ordenadas pelo melhor desempenho. Com base nessas métricas, é realizado a escolha do modelo com melhor desempenho entre eles e em seguida o script cria um modelo de aprendizado de máquina usando o algoritmo selecionado. Esta etapa é crucial para identificar o modelo mais adequado para previsão com base na variável alvo especificada e nas características do conjunto de dados.

A etapa final envolve treinar o modelo selecionado em todo o conjunto de dados. Uma vez finalizado o modelo, ele é aplicado para prever valores futuros. Esta etapa gera previsões para o horizonte de previsão especificado, permitindo uma avaliação do desempenho do modelo na previsão da variável alvo ('P/L') ao longo do tempo. O resultado desta etapa serve de base para análises adicionais e tomadas de decisão no domínio financeiro.

## 4.2 Coleta e Preparação de Dados

Os dados fundamentalistas históricos de ações da B3 (Bolsa de Valores do Brasil) utilizados para realizar a modelagem foram coletados a partir da plataforma Oceans14[19], conforme a Figura 3:

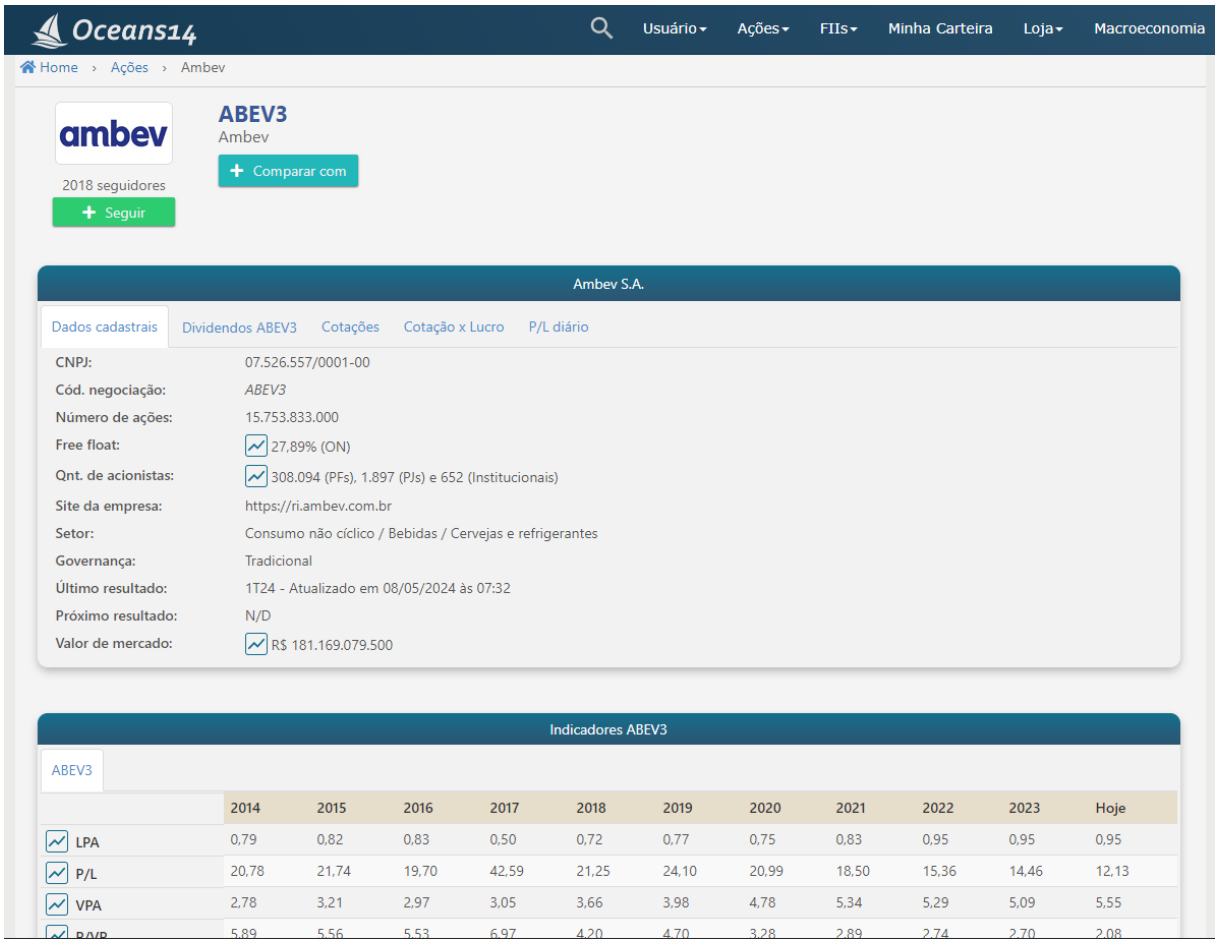



Figura 3 – Ocean14

Para realizar a coleta desses dados foram utilizadas técnicas de Scrapy juntamente com

manipulação de dados em Python, utilizando diversas bibliotecas. Assim sendo possível salvar esses dados em arquivos CSV e durante o processo utilizar ciclo proxies para evitar bloqueios durante as requisições.

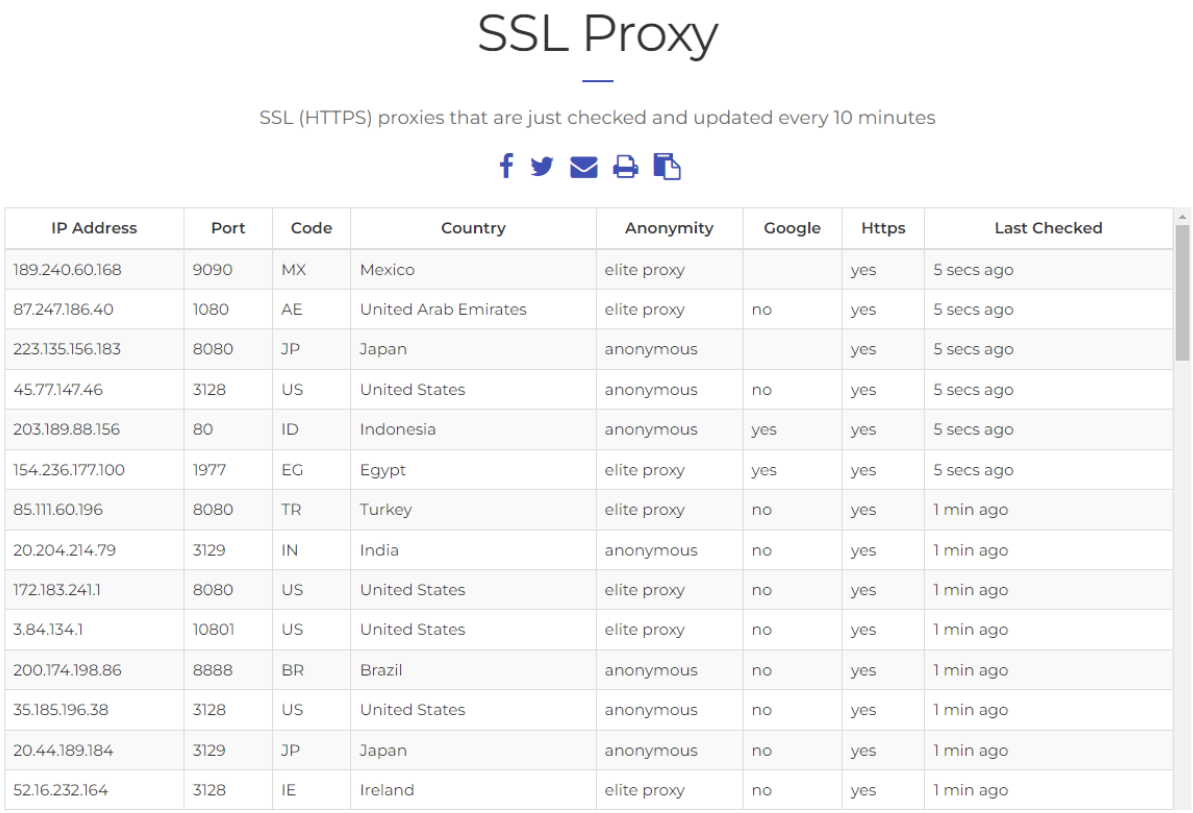
### 4.2.1 Bibliotecas Utilizadas

- requests: para realizar requisições HTTP.
- fromstring (da biblioteca lxml.html): para converter páginas HTML em strings.
- cycle (da biblioteca itertools): para criar um ciclo de proxies.
- BeautifulSoup (da biblioteca bs4): para filtrar e converter os conteúdos das tags HTML.
- pandas: para manipulação e exportação de dados em formato tabular.
- time e random: para randomizar a quantidade de requisições e o tempo entre elas.
- warnings: para gerenciar avisos durante a execução do código.

### 4.2.2 Ciclo Proxies

Para evitar o bloqueio das requisições e otimizar a utilização dos proxies, conforme na Figura 4, pela quantidade de requestes a serem realizadas, foi utilizado a técnica de ciclo de proxies. O ciclo de proxies envolve a utilização de uma lista de proxies, alternando entre eles para cada nova requisição. Essa técnica foi implementada da seguinte forma:

- **Coleta de Proxies**: Proxies são coletados de fontes públicas, como sites que disponibilizam listas de proxies (por exemplo, https://www.sslproxies.org/).
- **Armazenamento em um Ciclo**: A lista de proxies é armazenada em uma estrutura de dados que permite ciclar por eles, como itertools.cycle em Python. Isso cria um iterador que pode ser percorrido infinitamente.




| IP Address | Port | Code | Country | Anonymity | Google | Https | Last Checked |
|---|---|---|---|---|---|---|---|
| 189.240.60.168 | 9090 | MX | Mexico | elite proxy | | yes | 5 secs ago |
| 87.247.186.40 | 1080 | AE | United Arab Emirates | elite proxy | no | yes | 5 secs ago |
| 223.135.156.183 | 8080 | JP | Japan | anonymous | | yes | 5 secs ago |
| 45.77.147.46 | 3128 | US | United States | anonymous | no | yes | 5 secs ago |
| 203.189.88.156 | 80 | ID | Indonesia | anonymous | yes | yes | 5 secs ago |
| 154.236.177.100 | 1977 | EG | Egypt | elite proxy | yes | yes | 5 secs ago |
| 85.111.60.196 | 8080 | TR | Turkey | elite proxy | no | yes | 1 min ago |
| 20.204.214.79 | 3129 | IN | India | anonymous | no | yes | 1 min ago |
| 172.183.241.1 | 8080 | US | United States | elite proxy | no | yes | 1 min ago |
| 3.84.134.1 | 10801 | US | United States | elite proxy | no | yes | 1 min ago |
| 200.174.198.86 | 8888 | BR | Brazil | anonymous | no | yes | 1 min ago |
| 35.185.196.38 | 3128 | US | United States | anonymous | no | yes | 1 min ago |
| 20.44.189.184 | 3129 | JP | Japan | anonymous | no | yes | 1 min ago |
| 52.16.232.164 | 3128 | IE | Ireland | elite proxy | no | yes | 1 min ago |

Figura 4 – SSL Proxies

- **Requisições Alternadas**: Para cada requisição HTTP, um proxy diferente é selecionado do ciclo. Se um proxy falhar (por exemplo, estiver bloqueado ou não responder), o próximo proxy do ciclo é utilizado.
- **Troca de Proxy em Caso de Bloqueio**: Se um site bloquear um proxy ou a requisição falhar, o código automaticamente troca para o próximo proxy no ciclo e tenta novamente.

A função get_proxies(qtd_min) foi desenvolvida extrai proxies da página https://www.sslproxies.org/, esta página disponibiliza proxies com tecnologia SSL necessária. A função realiza requisições para coletar todos os proxies disponíveis e aguarda para que a página seja recarregada a cada 10 minutos com novos proxies.

```
1 def get_proxies(qtd_min):
2     url = 'https://www.sslproxies.org/'
3     proxies = set()
4     while True:
5         response = requests.get(url)
6         parser = fromstring(response.text)
7         for i in parser.xpath('//tbody/tr')[:100]:
8             if i.xpath('.//td[7][contains(text(),"yes")]'):
9                 proxy = ":".join([i.xpath('.//td[1]/text()')[0], i.xpath
   ('.//td[2]/text()')[0]])
```

