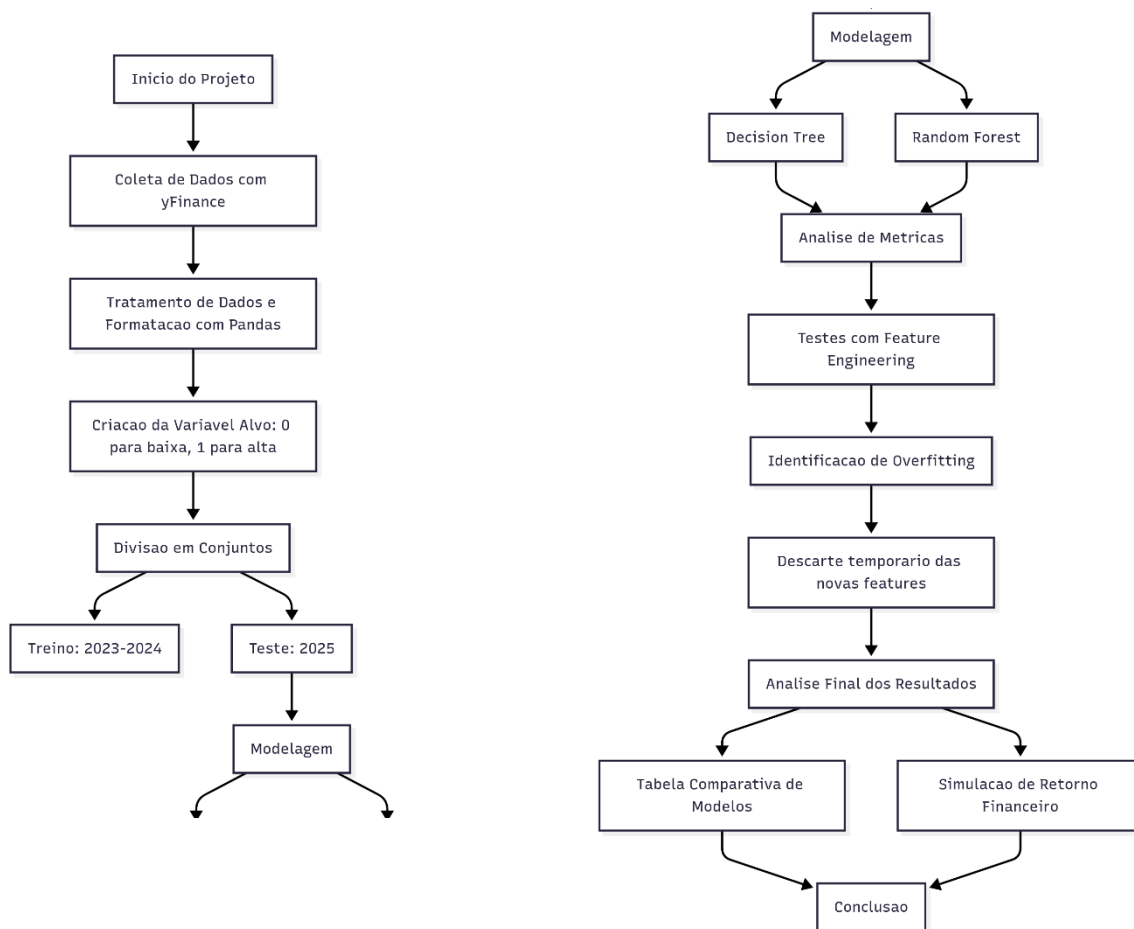
 <b>UNIFEI</b> <small>UNIVERSIDADE FEDERAL DE ITAJUBÁ</small>	<b>PROFESSOR (A):</b> EDUARDO JABBUR MACHADO	
	<b>CURSO:</b> BACHARELADO EM CIÊNCIA E TECNOLOGIA (BCTEC)	
	<b>DISCIPLINAS:</b> BCTEC015_2025 - PROGRAMAÇÃO PARA CIÊNCIA DE DADOS	
	<b>PERÍODO:</b> 3º	<b>DATA:</b> 27/06/2025
	<b>ALUNO:</b> KENNYSON CHAVES FLORENCIO	

## Relatório – Introdução ao Machine Learning

### Resumo

Este relatório apresenta o desenvolvimento de um modelo supervisionado de Machine Learning com o objetivo de prever a direção diária do preço do ativo BBAS3 (Banco do Brasil) na B3. Foram utilizados dados históricos de 2023 e 2024 para treinamento e dados de 2025 para teste. O modelo escolhido foi o Decision Tree, com avaliação baseada em métricas como acurácia, precisão, recall e F1-score. O algoritmo demonstrou melhor desempenho geral, com acurácia de 79,3%, se mostrando assertivo. Apesar da boa performance estatística, os retornos financeiros simulados foram baixos, reforçando a necessidade de abordagens complementares para melhorar a lucratividade.



