

UNIVERSIDADE FEDERAL DE SÃO CARLOS  
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA  
DEPARTAMENTO DE ESTATÍSTICA

**INTERVALOS DE PREDIÇÃO PARA O  
MERCADO DE AÇÕES**

**Henrique Ferreira da Silva Nobile**

**Trabalho de Conclusão de Curso**



UNIVERSIDADE FEDERAL DE SÃO CARLOS  
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA  
DEPARTAMENTO DE ESTATÍSTICA

# INTERVALOS DE PREDIÇÃO PARA O MERCADO DE AÇÕES

**Henrique Ferreira da Silva Nobile**

Orientador: Prof. Dr. Ricardo Felipe Ferreira

Trabalho de Conclusão de Curso apresentado  
como parte dos requisitos para obtenção do  
título de Bacharel em Estatística.

**São Carlos**  
**Março de 2025**



FEDERAL UNIVERSITY OF SÃO CARLOS  
EXACT AND TECHNOLOGY SCIENCES CENTER  
DEPARTMENT OF STATISTICS

## PREDICTION INTERVALS FOR THE STOCK MARKET

**Henrique Ferreira da Silva Nobile**

**Advisor: Prof. Dr. Ricardo Felipe Ferreira**

Bachelors dissertation submitted to the Department of Statistics, Federal University of São Carlos - DEs-UFSCar, in partial fulfillment of the requirements for the degree of Bachelor in Statistics.

São Carlos  
February 2025



Henrique Ferreira da Silva Nobile

## INTERVALOS DE PREDIÇÃO PARA O MERCADO DE AÇÕES

Este exemplar corresponde à redação final do trabalho de conclusão de curso devidamente corrigido e defendido por Henrique Ferreira da Silva Nobile e aprovado pela banca examinadora.

Aprovado em 14 de fevereiro de 2025.

Banca Examinadora:

- Prof. Dr. Ricardo Felipe Ferreira
- Profa. Dra. Andressa Cerqueira
- Dr. Victor Azevedo Coscrato



*Dedico este trabalho a Deus, aos meus familiares, amigos, professores e a todos que, de alguma forma, contribuíram para a realização deste projeto.*



# Agradecimentos

Primeiramente, desejo expressar minha sincera gratidão a Deus, cuja presença constante e misericórdia tornaram possível a superação dos desafios ao longo desta trajetória acadêmica. Foi nEle que encontrei a inspiração e a força necessárias para manter a perseverança, mesmo diante das maiores dificuldades.

Agradeço, em especial, à minha família — *Hisléia, Sérgio e Lígia* — cuja paciência, carinho e incentivo nunca me faltaram. O apoio que recebi de vocês foi essencial para me manter motivado, permitindo que eu me dedicasse com tranquilidade a cada etapa do meu estudo. Suas palavras de encorajamento e compreensão, nos momentos mais desafiadores, foram fundamentais para que este trabalho pudesse ser concluído. Sempre amarei vocês.

À minha namorada, *Maria Eduarda*, minha gratidão pela motivação contínua, por acreditar em mim e por estar sempre ao meu lado nos momentos mais desafiadores desta jornada. Obrigado por acreditar em mim, por celebrar cada pequena conquista e por me manter firme nos momentos em que as incertezas se faziam presentes.

Meus sinceros agradecimentos ao *Prof. Dr. Ricardo Felipe Ferreira*, cuja orientação, competência científica e dedicação foram fundamentais para o desenvolvimento deste Trabalho de Conclusão de Curso. Seu enorme conhecimento na área, paciência e, acima de tudo, humanidade, foram a base para que eu pudesse aprimorar minhas habilidades de pesquisa. Sua liderança inspiradora e seu comprometimento em estimular o pensamento crítico contribuíram para que este projeto fosse finalizado.

Enfim, expresso minha gratidão a todos que, de alguma forma, colaboraram para a realização deste projeto, seja através de apoio moral, conselhos técnicos ou palavras de incentivo. A cada um de vocês, meu muito obrigado.



*“O que sabemos é uma gota, o que ignoramos é um oceano.”*

(Sir Isaac Newton)



# Resumo

No mercado de ações, o processo de tomada de decisão de alto risco exige tanto boas previsões quanto a quantificação rigorosa da incerteza. Para lidar com a incerteza inerente às previsões, podemos utilizar intervalos de predição. Para lidar com tal incerteza, nesta monografia, utilizamos tanto intervalos de predição paramétricos quanto aqueles obtidos por meio de métodos de previsão conforme. A previsão conforme permite transformar previsões pontuais provenientes de qualquer modelo em intervalos de previsão com garantias não-assintóticas e sem hipóteses fortes sobre a distribuição dos dados. Na predição conformal clássica, em geral, a incerteza de um classificador é quantificada sob a suposição de permutabilidade (por exemplo, dados i.i.d.). Todavia, essa é uma hipótese não satisfeita para amostras de séries financeiras. Nesse caso, é necessário utilizar métodos de predição conforme que levem em consideração a natureza de dependência inerente às séries temporais. Neste trabalho, os métodos de predição conforme considerados foram o proposto por [Xu e Xie \(2021\)](#) e o proposto por [Gibbs e Candès \(2021\)](#). Utilizamos tanto a largura dos intervalos conformais quanto aqueles obtidos a partir de modelos paramétricos para construir carteiras de ações. Os resultados indicaram que, em contextos onde as suposições do modelo paramétrico são atendidas, o método paramétrico superou o método conforme em termos de desempenho, levando a carteiras com retornos maiores. Exemplos simulados nos quais tais hipóteses não são satisfeitas são apresentados para ilustrar a capacidade de predição intervalar dos métodos conformais quando comparada aos dos intervalos paramétricos.

**Palavras-chave:** *Previsão conforme; Séries temporais; Métodos baseados em árvores de decisão; Precificação de ativos.*



# Abstract

In the stock market, high-risk decision-making processes require both good forecasts and a rigorous quantification of uncertainty. To handle the uncertainty inherent in forecasts, we can use prediction intervals. To address such uncertainty, this thesis uses both parametric prediction intervals and those obtained through conformal prediction methods. Conformal prediction allows us to transform point forecasts from any model into prediction intervals with non-asymptotic guarantees and without strong assumptions about the data distribution. In classical conformal prediction, uncertainty of a classifier is typically quantified under the assumption of exchangeability (e.g., i.i.d. data). However, this assumption is not satisfied for financial time series samples. In this case, it is necessary to use conformal prediction methods that account for the dependency nature inherent in time series data. In this work, the conformal prediction methods considered were those proposed by [Xu e Xie \(2021\)](#) and [Gibbs e Candes \(2021\)](#). We used both the width of the conformal intervals and those obtained from parametric models to construct stock portfolios. The results indicated that, in contexts where the assumptions of the parametric model are met, the parametric method outperformed the conformal method in terms of performance, leading to portfolios with higher returns. Simulated examples where such assumptions are not satisfied are presented to illustrate the interval prediction capability of the conformal methods compared to the parametric intervals.

**Keywords:** *Conformal prediction; Time series; Decision tree-based methods; Asset pricing.*



# Lista de Figuras

- 2.1 **Desempenho de quatro algoritmos de aprendizado de máquina na previsão de dados sem dependência temporal.** São fornecidos quatro gráficos de pontos, um para cada algoritmo, em que no eixos das abscissas temos o identificador da unidade de teste e no eixo das ordenadas os possíveis valores da variável resposta. Os pontos azuis são os valores reais observados no conjunto de teste e os pontos laranjas são os valores preditos pelos algoritmos. . . . . 56
- 2.2 **Desempenho de quatro algoritmos de aprendizado de máquina na previsão de séries temporais.** São fornecidos quatro gráficos, um para cada algoritmo, em que no eixos das abscissas temos o identificador da unidade de tempo e no eixo das ordenadas os possíveis valores da variável resposta. A série temporal azul é a série real observada no conjunto de teste e as séries temporais laranjas são os séries preditas pelos algoritmos. . 70
- 2.3 **Previsão de quatro modelos de séries temporais.** Os modelos em consideração são: autoregressivo (AR), médias móveis (MV), autoregressivo e de médias móveis (ARMA) e autoregressivo integrado e de médias móveis (ARIMA). São fornecidos quatro gráficos, um para cada modelo, em que no eixos das abscissas temos o identificador da unidade de tempo e no eixo das ordenadas os possíveis valores da variável resposta. A série temporal azul é a série real observada no conjunto de teste e as séries temporais laranjas são as séries preditas pelos modelos. . . . . 83

3.1	<b>Comparação de intervalos de predição de quatro diferentes algoritmos de aprendizado de máquina para previsão de dados sem dependência temporal.</b> São apresentados quatro gráficos, um para cada algoritmo, em que no eixo das abscissas temos o identificador da unidade de teste e no eixo das ordenadas os possíveis valores da variável resposta. Os pontos azuis são os valores reais observados no conjunto de teste e os pontos laranjas são os valores preditos pelos algoritmos. As retas verticais são os intervalos de predição para cada unidade de teste. . . . .	102
4.1	<b>Evolução do valor da Carteira 1 ao longo dos últimos 5 anos.</b> O gráfico apresenta a variação do valor da carteira ao longo do tempo, considerando os ativos selecionados e sua respectiva alocação conforme a estratégia definida. . . . .	137
4.2	<b>Evolução do valor do ativo FLRY3.SA ao longo do tempo.</b> O gráfico mostra a variação do preço do ativo FLRY3.SA nos últimos 5 anos, com intervalos de predição conformais baseados na estratégia Boosting (ACI). A área sombreada representa o intervalo de predição, e a linha azul indica o preço do ativo durante o período analisado. . . . .	145
4.3	<b>Avaliação dos retornos - ABEV3.</b> O gráfico exibe a distribuição dos retornos transformados da ação ABEV3, comparando a distribuição empírica com a normal teórica. . . . .	146
4.4	<b>Avaliação dos retornos - BBSA3.</b> O gráfico mostra a distribuição dos retornos transformados da ação BBSA3, com o histograma e a linha de densidade indicando a aproximação com a distribuição normal. . . . .	147
4.5	<b>Avaliação dos retornos - BRF3.</b> Apresentação do histograma dos retornos transformados da ação BRF3, comparado com a normal teórica, e a linha de densidade suavizada. . . . .	147
4.6	<b>Avaliação dos retornos - CPLE6.</b> Exibição da distribuição dos retornos transformados da ação CPLE6, com a linha de densidade sobreposta ao histograma para visualização da normalidade. . . . .	147
4.7	<b>Avaliação dos retornos - CYRE3.</b> O gráfico exibe a distribuição empírica dos retornos transformados da ação CYRE3, com a linha de densidade representando a suavização da distribuição. . . . .	148

4.8	<b>Avaliação dos retornos - ELET6.</b> O histograma dos retornos transformados da ação ELET6, com a linha de densidade suavizada mostrando a aproximação com a normal teórica. . . . .	148
4.9	<b>Avaliação dos retornos - ENEV3.</b> O gráfico apresenta o histograma dos retornos transformados da ação ENEV3, comparado com a distribuição normal teórica, e a linha de densidade suavizada. . . . .	148
4.10	<b>Avaliação dos retornos - FLRY3.</b> Apresentação da distribuição dos retornos transformados da ação FLRY3, com a linha de densidade que suaviza a distribuição observada. . . . .	149
4.11	<b>Avaliação dos retornos - ITUB4.</b> Exibição do histograma e da linha de densidade para os retornos transformados da ação ITUB4, com uma visualização clara da distribuição normal. . . . .	149
4.12	<b>Avaliação dos retornos - JBSS3.</b> O gráfico de histograma dos retornos transformados da ação JBSS3, com a linha de densidade representando a suavização da distribuição dos dados. . . . .	149
4.13	<b>Avaliação dos retornos - PETR4.</b> O gráfico mostra o histograma dos retornos transformados da ação PETR4, com a linha de densidade indicando a aproximação com a distribuição normal teórica. . . . .	150
4.14	<b>Avaliação dos retornos - RADL3.</b> Apresentação do histograma de retornos transformados da ação RADL3, com a linha de densidade suavizada para verificar a aproximação com a normalidade. . . . .	150



# Lista de Tabelas

2.1	<b>Desempenho de quatro algoritmos de aprendizado de máquina na previsão de dados sem dependência temporal.</b> Comparação dos erros quadráticos médios para quatro diferentes algoritmos de aprendizado de máquina: árvore de decisão, florestas aleatórias, <i>bagging</i> e <i>boosting</i> . A comparação é feita tanto no conjunto de treino e validação quanto no conjunto de teste. . . . .	56
2.2	<b>Desempenho de quatro algoritmos de aprendizado de máquina na previsão de séries temporais.</b> Comparação dos erros quadráticos médios para quatro diferentes algoritmos de aprendizado de máquina: árvore de decisão, florestas aleatórias, <i>bagging</i> e <i>boosting</i> . A comparação é feita tanto no conjunto de treino e validação quanto no conjunto de teste. . . . .	69
2.3	<b>Desempenho de quatro modelos de séries temporais.</b> Comparação dos erros quadráticos médios para quatro diferentes modelos de séries temporais: autoregressivo (AR), médias móveis (MA), autoregressivo e de médias móveis (ARMA) e autoregressivo integrado e de médias móveis (ARIMA). A comparação é feita tanto no conjunto de treino e validação quanto no conjunto de teste. . . . .	82
3.1	<b>Desempenho dos preditores a partir de intervalos de predição conforme.</b> Para cada preditor, são construídos intervalos de predição conforme com nível significância $\alpha = 5\%$ . Em seguida, são calculados a proporção de cobertura do valor real de $Y$ no conjunto de teste e do tamanho médio dos intervalos. . . . .	102

3.2	<b>Desempenho dos preditores a partir de intervalos de predição conforme (Chernozhukov).</b> Para cada preditor, são construídos intervalos de predição conforme com nível de significância $\alpha = 5\%$ . Em seguida, são calculados a proporção de cobertura do valor real de $Y$ no conjunto de teste e o tamanho médio dos intervalos. . . . .	114
3.3	<b>Desempenho dos preditores a partir de intervalos de predição com EnbPI (Xu e Xie).</b> Para cada preditor, são construídos intervalos de predição com nível de significância $\alpha = 5\%$ . Em seguida, são calculados a proporção de cobertura do valor real de $Y$ no conjunto de teste e o tamanho médio dos intervalos. . . . .	114
3.4	<b>Desempenho dos preditores utilizando o algoritmo proposto por Gibbs e Cándes (2021).</b> Para cada preditor, são construídos intervalos de predição com nível de significância $\alpha = 5\%$ . Em seguida, são calculadas a proporção de cobertura do valor real de $Y$ no conjunto de teste e o tamanho médio dos intervalos. . . . .	115
3.5	<b>Desempenho dos preditores a partir de intervalos de predição paramétricos.</b> Para cada preditor, são construídos intervalos de predição com nível de significância $\alpha = 5\%$ . Em seguida, são calculados a proporção de cobertura do valor real de $Y$ no conjunto de teste e o tamanho médio dos intervalos. . . . .	121
4.1	<b>Treze ações selecionadas para avaliar a estratégia de alocação de cada carteira via intervalos de predição.</b> As ações foram escolhidas com base em critérios de volatilidade e liquidez. (Parte 1) . . . . .	133
4.2	<b>Treze ações selecionadas para avaliar a estratégia de alocação de cada carteira via intervalos de predição.</b> As ações foram escolhidas com base em critérios de volatilidade e liquidez. (Parte 2) . . . . .	133
4.3	<b>Distribuição percentual dos ativos na Carteira 1.</b> A tabela apresenta a alocação final dos ativos na carteira, representando a proporção de cada ativo no valor total da carteira após o período de cinco anos. (Parte 1) . . .	136
4.4	<b>Distribuição percentual dos ativos na Carteira 1.</b> A tabela apresenta a alocação final dos ativos na carteira, representando a proporção de cada ativo no valor total da carteira após o período de cinco anos. (Parte 2) . . .	136

4.5	<b>Média da largura dos intervalos de predição para alocação da Carteira 2.</b> A alocação inicial foi feita com um valor fixo de R\$ 7.692,00 por ação, seguindo a abordagem da Carteira 1. Pesos dos ativos foram ajustados com base nos intervalos de predição. (Parte 1) . . . . .	138
4.6	<b>Média da largura dos intervalos de predição para alocação da Carteira 2.</b> A alocação inicial foi feita com um valor fixo de R\$ 7.692,00 por ação, seguindo a abordagem da Carteira 1. Pesos dos ativos foram ajustados com base nos intervalos de predição. (Parte 2) . . . . .	139
4.7	<b>Alocação de capital inversamente proporcional à incerteza para a Carteira 2.</b> A tabela apresenta a distribuição do capital (em R\$) para cada ativo, considerando a incerteza associada a cada modelo. (Parte 1) . . . . .	139
4.8	<b>Alocação de capital inversamente proporcional à incerteza para a Carteira 2.</b> A tabela apresenta a distribuição do capital (em R\$) para cada ativo, considerando a incerteza associada a cada modelo. (Parte 2) . . . . .	139
4.9	<b>Métricas de desempenho da Carteira 2.</b> Essa tabela representa a estratégia que adota uma estratégia de alocação dinâmica de capital com base na incerteza associada à previsão dos preços das ações. (Parte 1) . . . . .	140
4.10	<b>Métricas de desempenho da Carteira 2.</b> Essa tabela representa a estratégia que adota uma estratégia de alocação dinâmica de capital com base na incerteza associada à previsão dos preços das ações. (Parte 2) . . . . .	140
4.11	<b>Distribuição percentual dos ativos para o Modelo ARIMA na Carteira 2.</b> A tabela apresenta a alocação final dos ativos na carteira, representando a proporção de cada ativo no valor total da carteira obtida pelo modelo ARIMA. (Parte 1) . . . . .	141
4.12	<b>Distribuição percentual dos ativos para o Modelo ARIMA na Carteira 2.</b> A tabela apresenta a alocação final dos ativos na carteira, representando a proporção de cada ativo no valor total da carteira obtida pelo modelo ARIMA. (Parte 2) . . . . .	141
4.13	<b>Média da largura dos intervalos de predição para alocação da Carteira 3.</b> A alocação inicial foi feita com um valor fixo de R\$ 7.692,00 por ação, seguindo a abordagem da Carteira 1, mas os pesos dos ativos foram ajustados com base nos intervalos de predição. (Parte 1) . . . . .	142

4.14	<b>Média da largura dos intervalos de predição para alocação da Carteira 3.</b> A alocação inicial foi feita com um valor fixo de R\$ 7.692,00 por ação, seguindo a abordagem da Carteira 1, mas os pesos dos ativos foram ajustados com base nos intervalos de predição. (Parte 2) . . . . .	142
4.15	<b>Alocação de capital inversamente proporcional à incerteza para a Carteira 3.</b> A tabela apresenta a distribuição do capital (em R\$) para cada ativo, considerando a incerteza associada a cada modelo. (Parte 1) . .	143
4.16	<b>Alocação de capital inversamente proporcional à incerteza para a Carteira 3.</b> A tabela apresenta a distribuição do capital (em R\$) para cada ativo, considerando a incerteza associada a cada modelo. (Parte 2) . .	143
4.17	<b>Métricas de desempenho da Carteira 3.</b> Essa tabela representa a estratégia que adota o uso de covariáveis e, assim como na Carteira 2, uma estratégia de alocação dinâmica de capital com base na incerteza associada à previsão dos preços das ações. (Parte 1) . . . . .	144
4.18	<b>Métricas de desempenho da Carteira 3.</b> Essa tabela representa a estratégia que adota o uso de covariáveis e, assim como na Carteira 2, uma estratégia de alocação dinâmica de capital com base na incerteza associada à previsão dos preços das ações. (Parte 2) . . . . .	144
4.19	<b>Distribuição percentual dos ativos na Carteira 3.</b> A tabela apresenta a alocação final dos ativos na carteira, representando a proporção de cada ativo no valor total da carteira obtida pelo modelo <b>ARIMA</b> . (Parte 1) . .	145
4.20	<b>Distribuição percentual dos ativos na Carteira 3.</b> A tabela apresenta a alocação final dos ativos na carteira, representando a proporção de cada ativo no valor total da carteira obtida pelo modelo <b>ARIMA</b> . (Parte 2) . .	145
4.21	<b>Resultados do teste Kolmogorov-Smirnov para verificação da normalidade dos retornos (Parte 1).</b> A tabela apresenta os resultados do teste Kolmogorov-Smirnov (KS) aplicado aos retornos dos ativos. . . . .	151
4.22	<b>Resultados do teste Kolmogorov-Smirnov para verificação da normalidade dos retornos (Parte 2).</b> A tabela apresenta os resultados do teste Kolmogorov-Smirnov (KS) aplicado aos retornos dos ativos. . . . .	151