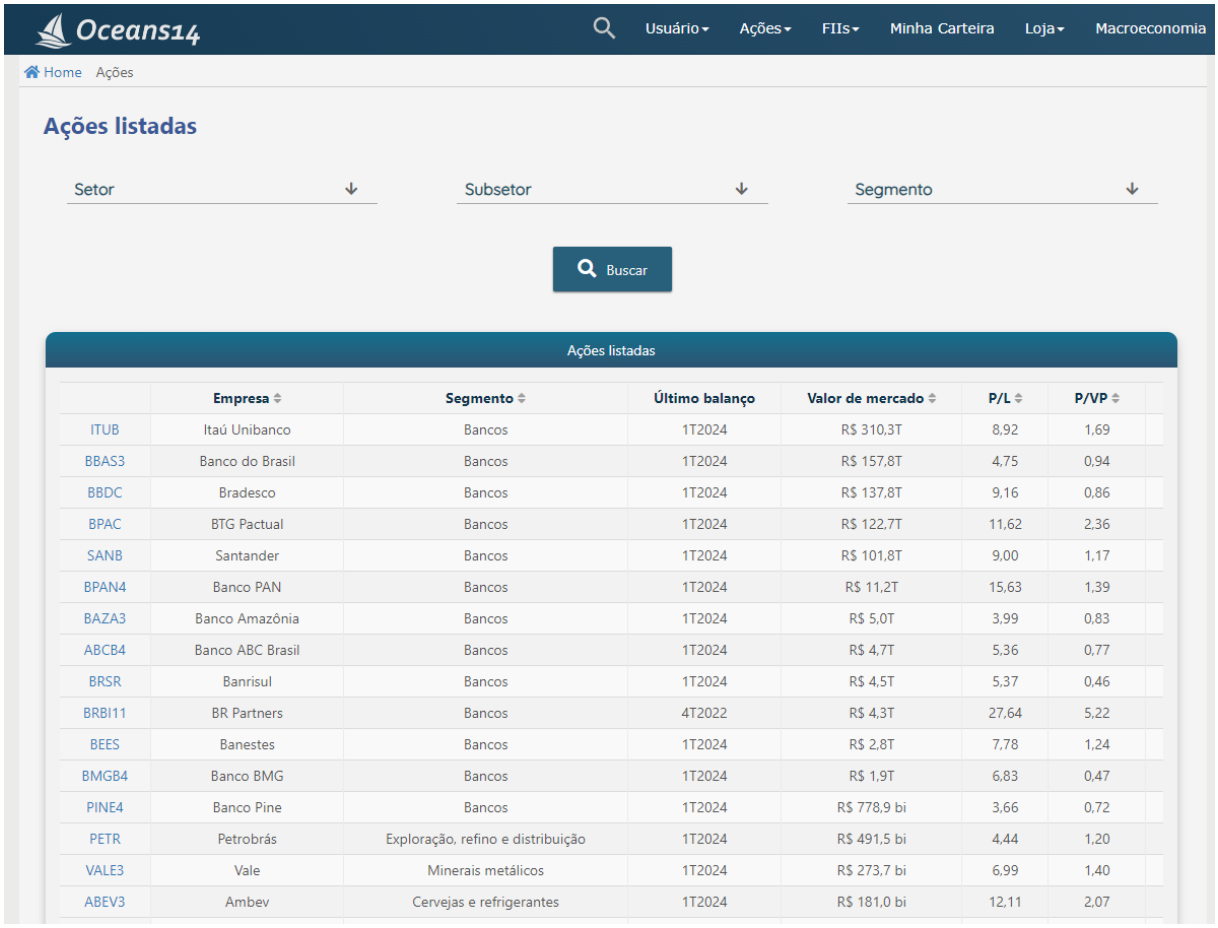
```
10                  proxies.add(proxy)
11          if len(proxies) < qtd_min:
12              for sleeping in range(600,0,-1):#Time to free proxies refresh
13                  time.sleep(1)
14                  clear_output()
15                  print(str(len(proxies)/qtd_min*100) + '%')
16                  print('New Request in ' + str(sleeping) + '...')
17              continue
18          else:
19              break
20      clear_output()
21      print('Done...' + str(len(proxies)) + 'proxies!')
22      return proxies
```

Esse processo se repete até que a quantidade necessária de proxies (passada como parâmetro), mais uma porcentagem extra para compensar a baixa qualidade de alguns proxies, seja atingida, além de armazenar os proxies em um conjunto para garantir que não haja duplicatas.

Para evitar a perda, código salva os proxies em arquivos CSV para uso futuro, facilitando a continuidade do processo em caso de interrupções.

### 4.2.3 Extração de Dados das Ações

Para realizar a modelagem dos dados fundamentalistas das ações, é essencial coletar as URLs que direcionam para as páginas específicas de cada ação no site Oceans14. A opção por utilizar o Ocean14 em vez das APIs da B3 deve-se à instabilidade das APIs da B3, à sua indisponibilidade para pessoas físicas e à falta de dados fundamentalistas consolidados. Primeiro, o código faz uma requisição HTTP para a página principal das ações (https://www.oceans14.com.br/acoes).




| | Empresa | Segmento | Último balanço | Valor de mercado | P/L | P/VP |
|---|---|---|---|---|---|---|
| ITUB | Itaú Unibanco | Bancos | 1T2024 | R$ 310,3T | 8,92 | 1,69 |
| BBAS3 | Banco do Brasil | Bancos | 1T2024 | R$ 157,8T | 4,75 | 0,94 |
| BBDC | Bradesco | Bancos | 1T2024 | R$ 137,8T | 9,16 | 0,86 |
| BPAC | BTG Pactual | Bancos | 1T2024 | R$ 122,7T | 11,62 | 2,36 |
| SANB | Santander | Bancos | 1T2024 | R$ 101,8T | 9,00 | 1,17 |
| BPAN4 | Banco PAN | Bancos | 1T2024 | R$ 11,2T | 15,63 | 1,39 |
| BAZA3 | Banco Amazônia | Bancos | 1T2024 | R$ 5,0T | 3,99 | 0,83 |
| ABCB4 | Banco ABC Brasil | Bancos | 1T2024 | R$ 4,7T | 5,36 | 0,77 |
| BRSR | Banrisul | Bancos | 1T2024 | R$ 4,5T | 5,37 | 0,46 |
| BRBI11 | BR Partners | Bancos | 4T2022 | R$ 4,3T | 27,64 | 5,22 |
| BEES | Banestes | Bancos | 1T2024 | R$ 2,8T | 7,78 | 1,24 |
| BMGB4 | Banco BMG | Bancos | 1T2024 | R$ 1,9T | 6,83 | 0,47 |
| PINE4 | Banco Pine | Bancos | 1T2024 | R$ 778,9 bi | 3,66 | 0,72 |
| PETR | Petrobrás | Exploração, refino e distribuição | 1T2024 | R$ 491,5 bi | 4,44 | 1,20 |
| VALE3 | Vale | Minerais metálicos | 1T2024 | R$ 273,7 bi | 6,99 | 1,40 |
| ABEV3 | Ambev | Cervejas e refrigerantes | 1T2024 | R$ 181,0 bi | 12,11 | 2,07 |

Figura 5 – Lista De Ações

Para isso, utilizamos a biblioteca requests configurada com cabeçalhos de requisição (headers) que simulam um navegador real, permitindo que as requisições pareçam ser feitas por um usuário humano, o que ajuda a evitar bloqueios pelo servidor do site.

Após obter o conteúdo da página, utilizamos a biblioteca BeautifulSoup para analisar o HTML retornado. A análise busca por todos os elementos <a> (links) no HTML que contenham a substring balanco-dividendos em seus atributos href. Esses links específicos são importantes porque direcionam para as páginas que contêm os dados fundamentalistas necessários para nossa modelagem.

```
url = 'https://www.oceans14.com.br/acoes'
page = req.get(url, headers=headers, timeout=15)
soup = BeautifulSoup(page.content, 'html.parser')
url_list = []
for url in soup.find_all('a'):
    if 'balanco-dividendos' in url.get('href'):
        url_list.append('https://www.oceans14.com.br' + url.get('href'))

url_list = list(set(url_list))
```

O código armazena todas essas URLs em uma lista (url_list). Para garantir a eficiência do processo e evitar a necessidade de repetir a coleta das URLs em execuções futuras,

a lista de URLs é salva em um arquivo de texto. Isso permite que o código leia esse arquivo em vez de fazer uma nova requisição e análise da página principal em execuções subsequentes.

Com a lista de URLs das páginas das ações e os proxies previamente obtidos, o próximo passo é coletar os dados fundamentalistas de cada ação. Para cada URL na lista, foi implementado um loop que segue as seguintes etapas:

- **Seleção de Proxy**: A partir de um ciclo de proxies disponíveis, foi selecionado um proxy para ser utilizado na requisição seguinte. Essa prática ajuda a diversificar os endereços IP utilizados, reduzindo a probabilidade de bloqueios por parte dos servidores.

- **Requisição HTTP**: Utilizando o proxy selecionado, foi realizada uma requisição HTTP para a URL da ação. Nesse processo, foram empregados cabeçalhos configurados previamente para simular o comportamento de um navegador convencional.

- **Análise do Conteúdo HTML**: Após obter a resposta da requisição, o conteúdo HTML foi analisado com auxílio da biblioteca BeautifulSoup. Os dados tabulares presentes nas páginas foram extraídos e manipulados utilizando o pandas, o que facilitou a conversão dessas informações em dataframes, garantindo assim uma estrutura organizada e de fácil análise.

- **Armazenamento dos Dados**: Os dados fundamentalistas obtidos, juntamente com os tickers das ações correspondentes, foram armazenados em arquivos CSV separados. Cada arquivo foi nomeado de acordo com o ticker da ação, proporcionando uma organização clara e acessível dos dados coletados.

- **Tratamento de Bloqueios**: Em situações onde o proxy utilizado foi bloqueado ou ocorreu falha na requisição, o código foi programado para alternar automaticamente para o próximo proxy disponível e tentar novamente. Esse mecanismo de tratamento de exceções contribuiu para a resiliência do processo de coleta de dados, minimizando interrupções e garantindo a continuidade da operação.

Ao longo do processo de coleta, as URLs já processadas foram removidas da lista para evitar requisições redundantes, garantindo assim a obtenção dos dados fundamentalistas das ações sem requisições desnecessárias.

Este processo de scraping garante a coleta completa dos dados fundamentalistas das ações da B3. A utilização de proxies, juntamente com a manipulação cuidadosa dos dados coletados, permite superar as restrições impostas pelos sites e garantir que os dados necessários sejam obtidos de maneira confiável.

### 4.2.4 Preparação de Dados

Após a captura dos dados foi realizado a manipulação e análise de dados para que os mesmos estivessem padronizados e normalizados de modo a serem utilizados para criação do modelo.

As bibliotecas pandas e glob foram utilizadas para manipulação e análise de dados e para buscar arquivos no sistema de arquivos, respectivamente. A biblioteca os.path foi utilizada para manipulação de caminhos de arquivos. As bibliotecas matplotlib, seaborn e plotly foram utilizadas para visualização de dados.E a biblioteca pandas_profiling foi utilizada para gerar relatórios exploratórios de dados.

```
data_files = [path.normpath(i).replace('\\','/') for i in glob.glob(
    raw_data + '*.csv')]

full_df = pd.DataFrame()
for file_name in data_files:
    df = pd.read_csv(file_name, sep=';', header=None)
    df.iloc[0, 0] = 'Ano'
    df.iloc[:, 0] = df.iloc[:, 0].str.replace(' ', '')
    df = df.transpose()
    df.columns = df.iloc[0]
    df = df.drop(df.index[0])
    df['Ticker'] = path.splitext(path.basename(file_name))[0]
    full_df = pd.concat([full_df, df])
```

Para realizar a leitura dos arquivos CSV definimos o diretório onde os arquivos CSV estavam armazenados e utilizamos uma função para encontrar todos os arquivos CSV nesse diretório. Criamos um DataFrame vazio para armazenar os dados e, para cada arquivo CSV, lemos o arquivo, ajustamos os cabeçalhos e removemos caracteres indesejados. Adicionamos uma coluna para identificar o nome do arquivo e concatenamos os dados ao DataFrame principal, conforme a Tabela 2:

| Ano | LPA | P/L | VPA | P/VP | ... | Ticker |
|---|---|---|---|---|---|---|
| 2020 | 0.09 | 134.45 | 3.45 | 3.67 | ... | BOAS3 |
| 2021 | 0.09 | 130.65 | 3.49 | 3.31 | ... | BOAS3 |
| 1998 | 19.53 | 4.69 | 302.40 | 0.30 | ... | SBSP3 |
| 1999 | -8.23 | -25.78 | 290.85 | 0.73 | ... | SBSP3 |
| 2000 | 18.33 | 9.49 | 290.32 | 0.60 | ... | SBSP3 |

Tabela 2 – Topo do DataFrame com dados financeiros por ano

Como limpeza dos dados, as colunas foram renomeadas para nomes mais compatíveis com padrões comuns, transformamos valores percentuais em valores numéricos e convertemos outras colunas relevantes para tipos numéricos apropriados. Além disso, foi verificado a quantidade de valores nulos em cada coluna, exibimos as linhas que continham valores

nulos e removemos todas essas linhas. A Tabela 3 apresenta o resumo estatístico dos dados limpos e salvos como um arquivo CSV.

| | Ano | LPA | P/L | VPA | P/VP | ... | Volume_diario |
|---|---|---|---|---|---|---|---|
| count | 3187.00 | 3187.00 | 3187.00 | 3187.00 | 3187.00 | ... | 3187.00 |
| mean | 2013.77 | 6.27 | 11.54 | 67.91 | 2.12 | ... | 35.15 |
| std | 5.46 | 319.52 | 133.11 | 2940.51 | 30.99 | ... | 114.81 |
| min | 1998.00 | -666.67 | -3856.92 | -695.02 | -1714.19 | ... | 0.00 |
| 25% | 2010.00 | 0.16 | 4.24 | 4.22 | 0.82 | ... | 1.00 |
| 50% | 2015.00 | 0.88 | 9.88 | 8.88 | 1.52 | ... | 5.00 |
| 75% | 2018.00 | 2.26 | 20.01 | 18.39 | 2.85 | ... | 27.00 |
| max | 2021.00 | 18000.00 | 928.23 | 166000.00 | 133.39 | ... | 2913.00 |

Tabela 3 – Estatísticas descritivas dos dados financeiros

Por último foi utilizado a biblioteca pandas_profiling para gerar relatórios exploratórios e exibindo-o dentro do notebook, o mesmo estará disponível como apêndice A. Mas dentre as análises geradas está a análise de Coeficiente de correlação de Pearson entre as colunas, que podem ser usadas como entradas adicionais na criação do modelo. Na Figura 6 é possível ver a correlação de forma gráfica em um mapa de calor, onde é possível verificar a alta correlação positiva entre algumas das colunas.