Fluxograma 1 – Resumo das aplicações

## Introdução

A análise de dados é cada vez mais presente e necessária no nosso dia a dia, principalmente no mercado financeiro em que as decisões são cada vez mais tomadas com base em dados, do que em especulação como já foi um dia. Sendo assim surge a necessidade da criação de algoritmos inteligentes capazes de analisar dados em massa e gerar resultados assertivos e relevantes ao fim desejado.

Este experimento tem como objetivo o desenvolvimento e avaliação de um modelo supervisionado de Machine Learning que seja capaz de prever a direção do movimento diário de um ativo da B3 (bolsa de valores brasileira). A meta é identificar, com base em dados históricos diários de preços, se o ativo irá fechar em alta (1) ou baixa (0) no mesmo dia.

O ativo BBAS3 foi o escolhido para a análise, usando seus dados históricos de 2023 e 2024 para treinamento do modelo e 2025 para teste de eficácia. A classe alvo foi definida com base na diferença entre os preços de fechamento e abertura de cada dia. Os modelos de algoritmo supervisionados usados nessa análise foram o Decision Tree e Random Forest, além de testes com Feature Engineering

## Metodologia

### 1. Coleta e Preparação dos Dados

Através da biblioteca yfinance utilizada no python foi coletada a base histórica do ativos BBAS3 (Banco do Brasil, com granularidade diária, contendo os dados de Close, High, Low, Open e Volume do ativo.

### 2. Tratamento dos dados e criação do alvo

Após a extração dos dados, foi necessário o tratamento dos mesmos, para facilitar o uso e a implementação dos dados em dataframe da lib Pandas. No mesmo tratamento e ajuste de colunas também foi adicionado o “Target” que seria o alvo do modelo de aprendizagem para sua análises, sendo ele binário 0 para baixa e 1 para alta.

### 3. Modelos de Aprendizagem

Como ponto de partida, foi utilizada uma abordagem básica, empregando apenas os atributos originais da base: Open, High, Low, Close, Volume e Target. Essa estrutura inicial permitiu o treinamento de dois modelos supervisionados clássicos fornecido pelo SkLearn:

- Decision Tree (Árvore de Decisão) – modelo simples e interpretável, usado como benchmark inicial;
- Random Forest (Floresta Aleatória) – modelo mais robusto, baseado em múltiplas árvores, utilizado para comparação.

Os modelos foram avaliados utilizando os dados de teste (2025), considerando as seguintes métricas de desempenho:

- Acurácia
- Precisão
- Recall (Sensibilidade)

- F1-Score
- Matriz de Confusão

---

#### 4. Engenharia de Atributos (Feature Engineering)

Com base na acurácia do modelo, foi verificada a possibilidade de melhoria do algoritmo, então foram acrescentados atributos como retorno diário, amplitude, retorno acumulados, volatilidade e volume relativo. A criação dessas features teve como objetivo fornecer ao modelo contexto histórico e comportamental, permitindo que ele identificasse padrões e tendências com maior precisão.

Porém as mesmas geraram overfitting, ou seja entregaram dados além do que o modelo podia receber, gerando uma acurácia de 100% no período de teste. Sendo assim não seria aplicável em um cenário real, pelo menos da maneira testada até o momento, visto isso o feature engineering foi deixado de lado nessa análise para fins de simplificação.

### Resultados e Discussões

Nesta etapa, foram avaliados dois modelos supervisionados: Decision Tree e Random Forest, ambos treinados com os dados de 2023 e 2024 e testados com dados reais de 2025. As métricas de desempenho obtidas estão resumidas na tabela abaixo:

Métrica	Decision Tree	Random Forest
Acurácia	0.7931	0.6983
Precisão (Classe 0)	0.90	0.78
Recall (Classe 0)	0.70	0.62
F1-score (Classe 0)	0.79	0.69
Precisão (Classe 1)	0.72	0.64
Recall (Classe 1)	0.91	0.79
F1-score (Classe 1)	0.80	0.71
Média macro F1-score	0.79	0.70

Tabela 1 – Comparação de Desempenho

Como é possível ver na tabela o modelo Decision Tree apresentou desempenho superior em praticamente todas as métricas. A acurácia foi de 79,3%, contra 69,8% do Random Forest. Esse resultado indica que, mesmo com uma estrutura simples, o Decision Tree foi mais eficaz em identificar padrões relevantes nos dados históricos do ativo analisado.

