Previsão do IBOVESPA: Modelo Preditivo com Machine Learning para Apoiar Decisões em Fundos de Investimento.

1. **Introdução**

Durante **o TechChallenge da pós-graduação em Data Analytics da FIAP**, fui desafiada a atuar como cientista de dados em um cenário simulado para um grande fundo de investimentos. O principal objetivo era desenvolver um modelo de Machine Learning capaz de prever, com pelo menos 75% de acurácia, se o índice IBOVESPA (IBOV) fecharia em alta ou baixa no dia seguinte.

Para isso, trabalhei com dados históricos do IBOVESPA obtidos na plataforma Investing.com — uma das maiores referências globais em informações financeiras, reconhecida por rankings como o SimilarWeb e a Alexa.

*Fonte:* [*https://www.investing.com/blog/alexa-top-500-global-sites-240*](https://www.investing.com/blog/alexa-top-500-global-sites-240)

1. **Conhecendo os dados**

O IBOVESPA é o **principal índice da Bolsa Brasileira (B3)** e reflete o desempenho das ações mais negociadas do país. Ele funciona como um termômetro do mercado, sobe quando a maioria dessas ações se valoriza e cai quando perdem valor. Serve como referência para que investidores avaliem se o mercado está otimista ou pessimista.

Para este projeto, trabalhei com uma base de dados que traz o histórico diário detalhado do IBOVESPA. Ela inclui as informações: **preço de abertura, fechamento, mínimo e máximo do dia, o volume negociado e a variação do índice** em relação ao dia anterior.

Inicialmente, o período analisado é de **2006 a 2025**.

Tabela

O conteúdo gerado por IA pode estar incorreto.  
*Figura 1 — Prévia dos dados da base utilizada na análise.*

*Fonte de dados:* [*https://br.investing.com/indices/bovespa-historical-data*](https://br.investing.com/indices/bovespa-historical-data)

1. **Análise Exploratória**

**Tendência:**  
Ao analisar o comportamento do IBOVESPA entre 2006 e 2025, fica evidente que o índice operava em **níveis bem mais baixos até 2017**. Isso aparece com clareza no gráfico, onde ele se mantém abaixo dos 80 mil pontos por um bom tempo.   
Já entre 2016 e 2019, o cenário muda: o índice passa a **subir de forma mais acelerada**, chegando perto dos 120 mil pontos.

Essas diferenças de contexto podem distorcer os padrões mais recentes e acabar afetando a precisão do modelo preditivo. Por isso, optei por focar nos dados dos **últimos seis anos**, de 2019 a 2025. Apesar da queda atípica em 2020, causada pela pandemia de COVID-19, o IBOVESPA se manteve em um patamar mais elevado nos anos seguintes.

Isso permite que o modelo esteja mais alinhado com as dinâmicas atuais do mercado, o que contribui para um desempenho mais estável e adaptável nas previsões futuras.

  
*Figura 2 — Gráfico de tendência do índice de fechamento IBOVESPA de 2006 a 2025.*

*Gráfico gerado no notebook do projeto:* [*https://github.com/gesianne9/TechChallenge2*](https://github.com/gesianne9/TechChallenge2)

**Correlação:**O Heatmap de Correlação oferece uma visão clara de como as variáveis do IBOVESPA se conectam. Os preços — abertura, fechamento, máxima e mínima — mostram uma forte correlação entre si, o que é esperado, já que são diferentes formas de analisar o mesmo índice. O volume negociado também se relaciona positivamente com os preços, embora de forma mais moderada. Isso indica que ele tem influência nas oscilações, ajudando a entender a força e a intensidade dos movimentos. Já a variação diária se destaca por seguir um caminho mais independente. Sua baixa correlação com as demais variáveis reforça seu papel único como indicador de mudança e oscilação.  
  
Considerando essas relações, fica claro que cada variável traz uma contribuição única para a análise e são fundamentais para a construção do modelo.

Gráfico

O conteúdo gerado por IA pode estar incorreto.  
*Figura 3 — Gráfico Heatmap de Correlação das variáveis da base de dados.*

*Gráfico gerado no notebook do projeto:* [*https://github.com/gesianne9/TechChallenge2*](https://github.com/gesianne9/TechChallenge2)

1. **Engenharia de Atributos**

Para oferecer ao modelo uma visão mais completa e contextualizada do comportamento do IBOVESPA, criei novas variáveis (features) a partir dos dados originais:

* **Amplitude Diária:** Representa a volatilidade do dia, calculada pela diferença entre o preço máximo e mínimo, tanto em valores absolutos quanto percentuais.
* **Valores Defasados (Lagged Features):** Incluem os preços de fechamento, abertura, máximo, mínimo, volume e variação dos dias anteriores (1, 2 e 3 dias).
* **Médias Móveis:** Cálculo de médias dos últimos 5, 10 e 20 dias para preços, volume e variação.
* **Variações Percentuais de Períodos:** Indicam o crescimento ou declínio das variáveis ao longo de 5, 10 e 20 dias.