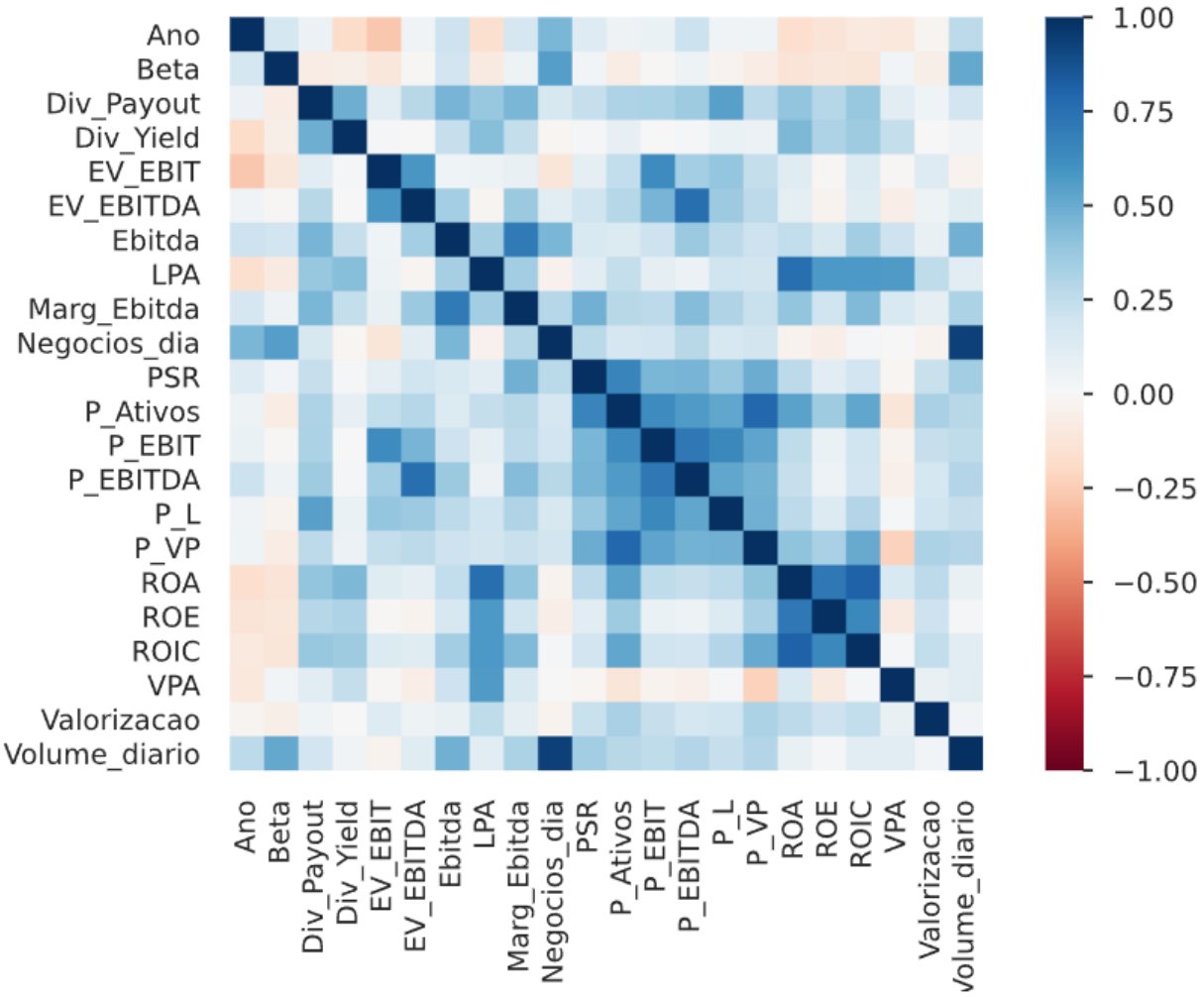



Figura 6 – Mapa de Calor - Correlação

## 4.3 Comparação e Avaliação de Modelos

Para realizar a criação dos modelos foram utilizados duas bibliotecas públicas, sendo elas PyCaret e Prophet e para ter um maior histórico foi escolhido a métrica lucro/prejuízo (P/L) como foco da previsão para a ação ABEV3, para ambos os modelos.

PyCaret e Prophet são duas poderosas ferramentas utilizadas na análise de dados e previsão, cada uma com suas particularidades e vantagens. PyCaret é uma biblioteca de aprendizado de máquina de baixo código que visa simplificar o ciclo de vida do desenvolvimento de modelos. Ela automatiza as tarefas mais comuns de pré-processamento de dados, seleção de modelos, ajuste de hiperparâmetros e avaliação de desempenho. Com uma interface amigável e intuitiva, PyCaret permite que tanto iniciantes quanto profissionais experientes implementem soluções de aprendizado de máquina de forma rápida e eficiente. Além disso, a biblioteca oferece suporte a uma ampla gama de algoritmos de classificação, regressão, clustering e séries temporais, tornando-se uma escolha versátil para diversas aplicações.

Por outro lado, Prophet é uma ferramenta específica para previsão de séries temporais, desenvolvida pelo Facebook. É especialmente adequada para dados que exibem padrões sazonais e tendências não lineares. A grande vantagem do Prophet é sua facilidade de uso e capacidade de lidar com séries temporais com faltas de dados e mudanças abruptas nos padrões históricos. Ele permite a inclusão de componentes adicionais como feriados e eventos especiais, que podem influenciar a previsão. Diferentemente do PyCaret, que é uma plataforma de aprendizado de máquina mais generalista, Prophet é focado exclusivamente em séries temporais, proporcionando uma abordagem robusta e especializada para esse tipo de análise. Em resumo, enquanto PyCaret oferece uma solução abrangente e automatizada para diversas tarefas de aprendizado de máquina, Prophet se destaca como uma ferramenta poderosa e intuitiva para a previsão de séries temporais complexas.

- **Carregamento e Pré-processamento dos Dados**: O primeiro passo foi o carregamento dos dados da série histórica da ação ABEV3 a partir de um arquivo CSV. Em seguida, foi realizada algumas etapas de pré-processamento para garantir a qualidade e consistência dos dados. Isso incluiu a remoção de colunas irrelevantes, como o 'Ticker', e a conversão da coluna 'Ano' para o formato datetime, representando o último dia do ano, no caso do PyCaret. E a renomeação das colunas 'Ano' para 'ds' e 'P_L' para 'y' para atender ao formato esperado pelo Prophet.

- **Configuração do Ambiente de Modelagem**: Utilizando PyCaret foi necessário configurar o ambiente de modelagem. A função setup() foi utilizada para definir o conjunto de dados, a variável alvo ('P_L' - lucro/prejuízo), o horizonte de previsão (3 períodos) e o período sazonal (anual). Isso preparou o cenário para a modelagem e avaliação dos modelos de previsão disponíveis. Já para Prophet foi apenas necessário configurar o intervalo de confiança, definido como 95% (interval_width=0.95).

- **Comparação e Seleção de Modelos**: Após a configuração do ambiente, foi realizada uma comparação entre diferentes modelos de previsão disponíveis no PyCaret utilizando a função compare_models(). Esse processo permitiu identificar o modelo mais adequado para os nossos dados e objetivos de previsão. Essa etapa foi apenas necessária utilizando PyCaret já que essa biblioteca low-code disponibiliza vários tipos de modelos a partir de uma execução.

| | Model | MASE | RMSSE | MAE | RMSE | MAPE | SMAPE | R2 | TT (Sec) |
|---|---|---|---|---|---|---|---|---|---|
| ada_cds_dt | AdaBoost w/ Cond. Deseasonalize & Detrending | 0.3314 | 0.2455 | 4.0359 | 4.7855 | 0.1528 | 0.1534 | -1.0488 | 0.1033 |
| dt_cds_dt | Decision Tree w/ Cond. Deseasonalize & Detrending | 0.3693 | 0.2764 | 4.6688 | 5.4858 | 0.1886 | 0.1803 | -5.4955 | 0.0667 |
| gbr_cds_dt | Gradient Boosting w/ Cond. Deseasonalize & Detrending | 0.4071 | 0.3011 | 5.3934 | 6.1367 | 0.2279 | 0.2109 | -4.6135 | 0.0867 |
| croston | Croston | 0.4855 | 0.4069 | 6.1327 | 8.0428 | 0.2189 | 0.2634 | -5.5487 | 0.9700 |
| omp_cds_dt | Orthogonal Matching Pursuit w/ Cond. Deseasonalize & Detrending | 0.4935 | 0.3868 | 6.4503 | 7.8336 | 0.2483 | 0.2905 | -6.1378 | 0.0733 |
| rf_cds_dt | Random Forest w/ Cond. Deseasonalize & Detrending | 0.5188 | 0.3735 | 7.0682 | 7.7532 | 0.3080 | 0.2693 | -6.8496 | 0.1600 |
| knn_cds_dt | K Neighbors w/ Cond. Deseasonalize & Detrending | 0.5239 | 0.3893 | 6.9852 | 7.8678 | 0.2955 | 0.2673 | -4.1734 | 0.1167 |
| et_cds_dt | Extra Trees w/ Cond. Deseasonalize & Detrending | 0.8405 | 0.5942 | 12.4053 | 12.9886 | 0.5694 | 0.3912 | -31.1743 | 0.1333 |
| lightgbm_cds_dt | Light Gradient Boosting w/ Cond. Deseasonalize & Detrending | 0.9316 | 0.6142 | 12.7937 | 12.8881 | 0.5872 | 0.4510 | -34.1359 | 0.1500 |
| huber_cds_dt | Huber w/ Cond. Deseasonalize & Detrending | 2.5535 | 2.3093 | 38.9337 | 51.8814 | 1.6820 | 0.6192 | -700.2671 | 0.0733 |
| br_cds_dt | Bayesian Ridge w/ Cond. Deseasonalize & Detrending | 5.1620 | 4.1492 | 71.5320 | 87.4444 | 3.2847 | 1.0174 | -7103.4816 | 0.0667 |
| lasso_cds_dt | Lasso w/ Cond. Deseasonalize & Detrending | 5.9755 | 4.9127 | 84.4907 | 104.9166 | 3.8626 | 1.0199 | -7905.8301 | 0.0767 |
| llar_cds_dt | Lasso Least Angular Regressor w/ Cond. Deseasonalize & Detrending | 5.9756 | 4.9128 | 84.4928 | 104.9191 | 3.8627 | 1.0199 | -7906.4499 | 0.0867 |
| en_cds_dt | Elastic Net w/ Cond. Deseasonalize & Detrending | 6.2986 | 5.1771 | 88.9072 | 110.4189 | 4.0693 | 1.0273 | -9061.0502 | 0.1967 |
| arima | ARIMA | 6.4860 | 5.5303 | 95.4667 | 121.2174 | 4.2967 | 0.8982 | -8458.0878 | 1.3033 |
| auto_arima | Auto ARIMA | 7.2202 | 6.0812 | 105.3001 | 132.5195 | 4.7567 | 0.9265 | -8816.2072 | 1.3767 |
| ridge_cds_dt | Ridge w/ Cond. Deseasonalize & Detrending | 10.1477 | 8.8248 | 141.2764 | 185.8697 | 6.5565 | 1.0841 | -35227.2816 | 0.1833 |
| lr_cds_dt | Linear w/ Cond. Deseasonalize & Detrending | 15.0229 | 14.1070 | 207.7944 | 295.0811 | 9.7314 | 1.1692 | -103778.8168 | 0.1767 |

Figura 7 – Tabela de Comparação PyCaret

- **Treinamento e Avaliação do Modelo Escolhido**: O modelo escolhido no PyCaret foi o 'ada_cds_dt' (AdaBoost w/ Cond. Deseasonalize & Detrending). Ele foi treinado utilizando todos os dados disponíveis, visando capturar padrões e tendências relevantes para a previsão do P/L da ação ABEV3. Essa etapa não foi necessária utilizando Prophet já que o mesmo possue sua própria estrutura de modelagem.

```
# Treina e avalia o desempenho do modelo AdaBoost w/ Cond.
    Deseasonalize & Detrending
model = create_model('ada_cds_dt')
# Treina o modelo em todo o conjunto de dados
final = finalize_model(model)
```

Foram utilizadas as métricas MAE, RMSE e MAPE para avaliar e comparar os modelos criados por ambas as blibliotecas (PyCaret e Prophet). Como é possivel ver na Tabela 4 abaixo a comparação dos resultados:

| Modelo | MAE | RMSE | MAPE |
|---|---|---|---|
| PyCaret (AdaBoost w/ Cond. Deseasonalize & Detrending) | 4.4108 | 5.2234 | 0.1756 |
| Prophet | 11.35 | 12.86 | 0.51 |

Tabela 4 – Comparação de métricas entre modelos

Comparando os dois modelos, o PyCaret apresentou um MAE de 4.4108, enquanto o Prophet teve um MAE de 11.35, isso indica que o PyCaret é significativamente mais preciso, com erros médios absolutos menores. Em relação ao RMSE, o PyCaret registrou 5.2234, enquanto o Prophet registrou 12.86, mostrando que o PyCaret tem menor variância nos erros. No que se refere ao MAPE, o PyCaret obteve 0.1756 (15.28%) comparado

aos 0.51 (51%) do Prophet, indicando que o PyCaret possui uma menor porcentagem de erro absoluto.

Desse modo podemos concluir que o modelo PyCaret (AdaBoost com Cond. Deseasonalize & Detrending) apresenta uma qualidade muito superior ao Prophet em todas as métricas avaliadas. Os erros absolutos e percentuais do PyCaret são significativamente menores, indicando maior precisão e menor variabilidade nos erros. Portanto, o PyCaret é claramente o modelo de melhor qualidade baseado nas métricas fornecidas.

## 4.4 Análise dos Resultados com Novos Dados

Após a geração dos modelos foram realizados novas execuções para prever o valor do 'P_L' e realizar a comparação dos resultados com os valores reais e assim pode avaliar como os modelos se comportam em termos práticos.

Para realizar previsões futuras, foi preparado os dados de entrada utilizando um conjunto separado de dados específico para previsão. Novamente, foi realizada etapas de pré-processamento, como a remoção de colunas desnecessárias e a conversão da coluna 'Ano' para o formato datetime, para ambas as bibliotecas.

```
1 pycaretResults = predict_model(final, fh = 3, X=ABEV3_df_topredict)
```

Com os modelos treinados e os dados de entrada preparados, foi feita a previsão do P/L da ação ABEV3 para os próximos 3 períodos. Os resultados da previsão foram armazenados em um DataFrame chamado pycaretResults, fornecendo os valores absolutos.

| Modelo/Ano | 2022 | 2023 | 2024 |
|---|---|---|---|
| Valor Real | 15.36 | 14.46 | 13.05 |
| PyCaret (AdaBoost w/ Cond. Deseasonalize & Detrending) | 33.6254 | 31.3977 | 31.9997 |
| Prophet | 25.005587 | 21.076038 | 34.900937 |

Tabela 5 – Comparação das previsões entre modelos

Primeiro, analisamos o modelo PyCaret (AdaBoost w/ Cond. Deseasonalize & Detrending). Em 2022, a previsão foi de 33.6254, enquanto o valor real foi de 15.36, resultando em uma diferença absoluta de 18.2654. Em 2023, a previsão foi de 31.3977, com um valor real de 14.46, resultando em uma diferença absoluta de 16.9377. Em 2024, a previsão foi de 31.9997, com um valor real de 13.05, resultando em uma diferença absoluta de 18.9497.