# Sumário

<b>1</b>	<b>Introdução</b>	<b>29</b>
<b>2</b>	<b>Modelos e algoritmos preditivos</b>	<b>33</b>
2.1	Uma visão geral sobre o aprendizado estatístico supervisionado . . . . .	33
2.1.1	Função de Regressão . . . . .	35
2.1.2	Função de Risco . . . . .	36
2.1.3	Seleção de Modelos . . . . .	38
2.2	Métodos baseados em árvores . . . . .	43
2.3	<i>Bagging</i> , florestas aleatórias e <i>boosting</i> . . . . .	48
2.3.1	<i>Bagging</i> . . . . .	48
2.3.2	Florestas aleatórias . . . . .	50
2.3.3	<i>Boosting</i> . . . . .	52
2.3.4	Exemplo numérico . . . . .	54
2.4	Aprendizado estatístico para previsão de séries temporais . . . . .	57
2.4.1	Noções básicas, definições e suposições . . . . .	58
2.4.2	Métodos baseados em árvores de regressão . . . . .	59
2.4.3	Exemplo numérico . . . . .	67
2.5	Séries Temporais . . . . .	70
2.5.1	Modelo Autoregressivo . . . . .	71
2.5.2	Modelo de Médias Móveis . . . . .	73
2.5.3	Modelo Autoregressivo e de Médias Móveis . . . . .	75
2.5.4	Modelo Autoregressivo Integrado e de Médias Móveis . . . . .	77
2.5.5	Exemplo numérico . . . . .	80
<b>3</b>	<b>Intervalos de predição</b>	<b>85</b>
3.1	Uma visão geral sobre a predição conforme . . . . .	85

3.1.1	O problema de construir intervalos de predição . . . . .	87
3.1.2	A suposição de permutabilidade . . . . .	88
3.1.3	Medidas de não-conformidade . . . . .	89
3.1.4	Predição conforme . . . . .	90
3.1.5	Validade e eficiência . . . . .	91
3.2	Predição conforme para dados permutáveis . . . . .	93
3.2.1	Algoritmo proposto por Vovk <i>et al.</i> (2005) . . . . .	93
3.2.2	O papel da permutabilidade na predição conforme . . . . .	94
3.2.3	Complexidade computacional do algoritmo proposto por Vovk <i>et al.</i> (2005) . . . . .	96
3.2.4	<i>Split conformal prediction</i> . . . . .	97
3.2.5	<i>Quantile conformal prediction</i> . . . . .	98
3.2.6	Exemplo Numérico . . . . .	100
3.3	Relaxando a hipótese de permutabilidade . . . . .	104
3.3.1	Algoritmo proposto por Chernozhukov <i>et al.</i> (2018) . . . . .	105
3.3.2	Algoritmo proposto por Xu e Xie (2021) . . . . .	108
3.3.3	Algoritmo proposto por Gibbs e Cándes (2021) . . . . .	110
3.3.4	Exemplo Numérico . . . . .	112
3.4	Intervalos de predição paramétricos . . . . .	116
3.4.1	Exemplo numérico . . . . .	119
3.5	Discussão . . . . .	122
<b>4</b>	<b>Aplicação dos intervalos de predição no mercado de ações</b>	<b>125</b>
4.1	Mercado Financeiro . . . . .	125
4.2	O problema de quantificar a incerteza dos preços das ações . . . . .	128
4.3	Banco de dados . . . . .	130
4.4	Descrição do problema . . . . .	131
4.5	Resultados . . . . .	135
4.5.1	Carteira 1 . . . . .	135
4.5.2	Carteira 2 . . . . .	137
4.5.3	Carteira 3 . . . . .	141
4.5.4	Discussão . . . . .	146
<b>5</b>	<b>Considerações Finais</b>	<b>155</b>





# Capítulo 1

## Introdução

O mercado financeiro de capitais em renda variável é o ambiente no qual os agentes econômicos negociam de forma eficiente a renda excedente, a qual é direcionada para financiar projetos de empresas sem intermediação bancária (Qian e Rasheed, 2007). Os títulos emitidos por essas empresas são as ações, que são uma fração do seu capital. Assim, os detentores das ações de uma empresa são ditos sócios desta empresa. Um dos principais problemas relacionado ao investimento em ações é o de compor uma carteira, ou portfólio, que seja eficiente, i.e., que satisfaça de alguma forma os desejos do investidor. A satisfação dos investidores está atrelada, em geral, à composição de um portfólio que maximize o retorno esperado das ações a um certo nível de risco ou que minimize o risco para qualquer retorno esperado fixado. Nesse sentido, a previsão dos preços das ações permite a alocação eficiente de recursos no mercado, orientando investimentos em direção às empresas mais promissoras (Hussain *et al.*, 2016).

O mercado de ações representa um exemplo de uma área na qual decisões de alto risco exigem tanto boas previsões quanto a quantificação da incerteza (Romano, 2022). No entanto, analisar o preço de ações é uma tarefa difícil devido ao ambiente ruidoso e dinâmico do mercado (Patel *et al.*, 2015; Shah *et al.*, 2019; Bastos, 2024), o que sugere que as previsões no mercado de ações sejam feitas a partir de ferramentas estatísticas e de algoritmos de aprendizado de máquina (Kelly *et al.*, 2019; Freyberger *et al.*, 2020; Lettau e Pelger, 2020; Karolyi e Van Nieuwerburgh, 2020; Giglio e Xiu, 2021; Kumar *et al.*, 2022). Sob a perspectiva de gestão de riscos, focar apenas nas previsões pontuais dos algoritmos de aprendizado de máquina, negligenciando a quantificação da incerteza associada, pode ser uma abordagem inadequada para certas aplicações financeiras (Dixon *et al.*, 2020; Bastos, 2024). Para lidar com essa incerteza inerente às previsões, podemos

utilizar intervalos de predições. Quando conhecemos o preço da ação, esses intervalos podem servir como uma ferramenta especulativa valiosa (Bastos, 2024). De fato, se o preço da ação estiver abaixo do extremo inferior do intervalo de predição, observa-se uma subavaliação, indicando uma possível posição de compra; por outro lado, se o preço da ação estiver acima do extremo superior do intervalo de predição, observa-se uma sobreavaliação, sugerindo uma possível posição de venda.

Dixon *et al.* (2020) mencionam que o uso de processos Gaussianos é uma alternativa muito utilizada em aplicações financeiras, uma vez que geram intervalos de predição (Tegner e Roberts, 2021; Liu *et al.*, 2020; Cousin *et al.*, 2016). Todavia, os processos Gaussianos assumem que os dados são condicionalmente distribuídos de forma normal e, quando essa suposição não é atendida, levam a modelos inválidos. Em contraste, métodos de previsão conforme podem ser utilizados sem suposição alguma sobre a distribuição dos dados (Papadopoulos *et al.*, 2002; Vovk *et al.*, 2005; Gammerman e Vovk, 2007; Shafer e Vovk, 2008; Vovk *et al.*, 2009; Lei e Wasserman, 2014; Sadinle *et al.*, 2019; Foygel Barber *et al.*, 2021; Barber *et al.*, 2021). A partir de tais métodos, é possível obter intervalos de predição não-paramétricos para qualquer modelo de aprendizado de máquina. Tradicionalmente, assume-se que os dados sejam intercambiáveis, o que significa, a grosso modo, que a ordem das unidades amostrais não importam. No entanto, tal premissa não é atendida para dados de séries financeiras. Assim, para quantificar a incerteza das previsões dos preços das ações é necessário considerar métodos de predição conforme que sejam robustos a mudanças na distribuição marginal dos dados.

Adaptar métodos de previsão conforme para além de dados intercambiáveis tem ganhado um interesse significativo na literatura e vem sendo utilizados desde Vovk *et al.* (2005). Gibbs e Candes (2021) desenvolveram uma abordagem adaptativa geral para a previsão conforme que não requer suposições distribucionais, exigindo que os conjuntos preditivos sejam atualizados *online* a cada etapa reponderando o nível de significância. O trabalho subsequente Zaffran *et al.* (2022) propõe técnicas de reponderação mais sofisticadas. No entanto, permanece incerto se tais ajustes são aplicáveis a dados com dependência geral. Nesse cenário mais geral, o primeiro trabalho específico para séries temporais parece ser Chernozhukov *et al.* (2018) que emprega um método baseado em blocos cujo aprendizado depende da qualidade do ajuste do modelo e exige que o processo estocástico subjacente seja fortemente misturador. Xu e Xie (2021) consideram outra abordagem para séries temporais, baseada em conjunto de regressores treinados sobre subamostras

com *bootstrap*, cujas garantias teóricas requerem uma suposição de mistura mais fraca que a de Chernozhukov *et al.* (2018), porém exigem a forte suposição de que a função de regressão populacional é aprendida consistentemente a partir dos dados de treinamento. Jensen *et al.* (2022) consideram uma variante de séries temporais da regressão quantílica conformizada (ver Romano *et al.* (2019)). Como em Xu e Xie (2021), a abordagem de Jensen *et al.* (2022) também envolve treinar um conjunto de métodos sobre amostras com *bootstrap*, porém não são dadas garantias teóricas. Por fim, Oliveira *et al.* (2022) propõe um método de predição conforme dividida que exige que a série temporal seja  $\beta$ -misturadora, uma suposição de mistura mais fraca que a de Chernozhukov *et al.* (2018) e mais forte que a de Xu e Xie (2021). Uma vantagem do método proposto por Oliveira *et al.* (2022) é que ele é agnóstico em relação à qualidade do modelo treinado e possui garantias teóricas.

Aplicações de previsão conforme a dados financeiros não são inéditas, mas poucos são os trabalhos que consideram essa abordagem. Wisniewski *et al.* (2020) utilizam previsão conforme para gerar conjuntos de previsão para a posição líquida de um formador de mercado ao longo do tempo. Kath e Ziel (2021) preveem preços de eletricidade de curto prazo. Chernozhukov *et al.* (2021) preveem retornos de ações com base em sua volatilidade realizada. Gibbs e Candes (2021) preveem a volatilidade do mercado. Bastos (2023) quantificam a incerteza da previsão do preço de opções. Por fim, Bastos (2024) considera a previsão de taxa de juros, preço futuro do óleo e índices de futuros.

Nesta monografia, abordamos o problema de quantificar a incerteza nas previsões de preços de ações usando a predição conforme. Para alcançar esse objetivo, empregaremos os métodos propostos por Gibbs e Candes (2021) e por Xu e Xie (2021). Nesses casos, os modelos preditivos a serem utilizados são aqueles baseados em árvore de decisão (Breiman, 1996; Friedman, 2001) como, por exemplo, florestas aleatórias, *bagging* e *boosting*. Estudos anteriores mostram que esses tipos de modelo preveem bem o preço de ações quando comparado a outros métodos de aprendizado de máquina (Ivaşcu, 2021; Grinsztajn *et al.*, 2022). Esse estudo comparativo será realizado por meio de dados simulados e séries financeiras do mercado de ações brasileiro. Para garantir a reprodutibilidade, os dados reais são obtidos de um fornecedor aberto: <https://finance.yahoo.com/>.

Esta monografia está estruturada da seguinte forma. No Capítulo 2, são apresentados os modelos e algoritmos preditivos que utilizamos ao longo da monografia. No Capítulo 3, dissertamos sobre a predição conforme tanto quando a hipótese de permutabilidade

é satisfeita quanto quando ela não é. Ainda no Capítulo 3, apresentamos o algoritmos propostos por Gibbs e Candes (2021) e Xu e Xie (2021) para lidar com o problema de predição conforme para séries temporais dinâmicas. No Capítulo 4, realizamos a aplicação prática dos modelos e discutimos os resultados obtidos. Por fim, concluimos a monografia com o Capítulo 5, onde apresentamos considerações finais do trabalho.

# Capítulo 2

## Modelos e algoritmos preditivos

Neste capítulo, exploramos diferentes abordagens para a previsão de séries temporais financeiras, focando especificamente em modelos preditivos baseados em árvores de decisão. A análise é centrada em três métodos distintos: árvores de decisão simples, florestas aleatórias, *bagging* (agregação de *bootstrap*) e *boosting* (impulsionamento). Esses métodos são comparados com o tradicional modelo ARIMA (do inglês, *AutoRegressive Integrated Moving Average*), utilizado em previsões financeiras. A comparação visa avaliar a eficácia e precisão dos modelos baseados em árvores, que têm ganhado destaque por sua capacidade de capturar relações complexas e não-lineares nos dados, em contraste com a abordagem linear e estatística do ARIMA. Ao longo do capítulo, apresentamos a fundamentação teórica, as implementações práticas e uma breve análise comparativa, proporcionando uma compreensão das vantagens e limitações de cada método na previsão de séries temporais financeiras.

### 2.1 Uma visão geral sobre o aprendizado estatístico supervisionado

O aprendizado estatístico surgiu da união de várias disciplinas, tais como estatística, ciência da computação, matemática e engenharia. Nos últimos anos, este campo ganhou um impulso significativo com o aumento da capacidade computacional e a disponibilidade de grandes conjuntos de dados. Além disso, o desenvolvimento de algoritmos avançados tem permitido a criação de modelos mais complexos e precisos. Por essa razão, o aprendizado estatístico é explorado não apenas na aplicação de métodos estatísticos tradicionais

para interpretar e prever dados, mas também no desenvolvimento de técnicas como regressão, classificação, agrupamento e redes neurais, que permitem a criação de sistemas capazes de aprender e tomar decisões baseadas em dados.

Define-se, então, o aprendizado estatístico como o processo de construção de algoritmos baseados em dados e de desenvolvimento de modelos matemáticos que possibilitam a inferência de padrões e a predição de futuras observações. Essencialmente, essa área busca criar modelos que possam “aprender” a partir de conjuntos de dados previamente obtidos e aplicar esse aprendizado para prever ou explicar observações futuras ou desconhecidas. Além disso, o aprendizado estatístico envolve a validação e a otimização desses modelos para garantir sua eficácia e robustez em diferentes contextos e aplicações.

Os modelos e algoritmos em aprendizado de máquina podem ser classificados em duas categorias principais: supervisionados e não supervisionados. No aprendizado supervisionado, os algoritmos são treinados com um conjunto de dados rotulados, o que significa que cada exemplo de treinamento é composto por uma entrada e a saída desejada. O objetivo é aprender uma função que mapeia entradas para saídas com precisão, permitindo previsões em novos dados não vistos. Exemplos comuns de algoritmos supervisionados incluem regressão linear, árvores de decisão e máquinas de vetores de suporte (SVM, do inglês *Support Vector Machines*). Dentro do aprendizado supervisionado, os problemas podem ser de regressão ou classificação. Problemas de regressão envolvem prever um valor contínuo, como prever o preço de uma casa com base em suas características; já problemas de classificação envolvem prever uma categoria discreta, como determinar se um e-mail é *spam* ou não *spam*. Por outro lado, no aprendizado não-supervisionado, os algoritmos são utilizados para encontrar padrões ou estrutura nos dados sem rótulos predefinidos. Isso é útil para tarefas como agrupamento e redução de dimensionalidade. Algoritmos como *k-means* e análise de componentes principais (PCA, do inglês *Principal Component Analysis*) são exemplos de métodos não-supervisionados. Esses dois tipos de aprendizado são fundamentais para diferentes aplicações e ajudam a abordar uma ampla variedade de problemas no campo da ciência de dados.

Dada sua capacidade de fornecer previsões precisas em uma vasta gama de aplicações, o aprendizado estatístico vem ganhando espaço no desenvolvimento da sociedade moderna. Na área financeira, por exemplo, a previsão de preços de ações utilizando algoritmos de aprendizado supervisionado tem se mostrado bastante eficaz. Esses algoritmos são treinados com dados históricos do mercado, como preços passados de ações, volumes de

negociação e indicadores econômicos. Com base nesse treinamento, eles podem identificar padrões e relações nos dados e, assim, fazer previsões sobre os preços futuros das ações. Dessa forma, os modelos desenvolvidos através do aprendizado estatístico são essenciais para a tomada de decisões baseadas em dados

Nota-se, portanto, que em um mundo cada vez mais dominado por grandes volumes de dados, a capacidade de extrair informações realmente relevantes e confiar em modelos preditivos precisos é extremamente importante para manter vantagens competitivas e promover a inovação em diversas áreas.

### 2.1.1 Função de Regressão

Em um problema de aprendizado estatístico, estamos interessados em prever o valor de uma variável resposta  $Y$  com base na observação de um conjunto de variáveis preditoras  $X_1, \dots, X_p$ . Os problemas de aprendizado estatístico podem ser classificados como supervisionados e não-supervisionados. No aprendizado supervisionado, o valor da variável resposta é conhecido para todas as unidades amostrais do conjunto de dados, o que serve como um guia para o modelo aprender a relacionar as características de entrada com essa resposta esperada. Isso é importante para que o modelo possa discernir padrões nos dados e fazer previsões precisas. Dois cenários comuns nesse contexto são a regressão, em que a variável resposta  $Y$  é contínua e estamos interessados em prever valores numéricos; e a classificação, em que  $Y$  é categórica e o objetivo é atribuir cada entrada a uma categoria específica. Esses exemplos representam casos distintos em que o modelo busca entender e generalizar padrões nos dados. No aprendizado não-supervisionado, por outro lado, não são fornecidas variáveis resposta juntamente com os dados de entrada. Em vez disso, o objetivo é explorar a estrutura dos dados para identificar padrões, agrupamentos ou regras de associação. Nesta monografia, estamos interessados em prever o preço de uma ação com base em algumas variáveis do mercado financeiro. Portanto, é um problema de aprendizado supervisionado no qual a variável resposta é quantitativa.

Assim, para realizar a previsão da variável  $Y$  com base na observação de um vetor de covariáveis  $\mathbf{X} = (X_1, \dots, X_p)$ , consideramos uma função  $r : \mathcal{X} \rightarrow \mathcal{Y}$  que relaciona às observações  $\mathbf{x} \in \mathcal{X}$  com valores do conjunto  $\mathcal{Y}$ , em que  $\mathcal{X} \subset \mathbb{R}^p$  é o conjunto dos possíveis valores do vetor  $\mathbf{X}$  e  $\mathcal{Y} \subset \mathbb{R}$  é o conjunto dos possíveis valores da variável resposta  $Y$ . Essa função é denominada função de regressão e é definida como a esperança condicional

da variável resposta  $Y$  dado que o vetor  $\mathbf{X}$  de variáveis preditoras é igual a  $\mathbf{x}$ , i.e.,

$$r(\mathbf{x}) := E[Y|\mathbf{X} = \mathbf{x}].$$

Nesse sentido, pode-se dizer que a principal utilidade de estimar a função de regressão  $r(\mathbf{x})$  é a capacidade de prever valores da variável resposta  $Y$  a partir de novas observações das variáveis preditoras  $\mathbf{X}$ . Isso é essencial em contextos em que  $Y$  é difícil de obter diretamente, como ao tentar antecipar a flutuação do preço de um ativo com base em preditores como volume de negociação e múltiplos fundamentalistas, ou ao avaliar o risco de inadimplência de uma empresa com base em métricas financeiras e indicadores macroeconômicos. Do ponto de vista inferencial, estimar  $r(\mathbf{x})$  permite aos interessados entender as relações e influências entre as variáveis. Isso inclui identificar quais preditores são significativos, entender a natureza das relações (se são lineares ou não) e quantificar os efeitos das mudanças nos preditores sobre a resposta. Em muitas aplicações, os modelos são usados tanto para prever quanto para inferir. Uma situação onde isso ocorre é no mercado financeiro, especialmente na precificação de ativos. Por exemplo, ao tentar prever o preço de ações, investidores e analistas buscam entender quais fatores têm maior influência sobre as flutuações de preço. Eles consideram uma variedade de indicadores, como resultados financeiros da empresa, condições econômicas gerais e taxas de juros. A análise desses fatores ajuda a determinar o valor justo de um ativo e no processo de tomada de decisão. Outro ponto que justifica a estimação de  $r(\mathbf{x})$  é que modelos de regressão são flexíveis e aplicáveis em uma ampla variedade de campos, podendo ser ajustados para lidar com grandes volumes de dados e, adaptando-se bem a diferentes tipos de dados.