Essas variáveis serão usadas para aprimorar a capacidade de aprendizado do modelo e aumentar sua precisão nas previsões.

1. **Resultado Esperado (Target)**

O principal objetivo desta análise é prever se o IBOVESPA fechará em alta ou baixa no dia seguinte. Para isso, criei uma **variável-alvo binária**, que assume o valor 1 quando o preço de fechamento do dia seguinte (D+1) é maior que o do dia atual, e 0 quando é menor ou igual.   
  
Essa variável indica a tendência de movimento do índice, servindo como base para o modelo aprender a prever o comportamento do mercado com base nas variáveis disponíveis.

1. **Limpeza de Dados**Após a criação de novas variáveis (features) que envolvem valores passados e futuros, identifiquei algumas linhas incompletas por não possuírem dados correspondentes. Além disso, alguns cálculos realizados na engenharia de atributos geraram valores infinitos — provavelmente devido a divisões por zero. Para garantir a qualidade e a integridade do conjunto de dados, realizei uma etapa de limpeza.

Nessa etapa, identifiquei e removi todas as linhas que continham valores em branco ou infinitos.

Resultados da Limpeza:

* Total de linhas antes da limpeza: 1.494
* Total de linhas após a exclusão: 1.464
* Número de linhas excluídas: 30

Considerando o volume total da base, a exclusão desses registros não causará impacto significativo ao modelo, especialmente porque o período analisado — de 2019 a 2025 — foi mantido.  
  
Interface gráfica do usuário, Gráfico

O conteúdo gerado por IA pode estar incorreto.  
*Figura 4 — Gráfico de tendência do índice de fechamento IBOVESPA de 2019 a 2025.*

*Gráfico gerado no notebook do projeto:* [*https://github.com/gesianne9/TechChallenge2*](https://github.com/gesianne9/TechChallenge2)

1. **Escolha do Modelo**

O modelo escolhido para este projeto foi o **Random Forest**.

Esse modelo é capaz de capturar padrões não lineares com precisão, o que é perfeito para o contexto IBOVESPA, onde as relações entre variáveis raramente seguem uma lógica simples ou direta. Além disso, o modelo lida bem com diferentes tipos de variáveis e fornece uma estimativa da importância de cada uma, o que é valioso para entender os fatores que mais influenciam a previsão. Outro ponto positivo é que ele não exige normalização das variáveis, o que simplifica o pré-processamento e torna o modelo mais resistente a ruídos nos dados.

1. **Construção e Treino do Modelo**

**Separação e divisão dos dados:**  
O primeiro passo para construir o modelo foi **organizar os dados** de forma adequada. A base foi dividida em duas partes principais:

* **Features (X):** Todas as variáveis criadas na Etapa 4 (Engenharia de Atributos), que o modelo usará para aprender padrões.
* **Variável-Alvo (Y):** A variável binária definida na Etapa 5 (Resultado Esperado), que indica se o IBOVESPA fechará em alta ou baixa no dia seguinte.

Essa separação foi feita com base no mesmo intervalo temporal: de 19/08/2019 a 18/07/2025.

Em seguida, para avaliar o desempenho do modelo, dividi os dados em dois períodos:

* **Treino:** De 19/08/2019 a 05/06/2025 (1.434 dias), usado para o modelo aprender.
* **Teste:** De 06/06/2025 a 18/07/2025 (30 dias), usado para validar se o modelo consegue prever corretamente.

Com isso, obtive quatro conjuntos de dados: ***X\_treino, Y\_treino, X\_teste e Y\_teste***, com uma divisão de aproximadamente 98% para treino e 2% para teste, garantindo um volume robusto para aprendizado e um período recente para validação.  
  
 **Balanceamento da Variável-Alvo:**

Após a separação, verifiquei o balanceamento da variável-alvo (Y) nos conjuntos de treino e teste. Isso é essencial para evitar que o modelo favoreça a classe mais frequente.

Os resultados mostraram um bom equilíbrio:

* **Treino:** Classe 1 (alta) = **51,74%** | Classe 0 (baixa) = **48,26%**
* **Teste:** Classe 1 (alta) = **43,33%** | Classe 0 (baixa) = **56,67%**

Esse balanceamento garante que o modelo tenha exemplos suficientes para aprender a prever ambas as situações.