Para o modelo Prophet, em 2022, a previsão foi de 25.005587, enquanto o valor real foi de 15.36, resultando em uma diferença absoluta de 9.645587. Em 2023, a previsão foi de 21.076038, com um valor real de 14.46, resultando em uma diferença absoluta de 6.616038. Em 2024, a previsão foi de 34.900937, com um valor real de 13.05, resultando em uma diferença absoluta de 21.850937.

Resumindo as diferenças absolutas, o modelo PyCaret teve um erro absoluto de 18.2654 em 2022, 16.9377 em 2023 e 18.9497 em 2024, resultando em um total de 54.1528 e uma

média de 18.0509. O modelo Prophet teve um erro absoluto de 9.645587 em 2022, 6.616038 em 2023 e 21.850937 em 2024, resultando em um total de 38.112562 e uma média de 12.704187.

| Modelo/Ano | 2022 | 2023 | 2024 | Média |
|---|---|---|---|---|
| PyCaret | 18.2654 | 16.9377 | 18.9497 | 18.0509 |
| Prophet | 9.645587 | 6.616038 | 21.850937 | 12.704187 |

Tabela 6 – Erros Absolutos Médios (MAE) dos Modelos de Série Temporal

Em termos de erro absoluto médio (MAE), o modelo Prophet teve um desempenho melhor, com um MAE de 12.704187 comparado ao MAE de 18.0509 do modelo PyCaret. Analisando cada ano separadamente, em 2022, o Prophet teve um desempenho significativamente melhor que o PyCaret. Em 2023, o Prophet novamente teve um desempenho melhor. Em 2024, ambos os modelos apresentaram grandes erros absolutos, mas o Prophet teve o maior erro.

Em conclusão, o modelo Prophet apresentou um desempenho geral melhor, com menores diferenças absolutas em dois dos três anos analisados. No entanto, ambos os modelos apresentaram grandes desvios em 2024, indicando possíveis limitações na captura de padrões futuros ou mudanças bruscas nas séries temporais.

Para melhorar a análise, seria interessante testar outros modelos ou ajustar parâmetros para melhorar as previsões.

# 5 Conclusão

Neste trabalho, foi desenvolvido e avaliado modelos preditivos para gerar um portfólio de investimento previdenciário alinhado ao perfil do usuário, buscando o melhor retorno possível dentro dos níveis de volatilidade permitidos.

Para atingir os objetivos específicos, foi realizada uma análise abrangente do estado da arte em ciência de dados e séries temporais, seguida pela coleta e tratamento dos dados necessários para a análise. Foram desenvolvidos modelos preditivos baseados em dados fundamentalistas, utilizando bibliotecas como PyCaret e Prophet, e conduzidos experimentos para testar a eficácia desses modelos. Finalmente, foram analisados os resultados obtidos com os experimentos para avaliar o desempenho dos modelos.

O processo de desenvolvimento foi dividido em três fases principais. A primeira fase, de coleta e preparação de dados, envolveu a obtenção de dados fundamentalistas históricos de ações da B3 a partir da plataforma Oceans14. Foram utilizadas técnicas de scraping e manipulação de dados em Python para coletar e preparar os dados, garantindo a padronização e normalização necessários para a criação dos modelos.

Na segunda fase, os esforços foram concentrados na comparação e avaliação de modelos. Utilizando a biblioteca PyCaret, foi configurado o ambiente de modelagem e realizada a comparação de vários modelos preditivos. O modelo AdaBoost com Cond. Deseasonalize & Detrending foi escolhido como o de melhor desempenho, com base nas métricas de avaliação como MAE, RMSE e MAPE. Paralelamente, foi utilizada a biblioteca Prophet para desenvolver um modelo alternativo e compará-lo com o modelo do PyCaret.

Os resultados mostraram que o modelo PyCaret superou significativamente o modelo Prophet em todas as métricas avaliadas. O PyCaret apresentou menor erro absoluto médio (MAE), menor raiz do erro quadrático médio (RMSE) e menor porcentagem de erro absoluto médio (MAPE), indicando maior precisão e menor variabilidade nos erros. Portanto, foi concluído que o modelo PyCaret é mais adequado para previsão de P/L da ação ABEV3.

Na fase final, foram realizadas novas previsões com os modelos treinados utilizando novos dados. O desempenho prático dos modelos foi avaliado, observando que o Prophet obteve leve superioridade em termos de precisão, mas sem consistência das previsões. Assim, de forma limitada pelo aumento da imprecisão conforme o aumento do horizonte de previsão, é possível utilizar o modelo para realizar previsões dos indicadores fundamentalistas para serem utilizados nas análises microeconômicas das companhias e tomar decisões de compra ou venda mais bem informadas.

Em resumo, o modelo PyCaret (AdaBoost com Cond. Deseasonalize & Detrending) demonstrou ser uma ferramenta poderosa para previsão de séries temporais no contexto

de investimentos previdenciários, mas que possui muitas limitações a serem aperfeiçoadas para serem viáveis e entreguem resultados úteis. A utilização de técnicas avançadas de coleta e preparação de dados, combinadas com uma análise detalhada de modelos preditivos, permitiu desenvolver um modelo robusto, mas pouco preciso em termos absolutos, o que é um ponto crucial para este caso. Esses resultados destacam a importância de escolher modelos adequados e realizar uma avaliação rigorosa para alcançar previsões confiáveis e úteis no mercado financeiro.

Como sugestões para trabalhos futuros, para melhorar a precisão e eficácia do modelo, sugere-se explorar outros modelos de aprendizado de máquina e técnicas de otimização de hiperparâmetros, utilizando uma variedade maior de ações além da ABEV3. Além disso, a inclusão de mais variáveis fundamentalistas e técnicas avançadas de tratamento de dados pode ajudar a capturar melhor as nuances do mercado financeiro. Finalmente, a realização de estudos comparativos com diferentes horizontes de previsão e períodos sazonais pode fornecer percepções adicionais sobre o desempenho dos modelos preditivos.

# Referências Bibliográficas

1 GOETZMANN W. N. & ROUWENHORST, K. G. The origins of value: The financial innovations that created modern capital markets. In: *The Origins of Value: The Financial Innovations that Created Modern Capital Markets*. [S.l.]: Oxford University Press, 2005. 14

2 BARCELLOS M. & AZEVEDO, S. Histórias do mercado de capitais no brasil: depoimentos inéditos de personalidades que marcaram a trajetória das bolsas de valores no país. In: *Histórias do mercado de capitais no Brasil: depoimentos inéditos de personalidades que marcaram a trajetória das bolsas de valores no país*. [S.l.]: Alta Books, 2018. 14

3 AALST, W. Process mining. In: *Process mining*. [S.l.]: Springer, 2016. 14

4 DHAR, V. Data science and prediction. *Communications of the ACM*, Association for Computing Machinery, v. 56 (12), p. 64–73, 2013. 14

5 SEGAL, T. *Fundamental Analysis*. 2022. <https://www.investopedia.com/terms/f/fundamentalanalysis.asp>. 14

6 ELMERRAJI, J. *Guide to Financial Ratios*. 2022. <https://www.investopedia.com/articles/stocks/06/ratios.asp>. 14

7 KAHNEMAN, D. Thinking, fast and slow. In: *Thinking, Fast and Slow*. [S.l.]: Farrar, Straus and Giroux, 2011. 15

8 LAMPENIUS, N.; ZICKAR, M. Development and validation of a model and measure of financial risk-taking. *Journal of Behavioral Finance*, v. 6, 01 2005. 18, 19

9 ROUF, N. et al. Stock market prediction using machine learning techniques: A decade survey on methodologies, recent developments, and future directions. *Electronics*, v. 10, n. 21, 2021. ISSN 2079-9292. Disponível em: <https://www.mdpi.com/2079-9292/10/21/2717>. 21

10 EXAME, R. Knight capital perde us$440 milhões por falha em robô. *EXAME*, 2012. Disponível em: <https://exame.com/invest/mercados/knight-capital-perde-us-440-milhoes-por-falha-em-robo/>. 21

11 BROCKWELL, P. J.; DAVIS, R. A. *Introduction to Time Series and Forecasting*. [S.l.]: Springer, 2016. 22

12 CHATFIELD, C. *The Analysis of Time Series: An Introduction*. [S.l.]: Chapman and Hall/CRC, 2004. 24

13 NTI, I.; ADEKOYA, A.; WEYORI, B. A systematic review of fundamental and technical analysis of stock market predictions. *Artificial Intelligence Review*, v. 53, 04 2020. 26, 27, 33

14 HUANG, Y.; CAPRETZ, L. F.; HO, D. Neural network models for stock selection based on fundamental analysis. In: IEEE. *2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)*. [S.l.], 2019. p. 1–4. 27, 28, 33

15 BEYAZ, E. et al. Comparing technical and fundamental indicators in stock price forecasting. In: *2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*. [S.l.: s.n.], 2018. p. 1607–1613. 29, 30, 33

16 JUNIOR, P. R. et al. Arima: An applied time series forecasting model for the bovespa stock index. *Applied Mathematics*, Scientific Research Publishing, v. 5, n. 21, p. 3383, 2014. 30, 31, 33

17 LEBARON, B.; ARTHUR, W. B.; PALMER, R. Time series properties of an artificial stock market. *Journal of Economic Dynamics and control*, Elsevier, v. 23, n. 9-10, p. 1487–1516, 1999. 31, 32, 33

18 DANIEL, F. Financial time series data processing for machine learning. *arXiv preprint arXiv:1907.03010*, 2019. 32, 33

19 OCEANS14. 2024. <https://www.oceans14.com.br/>. Accessed: 2024-06-05. 37

# A Relatórios Exploratórios - Pandas Profiling

Para acessar o relatório basta clicar o link abaixo ou para baixar o mesmo localmente basta clicar com botão direito e escolher a opção "Salvar link como...".

clean_data_report.html

# B Código Fonte

Para acessar o código-fonte basta clicar o link abaixo.
investHub

# Modelo Analítico para Gerenciamento de Portfólio

**Felipe Longarai Trisotto**[1]

[1]Departamento de Informática e Estatística
Universidade Federal de Santa Catarina (UFSC)
Caixa Postal 476 – 88040-370 – Florianópolis – SC – Brasil

***Abstract.*** *In the field of finance and investments, it is known that the capital market is the most inclusive way to invest in successful businesses, build and preserve assets over time, especially with a social security focus. In the current situation where a large part of the population has little financial education to manage their investments autonomously, a model capable of generating automated recommendations from the analysis of the investor profile and financial data of the assets available in the market can be seen as a way to facilitate the process of managing investments so that the user can focus on their personal and professional development. In this work, predictive models were developed and evaluated to generate a retirement investment portfolio aligned with the user's profile, seeking the best possible return within the permitted volatility levels. After a comprehensive analysis of the state of the art in data science and time series, and the collection and preparation of fundamental historical data of B3 stocks, various predictive models were tested using the PyCaret and Prophet libraries. The AdaBoost model with Cond. Deseasonalize & Detrending, developed with PyCaret, was identified as the best performing, significantly outperforming the Prophet model in all evaluated metrics (MAE, RMSE, and MAPE). Although Prophet showed slight superiority in practical precision, its predictions were not consistent. It was concluded that the PyCaret model is more suitable for predicting the P/E of ABEV3 stock, highlighting the need to choose appropriate models and conduct rigorous evaluations to achieve reliable forecasts in the financial market. It is suggested, for future work, to explore other machine learning models, hyperparameter optimization, inclusion of more fundamental variables, and conducting comparative studies with different forecasting horizons.*

***Resumo.*** *No âmbito de finanças e investimentos sabe-se que o mercado de capitais é o meio mais inclusivo para se investir em negócios de sucesso, construir e preservar patrimônio ao longo do tempo, principalmente com foco previdenciário. Na situação atual, aonde grande parte da população possui pouca educação financeira para gerir seus investimentos de forma autônoma, um modelo capaz de gerar recomendações automatizadas a partir da análise do perfil do investidor e dados financeiros dos ativos disponíveis no mercado pode ser vista como uma forma de facilitar o processo de gerenciar investimentos de maneira que o usuário foque em seu desenvolvimento pessoal e profissional. Neste trabalho, foram desenvolvidos e avaliados modelos preditivos para gerar um portfólio de investimento previdenciário alinhado ao perfil do usuário, buscando o melhor retorno possível dentro dos níveis de volatilidade permitidos. Após uma análise abrangente do estado da arte em ciência de dados e*

*séries temporais e a coleta e preparação de dados fundamentalistas históricos de ações da B3, foram testados diversos modelos preditivos utilizando as bibliotecas PyCaret e Prophet. O modelo AdaBoost com Cond. Deseasonalize & Detrending, desenvolvido com PyCaret, foi identificado como o de melhor desempenho, superando significativamente o modelo Prophet em todas as métricas avaliadas (MAE, RMSE e MAPE). Embora o Prophet tenha demonstrado leve superioridade em termos de precisão prática, suas previsões não foram consistentes. Concluiu-se que o modelo PyCaret é mais adequado para previsão de P/L da ação ABEV3, destacando a necessidade de escolher modelos adequados e realizar avaliações rigorosas para alcançar previsões confiáveis no mercado financeiro. Sugere-se, para trabalhos futuros, explorar outros modelos de aprendizado de máquina, otimização de hiperparâmetros, inclusão de mais variáveis fundamentalistas e a realização de estudos comparativos com diferentes horizontes de previsão.*

## 1. Introdução

Desde o início do século XVII, com as primeiras emissões de títulos, dívidas e ações para custear as grandes navegações de companhias como a Companhia Holandesa das Índias Orientais[GOETZMANN 2005], essa se mantém uma das principais formas de investimento de capital mundial inclusive no Brasil, que teve início após a criação da Bolsa de Valores Bahia Sergipe Alagoas (BOVESBA), que se encontra desativada hoje[BARCELLOS 2018].

Com a evolução dos mecanismos de integração, como home brokers e internet banking, que facilitaram o acesso de muitos investidores a essa categoria de ativo e a constante queda da taxa básica de juros (Taxa Selic) que alcançou patamares de juros reais negativos, criou uma migração de muitos pequenos investidores para produtos de renda variável a procura por maiores rentabilidades como alternativa a produtos de renda fixa que não mais representavam um ganho adequado com anteriormente.