### 2.1.2 Função de Risco

Em termos muito gerais, em um ambiente de aprendizado supervisionado, qualquer função  $g : \mathcal{X} \rightarrow \mathcal{Y}$  é dita ser uma função de predição. É natural pensar que, dada uma observação  $(\mathbf{x}, y)$ , uma boa função de predição  $g$  é aquela que consegue produzir uma previsão  $g(\mathbf{x})$  suficientemente próxima de  $y$ . De fato, toda previsão está sujeita a algum nível de erro. Esse erro é determinado por uma função de perda  $L(g(\mathbf{X}), Y)$ , que quantifica a discrepância entre a previsão  $g(\mathbf{X})$  e o valor verdadeiro da variável de interesse  $Y$ . Nesse sentido, é razoável definir um critério para avaliar a capacidade preditiva da função  $g$  de acordo com uma função de perda pré-especificada. Em um cenário de regressão, isso é

geralmente feito a partir de uma função de perda quadrática  $L(g(\mathbf{X}), Y) = (Y - g(\mathbf{X}))^2$  e da minimização de uma função  $R_{pred} : \mathcal{G} \rightarrow [0, +\infty)$  tal que

$$R_{pred}(g) := E [L(g(\mathbf{X}), Y)] = E [(Y - g(\mathbf{X}))^2],$$

em que  $\mathcal{G}$  é o conjunto de todas as possíveis funções de predição. A função  $R_{pred}$  é denominada risco quadrático da predição e a melhor função de predição é aquela que minimiza  $R_{pred}$ .

Primeiramente, é importante associar o conceito de função de regressão com função de predição. A conexão entre a função de regressão e a função de predição está intrinsicamente relacionada à minimização do risco associado às previsões. Quanto menor o risco de uma função de predição, melhor ela é considerada em termos de sua capacidade de fazer previsões precisas. Ao criar uma boa função de predição, o objetivo é encontrar um estimador eficaz para a função de regressão. Isso significa que estamos buscando os parâmetros da função de regressão que resultarão em previsões com baixo risco. Em outras palavras, uma função de predição de alta qualidade é aquela que é derivada de uma função de regressão que se ajusta bem aos dados e é capaz de fazer previsões precisas. Portanto, a função  $g \in \mathcal{G}$  que minimiza a função de risco quadrática  $R_{pred}$  é a função de regressão  $r$  (Izbicki e dos Santos, 2020).

Nesse sentido, para estimar a função de risco  $R_{pred}(g) = \mathbb{E}[(Y - g(\mathbf{X}))^2]$  em contextos de modelagem preditiva, é importante adotar métodos que forneçam uma medida confiável do desempenho esperado de uma função de predição  $g : \mathcal{X} \rightarrow \mathcal{Y}$ . Para isso, podemos utilizar abordagens paramétricas ou não-paramétricas. A escolha entre métodos paramétricos e não-paramétricos depende do contexto do problema, da quantidade de dados disponíveis e das características dos dados. Modelos paramétricos são adequados quando a forma de  $g$  é bem compreendida e os dados são abundantes, enquanto modelos não-paramétricos são preferíveis para formas complexas e quando há dados suficientes para suportar um modelo flexível. Esta decisão é fundamental para garantir que a função estimada se ajuste bem aos dados atuais e generalize bem para novos dados, alinhando-se ao objetivo principal do aprendizado estatístico. Nesta monografia, utilizamos tanto abordagens paramétricas quanto não-paramétricas para abordar o problema de predição do preço de uma ação, as quais são exploradas em mais detalhes nas seções seguintes.

### 2.1.3 Seleção de Modelos

A seleção de modelos é um processo importante no aprendizado de máquina e na estatística, consistindo em escolher o modelo que melhor se adapta aos dados disponíveis para realizar previsões ou interpretações precisas. Esse processo inclui a avaliação de diversos modelos candidatos com base em critérios como acurácia, complexidade, interpretabilidade e desempenho em dados de validação. A escolha do modelo ideal impacta significativamente a capacidade de generalização do modelo para novos dados, garantindo que as previsões sejam confiáveis e as análises sejam robustas. Além disso, a seleção adequada do modelo influencia como as decisões baseadas nos resultados do modelo serão compreendidas e justificadas.

Na seleção de modelos, busca-se equilibrar a acurácia da predição com a interpretabilidade do modelo. Conforme destacado por [James et al. \(2013\)](#), modelos mais complexos tendem a oferecer previsões mais precisas, porém, frequentemente sacrificam a interpretabilidade, um aspecto fundamental em áreas como finanças, por exemplo. Assim, ao escolher entre modelos complexos, como redes neurais profundas e modelos mais simples, como a regressão linear, é necessário ponderar o *trade-off* entre precisão e compreensibilidade. [James et al. \(2013\)](#) explicam que modelos mais restritivos, como a regressão linear, são altamente interpretáveis, facilitando a compreensão das relações entre as variáveis. Por outro lado, abordagens mais flexíveis, como *splines* e métodos de *boosting*, podem produzir estimativas complexas de forma que a relação entre as variáveis se torne menos clara. Essa escolha entre flexibilidade e interpretabilidade é essencial para garantir que o modelo selecionado atenda às necessidades específicas da análise, seja para fins de previsão precisa ou para uma interpretação clara das relações entre as variáveis envolvidas.

Após considerarmos que, quando o objetivo é inferência, há claras vantagens em utilizar métodos estatísticos simples e relativamente inflexíveis e quando estamos interessados apenas na predição, modelos mais complexos podem ter um desempenho melhor, poderíamos esperar que o melhor seja usar o modelo mais flexível disponível para fins de predição. No entanto, esse nem sempre é o caso. Muitas vezes, obtemos previsões mais precisas usando um método menos flexível. Esse fenômeno, que pode parecer contraintuitivo à primeira vista, está relacionado ao potencial de sobreajuste em métodos altamente flexíveis. Assim é necessário também encontrar um equilíbrio entre subajuste e superajuste.

Conforme também discutido em [Izbicki e dos Santos \(2020\)](#), em geral, o subajuste

ocorre quando o modelo é tão simplificado que não consegue captar a complexidade dos dados, levando a um desempenho insatisfatório tanto em amostras de treinamento quanto em testes. Isso muitas vezes resulta em erros sistemáticos e previsões imprecisas. Por outro lado, o superajuste acontece quando um modelo é excessivamente complexo, ajustando-se perfeitamente aos dados de treinamento, incluindo o ruído e as anomalias, o que reduz sua eficácia em generalizar para novos conjuntos de dados. Este modelo, embora praticamente perfeito em um cenário de treinamento, falha ao enfrentar dados reais.

Para mitigar o risco de superajuste, várias técnicas podem ser empregadas no processo de modelagem. Uma abordagem comum é a regularização, que adiciona um termo de penalidade à função de custo durante o treinamento do modelo. Isso ajuda a evitar coeficientes de modelo excessivamente grandes, reduzindo assim a complexidade do modelo e diminuindo a tendência ao superajuste. Alguns métodos populares de regularização incluem a regressão *Ridge*, que adiciona uma penalidade  $\ell_2$  aos coeficientes do modelo e a regressão LASSO (do inglês, *Least Absolute Shrinkage and Selection Operator*), que utiliza uma penalidade  $\ell_1$ . Além disso, a validação cruzada é uma técnica fundamental para avaliar o desempenho do modelo e selecionar hiperparâmetros adequados. Ao dividir o conjunto de dados em conjuntos de treinamento e múltiplos conjuntos de teste, a validação cruzada permite estimar como o modelo se sairia em dados não vistos, ajudando a detectar e mitigar o superajuste. Outras estratégias incluem o uso de conjuntos de dados maiores para treinamento, que podem fornecer uma representação mais abrangente da variabilidade dos dados e reduzir a probabilidade de superajuste. Em modelos mais complexos, como redes neurais, ajustar hiperparâmetros como o número de camadas e a taxa de aprendizagem pode ter um impacto significativo na capacidade de previsão do modelo. Em conjunto, essas técnicas fornecem ferramentas eficazes para lidar com o superajuste e desenvolver modelos mais robustos e generalizáveis.

Ainda no que diz respeito à redução da complexidade do modelo, i.e., a seleção de variáveis e, conseqüentemente, a diminuição do número de parâmetros, outro componente importante a ser considerado é o *trade-off* entre viés e variância. Um viés alto indica uma simplificação excessiva do modelo, incapaz de capturar a complexidade dos dados, o que resulta em subajuste. Por outro lado, uma alta variância sugere uma sensibilidade excessiva aos dados de treinamento, levando ao fenômeno do superajuste. Portanto, encontrar o ponto ótimo entre esses dois extremos é fundamental para garantir que o modelo seja capaz de fazer previsões precisas e generalizáveis para novos conjuntos de dados.

Em resumo, diversos modelos são ajustados para a função de regressão  $r(\mathbf{x})$  e busca-se por aquele com o maior poder preditivo, i.e., àquele que possui o menor risco. Logo, trata-se da seleção da função de predição  $g \in \mathcal{G}$  tal que  $R_{pred}(g)$  seja o menor possível. Uma vez que  $R_{pred}(g)$  é desconhecido, precisamos estimá-lo para avaliar o desempenho da função  $g$  na predição de novas unidades amostrais. Para isso, utilizamos o erro quadrático médio empírico nos conjuntos de treinamento, validação e teste. Nas próximas subseções, apresentamos algumas técnicas utilizadas para seleção de modelo de modo a suavizar a diferença entre os riscos observados nesses subconjuntos de dados.

### Penalização: Critério de Informação de Akaike (AIC) e Critério de Informação Bayesiano (BIC)

Uma forma de realizar a seleção de modelos é utilizando uma medida de penalização ou complexidade. Existem diversas funções de penalização, dois critérios utilizados para esse propósito são o Critério de Informação Akaike (AIC, do inglês *Akaike Information Criterion*) e o Critério de Informação Bayesiano (BIC, do inglês *Bayesian Information Criterion*). Ambos os critérios adicionam um termo de penalidade ao ajuste do modelo para desencorajar a complexidade excessiva.

O AIC penaliza o número de parâmetros no modelo de forma mais leve, focando na qualidade do ajuste aos dados observados. Essa medida é definida por:

$$AIC = 2(p - 1) - 2 \log(\mathcal{L})$$

em que  $p - 1$  é o número de parâmetros do modelo e  $\mathcal{L}$  é a função de verossimilhança do modelo ajustado aos dados.

Por outro lado, o BIC impõe uma penalidade mais forte para modelos com muitos parâmetros, que tende a selecionar modelos mais simples quando o número de observações é grande. Essa medida é definida por

$$BIC = \log(n)(p - 1) - 2 \log(\mathcal{L}),$$

em que  $n$  é o número de observações.

A escolha do modelo com o menor valor de AIC ou BIC ajuda a evitar o superajuste, pois busca pela seleção do modelo que explica bem os dados com um número mínimo de parâmetros. Além disso, é possível ajustar tanto o AIC quanto o BIC de modo que

o risco observado seja dado pelo erro quadrático médio empírico ao invés menos a log-verossimilhança, o que os torna aplicável tanto em um paradigma paramétrico quanto não-paramétrico.

### Divisão dos dados (*Data splitting*) e Validação Cruzada (*Cross-Validation*)

Em problemas de regressão, o risco observado de uma função de predição é, em geral, dado pelo erro quadrático empírico do conjunto de treinamento. Esse é um estimador otimista do risco real, podendo resultar em superajuste dos dados, pois não leva em consideração a capacidade do modelo de generalização para novas unidades amostrais. Uma maneira de solucionar este problema é criar uma partição do conjunto de dados originais (*data splitting*). O processo envolve separar o conjunto de dados disponível em duas ou mais partes, geralmente em conjuntos de treinamento, validação e teste. O conjunto de treinamento é utilizado para ajustar os parâmetros do modelo, enquanto o conjunto de validação, quando usado, ajuda na seleção e ajuste dos hiperparâmetros do modelo. Finalmente, o conjunto de teste é empregado para avaliar o desempenho final do modelo, fornecendo uma estimativa de como ele irá se comportar em dados não vistos. Esta abordagem ajuda a identificar problemas como o superajuste e subajuste, garantindo que o modelo tenha a capacidade de generalizar bem para novos dados. A prática comum é dividir os dados de maneira aleatória, com proporções típicas de 70–80% para treinamento, 10–15% para validação e 10–15% para teste, embora essas proporções possam variar dependendo do tamanho do conjunto de dados e do contexto específico da aplicação.

Uma variação do método descrito anteriormente é a validação cruzada (*cross-validation*). O processo envolve dividir o conjunto de dados disponível em vários subconjuntos menores ou *folds*. Em um procedimento comum, conhecido como *k-fold cross-validation*, os dados são divididos em  $k$  partes iguais. O modelo é treinado  $k$  vezes, cada vez utilizando  $k - 1$  *folds* como dados de treinamento e o *fold* restante como dados de validação. O desempenho do modelo é então avaliado em cada iteração e os resultados são agregados para fornecer uma estimativa média do desempenho do modelo. Em outras palavras,

1. Divida os dados em  $k$  *folds*.
2. Para cada *fold*  $i \in \{1, 2, \dots, k\}$ :
  - Treine o modelo nos  $k - 1$  *folds* restantes,

- Teste o modelo no *fold i*,
  - Calcule o erro de predição (risco observado) do *fold i*,
3. Calcule a média dos erros de predição considerando os  $k$  *folds*.

Esse método ajuda a detectar problemas de superajuste e subajuste, pois permite que o modelo seja testado em diferentes subconjuntos dos dados. Além disso, a validação cruzada é útil para selecionar os melhores hiperparâmetros e para comparar a eficácia de diferentes modelos, fornecendo uma medida mais confiável de como o modelo se comportará em dados não vistos.

### Regularização $\ell_1$ nos coeficientes do modelo (LASSO)

O método LASSO (do inglês *Least Absolute Shrinkage and Selection Operator*) é uma técnica de seleção de modelos que realiza tanto a regularização quanto a seleção de variáveis, impondo uma penalidade  $\ell_1$  nos coeficientes dos preditores. O objetivo do LASSO é minimizar a soma dos erros quadráticos e um termo de penalização que é proporcional à soma das magnitudes dos coeficientes. Nesse sentido, a estimativa LASSO  $\hat{\boldsymbol{\beta}}$  é obtida resolvendo o problema de otimização:

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left\{ \frac{1}{n} \sum_{i=1}^n (y_i - g_{\boldsymbol{\beta}}(\mathbf{x}_i))^2 + \lambda \sum_{j=1}^p |\beta_j| \right\}$$

em que  $n$  é o número de observações no conjunto de treinamento,  $p$  é o número de parâmetros,  $y_i$  é o valor observado da variável resposta para a  $i$ -ésima observação,  $\mathbf{x}_i$  é o valor do vetor de variáveis preditoras,  $\boldsymbol{\beta} = (\beta_0, \beta_1, \dots, \beta_p)^\top$  é o vetor de coeficientes,  $g_{\boldsymbol{\beta}}(\mathbf{x}_i)$  é um função de predição parametrizada pelo vetor  $\boldsymbol{\beta}$  e  $\lambda \geq 0$  é o parâmetro de regularização que controla a penalização dos coeficientes.

O primeiro termo na função objetivo é a soma dos erros quadráticos, que mede a qualidade do ajuste do modelo aos dados. O segundo termo é a penalização  $\ell_1$ , que promove a esparsidade - propriedade de um vetor ou matriz ter muitos elementos iguais a zero - no vetor de coeficientes  $\boldsymbol{\beta}$ , fazendo com que alguns coeficientes sejam exatamente zero. Isso resulta na seleção automática de variáveis, pois variáveis com coeficientes zero são excluídas do modelo.

À medida que  $\lambda$  aumenta, a penalização se torna mais forte e mais coeficientes são reduzidos a zero, levando a modelos mais simples com menos variáveis preditoras. Quando

$\lambda = 0$ , o problema se reduz à regressão linear ordinária sem penalização. Quando  $\lambda$  é suficientemente grande, todos os coeficientes, exceto o intercepto, podem ser reduzidos a zero.

## 2.2 Métodos baseados em árvores

Os métodos baseados em árvores são ferramentas de modelagem estatística e de aprendizado de máquina que segmentam o espaço de covariáveis em regiões mais simples, dentro das quais as previsões se tornam mais interpretáveis. As árvores de regressão, por exemplo, são um tipo popular de métodos baseados em árvores, utilizadas para modelar relações não-lineares entre variáveis preditoras e uma variável resposta contínua. Neste contexto, os textos de [James \*et al.\* \(2013\)](#) e [Izbicki e dos Santos \(2020\)](#) oferecem perspectivas complementares sobre a aplicação e teoria por trás das árvores de regressão.

Considere uma variável resposta  $Y$  e um vetor de covariáveis  $\mathbf{X} = (X_1, \dots, X_p)$ . A função de regressão  $r : \mathcal{X} \rightarrow \mathcal{Y}$ , em que  $\mathcal{X} \subset \mathbb{R}^p$  e  $\mathcal{Y} \subset \mathbb{R}$ , é definida como:

$$r(\mathbf{x}) := E[Y | \mathbf{X} = \mathbf{x}].$$

Uma árvore de regressão particiona o espaço de covariáveis  $\mathcal{X}$  em  $M$  regiões distintas  $R_1, R_2, \dots, R_M$ , cada uma das quais é, geralmente, um hiper-retângulo no espaço das covariáveis. A previsão da variável resposta  $Y$  para uma nova observação  $\mathbf{x}$  é dada da seguinte forma: primeiramente, verifica-se a qual das  $M$  regiões, a observação  $\mathbf{x}$  pertence e, em seguida, calcula-se a média aritmética dos valores da variável resposta  $Y$  de todas as unidades amostrais de treinamento pertencentes à região em questão. Logo, a função de regressão para métodos baseados em árvores pode ser escrita da seguinte maneira

$$r(\mathbf{x}) = \sum_{m=1}^M c_m \cdot \mathbf{1}(\mathbf{x} \in R_m)$$

em que

$$c_m := \frac{1}{n_m} \sum_{\{i: \mathbf{x}_i \in R_m\}} y_i$$

é a média das respostas  $y_i$  para as observações no conjunto de treinamento que caem na região  $R_m$ ,  $n_m$  é o número de observações de  $R_m$  e  $\mathbf{1}(\cdot)$  denota a função indicadora.

Os parâmetros  $R_m$  e  $c_m$  são determinados durante o processo de treinamento da árvore,

que envolve a escolha de variáveis e pontos de corte que minimizam uma função de custo, tipicamente o erro quadrático médio (EQM), em cada etapa de bifurcação.

Inicialmente, o objetivo é desenvolver uma árvore que resulte em subdivisões “puras”, ou seja, subdivisões em que os valores de  $Y$  nas observações do conjunto de treinamento em cada folha sejam homogêneos (Izbicki e dos Santos, 2020). Isso é avaliado considerando o quão apropriada é uma determinada árvore  $T$ . Para isso, utilizamos o erro quadrático médio:

$$P(T) = \sum_{m=1}^M \frac{1}{n_m} \sum_{\{i: \mathbf{x}_i \in R_m\}} (y_i - \hat{y}_{R_m})^2$$

em que  $\frac{1}{n_m}$  é o inverso do número de observações na região  $R_m$ , utilizado para calcular a média dos quadrados das diferenças entre os valores reais e previstos de  $Y$  em cada região,  $\sum_{i: \mathbf{x}_i \in R_m}$  representa a soma sobre todas as observações  $i$  cujas variáveis preditoras  $\mathbf{x}_i$  pertencem à região  $R_m$  e, por fim,  $(y_i - \hat{y}_{R_m})^2$  é o quadrado da diferença entre o valor real  $y_i$  da variável de resposta e o valor previsto  $\hat{y}_{R_m}$  de  $Y$  na região  $R_m$  correspondente à observação  $i$ .