Sendo os resultados do Decision Tree mais assertivos ele foi usado como principal. Abaixo seguem os gráficos com as linhas temporais do período de treinamento e testes (com as marcações em verde de quanto as predições foram assertivas e em vermelho quando as predições foram errôneas), além do gráfico de classes em percentual e a tabela 2 com os retornos percentuais.



Figura 1 - Linha de Serie Temporal com Marcação de Acertos e Erros

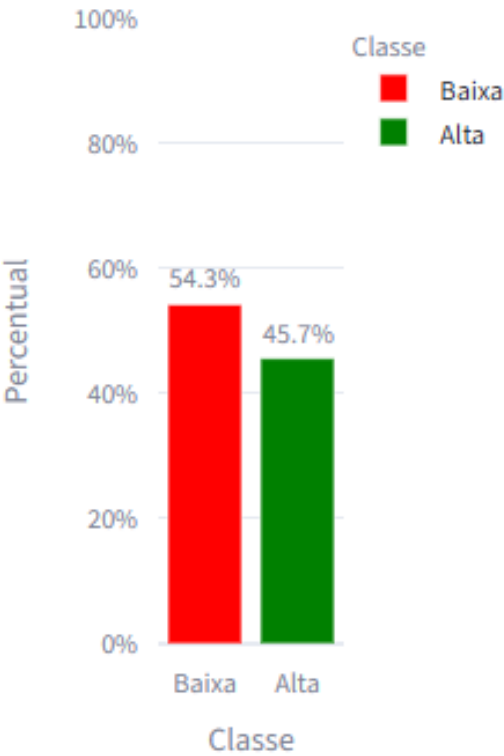


Figura 2 - Distribuição de Classes

<b>Métrica</b>	<b>Decision Tree</b>
<b>Perdas</b>	<b>-0.57%</b>
<b>Ganhos</b>	<b>0.15%</b>
<b>Retorno Geral</b>	<b>-0.43%</b>

Tabela 2 – Perdas, ganhos e retorno geral

Diante dos resultados apresentados é possível notar que para a análise do ativo BBAS3 o algoritmo de Decision Tree entregou uma acurácia muito satisfatória e mais assertiva que até mesmo outros modelos mais complexos. Porém apesar da precisão quando se verifica o retorno geral da tabela 2, o ganho ou perda não é tão expressivo, o que gera a necessidade de outras análises agregadas para que os ganhos sejam incrementais e as perdas reduzidas.

## Conclusões

O desenvolvimento deste projeto proporcionou uma aplicação prática dos conceitos de Machine Learning supervisionado no contexto do mercado financeiro, utilizando dados históricos do ativo BBAS3 da B3. A partir da comparação entre dois modelos clássicos — Decision Tree e Random Forest — foi possível observar que, mesmo com uma estrutura simples, o modelo de árvore de decisão demonstrou maior capacidade de identificar corretamente os padrões de variação diária do ativo.

Além das análises estatísticas, foi realizada uma simulação de retorno financeiro com base nas previsões realizadas, a qual evidenciou que, embora o modelo fosse estatisticamente eficiente, os retornos financeiros obtidos ainda foram limitados, com leve tendência negativa. Isso ressalta que uma boa performance técnica do modelo nem sempre se traduz em lucratividade direta, principalmente em ambientes complexos como o mercado de ações.

Outro ponto importante foi a identificação de overfitting durante a etapa de engenharia de atributos, causada pelo uso de variáveis construídas com base em informações futuras. Essa experiência reforça a importância de se atentar e respeitar a estrutura temporal dos dados, evitando erros no desenvolvimento de modelos preditivos.

Por fim, o experimento mostrou que é possível obter bons resultados mesmo com abordagens simples e dados acessíveis, desde que se mantenha uma atenção rigorosa à integridade dos dados e à lógica da modelagem. Porém é desejável a melhoria do modelo, agregando outros dados ao aprendizado para uma maior acurácia, bem como estratégias de investimentos que aumentem a lucratividade gerando maior retorno financeiro.

## Referencias

- [1] Scikit-learn. **1.10.1 - Decision Trees**. Disponível em: <https://scikit-learn.org/stable/modules/tree.html>. Acesso em: 27 jun. 2025.
- [2] Scikit-learn. **1.10.1 - Ensemble Methods (Random Forests)**. Disponível em: <https://scikit-learn.org/stable/modules/ensemble.html>. Acesso em: 27 jun. 2025.
- [3] YouTube – Sigmoide. **Aprendizado Supervisionado com Scikit-Learn | Curso de Machine Learning em Python**. Disponível em: <https://www.youtube.com/watch?v=QSR0RCC3Mes>. Acesso em: 27 jun. 2025.
- [4] Roussi, R. **YFinance: Yahoo! Finance market data downloader**. Disponível em: <https://ranaroussi.github.io/yfinance/>. Acesso em: 27 jun. 2025.