Entretanto, hoje temos um pouco mais de 91,5 milhões (IBGE 2010) de pessoas economicamente ativas no Brasil e apesar desse número apenas 3 milhões, pouco mais de 3%, possuem investimentos registrados na B3 (Brasil, Bolsa, Balcão), única bolsa de valores em funcionamento no Brasil, sendo os dois dos principais fatores para isso a complexidade para investir e administrar recursos em ativos de risco e a ausência de educação financeira em todos os âmbitos desde ensino fundamental até superior.

Nos últimos anos, a ciência de dados surgiu como uma disciplina nova e importante sendo uma mescla de disciplinas, como estatística, mineração de dados, bancos de dados e sistemas distribuídos[AALST 2016]. Ciência de dados é o estudo da extração de conhecimento a partir de dados, um requisito básico para avaliar se o novo conhecimento é útil para a tomada de decisão e seu poder preditivo, não apenas sua habilidade de explicar o passado[DHAR 2013].

Dentro do âmbito do mercado de capitais, temos a análise fundamentalista um método para medir o valor intrínseco de títulos por meio do estudo de indicadores macro e microeconômicos e posteriormente comparar com preço de mercado do mesmo[SEGAL 2022]. Durante essas análises, indicadores fundamentalistas são amplamente usados por analistas, são índices de desempenho calculados a partir de dados

administrativos e contábeis fornecidos publicamente que variam entre indicadores de lucratividade, liquidez, alavancagem e avaliativos[ELMERRAJI 2022].

Tendo esse cenário em perspectiva, podemos unir essa necessidade à possibilidade que estudos de máquina e modelos analíticos no campo da ciência de dados, que tem apresentado rápido avanço por meio novas bibliotecas em linguagens como Python e R, para captar, modelar e analisar a ampla quantidade de dados fundamentalistas disponíveis ao público para apoiar o gerenciamento de investimentos de forma simples e prática, possibilitando assim o investidor iniciante a dar seus primeiros passos e escapar de vieses comportamentais comuns a investidores iniciantes, como viés de confirmação e ancoragem[KAHNEMAN 2011].

Sendo assim, as próximas etapas dessa pesquisa serão, primeiramente, apresentar o tema de investimento focado para pessoas físicas e como modelos analíticos podem ser usados nessa área. Em seguida, será realizado um estudo de estratégias para análise de perfil e correlação entre retorno e volatilidade para algumas categorias de investimento, além da captura dos indicadores fundamentalistas para os mesmos. E por fim, será realizado o desenvolvimento do modelo capaz de recomendar um portfólio adequado ao perfil do investidor.

## 2. Desenvolvimento

Para atingir os objetivos traçados neste projeto, o fluxo de trabalho desdobra-se em três fases principais:

1. **Coleta e Preparação de Dados:**
    - A fase inicial envolve a coleta e preparação de dados de séries temporais.
2. **Comparação e Avaliação de Modelos:**
    - A segunda fase concentra-se em testar e avaliar vários modelos preditivos disponíveis na biblioteca PyCaret. Visando identificar o modelo mais eficaz com base em métricas de avaliação pré-definidas, fornecendo insights sobre os pontos fortes e fracos de cada modelo.
3. **Análise dos Resultados com Novos Dados:**
    - Com base nos conhecimentos obtidos na comparação de modelos, a terceira fase concentra-se numa análise detalhada dos resultados do modelo preditivo com melhor desempenho usando novos dados.

### 2.1. Arquitetura da Solução

Na primeira etapa da arquitetura realizamos a coleta dos dados, o objetivo é coletar dados fundamentalistas históricos brutos de uma página web específica. Esse processo é executado por meio de um script, desenvolvido para navegar e recuperar informações da web. Algumas funções essenciais são implementadas para aumentar a eficiência e a confidencialidade deste processo de recolha de dados. O script tem como alvo páginas específicas relacionadas às ações da B3, extraindo URLs relevantes associadas à análise fundamental. Iterando por meio dessas URLs, o script faz solicitações e captura dados para cada ativo financeiro. O conjunto de dados brutos adquiridos é então armazenado em um formato padronizado, estabelecendo as bases para os estágios subsequentes de refinamento e análise de dados.

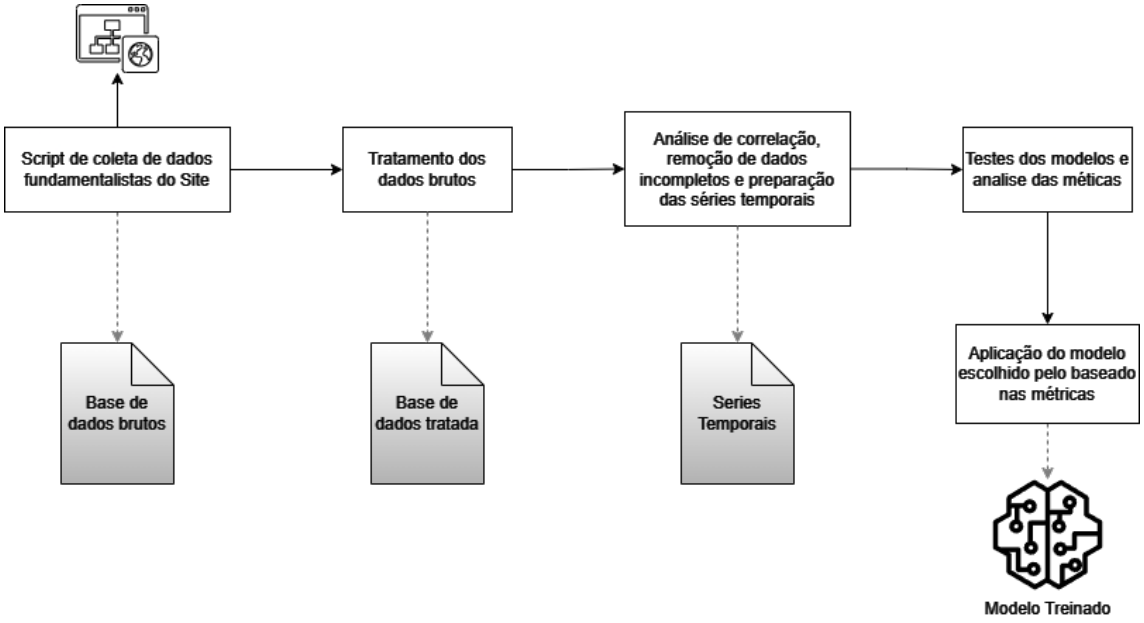



**Figura 1. Fluxograma da Arquitetura da Solução**

Após a coleta de dados, o foco subsequente está no processamento do conjunto de dados brutos obtido na etapa anterior. O código então inicia esta etapa incorporando bibliotecas de manipulação e analise necessárias, assim como ferramentas para visualizar as alterações. O script lê sistematicamente todos os arquivos da pasta do conjunto de dados brutos, padroniza os nomes das colunas e os reúne em um conjunto de dados consolidado. O conjunto de dados passa por procedimentos de limpeza de dados, abordando aspectos como substituição de valores faltantes, conversão de colunas para diversos formatos numéricos e gerenciamento de valores percentuais. Visualizações, incluindo mapas de calor de correlação e gráficos de distribuição, são elaboradas para oferecer insights sobre as nuances estruturais do conjunto de dados, facilitando assim a análise. O conjunto de dados refinado é preservado como um arquivo CSV em uma pasta designada, estabelecendo uma base para as fases subsequentes de análise de dados e criação de séries temporais. Além disso, o script incorpora a biblioteca ferramentas avançadas para gerar um relatório de perfil abrangente, melhorando a exploração e compreensão dos dados.

Com o conjunto de dados limpo e padronizado, a próxima etapa envolve carregar os dados e prepará-los para análise de série temporal. O código começa com a instalação da biblioteca de ML low-code, uma ferramenta poderosa para automatizar o fluxo de trabalho de aprendizado de máquina. O script então importa as bibliotecas necessárias e carrega o conjunto de dados consolidado. Posteriormente, concentra-se em uma ação específica (ABEV3 neste exemplo) filtrando o conjunto de dados com base no Ticker. As colunas relacionadas ao tempo são transformadas em um formato de data e hora, e o conjunto de dados é configurado para ter a coluna 'Ano' como índice, criando uma estrutura de série temporal. A configuração da série temporal é iniciada, especificando a variável alvo ('P/L'), o horizonte de previsão e o período sazonal anual. O script então compara vários modelos de série temporal usando a funções presentes da biblioteca gerando as métricas para comparação.

Com a geração da tabela de comparação, geradas pela comparação entre os mo-

delos preditivos disponibilizados pelo treinamento realizado com a biblioteca e a série temporal, obtemos todas as principais métricas utilizadas para verificar a eficácia de modelos de aprendizado de máquina ordenadas pelo melhor desempenho. Com base nessas métricas, é realizado a escolha do modelo com melhor desempenho entre eles e em seguida o script cria um modelo de aprendizado de máquina usando o algoritmo selecionado. Esta etapa é crucial para identificar o modelo mais adequado para previsão com base na variável alvo especificada e nas características do conjunto de dados.

A etapa final envolve treinar o modelo selecionado em todo o conjunto de dados. Uma vez finalizado o modelo, ele é aplicado para prever valores futuros. Esta etapa gera previsões para o horizonte de previsão especificado, permitindo uma avaliação do desempenho do modelo na previsão da variável alvo ('P/L') ao longo do tempo. O resultado desta etapa serve de base para análises adicionais e tomadas de decisão no domínio financeiro.

## 2.2. Coleta e Preparação de Dados

Os dados fundamentalistas históricos de ações da B3 (Bolsa de Valores do Brasil) utilizados para realizar a modelagem foram coletados a partir da plataforma Oceans14[OCEANS14 2024], conforme a Figura 2:

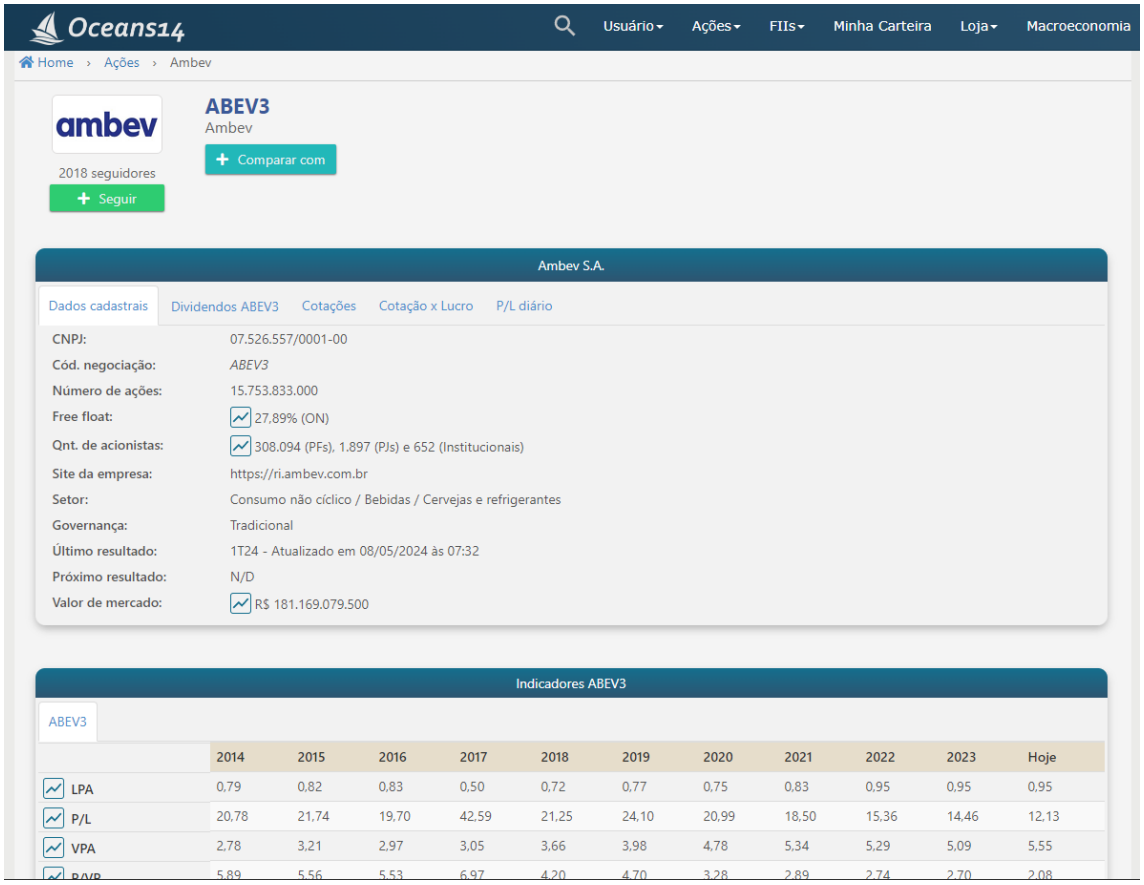



**Figura 2. Ocean14**

Para realizar a coleta desses dados foram utilizadas técnicas de Scrapy juntamente com manipulação de dados em Python, utilizando diversas bibliotecas. Assim sendo

possível salvar esses dados em arquivos CSV e durante o processo utilizar ciclo proxies para evitar bloqueios durante as requisições.