Devido à inviabilidade computacional de encontrar  $T$  que minimize  $P(T)$ , adota-se uma heurística para obter uma árvore com baixo erro quadrático médio. Assim, utiliza-se o seguinte algoritmo:

1. Inicialmente, particione o espaço de covariáveis em duas regiões distintas ( $R_1, R_2$ ).
  - (a) Para escolher essa partição, encontra-se a covariável  $X_j$ ,  $j = 1, 2, \dots, p$  e o ponto de corte  $t_1$  que minimize o erro quadrático:

$$\sum_{i: \mathbf{x}_i \in R_1} (y_i - \hat{y}_{R_1})^2 + \sum_{i: \mathbf{x}_i \in R_2} (y_i - \hat{y}_{R_2})^2,$$

em que  $\hat{y}_{R_k}$  é o valor predito na região  $R_k$ ,  $k = 1, 2$ .

- (b) Defina as regiões:

$$R_1 = \{\mathbf{x} : x_j < t_1\} \quad \text{e} \quad R_2 = \{\mathbf{x} : x_j \geq t_1\},$$

em que  $X_j$  é a variável escolhida e  $t_1$  é o corte definido no passo 1(a).

2. Fixe o nó inicial da árvore, i.e., a condição que permite a primeira subdivisão no espaço de covariáveis.

3. Particione  $R_1$  ou  $R_2$  em regiões menores:

- (a) Use a mesma estratégia de minimização do erro quadrático para escolher a nova divisão.
- (b) Escolha a região a ser particionada e defina as novas sub-regiões. Por exemplo, se  $R_1$  for particionada utilizando a variável  $X_k$ ,  $k \neq j$  e o corte  $t_2$ , as novas regiões serão:

$$R_{1,1} = \{\mathbf{x} : x_k < t_1, x_k < t_2\},$$

$$R_{1,2} = \{\mathbf{x} : x_k < t_1, x_k \geq t_2\},$$

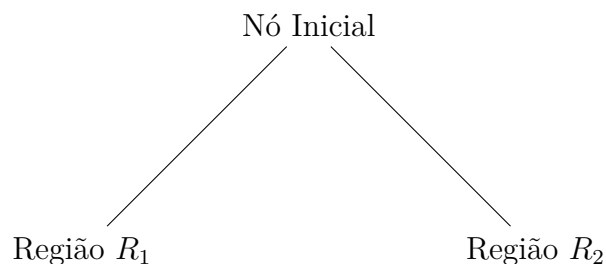
$$R_2 = \{\mathbf{x} : x_j \geq t_1\}.$$

4. Continue o procedimento recursivamente até que cada região tenha poucas observações (critério de parada definido arbitrariamente).

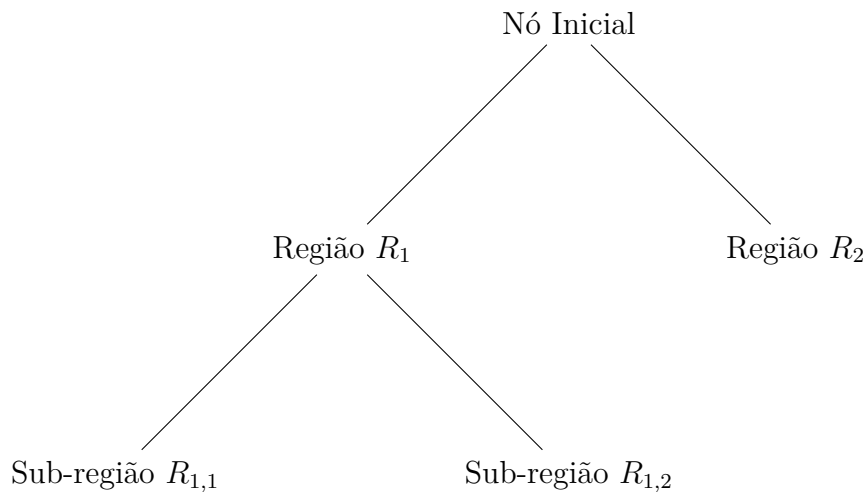
Dito isso, nota-se que a essência das árvores de regressão reside na segmentação do espaço de covariáveis em regiões menores e mais homogêneas, que são mais fáceis de analisar e prever. O processo inicia-se com o espaço de covariáveis completo e, sucessivamente, divide-se em duas regiões baseando-se em um critério que busca maximizar a homogeneidade interna de cada região em relação à variável resposta. Essas divisões são representadas graficamente como uma árvore, em que cada bifurcação representa um ponto de decisão baseado em uma variável preditora e cada folha da árvore representa uma região homogênea (Izbicki e dos Santos, 2020).

Visualmente, portanto, o algoritmo particiona o espaço de covariáveis da seguinte forma:

### Etapa 1: Particionamento inicial



### Etapa 2: Particionamento de $R_1$



Na sequência, temos o particionamento de  $R_2$  (etapa 3), seguido do particionamento de  $R_{1,1}$  (etapa 4), e, por fim, há o particionamento recursivo continuado (etapa 5).

De modo resumido, para determinar os melhores pontos de divisão, ou seja, aqueles que melhor separam os dados em grupos homogêneos, métodos como a redução da soma dos quadrados dos resíduos são comumente utilizados, como descrito acima. Após a construção inicial da árvore, é comum realizar uma poda para remover divisões que não contribuem significativamente para a precisão do modelo, evitando assim o problema de sobreajuste, em que um modelo é tão específico aos dados de treinamento que se torna ineficaz em prever novos dados. Dessa forma, podemos incluir no algoritmo supramencionado o seguinte passo:

5. Realize a poda para evitar o super-ajuste:

- (a) Retire cada nó de baixo para cima, um por vez e observe a variação do erro de predição no conjunto de validação.
- (b) Decida, com base em um critério de parada, por exemplo, quais nós manter para minimizar a variância do estimador e melhorar a performance preditiva em novas observações.

A interpretação de uma árvore de regressão é direta: seguindo-se os caminhos da raiz até as folhas, pode-se entender como as diferentes variáveis preditoras afetam a variável resposta. Tal clareza e facilidade de interpretação tornam as árvores de regressão uma ferramenta atraente em muitas áreas, incluindo finanças, medicina e ciências sociais, em que compreender as relações entre variáveis é tão importante quanto fazer previsões precisas.

Avaliações de desempenho, como a validação cruzada, são essenciais para testar a eficácia de uma árvore de regressão, garantindo que o modelo possa generalizar bem a partir de novos dados. Assim, as árvores de regressão não apenas fornecem um meio robusto para modelagem preditiva, mas também facilitam a interpretação e compreensão dos padrões nos dados.

Dito isso, é razoável pensar em vantagens e desvantagens do uso de árvores como preditor. Uma vantagem fundamental das árvores é a sua facilidade de interpretação. Isso as tornam acessíveis para indivíduos sem formação avançada em estatística, o que facilita a disseminação e utilização desses modelos. Somado a isso, as árvores de decisão mimetizam processos de tomada de decisão humana, segmentando decisões complexas em etapas binárias mais simples. Essa abordagem hierárquica e sequencial é frequentemente vista como um paralelo natural aos processos decisórios intuitivos. Outra característica relevante é a capacidade de representação gráfica das árvores, que permite uma visualização clara e direta do modelo, favorecendo a interpretação mesmo por parte de usuários não especializados, especialmente quando as árvores são pequenas. Além disso, as árvores gerenciam eficazmente variáveis qualitativas, evitando a necessidade de conversão dessas variáveis em formatos *dummy*, uma prática comum em outros modelos estatísticos. A seleção automática de variáveis é outra vantagem significativa dos modelos baseados em árvores. Durante o processo de construção da árvore, o algoritmo automaticamente seleciona as variáveis mais relevantes, reduzindo assim a necessidade de pré-processamento extenso dos dados e simplificando o modelo. Essa característica não só economiza tempo, mas também melhora a robustez do modelo ao focar nas variáveis que realmente contribuem para a previsão, descartando aquelas que são irrelevantes ou redundantes. Isso é particularmente útil em conjuntos de dados com um grande número de variáveis, em que a identificação manual das variáveis mais importantes pode ser impraticável e sujeita a erros.

No entanto, as árvores de decisão enfrentam desafios significativos em termos de precisão preditiva, muitas vezes não alcançando a eficácia de outras técnicas estatísticas. Esta limitação é particularmente problemática em aplicações que exigem alta precisão. A robustez também é uma preocupação, uma vez que pequenas alterações nos dados podem resultar em grandes mudanças na estrutura da árvore, o que pode levar a instabilidades significativas do modelo. Esta sensibilidade a variações nos dados de treinamento pode comprometer a confiabilidade do modelo sob diferentes condições operacionais ou

em conjuntos de dados distintos. Para mitigar essas limitações, métodos de agregação como *bagging*, florestas aleatórias e *boosting* são empregados para melhorar o desempenho preditivo das árvores. Essas técnicas, que combinam múltiplas árvores de decisão em um único modelo, proporcionam uma estabilidade maior e ampliam a capacidade de generalização do modelo. A aplicação dessas estratégias é fundamental e é explorada em mais detalhes nas seções subsequentes sobre métodos baseados em árvores.

## 2.3 *Bagging*, florestas aleatórias e *boosting*

Anteriormente, definimos que, para realizar a previsão da variável  $Y$  com base na observação de um vetor de covariáveis  $\mathbf{X} = (X_1, \dots, X_p)$ , consideramos uma função  $r : \mathcal{X} \rightarrow \mathcal{Y}$  que relaciona as observações  $\mathbf{x} \in \mathcal{X}$  com valores do conjunto  $\mathcal{Y}$ , em que  $\mathcal{X} \subset \mathbb{R}^p$  é o conjunto dos possíveis valores do vetor  $\mathbf{X}$  e  $\mathcal{Y} \subset \mathbb{R}$  é o conjunto dos possíveis valores da variável resposta  $Y$ . Essa função é denominada função de regressão e é definida como a esperança condicional da variável resposta  $Y$  dado que o vetor  $\mathbf{X}$  de variáveis preditoras é igual a  $\mathbf{x}$ , i.e.,

$$r(\mathbf{x}) := \mathbb{E}[Y | \mathbf{X} = \mathbf{x}].$$

Nesse sentido, as técnicas de agregação em aprendizado de máquina, popularmente conhecidas pelo termo do inglês *ensemble techniques*, são métodos que combinam as previsões de vários modelos com o objetivo de gerar uma previsão final mais precisa e robusta do que previsões individuais dos modelos agregados. A principal motivação por trás dessas técnicas é que a agregação de múltiplos modelos reduz o erro e aumenta a confiabilidade das previsões, lidando melhor com a variabilidade dos dados ([Izbicki e dos Santos, 2020](#)).

Diante disso, *Bagging*, Florestas Aleatórias e *Boosting* são técnicas de agregação que buscam melhorar, de algum modo, a precisão e a robustez dos modelos preditivos ao combinar múltiplos modelos menores e menos complexos. Essas técnicas são fundamentais para enfrentar problemas complexos de predição que podem ser difíceis de resolver com um único modelo.

### 2.3.1 *Bagging*

*Bagging*, ou *bootstrap aggregating*, é uma técnica de aprendizado agregado que foi proposta por [Breiman \(1996\)](#). O autor adaptou o conceito de *bootstrap* para construir

múltiplos conjuntos de dados de treinamento a partir de uma reamostragem com reposição dos dados originais. Em seguida, um modelo de aprendizado de máquina, como árvores de decisão, é treinado em cada conjunto de dados de treinamento gerado a partir do processo de reamostragem. Por fim, as previsões desses modelos são combinadas, geralmente por meio de votação, média ou outras técnicas de agregação, para produzir uma previsão final mais robusta e geralmente mais precisa do que qualquer um dos modelos individuais.

No mercado financeiro, o *Bagging* pode ser usado para construir modelos de otimização de carteiras que identificam a combinação ideal de ativos para maximizar o retorno esperado de uma carteira sujeita a restrições de risco. Esses modelos podem incluir técnicas como a teoria moderna do portfólio [Lehmkühl et al. \(2022\)](#) e o modelo de precificação de ativos financeiros [Forster \(2010\)](#).

O *Bagging* visa reduzir a variância dos estimadores de árvores de decisão, através do treinamento de múltiplas árvores em subconjuntos do conjunto de treinamento original, gerados via amostragem *bootstrap*. Seja  $B$  um número inteiro positivo que representa a quantidade de reamostras *bootstrap* a serem geradas. Para cada réplica *bootstrap*  $b$ , em que  $b = 1, \dots, B$ , procedemos da seguinte forma:

1. Selecionar uma amostra  $\mathcal{T}_b$  do conjunto de treinamento original  $\mathcal{T}$ , com reposição.
2. Treinar uma árvore de decisão  $r_b : \mathcal{X} \rightarrow \mathcal{Y}$  sobre a amostra  $\mathcal{T}_b$ .

O estimador *Bagging* é então definido como a média aritmética das previsões das árvores individuais

$$g(\mathbf{x}) = \frac{1}{B} \sum_{b=1}^B r_b(\mathbf{x}),$$

em que  $g : \mathcal{X} \rightarrow \mathcal{Y}$  é a função de predição agregada. Este método assume que os estimadores  $r_b$  são independentes e identicamente distribuídos quando condicionados à amostra  $\mathcal{T}_b$ . Dessa forma, a variância do estimador *Bagging* é:

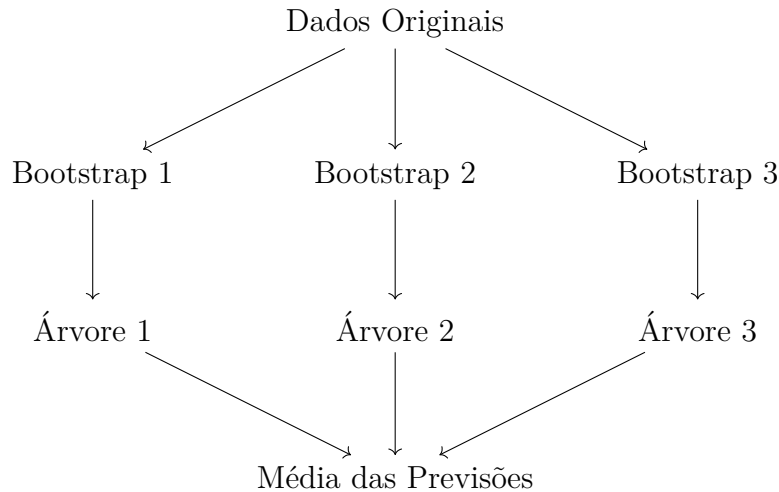
$$\text{Var}(g(\mathbf{X})) = \text{Var}\left(\frac{1}{B} \sum_{b=1}^B r_b(\mathbf{X})\right).$$

Utilizando a propriedade da variância da soma de variáveis aleatórias independentes, temos:

$$\text{Var}(g(\mathbf{X})) = \frac{1}{B^2} \sum_{b=1}^B \text{Var}(r_b(\mathbf{X})) = \frac{1}{B^2} B \text{Var}(r(\mathbf{X})) = \frac{1}{B} \text{Var}(r(\mathbf{X})).$$

Logo, a variância do estimador de *Bagging* diminui proporcionalmente a  $\frac{1}{B}$ .

Nesse sentido, o *Bagging* procura melhorar a estabilidade e a precisão dos modelos de previsão, especialmente em contextos em que a variância dos estimadores é uma preocupação predominante. Visualmente, temos:



Assim, ao combinar várias árvores de decisão treinadas em diferentes amostras *bootstrap* dos dados de treinamento original é esperado uma melhora na estabilidade e na precisão das previsões de preços de ações. Isso é alcançado pela redução da variância dos estimadores individuais, resultando em uma previsão mais robusta e confiável. Este método é particularmente útil em cenários em que há grande variabilidade nos dados financeiros.

### 2.3.2 Florestas aleatórias

Florestas Aleatórias, do inglês *Random Forests*, é uma técnica de aprendizado agregado também proposta por Breiman (1996). Tal método estende o conceito de *Bagging*, incorporando um elemento adicional de aleatoriedade ao processo de construção das árvores de decisão, o que geralmente resulta em um modelo final mais robusto e com maior capacidade de generalização.

Como o *Bagging*, as florestas aleatórias começam com a criação de múltiplos subconjuntos de treinamento através de uma reamostragem com reposição dos dados de treinamento originais. No entanto, ao treinar cada árvore de decisão, o método introduz uma aleatoriedade adicional ao selecionar um subconjunto aleatório das características disponíveis em cada divisão da árvore, tornando cada árvore construída fundamentalmente única.

Para cada árvore  $b$  estimada no subconjunto de treinamento  $\mathcal{T}_b$ , em que  $b = 1, \dots, B$ , as florestas aleatórias procedem da seguinte forma:

1. Selecionar uma amostra  $\mathcal{T}_b$  do conjunto de treinamento original  $\mathcal{T}$ , com reposição.
2. Ao construir cada nó da árvore, selecionamos  $m$  das  $p$  covariáveis disponíveis e utilizamos uma das  $m$  covariáveis para realizar a divisão do nó, cuja escolha é feita tal como no algoritmo de árvore de decisão apresentado na seção 2.2. A escolha do valor de  $m$  pode ser determinada usando validação cruzada. No campo das simulações, geralmente, um valor próximo a  $m \approx \frac{p}{3}$  resulta em um desempenho satisfatório. Vale ressaltar que, quando  $m = p$ , as florestas aleatórias torna-se equivalente ao *Bagging*.
3. Treinar a árvore de decisão  $r_b : \mathcal{X} \rightarrow \mathcal{Y}$  sobre a amostra  $\mathcal{T}_b$  com as características selecionadas.

O estimador de florestas aleatórias é então definido como:

$$g(\mathbf{x}) = \frac{1}{B} \sum_{b=1}^B r_b(\mathbf{x}),$$

em que  $g : \mathcal{X} \rightarrow \mathcal{Y}$  é a função de predição agregada. A agregação das previsões individuais pode ser feita por média, para regressão, ou por votação majoritária, para classificação.

Para calcular a variância do estimador de Florestas Aleatórias, assumimos que  $r_b$  são i.i.d., então:

$$\text{Var}(g(\mathbf{X})) = \text{Var}\left(\frac{1}{B} \sum_{b=1}^B r_b(\mathbf{X})\right).$$

Como os  $r_b$  são i.i.d., podemos usar a propriedade da variância da soma de variáveis aleatórias independentes:

$$\text{Var}\left(\frac{1}{B} \sum_{b=1}^B r_b(\mathbf{X})\right) = \frac{1}{B^2} \sum_{b=1}^B \text{Var}(r_b(\mathbf{X})) = \frac{1}{B} \text{Var}(r(\mathbf{X})).$$

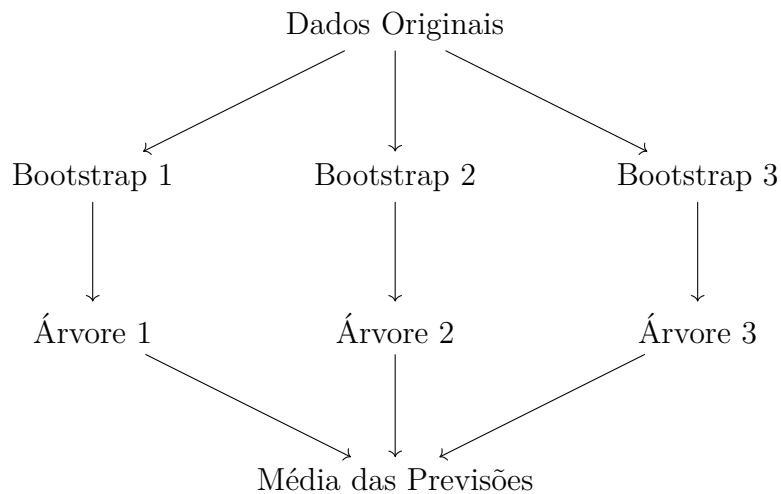
Assim, a variância do estimador de florestas aleatórias diminui inversamente proporcional ao número de árvores  $B$ .