**Definição dos parâmetros do modelo:**

Testei diferentes combinações de hiperparâmetros usando o GridSearchCV e ajustes manuais. A configuração que apresentou melhor desempenho e evitou overfitting foi:

* random\_state = 798,
* n\_estimators = 30
* max\_depth = 30
* min\_samples\_leaf = 3
* min\_samples\_split = 52
* max\_features = sqrt
* class\_weight = balanced.

Esses parâmetros foram definidos como padrão para o treinamento final.

**Treinamento do modelo:**

Com os dados organizados e os parâmetros definidos, realizei o treinamento do modelo. Nessa etapa, o Random Forest analisa os dados de treino (X\_treino e Y\_treino) e aprende os padrões que indicam se o IBOVESPA tende a subir ou cair no dia seguinte.

1. **Teste do Modelo**

**Primeiro teste:**Neste teste inicial, avaliei o modelo utilizando **todas as 56 variáveis** criadas na Etapa 4 (Engenharia de Atributos). A previsão foi realizada tanto com os dados de treino (que o modelo já conhece) quanto com os dados de teste (nunca vistos antes).   
Avaliar o desempenho no conjunto de treino ajuda a identificar possíveis sinais de overfitting.

O modelo atingiu **85% de acurácia** com os dados de treino e **57% com os dados de teste**. Isso indica que o modelo conseguiu aprender com os dados, mas ainda apresenta dificuldades para prever corretamente situações novas.   
  
Gráfico, Gráfico de barras

O conteúdo gerado por IA pode estar incorreto.  
*Figura 5 — Gráfico comparativo de acurácia (Teste1).*

*Gráfico gerado no notebook do projeto:* [*https://github.com/gesianne9/TechChallenge2*](https://github.com/gesianne9/TechChallenge2)

A acurácia oferece uma visão geral do desempenho, mas para entender melhor como o modelo se comporta em cada categoria, analisei a **Matriz de Confusão**. Nela, é possível observar que o modelo classificou todos os dados de teste como categoria 0 (baixa), revelando dificuldade em prever os dias de alta (categoria 1).

Gráfico, Gráfico de dispersão

O conteúdo gerado por IA pode estar incorreto.  
*Figura 6 — Gráfico Matriz da Confusão (Teste1).*

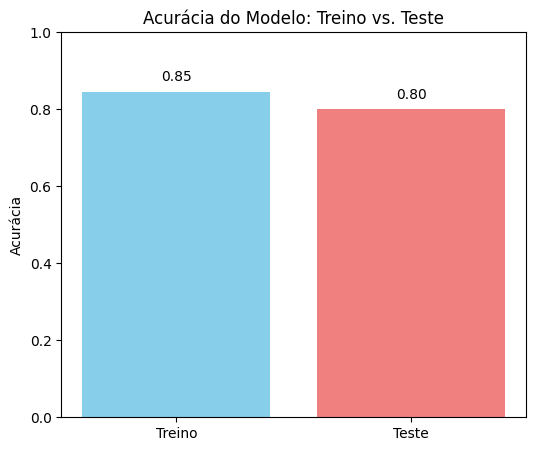
*Gráfico gerado no notebook do projeto:* [*https://github.com/gesianne9/TechChallenge2*](https://github.com/gesianne9/TechChallenge2)

Para melhorar o desempenho, avaliei a importância das variáveis e **selecionei as 20 que mais contribuíram** para a performance do modelo.   
  
Gráfico

O conteúdo gerado por IA pode estar incorreto.  
*Figura 7 — Gráfico de barras de importância das features (Top 20 - Teste1).*

*Gráfico gerado no notebook do projeto:* [*https://github.com/gesianne9/TechChallenge2*](https://github.com/gesianne9/TechChallenge2)

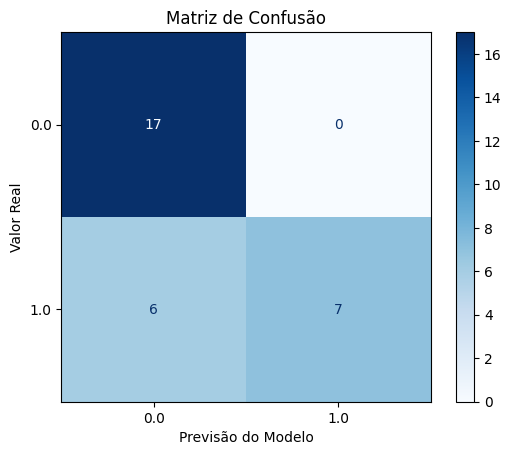
**Segundo teste:**Neste teste, utilizei apenas **as 20 variáveis mais relevantes** identificadas no teste anterior. O modelo apresentou **85% de acurácia no treino** e **80% no teste**, mostrando uma melhora significativa na capacidade de previsão.