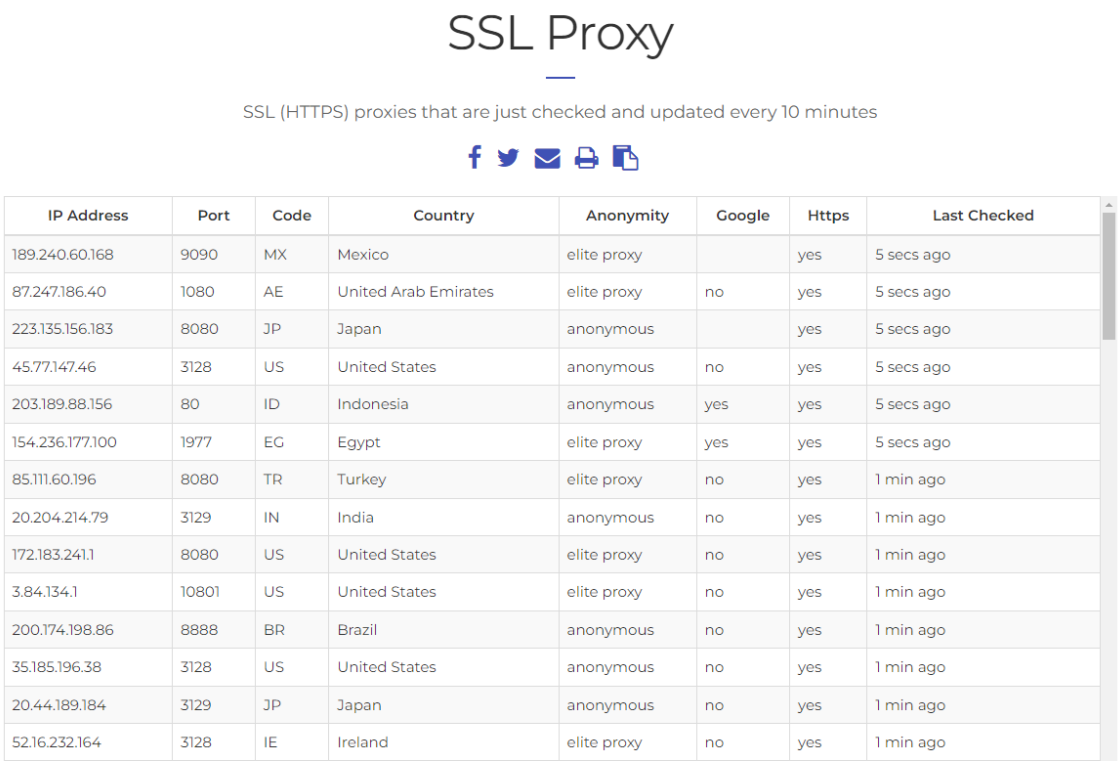
### 2.2.1. Bibliotecas Utilizadas

- requests: para realizar requisições HTTP.
- fromstring (da biblioteca lxml.html): para converter páginas HTML em strings.
- cycle (da biblioteca itertools): para criar um ciclo de proxies.
- BeautifulSoup (da biblioteca bs4): para filtrar e converter os conteúdos das tags HTML.
- pandas: para manipulação e exportação de dados em formato tabular.
- time e random: para randomizar a quantidade de requisições e o tempo entre elas.
- warnings: para gerenciar avisos durante a execução do código.

### 2.2.2. Ciclo Proxies

Para evitar o bloqueio das requisições e otimizar a utilização dos proxies, conforme na Figura 3, pela quantidade de requestes a serem realizadas, foi utilizado a técnica de ciclo de proxies. O ciclo de proxies envolve a utilização de uma lista de proxies, alternando entre eles para cada nova requisição. Essa técnica foi implementada da seguinte forma:

- **Coleta de Proxies**: Proxies são coletados de fontes públicas, como sites que disponibilizam listas de proxies (por exemplo, https://www.sslproxies.org/).
- **Armazenamento em um Ciclo**: A lista de proxies é armazenada em uma estrutura de dados que permite ciclar por eles, como itertools.cycle em Python. Isso cria um iterador que pode ser percorrido infinitamente.



SSL Proxy

SSL (HTTPS) proxies that are just checked and updated every 10 minutes

| IP Address | Port | Code | Country | Anonymity | Google | Https | Last Checked |
|---|---|---|---|---|---|---|---|
| 189.240.60.168 | 9090 | MX | Mexico | elite proxy | | yes | 5 secs ago |
| 87.247.186.40 | 1080 | AE | United Arab Emirates | elite proxy | no | yes | 5 secs ago |
| 223.135.156.183 | 8080 | JP | Japan | anonymous | | yes | 5 secs ago |
| 45.77.147.46 | 3128 | US | United States | anonymous | no | yes | 5 secs ago |
| 203.189.88.156 | 80 | ID | Indonesia | anonymous | yes | yes | 5 secs ago |
| 154.236.177.100 | 1977 | EG | Egypt | elite proxy | yes | yes | 5 secs ago |
| 85.111.60.196 | 8080 | TR | Turkey | elite proxy | no | yes | 1 min ago |
| 20.204.214.79 | 3129 | IN | India | anonymous | no | yes | 1 min ago |
| 172.183.241.1 | 8080 | US | United States | elite proxy | no | yes | 1 min ago |
| 3.84.134.1 | 10801 | US | United States | elite proxy | no | yes | 1 min ago |
| 200.174.198.86 | 8888 | BR | Brazil | anonymous | no | yes | 1 min ago |
| 35.185.196.38 | 3128 | US | United States | anonymous | no | yes | 1 min ago |
| 20.44.189.184 | 3129 | JP | Japan | anonymous | no | yes | 1 min ago |
| 52.16.232.164 | 3128 | IE | Ireland | elite proxy | no | yes | 1 min ago |

**Figura 3. SSL Proxies**

- **Requisições Alternadas**: Para cada requisição HTTP, um proxy diferente é selecionado do ciclo. Se um proxy falhar (por exemplo, estiver bloqueado ou não responder), o próximo proxy do ciclo é utilizado.
- **Troca de Proxy em Caso de Bloqueio**: Se um site bloquear um proxy ou a requisição falhar, o código automaticamente troca para o próximo proxy no ciclo e tenta novamente.

A função get_proxies(qtd_min) foi desenvolvida extrai proxies da página https://www.sslproxies.org/, esta página disponibiliza proxies com tecnologia SSL necessária. A função realiza requisições para coletar todos os proxies disponíveis e aguarda para que a página seja recarregada a cada 10 minutos com novos proxies.

```
def get_proxies(qtd_min):
    url = 'https://www.sslproxies.org/'
    proxies = set()
    while True:
        response = requests.get(url)
        parser = fromstring(response.text)
        for i in parser.xpath('//tbody/tr')[:100]:
            if i.xpath('.//td[7][contains(text(),"yes")]'):
                proxy = ":".join([i.xpath('.//td[1]/text()'
                    )[0], i.xpath('.//td[2]/text()')[0]])
                proxies.add(proxy)
        if len(proxies) < qtd_min:
```

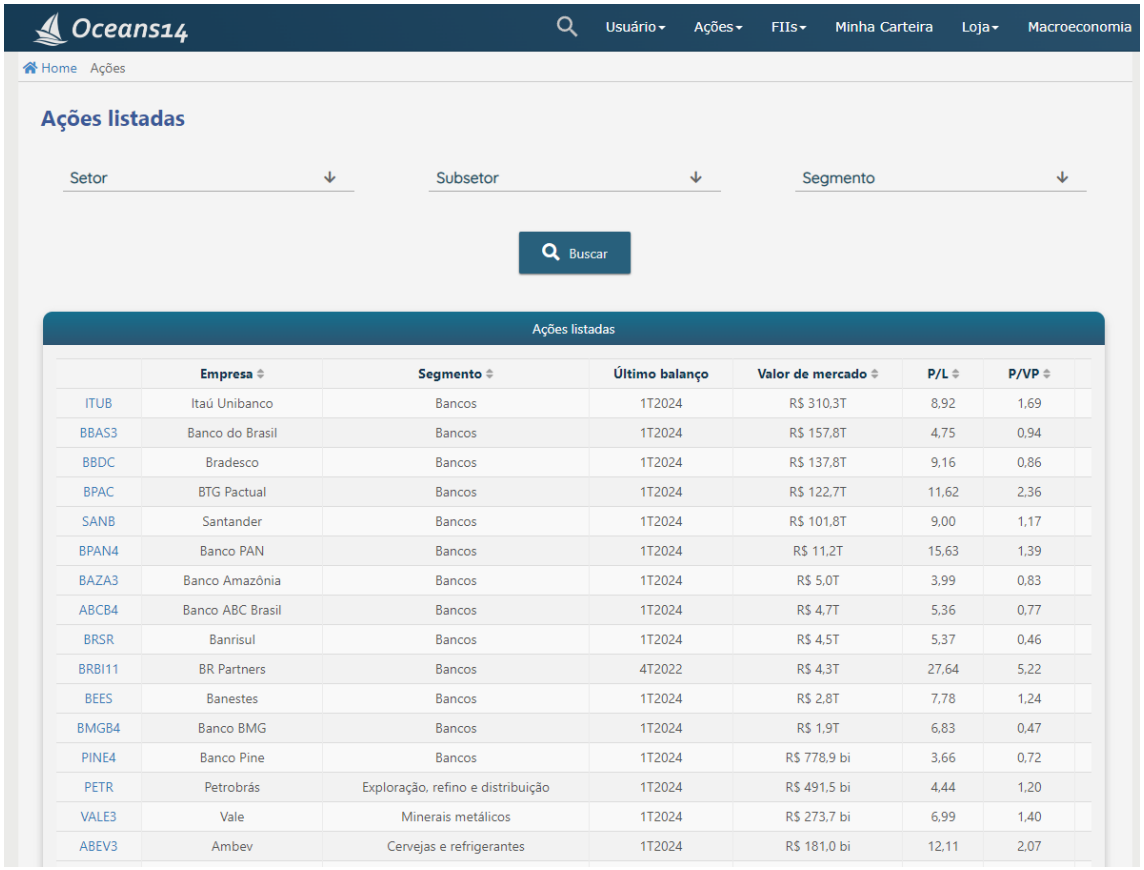
```
            for sleeping in range(600,0,-1):#Time to free
                proxies refresh
                time.sleep(1)
                clear_output()
                print(str(len(proxies)/qtd_min*100) + '%')
                print('New Request in ' + str(sleeping) + '
                    ...')
            continue
        else:
            break
    clear_output()
    print('Done...' + str(len(proxies)) + 'proxies!')
    return proxies
```

Esse processo se repete até que a quantidade necessária de proxies (passada como parâmetro), mais uma porcentagem extra para compensar a baixa qualidade de alguns proxies, seja atingida, além de armazenar os proxies em um conjunto para garantir que não haja duplicatas.

Para evitar a perda, código salva os proxies em arquivos CSV para uso futuro, facilitando a continuidade do processo em caso de interrupções.

### 2.2.3. Extração de Dados das Ações

Para realizar a modelagem dos dados fundamentalistas das ações, é essencial coletar as URLs que direcionam para as páginas específicas de cada ação no site Oceans14. A opção por utilizar o Ocean14 em vez das APIs da B3 deve-se à instabilidade das APIs da B3, à sua indisponibilidade para pessoas físicas e à falta de dados fundamentalistas consolidados. Primeiro, o código faz uma requisição HTTP para a página principal das ações (https://www.oceans14.com.br/acoes).




| | Empresa | Segmento | Último balanço | Valor de mercado | P/L | P/VP |
|---|---|---|---|---|---|---|
| ITUB | Itaú Unibanco | Bancos | 1T2024 | R$ 310,3T | 8,92 | 1,69 |
| BBAS3 | Banco do Brasil | Bancos | 1T2024 | R$ 157,8T | 4,75 | 0,94 |
| BBDC | Bradesco | Bancos | 1T2024 | R$ 137,8T | 9,16 | 0,86 |
| BPAC | BTG Pactual | Bancos | 1T2024 | R$ 122,7T | 11,62 | 2,36 |
| SANB | Santander | Bancos | 1T2024 | R$ 101,8T | 9,00 | 1,17 |
| BPAN4 | Banco PAN | Bancos | 1T2024 | R$ 11,2T | 15,63 | 1,39 |
| BAZA3 | Banco Amazônia | Bancos | 1T2024 | R$ 5,0T | 3,99 | 0,83 |
| ABCB4 | Banco ABC Brasil | Bancos | 1T2024 | R$ 4,7T | 5,36 | 0,77 |
| BRSR | Banrisul | Bancos | 1T2024 | R$ 4,5T | 5,37 | 0,46 |
| BRBI11 | BR Partners | Bancos | 4T2022 | R$ 4,3T | 27,64 | 5,22 |
| BEES | Banestes | Bancos | 1T2024 | R$ 2,8T | 7,78 | 1,24 |
| BMGB4 | Banco BMG | Bancos | 1T2024 | R$ 1,9T | 6,83 | 0,47 |
| PINE4 | Banco Pine | Bancos | 1T2024 | R$ 778,9 bi | 3,66 | 0,72 |
| PETR | Petrobrás | Exploração, refino e distribuição | 1T2024 | R$ 491,5 bi | 4,44 | 1,20 |
| VALE3 | Vale | Minerais metálicos | 1T2024 | R$ 273,7 bi | 6,99 | 1,40 |
| ABEV3 | Ambev | Cervejas e refrigerantes | 1T2024 | R$ 181,0 bi | 12,11 | 2,07 |

**Figura 4. Lista De Ações**

Para isso, utilizamos a biblioteca requests configurada com cabeçalhos de requisição (headers) que simulam um navegador real, permitindo que as requisições pareçam ser feitas por um usuário humano, o que ajuda a evitar bloqueios pelo servidor do site.

Após obter o conteúdo da página, utilizamos a biblioteca BeautifulSoup para analisar o HTML retornado. A análise busca por todos os elementos ¡a¿ (links) no HTML que contenham a substring balanco-dividendos em seus atributos href. Esses links específicos são importantes porque direcionam para as páginas que contêm os dados fundamentalistas necessários para nossa modelagem.

```
url = 'https://www.oceans14.com.br/acoes'
page = req.get(url, headers=headers, timeout=15)
soup = BeautifulSoup(page.content, 'html.parser')
url_list = []
for url in soup.find_all('a'):
    if 'balanco-dividendos' in url.get('href'):
        url_list.append('https://www.oceans14.com.br' + url
            .get('href'))

url_list = list(set(url_list))
```

O código armazena todas essas URLs em uma lista (url_list). Para garantir a

eficiência do processo e evitar a necessidade de repetir a coleta das URLs em execuções futuras, a lista de URLs é salva em um arquivo de texto. Isso permite que o código leia esse arquivo em vez de fazer uma nova requisição e análise da página principal em execuções subsequentes.