Nas florestas aleatórias, seleciona-se várias amostras *bootstrap* dos dados e, em cada nó de cada árvore, escolhe-se aleatoriamente um subconjunto de covariáveis para encontrar a melhor divisão. Ao treinar múltiplas árvores desta forma e combinar suas previsões por

média, obtém-se um modelo final mais robusto e capaz de generalizar melhor. Realiza-se, assim, os seguintes passos:

1. Selecionar várias amostras *bootstrap* dos dados.
2. Em cada nó de cada árvore, selecionar um subconjunto aleatório de características.
3. Treinar uma árvore de decisão em cada amostra com características selecionadas.
4. Combinar as previsões das árvores por média.

Visualmente:



Ao introduzir uma camada adicional de aleatoriedade na seleção das características em cada divisão da árvore, esse método não só reduz a variância, mas também o *overfitting*, tornando o modelo final mais robusto e generalizável. No contexto da precificação de ações, as florestas aleatórias oferecem previsões que aproveitam tanto a diversidade dos modelos quanto a riqueza dos dados históricos de preços e volumes.

### 2.3.3 *Boosting*

*Boosting*, por sua vez, é uma técnica de *ensemble learning* que visa melhorar a precisão dos modelos de previsão ao combinar vários modelos fracos para formar um modelo forte. Ao contrário do *Bagging* e das florestas aleatórias, o *Boosting* trabalha sequencialmente para corrigir os erros dos modelos anteriores na cadeia, aumentando a precisão geral.

O método de *Boosting* começa com um modelo inicial fraco,  $g_0(\mathbf{x})$ , geralmente um modelo que faz previsões constantes. A cada passo  $b$ , em que  $b = 1, \dots, B$ , o método ajusta um novo modelo  $f_b(\mathbf{x})$  para corrigir os erros do modelo agregado anterior,  $G_{b-1}(\mathbf{x})$ . Este processo é formalmente descrito da seguinte maneira:

1. Calcula-se os resíduos do modelo anterior:

$$e_b(\mathbf{x}_i) = Y_i - G_{b-1}(\mathbf{x}_i),$$

em que  $Y_i$  é a resposta observada e  $G_{b-1}(\mathbf{x}_i)$  é a previsão do modelo anterior para a observação  $\mathbf{x}_i$ .

2. Treina-se o modelo  $f_b(\mathbf{x})$  para prever os resíduos  $e_b(\mathbf{x})$ .
3. Atualiza-se o modelo agregado:

$$G_b(\mathbf{x}) = G_{b-1}(\mathbf{x}) + \alpha_b f_b(\mathbf{x}),$$

em que  $\alpha_b$  é a taxa de aprendizado, um parâmetro que controla o quanto cada modelo  $f_b(\mathbf{x})$  contribui para a atualização.

O modelo final, após  $B$  iterações, é dado por:

$$g(\mathbf{x}) = G_B(\mathbf{x}),$$

em que  $g : \mathcal{X} \rightarrow \mathcal{Y}$  é a função de predição agregada do *ensemble*.

Considerando que cada estimador  $f_b$  é condicionalmente independente e identicamente distribuído (i.i.d.) dado o conjunto de dados  $\mathcal{X}$ , a variância do estimador de *Boosting* é:

$$\text{Var}(g(\mathbf{x})) = \text{Var}\left(\sum_{b=1}^B \alpha_b f_b(\mathbf{x})\right).$$

Utilizando a propriedade da variância da soma de variáveis aleatórias independentes, temos:

$$\text{Var}(g(\mathbf{x})) = \text{Var}\left(\sum_{b=1}^B \alpha_b f_b(\mathbf{x})\right) = \sum_{b=1}^B \alpha_b^2 \text{Var}(f_b(\mathbf{x})).$$

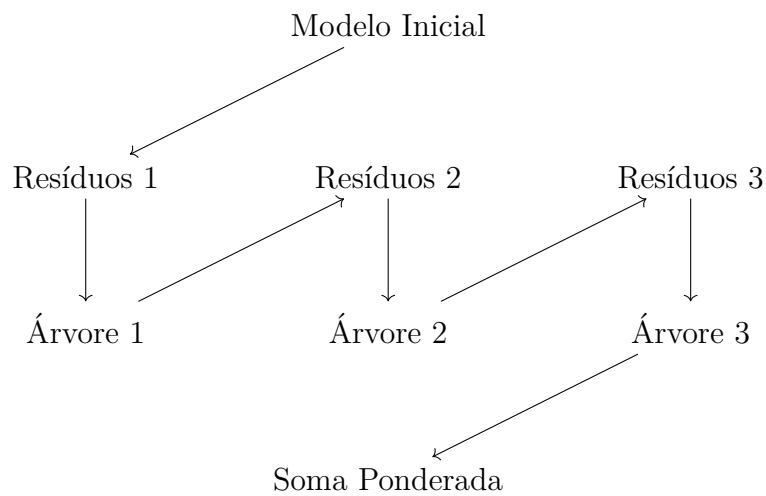
Se assumirmos que  $\alpha_b = \alpha$  para todos os  $b$ , a variância do estimador de *Boosting* pode ser simplificada para:

$$\text{Var}(g(\mathbf{x})) = \alpha^2 \sum_{b=1}^B \text{Var}(f_b(\mathbf{x})) = \alpha^2 B \text{Var}(f(\mathbf{x})).$$

Portanto, a variância do estimador de *Boosting* aumenta proporcionalmente a  $B$  e ao quadrado da taxa de aprendizado  $\alpha$ .

1. Inicializar com um modelo constante.
2. Para cada das hipotéticas 3 iterações:
  - (a) Calcular os resíduos do modelo atual.
  - (b) Treinar uma nova árvore nos resíduos.
  - (c) Atualizar o modelo adicionando a nova árvore com um peso  $\alpha$ .
3. O modelo final  $F_B(\mathbf{x})$  é a soma ponderada de todas as árvores.

Visualmente:



Ao focar nos resíduos das previsões anteriores, o *Boosting* melhora progressivamente a precisão das previsões de preços de ações, pois captura padrões "ocultos" nos dados financeiros, resultando em um modelo preciso para esse contexto.

### 2.3.4 Exemplo numérico

Nesta seção, vamos simular  $n$  carteiras financeiras, em que para cada unidade amostral  $i$ ,  $i = 1, 2, \dots, n$ , consideramos um conjunto de 4 variáveis preditoras ( $X_1, X_2, X_3$  e  $X_4$ ) e 1 a variável resposta ( $Y$ ). Assim, utilizamos as seguintes notações:

- $Y_i$ : rendimento da carteira  $i$ ;
- $X_{i1}$ : número de ações na carteira  $i$ ;
- $X_{i2}$ : volume de transações na carteira  $i$ ;
- $X_{i3}$ : volatilidade da carteira  $i$ ;

- $X_{i4}$ : receita dolarizadas  $i$  (0 para não, 1 para sim).

Para garantir a independência dos pares  $(\mathbf{X}_i, Y_i)$  é necessário tomar setores da economia diferentes em cada carteira. Com isso, definimos o número de amostras  $n = 500$  e, para cada  $i = 1, 2, \dots, 50$ , geramos as variáveis da seguinte forma:

- Número de ações  $X_{i1}$  da carteira  $i$ :  $X_{i1} \sim \text{Uniforme}(1, 50)$ , i.e., o número de ações varia de 1 a 50;
- Volume de transações  $X_{i2}$  da carteira  $i$ :  $X_{i2} \sim \text{Uniforme}(1000, 50000)$ , i.e., o volume de transações varia de 1000 a 50000;
- Volatilidade  $X_{i3}$  da carteira  $i$ :  $X_{i3} \sim \text{Uniforme}(0.1, 1.5)$ , i.e., a volatilidade do mercado varia de 0.1 a 1.5;
- Receita dolarizada  $X_{i4}$  da carteira  $i$ :  $X_{i4} \sim \text{Binomial}(1, 0.5)$ , i.e., a carteira é categorizada como 1 se é dolarizada e 0 caso contrário;
- Rendimento  $Y_i$  da carteira  $i$ : Calculado como uma função linear das variáveis acima mais um erro aleatório, i.e.,

$$Y_i = 0.5X_{i1} + 0.05X_{i2} - 2X_{i3} + 10X_{i4} + \epsilon_i$$

em que  $\epsilon_i \sim N(0, 10)$  é o termo de erro.

## Divisão dos Dados

Para avaliar o desempenho dos métodos de predição estudados ao longo desta seção, precisamos realizar uma divisão no conjunto de dados originais. Se denotarmos por  $\mathcal{O}$  esse conjunto original, a sua divisão é realizada a partir da criação de uma partição composta por três subconjuntos: o conjunto de treinamento  $\mathcal{T}$ , o conjunto de validação  $\mathcal{V}$  e o conjunto de teste  $\mathcal{U}$ . Para o exemplo em questão, tomamos  $|\mathcal{T}| = 350$  (70% das unidades amostrais),  $|\mathcal{V}| = 75$  (15% das unidades amostrais) e  $|\mathcal{U}| = 75$  (15% das unidades amostrais).

## Resultados

Uma vez feita a divisão no conjunto de dados originais, ajustamos quatro preditores no conjunto de treinamento  $\mathcal{T}$ : árvore de decisão, floresta aleatória, *bagging* e *boosting*.

Em seguida, realizamos a predição das unidades amostrais pertencentes tanto ao conjunto de validação  $\mathcal{V}$  quanto ao conjunto de teste  $\mathcal{T}$  e avaliamos o desempenho dos preditores por meio do erro quadrático médio empírico. Os resultados estão sumarizados na Tabela 2.1 e na Figura 2.2.

Tabela 2.1: **Desempenho de quatro algoritmos de aprendizado de máquina na previsão de dados sem dependência temporal.** Comparação dos erros quadráticos médios para quatro diferentes algoritmos de aprendizado de máquina: árvore de decisão, florestas aleatórias, *bagging* e *boosting*. A comparação é feita tanto no conjunto de treino e validação quanto no conjunto de teste.

Modelo	Erro de Treino	Erro de Validação	Erro de Teste
Árvore de Decisão	350.4891	413.8992	502.0395
Florestas Aleatórias	240.5287	258.7479	266.6582
<i>Bagging</i>	300.9234	325.0824	343.7787
<i>Boosting</i>	210.3201	218.5050	274.7805

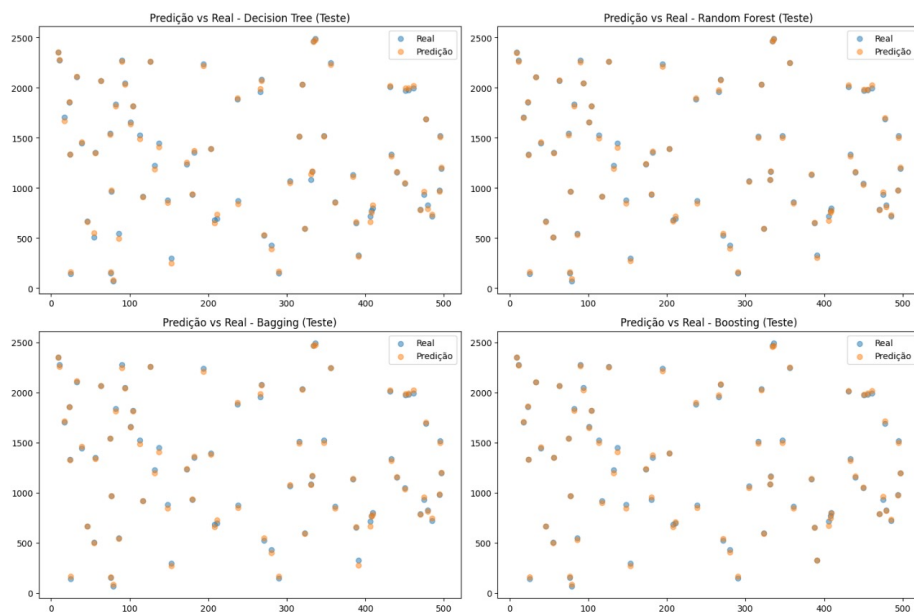


Figura 2.1: **Desempenho de quatro algoritmos de aprendizado de máquina na previsão de dados sem dependência temporal.** São fornecidos quatro gráficos de pontos, um para cada algoritmo, em que no eixos das abscissas temos o identificador da unidade de teste e no eixo das ordenadas os possíveis valores da variável resposta. Os pontos azuis são os valores reais observados no conjunto de teste e os pontos laranjas são os valores preditos pelos algoritmos.

Diante disso, é possível realizar as seguintes análises:

- **Árvore de Decisão:** Este algoritmo apresenta o maior erro de teste (502.039460) e um erro de validação também elevado (413.899178), indicando que o modelo pode estar superajustado aos dados de treino e não generaliza bem para novos dados. Isso demonstra a limitação de uma única árvore de decisão em capturar a complexidade dos dados.
- **Florestas Aleatórias:** Este algoritmo tem erros significativamente menores do que os da árvore de decisão tanto na validação (258.747886) quanto no teste (266.658228), sugerindo que ele generaliza melhor do que a árvore de decisão. A combinação de várias árvores de decisão reduz a variância e melhora a robustez do modelo.
- **Bagging:** O *Bagging* apresenta um erro intermediário entre a árvore de decisão e a floresta aleatória, com um erro de validação de 325.082425 e um erro de teste de 343.778706. Isso indica que o *Bagging* melhora a estabilidade do modelo, mas ainda não é tão eficaz quanto o *Random Forest*. O *Bagging* reduz a variância combinando várias árvores de decisão treinadas em subconjuntos diferentes dos dados originais.
- **Boosting:** O *Boosting* apresenta o menor erro de validação (218.504991), mas um erro de teste (274.780477) ligeiramente maior do que o *Random Forest*. Isso sugere que o *Boosting* é muito eficaz durante a validação, mas pode ser ligeiramente menos robusto no teste, possivelmente devido ao *overfitting*. O *Boosting* trabalha sequencialmente corrigindo os erros dos modelos anteriores, o que pode levar a uma melhor *performance* na validação, mas com o risco de superajuste.

## 2.4 Aprendizado estatístico para previsão de séries temporais

No estudo de aprendizado de máquina aplicado à previsão de séries temporais financeiras, é necessário compreender a transição do tratamento de amostras independentes  $(Y_i, X_i)$  para amostras dependentes  $(Y_t, X_t)$ . Nesse contexto, é importante destacar os avanços recentes nas técnicas de aprendizado de máquina voltadas para séries temporais, em que a dependência temporal das observações  $(Y_t, X_t)$  é intrínseca ao fenômeno estudado. Nesse contexto, os métodos supervisionados, incluindo aqueles baseados em árvores

e técnicas de agregação como *Bagging*, Florestas Aleatórias e *Boosting*, são de particular interesse devido à sua capacidade de lidar com dados não lineares e dependentes. Estas técnicas são aplicadas com sucesso em contextos em que a precisão preditiva é importante, como na otimização de carteiras e na previsão de preços de ações. A transição para o uso desses métodos em séries temporais requer uma adaptação dos algoritmos para considerar a natureza sequencial e dependente dos dados, garantindo previsões mais confiáveis em cenários econômicos e financeiros.

### 2.4.1 Noções básicas, definições e suposições

Sejam  $d$  um número inteiro positivo que representa o número de covariáveis consideradas no problema,  $(\mathbf{Y}, \mathbf{Z}) := \{(Y_t, \mathbf{Z}_t) : t \in \mathbb{N}\}$  um processo estocástico multivariado estacionário fracamente dependente tomando valores em  $\mathbb{R}^{d+1}$  e  $T$  um número inteiro positivo que representa o número de realizações do processo  $(\mathbf{Y}, \mathbf{X})$ . Dado um conjunto de treinamento  $\{(y_t, \mathbf{z}_t) : t = 1, 2, \dots, T\}$  composto por essas  $T$  realizações do processo  $(\mathbf{Y}, \mathbf{X})$ , queremos prever  $Y_{T+h}$  para  $h = 1, 2, \dots, H$ , em que  $H$  é um número inteiro positivo.

Em geral, podemos utilizar informações passadas da série gerado pelo processo  $\mathbf{Y}$  para prever os valores futuros dessa série, bem como podemos utilizar informações passadas das séries geradas pelas covariáveis nesse processo de previsão. Nesse sentido, para cada instante de tempo  $t \in \mathbb{N}$ , podemos considerar um vetor de variáveis preditoras da seguinte forma

$$\mathbf{X}_t := (Y_{t-1}, \dots, Y_{t-p}, \dots, \mathbf{Z}_t, \dots, \mathbf{Z}_{t-r}),$$

em que  $n = p + d(r + 1)$  com  $p$  e  $r$  sendo números inteiros positivos que representam, respectivamente, os números de observações passadas do processo  $\mathbf{Y}$  e do processo  $\mathbf{X}$  que serão utilizadas no processo de predição. Nesse contexto, para cada  $h \in \{1, \dots, H\}$ , a predição de  $Y_{t+h}$  é feita por meio de uma função de predição  $g_h : \mathcal{X} \rightarrow \mathcal{Y}$ , em que  $\mathcal{X} \subset \mathbb{R}^d$  e  $\mathcal{Y} \subset \mathbb{R}$  são, respectivamente, o conjunto dos possíveis valores do vetor de preditores  $\mathbf{X}_t$  e da variável aleatória  $Y_{t+h}$ . Como já discutimos, a melhor função de predição é aquela que minimiza o risco da predição. No caso, em que a função de perda é a quadrática, esse risco é minimizado quando a função de predição é a função de regressão  $r_h : \mathcal{X} \rightarrow \mathcal{Y}$  tal que

$$r_h(\mathbf{x}_t) := E[Y_{t+h} | \mathbf{X}_t = \mathbf{x}_t].$$

Como  $r_h$  é desconhecida, precisamos estimá-la a partir dos dados. Essa estimação pode ser realizada de forma paramétrica ou não-paramétrica, baseada em um único modelo e/ou algoritmo ou em um conjunto de modelos e/ou algoritmos (*ensemble*), podendo depender ou não do horizonte de predição  $h$ . Na prática, em um cenário de regressão, para avaliar o poder preditivo da função de regressão estimada  $\hat{r}_h$ , podemos analisar o risco observado o conjunto de teste por meio, por exemplo, do erro quadrático médio empírico, i.e.,

$$\hat{R}_{pred}(\hat{r}_h) := \frac{1}{H} \sum_{h=1}^H (\hat{r}(\mathbf{x}_{T+h}) - y_{T+h})^2,$$

em que  $\hat{R}_{pred}(\hat{r}_h)$  é o risco de predição estimado da função de regressão  $\hat{r}_h$ .

Ao abordar o problema de previsão em séries temporais, é importante observar que a dependência temporal das observações  $(Y_t, \mathbf{X}_t)$  impõe desafios adicionais em relação aos dados independentes e identicamente distribuídos (i.i.d.). Em particular, a correlação serial deve ser cuidadosamente considerada ao construir modelos preditivos para garantir a precisão das previsões.

Os métodos baseados em árvores, como árvores de regressão, florestas aleatórias, *bagging* e *boosting* podem ser adaptados para a previsão de séries temporais. Esses métodos, como explicado anteriormente, segmentam o espaço das covariáveis em regiões mais simples, permitindo previsões locais interpretáveis através do particionamento recursivo. A seguir exploramos essas técnicas em um pouco mais de detalhes.

## 2.4.2 Métodos baseados em árvores de regressão

A fim de incorporar a estrutura de dependência inerente às séries temporais, vamos substituir a etapa padrão de *bootstrap*, presente em florestas aleatórias, *bagging* e *boosting*, por um método de reamostragem baseado em blocos. Essa substituição é necessária, pois o método de reamostragem *bootstrap* assume que os dados da amostra original sejam i.i.d., o que não ocorre no caso de uma série temporal, então extensões deste método são necessárias para lidar com dados dependentes.