  
*Figura 8 — Gráfico comparativo de acurácia (Teste2).*

*Gráfico gerado no notebook do projeto:* [*https://github.com/gesianne9/TechChallenge2*](https://github.com/gesianne9/TechChallenge2)

A nova **Matriz de Confusão** revela que, para a classe 0 (baixa), o modelo foi extremamente eficaz: acertou todos os 17 dias em que o IBOVESPA realmente fechou em baixa. Isso representa uma segurança importante para decisões de venda dos investidores, evitando prejuízos.

Por outro lado, o modelo ainda apresenta dificuldades em prever fechamentos de alta. Dos 13 dias em que o IBOVESPA fechou em alta, ele acertou 7 e errou 6, classificando-os como baixa. Isso significa que, em alguns momentos, os investidores podem perder oportunidades de ganhos.

  
*Figura 9 — Gráfico Matriz da Confusão (Teste2).*

*Gráfico gerado no notebook do projeto:* [*https://github.com/gesianne9/TechChallenge2*](https://github.com/gesianne9/TechChallenge2)

Esse diagnóstico se confirma ao analisar o relatório de classificação da classe 1 (alta):

* Precisão (Precision): 100% — o modelo acertou todas as vezes que previu alta.
* Recall: 54% — o modelo identificou corretamente 54% dos dias que realmente fecharam em alta.
* F1 Score: 70% — indica um bom equilíbrio entre precisão e recall.

Ou seja, embora o modelo ainda não identifique todos os dias de alta, quando ele prevê que o mercado vai subir, essa previsão tem sido confiável.

Gráfico, Gráfico de barras

O conteúdo gerado por IA pode estar incorreto.  
*Figura 10 — Gráfico do Relatório de Classificação (Teste2).*

*Gráfico gerado no notebook do projeto:* [*https://github.com/gesianne9/TechChallenge2*](https://github.com/gesianne9/TechChallenge2)

1. **Conclusão sobre o Desempenho do Modelo:**

Ao avaliar o desempenho do modelo de previsão para o IBOVESPA, o principal destaque é a sua forte capacidade de proteger o investidor contra o principal risco: prever uma alta (Classe 1) quando, na realidade, o mercado fecha em baixa (Classe 0). Esse tipo de erro pode resultar em prejuízos significativos.

O modelo demonstra alta confiabilidade ao minimizar esse risco crítico, sendo especialmente recomendado para investidores com perfil conservador — aqueles que priorizam a preservação de capital e menor exposição ao risco, mesmo que isso signifique abrir mão de algumas oportunidades de lucro.

1. **Resumo Técnico:**

**Ferramentas utilizadas:**

* Editor: VSCode
* Linguagem: Python
* Bibliotecas: Pandas, Matplotlib, NumPy, Seaborn, Plotly, Scikit-learn

***Obs:*** *Todos as imagens de gráficos do artigo foram geradas no Notbook do projeto, disponível no GitHub mencionado abaixo.*

**Estruturação do Projeto:**

**-Pesquisa de Mercado**

Iniciei o projeto com uma pesquisa sobre o mercado de ações, buscando entender os principais termos, dinâmicas e características do setor. Essa etapa foi essencial para garantir que eu tivesse uma base sólida antes de trabalhar com os dados.

**-Avaliação da Base de Dados**

Avaliei a confiabilidade dos dados disponibilizados pelo site Investing.com, verificando a credibilidade da fonte e analisando cada coluna da base para entender seu significado. Isso me permitiu utilizar as informações de forma precisa e consciente.

**-Análise Exploratória**

Realizei ajustes no formato de algumas colunas, como variação e valor, que estavam em formato de string — converti para float para facilitar a visualização e análise. Em seguida, explorei as tendências das variáveis, com foco especial no valor de fechamento do IBOVESPA e suas correlações.

**-Preparação da Base para o Modelo**

Criei novas variáveis (features) a partir dos dados originais, com o objetivo de ampliar as possibilidades de aprendizado do modelo. Usei diferentes períodos e combinações para que ele pudesse “olhar para o passado” e prever o futuro. Também construí uma variável-alvo binária, que indica se o IBOV fechará em alta ou baixa. Após isso, realizei a limpeza dos dados, removendo valores nulos e inconsistentes que poderiam comprometer a performance do modelo.