Com a lista de URLs das páginas das ações e os proxies previamente obtidos, o próximo passo é coletar os dados fundamentalistas de cada ação. Para cada URL na lista, foi implementado um loop que segue as seguintes etapas:

- **Seleção de Proxy**: A partir de um ciclo de proxies disponíveis, foi selecionado um proxy para ser utilizado na requisição seguinte. Essa prática ajuda a diversificar os endereços IP utilizados, reduzindo a probabilidade de bloqueios por parte dos servidores.
- **Requisição HTTP**: Utilizando o proxy selecionado, foi realizada uma requisição HTTP para a URL da ação. Nesse processo, foram empregados cabeçalhos configurados previamente para simular o comportamento de um navegador convencional.
- **Análise do Conteúdo HTML**: Após obter a resposta da requisição, o conteúdo HTML foi analisado com auxílio da biblioteca BeautifulSoup. Os dados tabulares presentes nas páginas foram extraídos e manipulados utilizando o pandas, o que facilitou a conversão dessas informações em dataframes, garantindo assim uma estrutura organizada e de fácil análise.
- **Armazenamento dos Dados**: Os dados fundamentalistas obtidos, juntamente com os tickers das ações correspondentes, foram armazenados em arquivos CSV separados. Cada arquivo foi nomeado de acordo com o ticker da ação, proporcionando uma organização clara e acessível dos dados coletados.
- **Tratamento de Bloqueios**: Em situações onde o proxy utilizado foi bloqueado ou ocorreu falha na requisição, o código foi programado para alternar automaticamente para o próximo proxy disponível e tentar novamente. Esse mecanismo de tratamento de exceções contribuiu para a resiliência do processo de coleta de dados, minimizando interrupções e garantindo a continuidade da operação.

Ao longo do processo de coleta, as URLs já processadas foram removidas da lista para evitar requisições redundantes, garantindo assim a obtenção dos dados fundamentalistas das ações sem requisições desnecessárias.

Este processo de scraping garante a coleta completa dos dados fundamentalistas das ações da B3. A utilização de proxies, juntamente com a manipulação cuidadosa dos dados coletados, permite superar as restrições impostas pelos sites e garantir que os dados necessários sejam obtidos de maneira confiável.

### 2.2.4. Preparação de Dados

Após a captura dos dados foi realizado a manipulação e análise de dados para que os mesmos estivessem padronizados e normalizados de modo a serem utilizados para criação do modelo.

As bibliotecas pandas e glob foram utilizadas para manipulação e análise de dados e para buscar arquivos no sistema de arquivos, respectivamente. A biblioteca os.path foi

utilizada para manipulação de caminhos de arquivos. As bibliotecas matplotlib, seaborn e plotly foram utilizadas para visualização de dados.E a biblioteca pandas_profiling foi utilizada para gerar relatórios exploratórios de dados.

```
data_files = [path.normpath(i).replace('\\','/') for i in
    glob.glob(raw_data + '*.csv')]

full_df = pd.DataFrame()
for file_name in data_files:
    df = pd.read_csv(file_name, sep=';', header=None)
    df.iloc[0, 0] = 'Ano'
    df.iloc[:, 0] = df.iloc[:, 0].str.replace(' ', '')
    df = df.transpose()
    df.columns = df.iloc[0]
    df = df.drop(df.index[0])
    df['Ticker'] = path.splitext(path.basename(file_name))
        [0]
    full_df = pd.concat([full_df, df])
```

Para realizar a leitura dos arquivos CSV definimos o diretório onde os arquivos CSV estavam armazenados e utilizamos uma função para encontrar todos os arquivos CSV nesse diretório. Criamos um DataFrame vazio para armazenar os dados e, para cada arquivo CSV, lemos o arquivo, ajustamos os cabeçalhos e removemos caracteres indesejados. Adicionamos uma coluna para identificar o nome do arquivo e concatenamos os dados ao DataFrame principal, conforme a Tabela 1:

| Ano | LPA | P/L | VPA | P/VP | ... | Ticker |
|---|---|---|---|---|---|---|
| 2020 | 0.09 | 134.45 | 3.45 | 3.67 | ... | BOAS3 |
| 2021 | 0.09 | 130.65 | 3.49 | 3.31 | ... | BOAS3 |
| 1998 | 19.53 | 4.69 | 302.40 | 0.30 | ... | SBSP3 |
| 1999 | -8.23 | -25.78 | 290.85 | 0.73 | ... | SBSP3 |
| 2000 | 18.33 | 9.49 | 290.32 | 0.60 | ... | SBSP3 |

**Tabela 1. Topo do DataFrame com dados financeiros por ano**

Como limpeza dos dados, as colunas foram renomeadas para nomes mais compatíveis com padrões comuns, transformamos valores percentuais em valores numéricos e convertemos outras colunas relevantes para tipos numéricos apropriados. Além disso, foi verificado a quantidade de valores nulos em cada coluna, exibimos as linhas que continham valores nulos e removemos todas essas linhas. A Tabela 1 apresenta o resumo estatístico dos dados limpos e salvos como um arquivo CSV.

| | Ano | LPA | P/L | VPA | P/VP | ... | Volume_diario |
|---|---|---|---|---|---|---|---|
| count | 3187.00 | 3187.00 | 3187.00 | 3187.00 | 3187.00 | ... | 3187.00 |
| mean | 2013.77 | 6.27 | 11.54 | 67.91 | 2.12 | ... | 35.15 |
| std | 5.46 | 319.52 | 133.11 | 2940.51 | 30.99 | ... | 114.81 |
| min | 1998.00 | -666.67 | -3856.92 | -695.02 | -1714.19 | ... | 0.00 |
| 25% | 2010.00 | 0.16 | 4.24 | 4.22 | 0.82 | ... | 1.00 |
| 50% | 2015.00 | 0.88 | 9.88 | 8.88 | 1.52 | ... | 5.00 |
| 75% | 2018.00 | 2.26 | 20.01 | 18.39 | 2.85 | ... | 27.00 |
| max | 2021.00 | 18000.00 | 928.23 | 166000.00 | 133.39 | ... | 2913.00 |

**Tabela 2. Estatísticas descritivas dos dados financeiros**

Por último foi utilizado a biblioteca pandas_profiling para gerar relatórios exploratórios e exibindo-o dentro do notebook. Mas dentre as análises geradas está a análise de Coeficiente de correlação de Pearson entre as colunas, que podem ser usadas como entradas adicionais na criação do modelo. Na Figura 5 é possível ver a correlação de forma gráfica em um mapa de calor, onde é possível verificar a alta correlação positiva entre algumas das colunas.

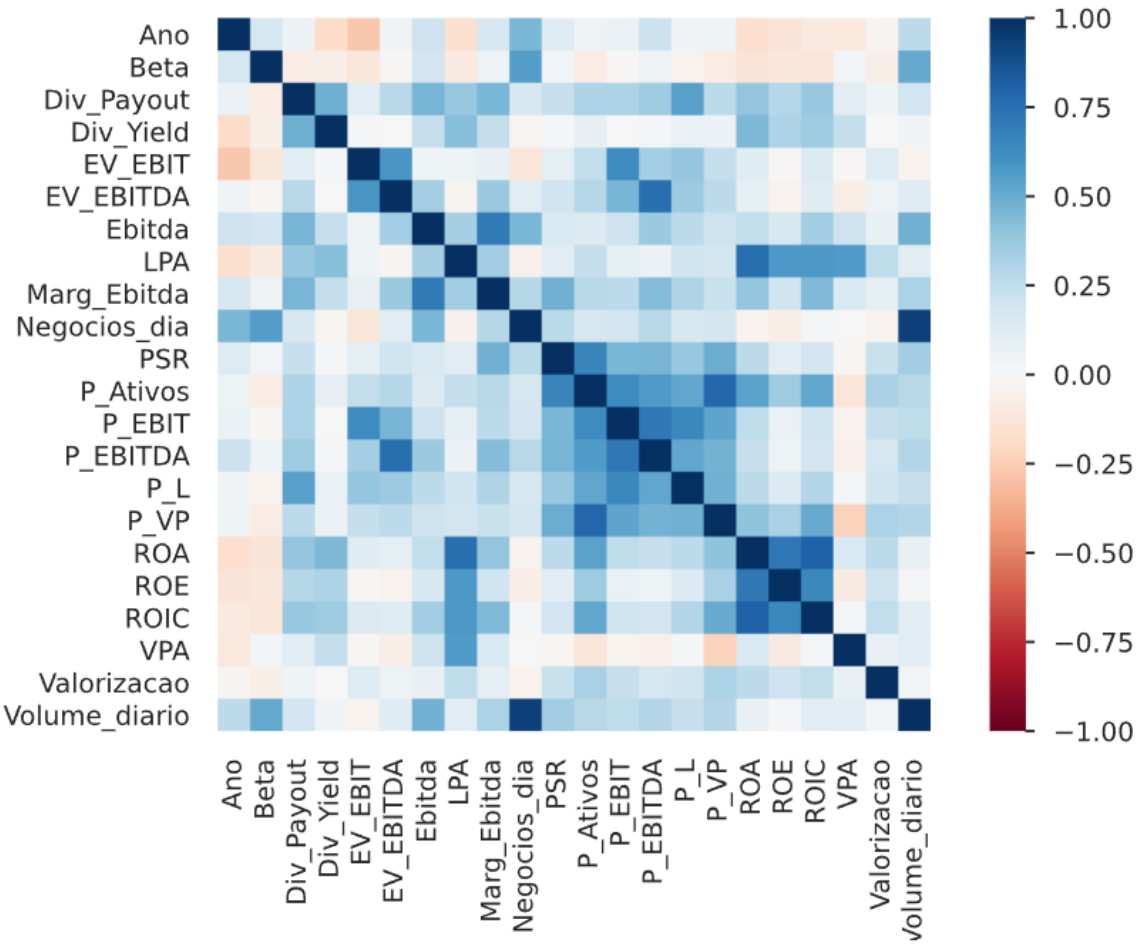



**Figura 5. Mapa de Calor - Correlação**

### 2.3. Comparação e Avaliação de Modelos

Para realizar a criação dos modelos foram utilizados duas bibliotecas públicas, sendo elas PyCaret e Prophet e para ter um maior histórico foi escolhido a métrica lucro/prejuízo (P/L) como foco da previsão para a ação ABEV3, para ambos os modelos.

PyCaret e Prophet são duas poderosas ferramentas utilizadas na análise de dados e previsão, cada uma com suas particularidades e vantagens. PyCaret é uma biblioteca de aprendizado de máquina de baixo código que visa simplificar o ciclo de vida do desenvolvimento de modelos. Ela automatiza as tarefas mais comuns de pré-processamento de dados, seleção de modelos, ajuste de hiperparâmetros e avaliação de desempenho. Com uma interface amigável e intuitiva, PyCaret permite que tanto iniciantes quanto profissionais experientes implementem soluções de aprendizado de máquina de forma rápida e eficiente. Além disso, a biblioteca oferece suporte a uma ampla gama de algoritmos de classificação, regressão, clustering e séries temporais, tornando-se uma escolha versátil para diversas aplicações.

Por outro lado, Prophet é uma ferramenta específica para previsão de séries temporais, desenvolvida pelo Facebook. É especialmente adequada para dados que exibem padrões sazonais e tendências não lineares. A grande vantagem do Prophet é sua facilidade de uso e capacidade de lidar com séries temporais com faltas de dados e mudanças

abruptas nos padrões históricos. Ele permite a inclusão de componentes adicionais como feriados e eventos especiais, que podem influenciar a previsão. Diferentemente do PyCaret, que é uma plataforma de aprendizado de máquina mais generalista, Prophet é focado exclusivamente em séries temporais, proporcionando uma abordagem robusta e especializada para esse tipo de análise. Em resumo, enquanto PyCaret oferece uma solução abrangente e automatizada para diversas tarefas de aprendizado de máquina, Prophet se destaca como uma ferramenta poderosa e intuitiva para a previsão de séries temporais complexas.

- **Carregamento e Pré-processamento dos Dados**: O primeiro passo foi o carregamento dos dados da série histórica da ação ABEV3 a partir de um arquivo CSV. Em seguida, foi realizada algumas etapas de pré-processamento para garantir a qualidade e consistência dos dados. Isso incluiu a remoção de colunas irrelevantes, como o 'Ticker', e a conversão da coluna 'Ano' para o formato datetime, representando o último dia do ano, no caso do PyCaret. E a renomeação das colunas 'Ano' para 'ds' e 'P_L' para 'y' para atender ao formato esperado pelo Prophet.
- **Configuração do Ambiente de Modelagem**: Utilizando PyCaret foi necessário configurar o ambiente de modelagem. A função setup() foi utilizada para definir o conjunto de dados, a variável alvo ('P_L' - lucro/prejuízo), o horizonte de previsão (3 períodos) e o período sazonal (anual). Isso preparou o cenário para a modelagem e avaliação dos modelos de previsão disponíveis. Já para Prophet foi apenas necessário configurar o intervalo de confiança, definido como 95% (interval_width=0.95).
- **Comparação e Seleção de Modelos**: Após a configuração do ambiente, foi realizada uma comparação entre diferentes modelos de previsão disponíveis no PyCaret utilizando a função compare_models(). Esse processo permitiu identificar o modelo mais adequado para os nossos dados e objetivos de previsão. Essa etapa foi apenas necessária utilizando PyCaret já que essa biblioteca low-code disponibiliza vários tipos de modelos a partir de uma execução.
- **Treinamento e Avaliação do Modelo Escolhido**: O modelo escolhido no PyCaret foi o 'ada_cds_dt' (AdaBoost w/ Cond. Deseasonalize & Detrending). Ele foi treinado utilizando todos os dados disponíveis, visando capturar padrões e tendências relevantes para a previsão do P/L da ação ABEV3. Essa etapa não foi necessária utilizando Prophet já que o mesmo possue sua própria estrutura de modelagem.

```
# Treina e avalia o desempenho do modelo AdaBoost w/
    Cond. Deseasonalize & Detrending
model = create_model('ada_cds_dt')
# Treina o modelo em todo o conjunto de dados
final = finalize_model(model)
```

Foram utilizadas as métricas MAE, RMSE e MAPE para avaliar e comparar os modelos criados por ambas as blibliotecas (PyCaret e Prophet). Como é possivel ver na Tabela 3 abaixo a comparação dos resultados:

Comparando os dois modelos, o PyCaret apresentou um MAE de 4.4108, enquanto o Prophet teve um MAE de 11.35, isso indica que o PyCaret é significativamente mais preciso, com erros médios absolutos menores. Em relação ao RMSE, o PyCaret

| | Model | MASE | RMSSE | MAE | RMSE | MAPE | SMAPE | R2 | TT (Sec) |
|---|---|---|---|---|---|---|---|---|---|
| ada_cds_dt | AdaBoost w/ Cond. Deseasonalize & Detrending | 0.3314 | 0.2455 | 4.0359 | 4.7855 | 0.1528 | 0.1534 | -1.0488 | 0.1033 |
| dt_cds_dt | Decision Tree w/ Cond. Deseasonalize & Detrending | 0.3693 | 0.2764 | 4.6688 | 5.4858 | 0.1886 | 0.1803 | -5.4955 | 0.0667 |
| gbr_cds_dt | Gradient Boosting w/ Cond. Deseasonalize & Detrending | 0.4071 | 0.3011 | 5.3934 | 6.1367 | 0.2279 | 0.2109 | -4.6135 | 0.0867 |
| croston | Croston | 0.4855 | 0.4069 | 6.1327 | 8.0428 | 0.2189 | 0.2634 | -5.5487 | 0.9700 |
| omp_cds_dt | Orthogonal Matching Pursuit w/ Cond. Deseasonalize & Detrending | 0.4935 | 0.3868 | 6.4503 | 7.8336 | 0.2483 | 0.2905 | -6.1378 | 0.0733 |
| rf_cds_dt | Random Forest w/ Cond. Deseasonalize & Detrending | 0.5188 | 0.3735 | 7.0682 | 7.7532 | 0.3080 | 0.2693 | -6.8496 | 0.1600 |
| knn_cds_dt | K Neighbors w/ Cond. Deseasonalize & Detrending | 0.5239 | 0.3693 | 6.9852 | 7.5678 | 0.2955 | 0.2673 | -4.1734 | 0.1167 |
| et_cds_dt | Extra Trees w/ Cond. Deseasonalize & Detrending | 0.8405 | 0.5942 | 12.4053 | 12.9886 | 0.5694 | 0.3912 | -31.1743 | 0.1333 |
| lightgbm_cds_dt | Light Gradient Boosting w/ Cond. Deseasonalize & Detrending | 0.9316 | 0.6142 | 12.7937 | 12.8881 | 0.5872 | 0.4510 | -34.1359 | 0.1500 |
| huber_cds_dt | Huber w/ Cond. Deseasonalize & Detrending | 2.5535 | 2.3093 | 38.9337 | 51.8814 | 1.6820 | 0.6192 | -700.2671 | 0.0733 |
| br_cds_dt | Bayesian Ridge w/ Cond. Deseasonalize & Detrending | 5.1620 | 4.1492 | 71.5320 | 87.4444 | 3.2847 | 1.0174 | -7103.4816 | 0.0667 |
| lasso_cds_dt | Lasso w/ Cond. Deseasonalize & Detrending | 5.9755 | 4.9127 | 84.4907 | 104.9166 | 3.8626 | 1.0199 | -7905.8301 | 0.0767 |
| llar_cds_dt | Lasso Least Angular Regressor w/ Cond. Deseasonalize & Detrending | 5.9756 | 4.9128 | 84.4928 | 104.9191 | 3.8627 | 1.0199 | -7906.4499 | 0.0867 |
| en_cds_dt | Elastic Net w/ Cond. Deseasonalize & Detrending | 6.2986 | 5.1771 | 88.9072 | 110.4189 | 4.0693 | 1.0273 | -9061.0502 | 0.1967 |
| arima | ARIMA | 6.4860 | 5.5303 | 95.4667 | 121.2174 | 4.2967 | 0.8982 | -6458.0878 | 1.3033 |
| auto_arima | Auto ARIMA | 7.2202 | 6.0812 | 105.3001 | 132.5195 | 4.7567 | 0.9265 | -8816.2072 | 1.3767 |
| ridge_cds_dt | Ridge w/ Cond. Deseasonalize & Detrending | 10.1477 | 8.8248 | 141.2764 | 185.8697 | 6.5565 | 1.0841 | -35227.2816 | 0.1833 |
| lr_cds_dt | Linear w/ Cond. Deseasonalize & Detrending | 15.0229 | 14.1070 | 207.7944 | 295.0811 | 9.7314 | 1.1692 | -103778.8168 | 0.1767 |

**Figura 6. Tabela de Comparação PyCaret**

| Modelo | MAE | RMSE | MAPE |
|---|---|---|---|
| PyCaret (AdaBoost w/ Cond. Deseasonalize & Detrending) | 4.4108 | 5.2234 | 0.1756 |
| Prophet | 11.35 | 12.86 | 0.51 |

**Tabela 3. Comparação de métricas entre modelos**

registrou 5.2234, enquanto o Prophet registrou 12.86, mostrando que o PyCaret tem menor variância nos erros. No que se refere ao MAPE, o PyCaret obteve 0.1756 (15.28%) comparado aos 0.51 (51%) do Prophet, indicando que o PyCaret possui uma menor porcentagem de erro absoluto.

Desse modo podemos concluir que o modelo PyCaret (AdaBoost com Cond. Deseasonalize & Detrending) apresenta uma qualidade muito superior ao Prophet em todas as métricas avaliadas. Os erros absolutos e percentuais do PyCaret são significativamente menores, indicando maior precisão e menor variabilidade nos erros. Portanto, o PyCaret é claramente o modelo de melhor qualidade baseado nas métricas fornecidas.

## 2.4. Análise dos Resultados com Novos Dados

Após a geração dos modelos foram realizados novas execuções para prever o valor do 'P_L' e realizar a comparação dos resultados com os valores reais e assim pode avaliar como os modelos se comportam em termos práticos.

Para realizar previsões futuras, foi preparado os dados de entrada utilizando um conjunto separado de dados específico para previsão. Novamente, foi realizada etapas de pré-processamento, como a remoção de colunas desnecessárias e a conversão da coluna 'Ano' para o formato datetime, para ambas as bibliotecas.

```
pycaretResults = predict_model(final, fh = 3, X=
    ABEV3_df_topredict)
```

Com os modelos treinados e os dados de entrada preparados, foi feita a previsão

do P/L da ação ABEV3 para os próximos 3 períodos. Os resultados da previsão foram armazenados em um DataFrame chamado pycaretResults, fornecendo os valores absolutos.

| Modelo/Ano | 2022 | 2023 | 2024 |
|---|---|---|---|
| Valor Real | 15.36 | 14.46 | 13.05 |
| PyCaret (AdaBoost w/ Cond. Deseasonalize & Detrending) | 33.6254 | 31.3977 | 31.9997 |
| Prophet | 25.005587 | 21.076038 | 34.900937 |

**Tabela 4. Comparação das previsões entre modelos**

Primeiro, analisamos o modelo PyCaret (AdaBoost w/ Cond. Deseasonalize & Detrending). Em 2022, a previsão foi de 33.6254, enquanto o valor real foi de 15.36, resultando em uma diferença absoluta de 18.2654. Em 2023, a previsão foi de 31.3977, com um valor real de 14.46, resultando em uma diferença absoluta de 16.9377. Em 2024, a previsão foi de 31.9997, com um valor real de 13.05, resultando em uma diferença absoluta de 18.9497.

Para o modelo Prophet, em 2022, a previsão foi de 25.005587, enquanto o valor real foi de 15.36, resultando em uma diferença absoluta de 9.645587. Em 2023, a previsão foi de 21.076038, com um valor real de 14.46, resultando em uma diferença absoluta de 6.616038. Em 2024, a previsão foi de 34.900937, com um valor real de 13.05, resultando em uma diferença absoluta de 21.850937.

Resumindo as diferenças absolutas, o modelo PyCaret teve um erro absoluto de 18.2654 em 2022, 16.9377 em 2023 e 18.9497 em 2024, resultando em um total de 54.1528 e uma média de 18.0509. O modelo Prophet teve um erro absoluto de 9.645587 em 2022, 6.616038 em 2023 e 21.850937 em 2024, resultando em um total de 38.112562 e uma média de 12.704187.

| Modelo/Ano | 2022 | 2023 | 2024 | Média |
|---|---|---|---|---|
| PyCaret | 18.2654 | 16.9377 | 18.9497 | 18.0509 |
| Prophet | 9.645587 | 6.616038 | 21.850937 | 12.704187 |

**Tabela 5. Erros Absolutos Médios (MAE) dos Modelos de Série Temporal**

Em termos de erro absoluto médio (MAE), o modelo Prophet teve um desempenho melhor, com um MAE de 12.704187 comparado ao MAE de 18.0509 do modelo PyCaret. Analisando cada ano separadamente, em 2022, o Prophet teve um desempenho significativamente melhor que o PyCaret. Em 2023, o Prophet novamente teve um desempenho melhor. Em 2024, ambos os modelos apresentaram grandes erros absolutos, mas o Prophet teve o maior erro.

Em conclusão, o modelo Prophet apresentou um desempenho geral melhor, com menores diferenças absolutas em dois dos três anos analisados. No entanto, ambos os modelos apresentaram grandes desvios em 2024, indicando possíveis limitações na captura de padrões futuros ou mudanças bruscas nas séries temporais.

Para melhorar a análise, seria interessante testar outros modelos ou ajustar parâmetros para melhorar as previsões.

## 3. Conclusão

Neste trabalho, foi desenvolvido e avaliado modelos preditivos para gerar um portfólio de investimento previdenciário alinhado ao perfil do usuário, buscando o melhor retorno possível dentro dos níveis de volatilidade permitidos.

Para atingir os objetivos específicos, foi realizada uma análise abrangente do estado da arte em ciência de dados e séries temporais, seguida pela coleta e tratamento dos dados necessários para a análise. Foram desenvolvidos modelos preditivos baseados em dados fundamentalistas, utilizando bibliotecas como PyCaret e Prophet, e conduzidos experimentos para testar a eficácia desses modelos. Finalmente, foram analisados os resultados obtidos com os experimentos para avaliar o desempenho dos modelos.

O processo de desenvolvimento foi dividido em três fases principais. A primeira fase, de coleta e preparação de dados, envolveu a obtenção de dados fundamentalistas históricos de ações da B3 a partir da plataforma Oceans14. Foram utilizadas técnicas de scraping e manipulação de dados em Python para coletar e preparar os dados, garantindo a padronização e normalização necessários para a criação dos modelos.

Na segunda fase, os esforços foram concentrados na comparação e avaliação de modelos. Utilizando a biblioteca PyCaret, foi configurado o ambiente de modelagem e realizada a comparação de vários modelos preditivos. O modelo AdaBoost com Cond. Deseasonalize & Detrending foi escolhido como o de melhor desempenho, com base nas métricas de avaliação como MAE, RMSE e MAPE. Paralelamente, foi utilizada a biblioteca Prophet para desenvolver um modelo alternativo e compará-lo com o modelo do PyCaret.

Os resultados mostraram que o modelo PyCaret superou significativamente o modelo Prophet em todas as métricas avaliadas. O PyCaret apresentou menor erro absoluto médio (MAE), menor raiz do erro quadrático médio (RMSE) e menor porcentagem de erro absoluto médio (MAPE), indicando maior precisão e menor variabilidade nos erros. Portanto, foi concluído que o modelo PyCaret é mais adequado para previsão de P/L da ação ABEV3.

Na fase final, foram realizadas novas previsões com os modelos treinados utilizando novos dados. O desempenho prático dos modelos foi avaliado, observando que o Prophet obteve leve superioridade em termos de precisão, mas sem consistência das previsões. Assim, de forma limitada pelo aumento da imprecisão conforme o aumento do horizonte de previsão, é possível utilizar o modelo para realizar previsões dos indicadores fundamentalistas para serem utilizados nas análises microeconômicas das companhias e tomar decisões de compra ou venda mais bem informadas.

Em resumo, o modelo PyCaret (AdaBoost com Cond. Deseasonalize & Detrending) demonstrou ser uma ferramenta poderosa para previsão de séries temporais no contexto de investimentos previdenciários, mas que possui muitas limitações a serem aperfeiçoadas para serem viáveis e entreguem resultados úteis. A utilização de técnicas avançadas de coleta e preparação de dados, combinadas com uma análise detalhada de modelos preditivos, permitiu desenvolver um modelo robusto, mas pouco preciso em termos absolutos, o que é um ponto crucial para este caso. Esses resultados destacam a importância de escolher modelos adequados e realizar uma avaliação rigorosa para alcançar previsões confiáveis e úteis no mercado financeiro.

Como sugestões para trabalhos futuros, para melhorar a precisão e eficácia do modelo, sugere-se explorar outros modelos de aprendizado de máquina e técnicas de otimização de hiperparâmetros, utilizando uma variedade maior de ações além da ABEV3. Além disso, a inclusão de mais variáveis fundamentalistas e técnicas avançadas de tratamento de dados pode ajudar a capturar melhor as nuances do mercado financeiro. Finalmente, a realização de estudos comparativos com diferentes horizontes de previsão e períodos sazonais pode fornecer percepções adicionais sobre o desempenho dos modelos preditivos.

## 4. Referências

## Referências


AALST, W. (2016). Process mining. In *Process mining*. Springer.

BARCELLOS, M. & AZEVEDO, S. (2018). Histórias do mercado de capitais no brasil: depoimentos inéditos de personalidades que marcaram a trajetória das bolsas de valores no país. In *Histórias do mercado de capitais no Brasil: depoimentos inéditos de personalidades que marcaram a trajetória das bolsas de valores no país*. Alta Books.

DHAR, V. (2013). Data science and prediction. *Communications of the ACM*, 56 (12):64–73.

ELMERRAJI, J. (2022). Guide to financial ratios. `https://www.investopedia.com/articles/stocks/06/ratios.asp`.

GOETZMANN, W. N. & ROUWENHORST, K. G. (2005). The origins of value: The financial innovations that created modern capital markets. In *The Origins of Value: The Financial Innovations that Created Modern Capital Markets*. Oxford University Press.

KAHNEMAN, D. (2011). Thinking, fast and slow. In *Thinking, Fast and Slow*. Farrar, Straus and Giroux.

OCEANS14 (2024). Oceans14. `https://www.oceans14.com.br/`. Accessed: 2024-06-05.

SEGAL, T. (2022). Fundamental analysis. `https://www.investopedia.com/terms/f/fundamentalanalysis.asp`.