### Reamostragem *bootstrap* em blocos

Os métodos de reamostragem *bootstrap* em blocos são técnicas adaptadas para séries temporais, visando preservar a estrutura de dependência entre observações consecutivas. Diferentemente do *bootstrap* tradicional, que assume independência entre as amostras, o

*bootstrap* em blocos segmenta a série temporal em blocos de dados consecutivos. Cada bloco é selecionado aleatoriamente com reposição para construir novas amostras de dados, mantendo a correlação serial dentro de cada bloco. Esse processo ajuda a capturar e refletir a estrutura de dependência temporal nos dados reamostrados, resultando em estimativas mais robustas e confiáveis para previsões e inferências em contextos em que a ordem temporal das observações é importante. É necessário ressaltar que este método tem como pressuposto os dados serem provenientes de uma série temporal fracamente estacionária e fracamente dependente, garantindo a validade das inferências realizadas a partir das amostras reamostradas.

Os métodos de reamostragem baseados em blocos existentes incluem diversas variantes projetadas para lidar com a dependência temporal em séries temporais. Entre os mais comuns estão o *Non-overlapping Block Bootstrap* (NOBB) (Carlstein, 1986), que divide a série temporal em blocos de comprimento fixo e seleciona esses blocos com reposição para criar novas amostras; o *Moving Block Bootstrap* (MBB) (Künsch, 1989; Liu *et al.*, 1992), onde os blocos sobrepõem-se parcialmente, oferecendo maior suavidade na transição entre blocos; e o *Circular Block Bootstrap* (CBB) (Politis e Romano, 1991), que trata a série temporal como circular para evitar problemas na borda dos blocos. Outras variações incluem o *Stationary Bootstrap* (Politis e Romano, 1994), que utiliza blocos de tamanho aleatório para manter a estacionariedade da série e o *Tapered Block Bootstrap* (Papadimitis e Politis, 2001), que aplica um peso decrescente aos dados dentro de cada bloco para mitigar efeitos de borda. Nesta monografia, o método de reamostragem utilizado é o MBB, pois em estudos passados foi o que apresentou o melhor desempenho em manter a estrutura da série original quando comparado a outros métodos Vogel e Shallcross (1996).

No procedimento de *Moving Block Bootstrap* (MBB), em vez de reamostrar uma única observação de cada vez, reamostramos blocos sobrepostos de observações consecutivas no tempo. Isso garante que a estrutura de dependência dentro de cada bloco seja preservada. Quando os dados são gerados por um processo fracamente dependente, o MBB captura a estrutura subjacente de dependência do processo de forma assintótica. Para garantir que a dependência temporal seja mantida adequadamente, o tamanho de cada bloco cresce proporcionalmente ao tamanho da amostra original, permitindo que o método capture eficazmente a correlação serial presente nos dados.

Vamos descrever o algoritmo do *Moving Block Bootstrap* (MBB) matematicamente para a série temporal observada  $\{(y_t, \mathbf{x}_t) : t = 1, 2, \dots, T\}$ .

1. **Escolha do tamanho do bloco:** Defina o tamanho do bloco  $\ell$ , Esse valor é escolhido com base em considerações teóricas ou práticas e deve ser tal que  $\ell \leq T$ , mas suficientemente grande para capturar a dependência temporal.

2. **Construção dos blocos:**

- Construa os blocos de dados da série temporal. Para uma série de tamanho  $T$ , temos  $T - \ell + 1$  blocos possíveis.
- Cada bloco  $B_i$ , para  $i = 1, 2, \dots, T - \ell + 1$ , é definido como:

$$B_i := \{(y_i, \mathbf{x}_i), (y_{i+1}, \mathbf{x}_{i+1}), \dots, (y_{i+\ell-1}, \mathbf{x}_{i+\ell-1})\}.$$

3. **Reamostragem dos blocos:**

- Realize a reamostragem dos blocos com reposição. Selecione  $K$  blocos aleatoriamente dentre os  $T - \ell + 1$  blocos construídos.
- Cada bloco selecionado no processo de reamostragem pode ser denotado por  $B_{i_k}$ , em que  $i_k \in \{1, 2, \dots, T - \ell + 1\}$  é o índice do  $k$ -ésimo bloco selecionado,  $k = 1, 2, \dots, K$ .

4. **Construção da série *bootstrap*:**

- Forme a série *bootstrap* concatenando os blocos reamostrados. A série reamostrada  $\{(y_t^*, \mathbf{x}_t^*) : t = 1, 2, \dots, T\}$  é composta pelos dados dos blocos selecionados, i.e.,

$$\{(y_t^*, \mathbf{x}_t^*) : t = 1, 2, \dots, T\} = B_{i_1} \cup B_{i_2} \cup \dots \cup B_{i_K}.$$

- Na tentativa de fazer com que a série reamostrada possua  $T$  observações assim como a série original, podemos tomar  $K = T/\ell$ . Todavia, se  $T/\ell$  não for um número inteiro, a série pode ser truncada ou blocos adicionais podem ser usados.

5. **Repetição do procedimento:** Repita  $B$  vezes os passos de 2 a 4 para gerar  $B$  séries *bootstrap*, cada uma das quais preserva a estrutura de dependência temporal dos dados originais.

Em resumo, o MBB funciona escolhendo blocos sobrepostos da série temporal original e reamostrando esses blocos com reposição para construir novas séries que mantêm a

estrutura de dependência temporal original. Desde que o tamanho do bloco cresça à medida em que o tamanho da amostra aumenta, qualquer distribuição de probabilidade conjunta finita-dimensional pode ser recuperada a partir das observações reamostradas [Künsch \(1989\)](#).

## Transformação da série temporal original antes do ajuste dos algoritmos

A transformação da série temporal original antes do ajuste dos algoritmos de predição é uma etapa importante para melhorar a precisão e a eficácia dos modelos preditivos. Esta transformação pode envolver diversas técnicas, como a diferenciação para eliminar tendências e sazonalidades, a normalização ou padronização para colocar as variáveis em escalas comparáveis e a aplicação de transformações como a Box-Cox para estabilizar a variância e aproximar a distribuição dos dados à normalidade. Ao transformar a série temporal, garantimos que as características indesejadas dos dados, como heterocedasticidade e não-linearidade, sejam atenuadas, facilitando o trabalho dos algoritmos de aprendizado de máquina ou estatísticos. Conseqüentemente, os modelos ajustados sobre os dados transformados tendem a produzir previsões mais robustas e precisas, refletindo melhor as estruturas subjacentes da série temporal. Nesta monografia, quanto tivermos séries financeiras muito voláteis, vamos aplicar a transformação Box-Cox e para mitigar o efeito da tendência e da sazonalidade da série vamos realizar uma decomposição da série original. A seguir, detalhamos como procederemos com o pré-processamento dos dados.

A transformação Box-Cox foi proposta por Box e Cox e é usada para estabilizar a variância de séries heterocedásticas. A partir dessa transformação pode-se obter uma série mais simétrica, com variância mais constante e tornar os dados mais parecidos com aqueles gerados por uma distribuição normal. Para aplicar a transformação Box-Cox à série temporal observada  $\{(y_t, \mathbf{x}_t) : t = 1, 2, \dots, T\}$  podemos seguir o algoritmo:

### 1. Escolha do parâmetro $\lambda$ :

- O parâmetro  $\lambda$  pode ser escolhido de modo a maximizar a verossimilhança da série transformada ou minimizando uma medida de erro como o erro quadrático médio.
- Isso geralmente envolve uma busca numérica para encontrar o valor de  $\lambda$  que otimiza o critério escolhido.

– Um exemplo prático é a escolha de  $\lambda$  via maximização da verossimilhança.

A função de verossimilhança para a transformação Box-Cox é dada por:

$$L(\lambda) = -\frac{T}{2} \log \left( \frac{1}{T} \sum_{t=1}^T (y_t^{(\lambda)} - \bar{y}^{(\lambda)})^2 \right) + (\lambda - 1) \sum_{t=1}^T \log y_t,$$

onde  $\bar{y}^{(\lambda)}$  é a média dos valores transformados.

– Para encontrar  $\lambda$  de forma numérica, pode-se testar valores dentro de um intervalo típico, como  $[-2, 2]$ , e escolher aquele que maximiza  $L(\lambda)$ . Por exemplo, considerando as observações

$$y = \{10, 15, 20, 25, 30\},$$

e utilizando um método numérico, como a otimização de Brent, obtemos  $\hat{\lambda} \approx 0.3$ . Assim, a série transformada é calculada como:

$$y_t^{(0.3)} = \frac{y_t^{0.3} - 1}{0.3}, \quad t = 1, \dots, 5.$$

## 2. Aplicação da transformação na série temporal a ser predita:

- Uma vez escolhido o valor ótimo de  $\lambda$  aplica a transformação Box-Cox na série temporal  $\mathbf{Y}$  a ser predita.

- Para  $\lambda \neq 0$

$$y_t^{(\lambda)} = \frac{y_t^{(\lambda)} - 1}{\lambda}, \quad t = 1, \dots, T.$$

- Para  $\lambda = 0$

$$y_t^{(\lambda)} = \log(y_t), \quad t = 1, \dots, T.$$

## 3. Aplicação da transformação nas variáveis preditoras (se necessário):

- Em algumas situações, as variáveis preditoras  $\mathbf{x}_t$  também podem ser transformadas usando a transformação Box-Cox.

- Se a transformação Box-Cox for aplicada às variáveis preditoras  $\mathbf{x}_t$ , então defina as variáveis preditoras transformadas como

$$x_{tj}^{\lambda_j} = \begin{cases} \frac{x_{tj}^{\lambda_j} - 1}{\lambda_j} & \text{se } \lambda_j \neq 0, \\ \log(x_{tj}) & \text{caso contrário,} \end{cases}$$

em que  $x_{tj}$  é o valor observado da variável preditora  $X_j$  no instante de tempo  $t$  e  $\lambda_j$  é o parâmetro de transformação para a  $j$ -ésima variável preditoras com  $j = 1, \dots, n$  e  $t = 1, \dots, T$ .

4. **Construção da série transformada:** A série observada transformada é dada por  $\{(y_t^{(\lambda)}, \mathbf{x}_t^{(\lambda)} : t = 1, 2, \dots, T)\}$ , em que  $\mathbf{x}_t^{(\lambda)}$  denota o vetor de variáveis preditoras transformadas no passo anterior.
5. **Transformação Inversa (se necessário):** Após a análise ou modelagem, pode ser necessário reverter a transformação para interpretar os resultados na escala original. Se esse for o caso, reverta a transformação.

Note que a transformação Box-Cox só é possível quando a série assume valores positivos. Dessa forma, antes de aplicá-la, é necessário avaliar se as variáveis em análise atendem a esse pressuposto.

A decomposição sazonal e de tendência é um método estatístico empregado na análise de séries temporais para desmembrar os vários componentes inerentes aos dados: tendência, sazonalidade e componente aleatório. A separação das flutuações sazonais e as tendências subjacentes dos componentes residuais, facilita a compreensão dos padrões e dinâmicas subjacentes nos dados. Essa técnica utiliza uma abordagem de regressão ponderada localmente (Cleveland, 1979; Cleveland e Devlin, 1988) para ajustar curvas suaves aos dados permitindo a identificação e extração de padrões cíclicos de curto prazo e tendências de longo prazo. Para aplicar essa decomposição à série temporal observada  $\{(y_t, \mathbf{x}_t) : t = 1, 2, \dots, T\}$  podemos seguir o algoritmo:

1. **Suavização da tendência:** Aplicamos uma técnica de suavização para estimar a tendência subjacente da série temporal. Isso pode ser feito usando um ajuste LOESS ou outra abordagem de suavização, como médias móveis ponderadas Carmo (2017).
2. **Remoção da tendência:** Subtraímos a tendência estimada dos dados originais para obter os resíduos iniciais.
3. **Estimativa da Sazonalidade:** Aplicamos LOESS ou outro método de ajuste local para estimar a componente sazonal dos resíduos.
4. **Remoção da Sazonalidade:** Subtraímos a componente sazonal dos resíduos para obter uma nova série de resíduos.

5. **Iteração ou Convergência:** Repetimos os passos 1 a 4 até que os resíduos não mostrem mais padrões significativos ou até que critérios de convergência sejam alcançados.

Este algoritmo recebe os valores observados da série temporal  $\mathbf{Y}$  e, se necessário, também aqueles das variáveis preditoras  $\mathbf{X}$ .

### ***Bagging***

Com isso, temos as ferramentas necessárias para apresentar a versão do algoritmo *bagging* que vamos utilizar para a previsão de séries temporais. Neste algoritmo, a reamostragem é feita aplicando-se a técnica *bootstrap* em blocos MBB no componente aleatório da série original de treino e então as previsões de cada série reamostrada são combinadas de modo a se obter uma única previsão final.

Para aplicar essa versão do método *Bagging* à série temporal observada  $\{(y_t, \mathbf{x}_t) : t = 1, 2, \dots, T + H\}$  podemos seguir o algoritmo:

1. **Divisão da série temporal original:** Divide-se a série temporal original em série de treinamento e série de teste.
  - A série de treinamento será formada pelas primeiras  $T$  realizações da série original e utilizada para obter as previsões.
  - A série de teste é formada pelas últimas  $H$  realizações da série original e será utilizada para medir o poder preditivo do método.
2. **Transformação Box-Cox na série de treinamento:** Aplica-se a transformação Box-Cox na série de treinamento com o intuito de estabilizar sua variância e torná-la mais simétrica. Séries com valores negativos ou iguais a zero não são transformadas.
3. **Decomposição da série de treinamento:** A série de treinamento é decomposta em três componentes: tendência, sazonalidade e componente aleatório.
4. **Remostragem em blocos do componente aleatório:** Aplica-se o método de reamostragem em blocos MBB no componente aleatório da série de treinamento, gerando  $B$  amostras *bootstrap*.
5. **Aplicação inversa da decomposição e da transformação nas amostras *bootstrap*.** Adiciona-se a tendência e a sazonalidade e aplica-se o inverso da trans-

formação Box-Cox em cada uma das  $B$  séries *bootstrap*, resultando em novas séries reamostradas a partir da série original de treinamento.

6. **Ajuste da árvore de regressão nas amostras *bootstrap*:** Uma árvore de regressão é ajustada em cada uma das  $B$  séries *bootstraps*.
7. **Previsão:** Realiza-se então as previsões  $h$  passos à frente para cada uma das  $B$  séries e calcula-se as médias das  $B$  previsões para cada um dos  $h$  passos,  $h = 1, 2, \dots, H$ .

Mesmo que a natureza de séries temporais dos dados seja esquecida após o passo de *bootstrap*, deve-se notar que incluir o tempo como uma variável dependente pode fornecer uma maneira indireta de levar em consideração, em certa medida, a natureza temporal dos dados.

## Florestas aleatórias

O algoritmo de florestas aleatórias para variáveis dependentes que utilizamos nesta monografia é análogo ao *bagging* com a exceção de que no passo 6 uma floresta aleatória é ajustada em cada uma das  $B$  séries *bootstraps*.

## Boosting

A ideia geral de como os métodos *boosting* funcionam foi descrita na seção 2.2. Diversas versões do método foram propostas pela literatura e utilizadas na predição do preço da ação. *A priori*, vamos utilizar nesse trabalho XGBoost (do inglês, *Extreme Gradient Boosting*) (Hartanto *et al.*, 2023). Para aplicar essa versão do método *Boosting* à série temporal observada  $\{(y_t, \mathbf{x}_t) : t = 1, 2, \dots, T + H\}$  podemos seguir o algoritmo geral descrito anteriormente. No entanto, o XGBoost utiliza a técnica de *boosting* por gradientes para construir modelos preditivos robustos e eficientes. Para isso, a função objetivo do XGBoost combina a função de perda com termos de regularização para evitar *overfitting*:

$$\mathcal{L}(F) := \sum_{i=1}^N L(y_t, F(\mathbf{x}_t)) + \sum_{k=1}^K \Omega(h_k),$$

em que  $L(\cdot, \cdot)$  é a função de perda (geralmente uma função convexa diferenciável) e  $\Omega(\cdot)$  é um termo de regularização que penaliza a complexidade do modelo. A regularização em XGBoost geralmente inclui a regularização  $\ell_1$  (Ridge) ou  $\ell_2$  (LASSO). Para otimizar a função objetivo, o XGBoost utiliza uma aproximação de segunda ordem da função de

perda, usando os gradientes (primeira derivada) e hessianos (segunda derivada). A função de perda aproximada na iteração  $m$  é

$$\mathcal{L}^{(m)} \approx \sum_{i=1}^N \left[ g_t h_m(\mathbf{x}_t) + \frac{1}{2} h_t h_m^2(\mathbf{x}_t) \right] + \Omega(h_m),$$

em que  $g_t$  e  $h_t$  são, respectivamente, o gradiente e o Hessiano da função de perda em relação à previsão na iteração anterior. A nova árvore  $h_m(\mathbf{x}_t)$  é ajustada minimizando a função de perda aproximada. Para cada possível divisão, calculamos o ganho de informação, que é a redução na função de perda após a divisão. Para evitar *overfitting*, o XGBoost realiza a poda das árvores após a construção. Uma árvore é podada se o ganho de informação da divisão não exceder um determinado limiar. Para fazer uma previsão com o modelo ajustado, somamos as contribuições de todas as árvores.

### 2.4.3 Exemplo numérico

Nesta seção, simulamos uma série temporal financeira com  $n = 500$  observações, em que cada observação representa o valor diário de variáveis do mercado. Definimos uma variável resposta ( $Y$ ) e quatro variáveis preditoras ( $X_1, X_2, X_3$  e  $X_4$ ) que influenciam o preço das ações. Assim, utilizamos as seguintes notações:

- $Y_t$ : rendimento da ação no tempo  $t$ ;
- $X_{t1}$ : volume de negociações no tempo  $t$  (simulado);
- $X_{t2}$ : Ibovespa no tempo  $t$  (simulado);
- $X_{t3}$ : taxa de juros no tempo  $t$  (simulado);
- $X_{t4}$ : ruído irrelevante no tempo  $t$ .

Considerando a dependência dos pares  $(\mathbf{X}_t, Y_t)$ , é necessário tomar amostras ao longo do tempo de maneira sequencial. Com isso, definimos o número de amostras  $n = 500$  e geramos as variáveis da seguinte forma:

- Volume de negociações  $X_{t1}$  no tempo  $t$ :  $X_{t1} \sim \text{Poisson}(1000)$ , i.e., o volume de negociações é modelado por uma distribuição de Poisson com média 1000;
- Ibovespa  $X_{t2}$  no tempo  $t$ :  $X_{t2} = \sin\left(\frac{20\pi t}{n}\right) + W$ , em que  $W \sim N(0, 0.1)$ , i.e., o índice de mercado é uma função senoidal com ruído normal adicionado;

- Taxa de juros  $X_{t3}$  no tempo  $t$ :  $X_{t3} = \frac{5t}{n} + K$ , em que  $K \sim N(0, 0.5)$ , i.e., a taxa de juros aumenta linearmente com ruído normal adicionado;
- Ruído irrelevante  $X_{4t}$  no tempo  $t$ :  $X_{4t} \sim N(5, 2)$ , i.e., o ruído irrelevante é modelado por uma distribuição normal com média 5 e desvio padrão 2;
- Rendimento da ação  $Y_t$  no tempo  $t$ : Calculado como uma função das variáveis acima mais um termo de ruído normal com média 0 e desvio padrão 0.5, i.e.,

$$Y_t = Y_{t-1} + 0.5 \log(X_{t1} + 1) + 2X_{t2} - 0.3X_{t3} + \epsilon_t$$

em que  $\epsilon_t \sim N(0, 0.5)$  é o termo de erro.

## Divisão dos Dados

Para avaliar o desempenho dos métodos de predição estudados ao longo desta seção, precisamos realizar uma divisão no conjunto de dados originais. Se denotarmos por  $\mathcal{O}$  esse conjunto original, a sua divisão é realizada a partir da criação de uma partição composta por três subconjuntos: o conjunto de treinamento  $\mathcal{T}$ , o conjunto de validação  $\mathcal{V}$  e o conjunto de teste  $\mathcal{U}$ . Para o exemplo em questão, tomamos  $|\mathcal{T}| = 350$  (70% das unidades amostrais),  $|\mathcal{V}| = 75$  (15% das unidades amostrais) e  $|\mathcal{U}| = 75$  (15% das unidades amostrais).