**-Escolha do Modelo**

Embora o foco do estudo tenha sido o Random Forest, testei outros modelos como ARIMA, SARIMA e Regressão Linear. No entanto, eles não apresentaram bom desempenho na previsão. Por isso, optei por seguir com o Random Forest, que mostrou maior precisão e estabilidade.

**-Criação e Teste do Modelo**

Dividi a base em dados de treino e teste, garantindo que a variável-alvo estivesse balanceada. Testei diversas combinações de parâmetros, pois algumas causavam overfitting. Ajustei os parâmetros até encontrar a configuração que ofereceu o melhor equilíbrio entre aprendizado e capacidade de previsão.

**-Análise de performance do Modelo**

Avaliei o desempenho do modelo por meio de métricas como acurácia, matriz de confusão e relatório de classificação. Essas análises me ajudaram a garantir que o modelo não apenas acertasse, mas também tivesse equilíbrio entre as classes (baixa e alta). Com base nos resultados, fui aprimorando o modelo até alcançar um desempenho confiável.

**-Publicação e Acesso ao Projeto**  
Para viabilizar o acesso o projeto disponibilizei em:

* GitHub: O projeto foi salvo no GitHub, permitindo que os usuários acessem o notebook e visualizem as técnicas aplicadas na análise (<https://github.com/gesianne9/TechChallenge2>).
* Conteúdo na Web: O artigo foi disponibilizado online, garantindo maior acessibilidade e comodidade de leitura para os usuários (<https://medium.com/@gesianne9/previs%C3%A3o-do-ibovespa-modelo-preditivo-com-machine-learning-para-apoiar-decis%C3%B5es-em-fundos-de-ff291d6ebdac>)

**Glossário Técnico:**

\***Correlação:** Ajuda a identificar se duas variáveis têm um relacionamento, se movendo juntas de alguma forma. Ela pode ser **positiva**, quando uma variável sobe e a outra também sobe; **negativa**, quando uma sobe e a outra desce; ou as variáveis podem **não ter nenhuma relação.**   
Para saber o grau de correlação, avalia-se o coeficiente de correlação, um número que vai de -1 a 1. Valores próximos de 1 indicam uma relação positiva e forte, enquanto valores próximos de -1 indicam uma relação negativa e forte.

\* **Variável-Alvo Binária:**  É uma variável que assume apenas dois valores possíveis, geralmente 0 e 1. No contexto de Machine Learning, ela é a variável que nosso modelo busca prever e que responde a uma pergunta de "sim ou não". Neste estudo, a variável-alvo responde à pergunta: "O Ibovespa vai subir amanhã?". O valor 1 representa a resposta "sim" (o Ibovespa sobe), enquanto o valor 0 representa "não" (ele cai ou se mantém estável)

**\*Random Forest:** É um modelo de Machine Learning que constrói várias “árvores de decisão”. Cada árvore funciona como um grupo de perguntas e respostas do tipo “sim ou não”, seguindo uma sequência de perguntas simples, como:

* “O fechamento anterior foi positivo?”
* “Houve variação negativa?”
* “O volume de negociações caiu?”

Cada árvore dá sua “opinião” sobre o resultado, e o modelo final escolhe a resposta mais comum entre elas — como uma votação. Isso reduz o risco de erro e torna a previsão mais confiável

\***Overfitting:** Acontece quando o modelo “decora” os dados em vez de aprender de verdade. Como quem decora respostas de prova sem entender: acerta nos exemplos conhecidos, mas falha nos novos por não conseguir aplicar o conhecimento em situações diferentes.

\***Parâmetros:** São ajustes que controlam o funcionamento do modelo. No Random Forest, por exemplo, defini quantas árvores seriam criadas, a profundidade máxima de cada uma e quantos dados cada árvore poderia usar para fazer previsões.

\***Acurácia:** Representa o percentual de acertos do modelo. Se ele fez 100 previsões e acertou 85, a acurácia é 85%.  
  
**\*Precisão, Recall e F1 Score:** São métricas que ajudam a entender como o modelo se comporta nos acertos e nos erros, especialmente em classificações.

* **Precisão**: É como perguntar entre todas as vezes que o modelo disse “vai subir”, quantas vezes ele acertou? Exemplo: se ele disse que o IBOV subiria 10 vezes, mas só acertou 6, a precisão é 60%.
* **Recall**: É como perguntar entre todas as vezes que o IBOV realmente subiu, quantas vezes o modelo acertou? Exemplo: se o IBOV subiu 13 vezes e o modelo só previu 7 dessas, o recall é 53,8%.
* **F1 Score**: É o equilíbrio entre precisão e recall. Avalia se o modelo está sendo justo — acertando com consistência, sem deixar passar muitos casos que deveria prever.