## Resultados

Ao realizar a divisão no conjunto de dados originais, ajusta-se quatro preditores no conjunto de treinamento  $\mathcal{T}$ : árvore de decisão, floresta aleatória, *bagging* e *boosting*. Em seguida, realizamos a predição das unidades amostrais pertencentes tanto ao conjunto de validação  $\mathcal{V}$  quanto ao conjunto de teste  $\mathcal{T}$  e avaliamos o desempenho dos preditores por meio do erro quadrático médio empírico. Os resultados estão sumarizados na Tabela 2.2 e na Figura 2.2, cuja análise nos levam as seguintes conclusões:

- **Árvore de Decisão:** Comparado aos demais, esse algoritmo apresenta um erro de teste (1.120920) elevado e um erro de validação também alto (0.769170), indicando que o modelo pode estar superajustado aos dados de treino e não generaliza bem para novos dados. Isso demonstra a limitação de uma única árvore de decisão em capturar a complexidade dos dados.

- **Florestas Aleatórias:** Este algoritmo tem um erro de validação (0.463779) significativamente menor do que o da árvore de decisão, sugerindo que ele generaliza melhor durante a validação. No entanto, o erro de teste (0.912690) é muito próximo ao da árvore de decisão, indicando que, embora mais robusto na validação, ele ainda enfrenta desafios semelhantes na generalização para novos dados.
- **Bagging:** O *Bagging* apresenta um erro de validação (0.484571) intermediário entre a árvore de decisão e a floresta aleatória e um erro de teste (0.908763) similar ao da árvore de decisão. Isso indica que o *Bagging* melhora a estabilidade do modelo, mas não é significativamente mais eficaz na generalização para novos dados comparado ao *Random Forest*. O método de *Bagging* reduz a variância combinando várias árvores de decisão treinadas em subconjuntos diferentes dos dados originais.
- **Boosting:** O *Boosting* mostra o menor erro de validação (0.357919) e também o menor erro de teste (0.624852). Isso sugere que o *Boosting* é muito eficaz tanto durante a validação quanto no teste, indicando uma excelente capacidade de generalização e um ajuste robusto aos dados de treino. O *Boosting* trabalha sequencialmente corrigindo os erros dos modelos anteriores, o que pode levar a uma melhor performance na validação e no teste.

Tabela 2.2: **Desempenho de quatro algoritmos de aprendizado de máquina na previsão de séries temporais.** Comparação dos erros quadráticos médios para quatro diferentes algoritmos de aprendizado de máquina: árvore de decisão, florestas aleatórias, *bagging* e *boosting*. A comparação é feita tanto no conjunto de treino e validação quanto no conjunto de teste.

Modelo	Erro de Treino	Erro de Validação	Erro de Teste
Árvore de Decisão	0.6401	0.7692	1.1209
Floresta Aleatória	0.4005	0.4638	0.9127
<i>Bagging</i>	0.4251	0.4846	0.9088
<i>Boosting</i>	0.3002	0.3579	0.6249

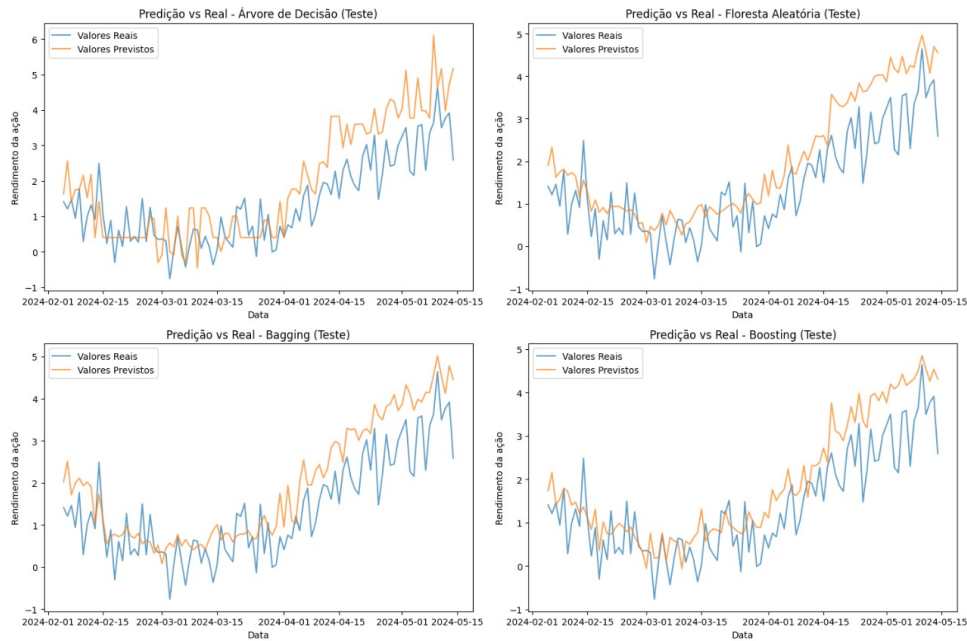


Figura 2.2: **Desempenho de quatro algoritmos de aprendizado de máquina na previsão de séries temporais.** São fornecidos quatro gráficos, um para cada algoritmo, em que no eixos das abscissas temos o identificador da unidade de tempo e no eixo das ordenadas os possíveis valores da variável resposta. A série temporal azul é a série real observada no conjunto de teste e as séries temporais laranjas são os séries preditas pelos algoritmos.

## 2.5 Séries Temporais

Séries temporais são sequências de observações de uma variável registradas em intervalos sucessivos no tempo. No contexto do mercado financeiro, as séries temporais são utilizadas para analisar e prever o comportamento de preços de ativos, taxas de juros, índices de mercado e outros indicadores econômicos. As séries temporais financeiras são essenciais para modelar a dinâmica do mercado e ajudar na tomada de decisões sobre investimentos. Um aspecto das séries temporais financeiras é a dependência temporal, onde os valores futuros estão correlacionados com os valores passados. Essa dependência deve ser considerada para a construção de modelos preditivos que ajudem a antecipar movimentos de mercado. Nesta seção, vamos considerar quatro modelos de séries temporais que buscam capturar essa dependência inerente ao mercado de ações: o modelo autoregressivo, o modelo de médias móveis, o modelo autoregressivo e de médias móveis e o modelo integrado e de médias móveis. A seguir, apresentamos os modelos em questão.

### 2.5.1 Modelo Autoregressivo

Os Modelos Autoregressivos (AR, do inglês *Autoregressive Model*) são amplamente utilizados para modelagem e previsão de séries temporais. A ideia principal de um modelo AR é que o valor atual de uma série temporal pode ser explicado como uma função linear de seus valores passados. Um modelo autoregressivo de ordem  $p$  ( $AR(p)$ ) é definido pela seguinte expressão:

$$Y_t = \phi_0 + \sum_{i=1}^p \phi_i Y_{t-i} + \epsilon_t, \quad t = p, p+1, \dots$$

em que  $Y_t$  é o valor da série temporal no tempo  $t$ ,  $\phi_0 \in \mathbb{R}$  é o termo de intercepto,  $\phi_i \in \mathbb{R}$  são os coeficientes autoregressivos para  $i = 1, 2, \dots, p$  e  $\epsilon_t$  é o termo de erro, assumido como ruído branco, ou seja,  $\epsilon_t \sim N(0, \sigma^2)$ , em que  $\sigma^2$  é um número real positivo ( $\sigma^2 \in \mathbb{R}^+$ ).

Este modelo assume que a série temporal é estacionária, ou seja, suas propriedades estatísticas como média e variância não mudam ao longo do tempo. Este pressuposto é importante para a validade do modelo e deve ser verificado durante a análise exploratória dos dados.

Além disso, esses modelos são razoáveis para a previsão do preço de ações porque capturam a dependência temporal nos dados financeiros. Em outras palavras, os preços das ações hoje são influenciados pelos preços passados. Isso se deve à inércia no comportamento dos mercados financeiros, em que as tendências e padrões históricos tendem a persistir por algum tempo. Ao considerar os valores passados para prever os valores futuros, os modelos AR conseguem incorporar essa dinâmica temporal.

#### Modelo Autoregressivo com Covariáveis

Para capturar a influência de fatores externos  $X_{kt}$  na série temporal  $Y_t$ , estendemos o modelo autoregressivo para incluir covariáveis, resultando em um modelo ARX (do inglês, *Autoregressive with Exogenous Inputs*), que é definido da seguinte maneira:

$$Y_t = \phi_0 + \sum_{i=1}^p \phi_i Y_{t-i} + \sum_{k=1}^m \beta_k X_{kt} + \epsilon_t, \quad t = p, p+1, \dots$$

em que  $Y_t$  é o valor da série temporal no tempo  $t$ ,  $\phi_0$  é o termo de intercepto,  $\phi_i$  são os coeficientes autoregressivos,  $X_{kt}$  são as covariáveis no tempo  $t$  para  $k = 1, 2, \dots, m$ ,  $\beta_k$  são os coeficientes das covariáveis e  $\epsilon_t$  é o termo de erro, assumido como ruído branco

( $\epsilon_t \sim N(0, \sigma^2)$ ), em que  $\sigma^2$  é a variância do erro.

Para facilitar a estimação, utilizamos uma notação matricial e vetorial, em que o vetor de observações da série temporal a partir do tempo  $p$  é  $\mathbf{Y} = [Y_p, Y_{p+1}, \dots, Y_T]^T$ , o vetor de covariáveis no tempo  $t$  é  $\mathbf{X}_t = [X_{1t}, X_{2t}, \dots, X_{mt}]^T$ , a matriz de covariáveis,  $\mathbf{X} \in \mathbb{R}^{(T-p+1) \times m}$  é  $\mathbf{X} = [\mathbf{X}_p, \mathbf{X}_{p+1}, \dots, \mathbf{X}_T]^T$ , o vetor de coeficientes autoregressivos é  $\boldsymbol{\phi} = [\phi_1, \phi_2, \dots, \phi_p]^T$ , o vetor de coeficientes das covariáveis é  $\boldsymbol{\beta} = [\beta_1, \beta_2, \dots, \beta_m]^T$ , a matriz contendo os valores passados da série temporal,  $\mathbf{Y}_{AR} \in \mathbb{R}^{(T-p+1) \times p}$  é  $\mathbf{Y}_{AR}$ .

Com isso, a matriz de *design*  $\mathbf{Z}$  que inclui as lags da série temporal e as covariáveis é dada por:

$$\mathbf{Z} = \begin{bmatrix} 1 & Y_{p-1} & Y_{p-2} & \cdots & Y_{p-p} & X_{1p} & X_{2p} & \cdots & X_{mp} \\ 1 & Y_p & Y_{p-1} & \cdots & Y_{p-p+1} & X_{1(p+1)} & X_{2(p+1)} & \cdots & X_{m(p+1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & Y_{T-1} & Y_{T-2} & \cdots & Y_{T-p} & X_{1T} & X_{2T} & \cdots & X_{mT} \end{bmatrix}.$$

Por outro lado, o vetor de coeficientes  $\boldsymbol{\theta}$  que inclui o intercepto, os coeficientes autoregressivos e os coeficientes das covariáveis é:

$$\boldsymbol{\theta}^T = [\phi_0 \quad \phi_1 \quad \phi_2 \quad \dots \quad \phi_p \quad \beta_1 \quad \beta_2 \quad \dots \quad \beta_m].$$

Logo, o modelo ARX pode ser representado de forma matricial como:

$$\mathbf{Y} = \mathbf{Z}\boldsymbol{\theta} + \boldsymbol{\epsilon}$$

em que  $\mathbf{Y}$  é o vetor de observações da série temporal,  $\mathbf{Z}$  é a matriz de *design*,  $\boldsymbol{\theta}$  é o vetor de coeficientes,  $\boldsymbol{\epsilon}$  é o vetor de termos de erro.

Os coeficientes  $\boldsymbol{\theta}$  são estimados minimizando a soma dos quadrados dos resíduos, de modo que essa é a função de custo a ser minimizada:

$$S(\boldsymbol{\theta}) = \sum_{t=p}^T (Y_t - \mathbf{Z}_t \boldsymbol{\theta})^2$$

em que  $Y_t$  é o valor observado da série temporal no tempo  $t$ ,  $\mathbf{Z}_t$  é a linha  $t$  da matriz de *design*  $\mathbf{Z}$ , que inclui o termo de intercepto, os valores passados da série temporal (lags) e as covariáveis no tempo  $t$  e  $\boldsymbol{\theta}$  é o vetor de coeficientes que inclui o intercepto  $\phi_0$ , os coeficientes autoregressivos  $\phi_i$  e os coeficientes das covariáveis  $\beta_k$ .

Com isso, resolvemos a equação normal  $(\mathbf{Z}^T \mathbf{Z})\boldsymbol{\theta} = \mathbf{Z}^T \mathbf{Y}$  e obtemos  $\hat{\boldsymbol{\theta}} = (\mathbf{Z}^T \mathbf{Z})^{-1} \mathbf{Z}^T \mathbf{Y}$ .

Portanto, nota-se que a inclusão de covariáveis em um modelo autoregressivo (AR) resulta no modelo ARX, que melhora a capacidade de captura de influências externas, além das dependências temporais intrínsecas na série.

## 2.5.2 Modelo de Médias Móveis

Ainda nessa linha, os Modelos de Médias Móveis (MA, do inglês *Moving Average*) são ferramentas para a análise e previsão de séries temporais. A característica central de um modelo MA é que o valor atual de uma série temporal é uma função linear dos termos de erros passados. Um modelo de médias móveis de ordem  $q$  ( $MA(q)$ ) é definido pela seguinte expressão:

$$Y_t = \mu + \sum_{j=1}^q \theta_j \epsilon_{t-j} + \epsilon_t, \quad t = q, q+1, \dots,$$

em que  $Y_t$  é o valor da série temporal no tempo  $t$ ,  $\mu \in \mathbb{R}$  é o termo de intercepto,  $\theta_j \in \mathbb{R}$  são os coeficientes de médias móveis para  $j = 1, 2, \dots, q$ ,  $\epsilon_t$  é o termo de erro no tempo  $t$ , assumido como ruído branco, ou seja,  $\epsilon_t \sim N(0, \sigma^2)$ , e  $\epsilon_{t-j}$  são os termos de erro passados para  $j = 1, \dots, q$ .

O modelo MA é especialmente útil quando a série temporal mostra um comportamento de “ruído branco” em seus erros, sugerindo que flutuações aleatórias em períodos passados podem afetar significativamente os valores futuros. Este modelo, como o AR, geralmente requer que a série seja estacionária.

Enquanto os modelos AR consideram a dependência dos valores passados da própria série, os modelos MA focam na estrutura dos erros passados, permitindo a identificação e correção de padrões de erro recorrentes. Isso é particularmente relevante para séries financeiras, em que choques de mercado ou eventos inesperados podem ter efeitos duradouros sobre os preços. Ao incorporar esses efeitos de erro, os modelos MA ajudam a refinar as previsões, especialmente quando a série apresenta comportamento de “ruído branco” nos erros.

### Modelo de Médias Móveis com Covariáveis

De modo análogo, para capturar a influência de fatores externos  $X_{kt}$  na série temporal  $Y_t$ , estendemos o modelo de médias móveis para incluir covariáveis, resultando em um modelo MAX (do inglês, *Moving Average with Exogenous Inputs*), que é definido da

seguinte maneira:

$$Y_t = \mu + \sum_{j=1}^q \theta_j \epsilon_{t-j} + \sum_{k=1}^m \beta_k X_{kt} + \epsilon_t, \quad t = q, q+1, \dots,$$

em que  $Y_t$  é o valor da série temporal no tempo  $t$ ,  $\mu$  é o termo de intercepto,  $\theta_j$  são os coeficientes de médias móveis,  $X_{kt}$  são as covariáveis no tempo  $t$  para  $k = 1, 2, \dots, m$ ,  $\beta_k$  são os coeficientes das covariáveis e  $\epsilon_t$  é o termo de erro, assumido como ruído branco ( $\epsilon_t \sim N(0, \sigma^2)$ ), em que  $\sigma^2$  é a variância do erro.

Para facilitar a estimação, utilizamos uma notação matricial e vetorial, em que o vetor de observações da série temporal a partir do tempo  $q$  é  $\mathbf{Y} = [Y_q, Y_{q+1}, \dots, Y_T]^T$ , o vetor de covariáveis no tempo  $t$  é  $\mathbf{X}_t = [X_{1t}, X_{2t}, \dots, X_{mt}]^T$ , a matriz de covariáveis,  $\mathbf{X} \in \mathbb{R}^{(T-q+1) \times m}$  é  $\mathbf{X} = [\mathbf{X}_q, \mathbf{X}_{q+1}, \dots, \mathbf{X}_T]^T$ , o vetor de coeficientes de médias móveis,  $\theta_j \in \mathbb{R}$  é  $\boldsymbol{\theta} = [\theta_1, \theta_2, \dots, \theta_q]^T$ , o vetor de coeficientes das covariáveis,  $\beta_k \in \mathbb{R}$  é  $\boldsymbol{\beta} = [\beta_1, \beta_2, \dots, \beta_m]^T$ , a matriz contendo os valores passados dos termos de erro,  $\mathbf{E}_{MA} \in \mathbb{R}^{(T-q+1) \times q}$  é  $\mathbf{E}_{MA}$ .

Com isso, a matriz de *design*  $\mathbf{Z}$  que inclui os termos de erro passados e as covariáveis é dada por:

$$\mathbf{Z} = \begin{bmatrix} 1 & \epsilon_{q-1} & \epsilon_{q-2} & \cdots & \epsilon_{q-q} & X_{1q} & X_{2q} & \cdots & X_{mq} \\ 1 & \epsilon_q & \epsilon_{q-1} & \cdots & \epsilon_{q-q+1} & X_{1(q+1)} & X_{2(q+1)} & \cdots & X_{m(q+1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \epsilon_{T-1} & \epsilon_{T-2} & \cdots & \epsilon_{T-q} & X_{1T} & X_{2T} & \cdots & X_{mT} \end{bmatrix}.$$

Por outro lado, o vetor de coeficientes  $\boldsymbol{\theta}$  que inclui o intercepto, os coeficientes de médias móveis e os coeficientes das covariáveis é:

$$\boldsymbol{\theta}^T = \left[ \mu \quad \theta_1 \quad \theta_2 \quad \dots \quad \theta_q \quad \beta_1 \quad \beta_2 \quad \dots \quad \beta_m \right].$$

Logo, o modelo MAX pode ser representado de forma matricial como:

$$\mathbf{Y} = \mathbf{Z}\boldsymbol{\theta} + \boldsymbol{\epsilon}$$

em que  $\mathbf{Y}$  é o vetor de observações da série temporal,  $\mathbf{Z}$  é a matriz de *design*,  $\boldsymbol{\theta}$  é o vetor de coeficientes,  $\boldsymbol{\epsilon}$  é o vetor de termos de erro.

Os coeficientes  $\boldsymbol{\theta}$  são estimados minimizando a soma dos quadrados dos resíduos, de modo que essa é a função de custo a ser minimizada:

$$S(\boldsymbol{\theta}) = \sum_{t=q}^T (Y_t - \mathbf{Z}_t \boldsymbol{\theta})^2$$

em que  $\mathbf{Z}_t$  é a linha  $t$  da matriz  $\mathbf{Z}$ .

Com isso, resolvemos a equação normal  $(\mathbf{Z}^T \mathbf{Z}) \boldsymbol{\theta} = \mathbf{Z}^T \mathbf{Y}$  e obtemos  $\hat{\boldsymbol{\theta}} = (\mathbf{Z}^T \mathbf{Z})^{-1} \mathbf{Z}^T \mathbf{Y}$ . Nesse sentido, nota-se que a inclusão de covariáveis em um modelo de médias móveis (MA) resulta no modelo MAX, que melhora a capacidade de captura de influências externas, além das dependências de erro passadas na série.

### 2.5.3 Modelo Autoregressivo e de Médias Móveis

Os Modelos Autoregressivos e de Médias Móveis (ARMA, do inglês *Autoregressive Moving Average Model*), por sua vez, combinam as características dos Modelos Autoregressivos (AR) e dos Modelos de Médias Móveis (MA). Esses modelos são utilizados para a análise e previsão de séries temporais, pois capturam tanto a dependência dos valores passados da série quanto a dos erros passados. Um modelo ARMA de ordem  $p, q$  ( $ARMA(p, q)$ ) é definido pela seguinte expressão:

$$Y_t = \phi_0 + \sum_{i=1}^p \phi_i Y_{t-i} + \sum_{j=1}^q \theta_j \epsilon_{t-j} + \epsilon_t, \quad t = \max\{p, q\}, \max\{p, q\} + 1, \dots$$

em que  $Y_t$  é o valor da série temporal no tempo  $t$ ,  $\phi_0 \in \mathbb{R}$  é o termo de intercepto,  $\phi_i \in \mathbb{R}$  são os coeficientes autoregressivos para  $i = 1, 2, \dots, p$ ,  $\theta_j \in \mathbb{R}$  são os coeficientes de médias móveis para  $j = 1, 2, \dots, q$ ,  $\epsilon_t$  é o termo de erro no tempo  $t$ , assumido como ruído branco, ou seja,  $\epsilon_t \sim N(0, \sigma^2)$ , em que  $\sigma^2 > 0$  e  $\epsilon_{t-j}$  são os termos de erro passados para  $j = 1, 2, \dots, q$ .

Os modelos ARMA são adequados para séries temporais estacionárias. Caso a série não seja estacionária, transformações como diferenciação podem ser aplicadas antes de ajustar o modelo ARMA. Essa flexibilidade faz do ARMA uma ferramenta para analisar e prever padrões em séries temporais em que as correlações tanto dos valores quanto dos erros são significativas.

Além disso, esse modelo captura tanto a dependência dos valores passados da série (como os modelos AR) quanto a influência dos erros passados (como os modelos MA). Isso significa que o modelo pode identificar e ajustar tanto as tendências subjacentes nos preços das ações quanto os impactos de eventos aleatórios e choques de mercado. Ao lidar

simultaneamente com os valores passados e os erros passados, os modelos ARMA oferecem uma abordagem mais abrangente e precisa para modelar séries temporais financeiras. Essa flexibilidade e capacidade de adaptação fazem do ARMA uma ferramenta para prever padrões nos preços das ações, provavelmente, melhorando a acurácia das previsões em comparação com modelos que consideram apenas um dos aspectos.

### Modelo Autoregressivo e de Médias Móveis com Covariáveis

Para capturar a influência de fatores externos  $X_{kt}$  na série temporal  $Y_t$ , podemos estender o modelo ARMA para incluir covariáveis, resultando em um modelo ARMAX (do inglês, *Autoregressive Moving Average with Exogenous Inputs*), que é definido da seguinte maneira:

$$Y_t = \phi_0 + \sum_{i=1}^p \phi_i Y_{t-i} + \sum_{j=1}^q \theta_j \epsilon_{t-j} + \sum_{k=1}^m \beta_k X_{kt} + \epsilon_t, \quad t = \max\{p, q\}, \max\{p, q\} + 1, \dots$$

em que  $Y_t$  é o valor da série temporal no tempo  $t$ ,  $\phi_0$  é o termo de intercepto,  $\phi_i$  são os coeficientes autoregressivos,  $\theta_j$  são os coeficientes de médias móveis,  $X_{kt}$  são as covariáveis no tempo  $t$  para  $k = 1, 2, \dots, m$ ,  $\beta_k$  são os coeficientes das covariáveis e  $\epsilon_t$  é o termo de erro, assumido como ruído branco ( $\epsilon_t \sim N(0, \sigma^2)$ ), em que  $\sigma^2$  é a variância do erro.

Para facilitar a estimação, utilizamos uma notação matricial e vetorial, em que o vetor de observações da série temporal a partir do tempo  $\max\{p, q\}$  é

$$\mathbf{Y} = [Y_{\max\{p, q\}}, Y_{\max\{p, q\}+1}, \dots, Y_T]^T.$$

O vetor de covariáveis no tempo  $t$ ,  $X_{kt} \in \mathbb{R}$ , é  $\mathbf{X}_t = [X_{1t}, X_{2t}, \dots, X_{mt}]^T$ . A matriz de covariáveis,  $\mathbf{X} \in \mathbb{R}^{(T-\max\{p, q\}+1) \times m}$ , é

$$\mathbf{X} = [\mathbf{X}_{\max\{p, q\}}, \mathbf{X}_{\max\{p, q\}+1}, \dots, \mathbf{X}_T]^T$$

O vetor de coeficientes autoregressivos,  $\phi_i \in \mathbb{R}$ , é  $\boldsymbol{\phi} = [\phi_1, \phi_2, \dots, \phi_p]^T$ , o vetor de coeficientes de médias móveis,  $\theta_j \in \mathbb{R}$ , é  $\boldsymbol{\theta} = [\theta_1, \theta_2, \dots, \theta_q]^T$  e o vetor de coeficientes das covariáveis,  $\beta_k \in \mathbb{R}$ , é  $\boldsymbol{\beta} = [\beta_1, \beta_2, \dots, \beta_m]^T$ .

Com isso, a matriz de *design*  $\mathbf{Z}$  que inclui as lags da série temporal, os termos de erro passados e as covariáveis é dada por:

$$\mathbf{Z} = \begin{bmatrix} 1 & Y_{\max\{p,q\}-1} & \cdots & Y_{\max\{p,q\}-p} & \epsilon_{\max\{p,q\}-1} & \cdots & \epsilon_{\max\{p,q\}-q} & X_{1\max\{p,q\}} & \cdots & X_{m\max\{p,q\}} \\ 1 & Y_{\max\{p,q\}} & \cdots & Y_{\max\{p,q\}-p+1} & \epsilon_{\max\{p,q\}} & \cdots & \epsilon_{\max\{p,q\}-q+1} & X_{1(\max\{p,q\}+1)} & \cdots & X_{m(\max\{p,q\}+1)} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 1 & Y_{T-1} & \cdots & Y_{T-p} & \epsilon_{T-1} & \cdots & \epsilon_{T-q} & X_{1T} & \cdots & X_{mT} \end{bmatrix}.$$

Por outro lado, o vetor de coeficientes  $\boldsymbol{\theta}$  que inclui o intercepto, os coeficientes autoregressivos, os coeficientes de médias móveis e os coeficientes das covariáveis é:

$$\boldsymbol{\theta}^T = [\phi_0, \phi_1, \phi_2, \dots, \phi_p, \theta_1, \theta_2, \dots, \theta_q, \beta_1, \beta_2, \dots, \beta_m].$$

Logo, o modelo ARMAX pode ser representado de forma matricial como:

$$\mathbf{Y} = \mathbf{Z}\boldsymbol{\theta} + \boldsymbol{\epsilon}$$

em que  $\mathbf{Y}$  é o vetor de observações da série temporal,  $\mathbf{Z}$  é a matriz de *design*,  $\boldsymbol{\theta}$  é o vetor de coeficientes,  $\boldsymbol{\epsilon}$  é o vetor de termos de erro.

Os coeficientes  $\boldsymbol{\theta}$  são estimados minimizando a soma dos quadrados dos resíduos, de modo que essa é a função de custo a ser minimizada:

$$S(\boldsymbol{\theta}) = \sum_{t=\max\{p,q\}}^T (Y_t - \mathbf{Z}_t\boldsymbol{\theta})^2$$

em que  $\mathbf{Z}_t$  é a linha  $t$  da matriz  $\mathbf{Z}$ .

Com isso, resolvemos a equação normal  $(\mathbf{Z}^T\mathbf{Z})\boldsymbol{\theta} = \mathbf{Z}^T\mathbf{Y}$  e obtemos  $\hat{\boldsymbol{\theta}} = (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T\mathbf{Y}$ . Portanto, nota-se que a inclusão de covariáveis em um modelo ARMA resulta no modelo ARMAX, que melhora a capacidade de captura de influências externas, além das dependências temporais e de erro passadas na série. Ao lidar simultaneamente com os valores passados, erros passados e covariáveis, o modelo ARMAX oferece uma abordagem mais abrangente e precisa para modelar séries temporais financeiras.

#### 2.5.4 Modelo Autoregressivo Integrado e de Médias Móveis

Os Modelos Autoregressivos Integrados e de Médias Móveis (ARIMA, do inglês *Autoregressive Integrated Moving Average Model*), por fim, são uma generalização dos modelos ARMA que incorporam a diferenciação para lidar com séries temporais não estacionárias. Esses modelos são utilizados para a análise e previsão de séries temporais,

especialmente quando há tendências ou sazonalidades. Um modelo ARIMA de ordem  $p, d, q$  ( $ARIMA(p, d, q)$ ) é definido pela seguinte expressão:

$$\Delta^d Y_t = \phi_0 + \sum_{i=1}^p \phi_i \Delta^d Y_{t-i} + \sum_{j=1}^q \theta_j \epsilon_{t-j} + \epsilon_t, \quad t = \max\{p, q\}, \max\{p, q\} + 1, \dots,$$

em que  $Y_t$  é o valor da série temporal no tempo  $t$ ,  $\Delta^d$  é o operador de diferenciação aplicado  $d$  vezes, para tornar a série estacionária,  $\phi_0 \in \mathbb{R}$  é o termo de intercepto,  $\phi_i \in \mathbb{R}$  são os coeficientes autoregressivos para  $i = 1, 2, \dots, p$ ,  $\theta_j \in \mathbb{R}$  são os coeficientes de médias móveis para  $j = 1, 2, \dots, q$ ,  $\epsilon_t$  é o termo de erro no tempo  $t$ , assumido como ruído branco, ou seja,  $\epsilon_t \sim N(0, \sigma^2)$ , com  $\sigma^2 > 0$  e  $\epsilon_{t-j}$  são os termos de erro passados.

O componente de diferenciação é importante para modelar séries que mostram tendências ou variações sazonais, permitindo que o modelo se ajuste às dinâmicas de longo prazo dos dados. Os modelos ARIMA são utilizados em previsões econômicas e financeiras, em que as séries frequentemente exibem padrões não estacionários.

Muitas vezes, os preços das ações exibem tendências ou sazonalidades e os modelos ARIMA, através da diferenciação, conseguem transformar essas séries em estacionárias, facilitando a modelagem e previsão. A combinação da diferenciação, componentes autoregressivos (AR) e de médias móveis (MA) permite que o modelo capture tanto as dependências nos valores passados quanto nos erros passados no contexto de previsão de ações.

### Modelo Autoregressivo Integrado e de Médias Móveis com Covariáveis

O modelo ARIMA (Autoregressive Integrated Moving Average) é uma extensão do modelo ARMA que pode lidar com séries temporais não estacionárias ao incorporar uma etapa de diferenciação. Quando adicionamos covariáveis a esse modelo, obtemos o modelo ARIMAX (Autoregressive Integrated Moving Average with Extra Input). O modelo ARIMAX de ordem  $p, d, q$  é definido pela seguinte expressão:

$$Y_t = \phi_0 + \sum_{i=1}^p \phi_i Y_{t-i} + \sum_{j=1}^q \theta_j \epsilon_{t-j} + \sum_{k=1}^m \beta_k X_{kt} + \epsilon_t, \quad t = \max\{p + d, q\}, \max\{p + d, q\} + 1, \dots$$

em que  $Y_t$  representa o valor da série temporal no tempo  $t$ , após  $d$  diferenciações para torná-la estacionária. Aqui,  $\phi_0$  é o termo constante,  $\phi_i$  são os coeficientes autoregressivos,

$\theta_j$  são os coeficientes das médias móveis,  $X_{kt}$  são as covariáveis no tempo  $t$ ,  $\beta_k$  são os coeficientes associados às covariáveis e  $\epsilon_t$  é o termo de erro, assumido como ruído branco ( $\epsilon_t \sim N(0, \sigma^2)$ ), em que  $\sigma^2 > 0$ .

Para estimar os parâmetros do modelo, utilizamos uma notação matricial e vetorial. O vetor de observações da série temporal a partir do tempo  $\max\{p + d, q\}$  é  $\mathbf{Y} = [Y_{\max\{p+d,q\}}, Y_{\max\{p+d,q\}+1}, \dots, Y_T]^T$ . O vetor de covariáveis no tempo  $t$  é  $\mathbf{X}_t = [X_{1t}, X_{2t}, \dots, X_{mt}]^T$  e a matriz de covariáveis é  $\mathbf{X} = [\mathbf{X}_{\max\{p+d,q\}}, \mathbf{X}_{\max\{p+d,q\}+1}, \dots, \mathbf{X}_T]^T$ . Os coeficientes autoregressivos formam o vetor  $\boldsymbol{\phi} = [\phi_1, \phi_2, \dots, \phi_p]^T$ , os coeficientes das médias móveis formam o vetor  $\boldsymbol{\theta} = [\theta_1, \theta_2, \dots, \theta_q]^T$  e os coeficientes das covariáveis formam o vetor  $\boldsymbol{\beta} = [\beta_1, \beta_2, \dots, \beta_m]^T$ .

A matriz de *design*  $\mathbf{Z}$  que inclui os valores passados da série temporal (após diferenciação), os termos de erro passados e as covariáveis é dada por:

$$\mathbf{Z} = \begin{bmatrix} 1 & Y_{\max\{p+d,q\}-1} & \cdots & Y_{\max\{p+d,q\}-p} & \epsilon_{\max\{p+d,q\}-1} & \cdots & \epsilon_{\max\{p+d,q\}-q} & X_{1 \max\{p+d,q\}} & \cdots & X_{m \max\{p+d,q\}} \\ 1 & Y_{\max\{p+d,q\}} & \cdots & Y_{\max\{p+d,q\}-p+1} & \epsilon_{\max\{p+d,q\}} & \cdots & \epsilon_{\max\{p+d,q\}-q+1} & X_{1(\max\{p+d,q\}+1)} & \cdots & X_{m(\max\{p+d,q\}+1)} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 1 & Y_{T-1} & \cdots & Y_{T-p} & \epsilon_{T-1} & \cdots & \epsilon_{T-q} & X_{1T} & \cdots & X_{mT} \end{bmatrix}.$$

O vetor de coeficientes  $\boldsymbol{\theta}$ , que inclui o intercepto, os coeficientes autoregressivos, os coeficientes das médias móveis e os coeficientes das covariáveis, é:

$$\boldsymbol{\theta} = [\phi_0, \phi_1, \phi_2, \dots, \phi_p, \theta_1, \theta_2, \dots, \theta_q, \beta_1, \beta_2, \dots, \beta_m]^T.$$

Dessa forma, o modelo ARIMAX pode ser representado de forma matricial como:

$$\mathbf{Y} = \mathbf{Z}\boldsymbol{\theta} + \boldsymbol{\epsilon}$$

em que  $\mathbf{Y}$  é o vetor de observações da série temporal,  $\mathbf{Z}$  é a matriz de *design*,  $\boldsymbol{\theta}$  é o vetor de coeficientes e  $\boldsymbol{\epsilon}$  é o vetor de termos de erro.

Os coeficientes  $\boldsymbol{\theta}$  são estimados minimizando a soma dos quadrados dos resíduos, de modo que a função de custo a ser minimizada é:

$$S(\boldsymbol{\theta}) = \sum_{t=\max\{p+d,q\}}^T (Y_t - \mathbf{Z}_t\boldsymbol{\theta})^2$$

em que  $\mathbf{Z}_t$  é a linha  $t$  da matriz  $\mathbf{Z}$ .

Ao resolver a equação normal  $(\mathbf{Z}^T \mathbf{Z})\boldsymbol{\theta} = \mathbf{Z}^T \mathbf{Y}$ , obtemos  $\hat{\boldsymbol{\theta}} = (\mathbf{Z}^T \mathbf{Z})^{-1} \mathbf{Z}^T \mathbf{Y}$ . Portanto, a inclusão de covariáveis em um modelo ARIMA resulta no modelo ARIMAX, que melhora a capacidade de capturar influências externas, além das dependências temporais e dos erros passados na série. Ao lidar simultaneamente com valores passados, erros passados e covariáveis, o modelo ARIMAX oferece uma abordagem mais abrangente e precisa para modelar séries temporais financeiras.

### 2.5.5 Exemplo numérico

Nesta subseção, assim como na Subseção 2.4.3, simulamos a mesma série temporal financeira com  $n = 500$  observações, em que cada observação representa um valor diário de variáveis do mercado. Definimos uma variável resposta ( $Y$ ) e quatro variáveis preditoras ( $X_1, X_2, X_3, X_4$ ) que influenciam o preço das ações. Assim, utilizamos as seguintes notações:

- $Y_t$ : rendimento da ação no tempo  $t$ ;
- $X_{t1}$ : volume de negociações no tempo  $t$  (simulado);
- $X_{t2}$ : Ibovespa no tempo  $t$  (simulado);
- $X_{t3}$ : taxa de juros no tempo  $t$  (simulado);
- $X_{t4}$ : ruído irrelevante no tempo  $t$ .

Considerando a dependência dos pares  $(\mathbf{X}_i, Y_i)$ , é necessário tomar amostras ao longo do tempo de maneira sequencial. Com isso, definimos o número de amostras  $n = 500$  e geramos as variáveis da seguinte forma:

- Volume de negociações  $X_{t1}$  no tempo  $t$ :  $X_{t1} \sim \text{Poisson}(1000)$ , i.e., o volume de negociações é modelado por uma distribuição de Poisson com média 1000;
- Ibovespa  $X_{t2}$  no tempo  $t$ :  $X_{t2} = \sin\left(\frac{20\pi t}{n}\right) + N(0, 0.1)$ , i.e., o índice de mercado é uma função senoidal com ruído normal adicionado;
- Taxa de juros  $X_{t3}$  no tempo  $t$ :  $X_{t3} = \frac{5t}{n} + N(0, 0.1)$ , i.e., a taxa de juros aumenta linearmente com ruído normal adicionado;
- Ruído irrelevante  $X_{t4}$  no tempo  $t$ :  $X_{t4} \sim N(5, 2)$ , i.e., o ruído irrelevante é modelado por uma distribuição normal com média 5 e desvio padrão 2;

- Preço da ação  $Y_t$  no tempo  $t$ : Calculado como uma função das variáveis acima mais um termo de ruído normal com média 0 e desvio padrão 0.5, i.e.,

$$Y_t = Y_{t-1} + 0.5 \log(X_{t1} + 1) + 2X_{t2} - 0.3X_{t3} + \epsilon_t$$

em que  $\epsilon_t \sim N(0, 0.5)$  é o termo de erro.

## Divisão dos Dados

Para avaliar o desempenho dos métodos de predição estudados ao longo desta seção, precisamos realizar uma divisão no conjunto de dados originais. Se denotarmos por  $\mathcal{O}$  esse conjunto original, a sua divisão é realizada a partir da criação de uma partição composta por três subconjuntos: o conjunto de treinamento  $\mathcal{T}$ , o conjunto de validação  $\mathcal{V}$  e o conjunto de teste  $\mathcal{U}$ . Para o exemplo em questão, tomamos  $|\mathcal{T}| = 350$  (70% das unidades amostrais),  $|\mathcal{V}| = 75$  (15% das unidades amostrais) e  $|\mathcal{U}| = 75$  (15% das unidades amostrais).

## Resultados

Ao realizar a divisão no conjunto de dados originais, ajusta-se quatro preditores no conjunto de treinamento  $\mathcal{T}$ : Modelo autoregressivo, modelo de médias móveis, modelo autoregressivo e de médias móveis e, por fim, modelo autoregressivo, integrado e de médias móveis. Em seguida, realizamos a predição das unidades amostrais pertencentes tanto ao conjunto de validação  $\mathcal{V}$  quanto ao conjunto de teste  $\mathcal{U}$  e avaliamos o desempenho dos preditores por meio do erro quadrático médio empírico. Os resultados estão sumarizados na Tabela 2.3 e na Figura 2.3.

Tabela 2.3: **Desempenho de quatro modelos de séries temporais.** Comparação dos erros quadráticos médios para quatro diferentes modelos de séries temporais: autoregressivo (AR), médias móveis (MA), autoregressivo e de médias móveis (ARMA) e autoregressivo integrado e de médias móveis (ARIMA). A comparação é feita tanto no conjunto de treino e validação quanto no conjunto de teste.

<b>Metodologia</b>	<b>Erro de Treino</b>	<b>Erro de Validação</b>	<b>Erro de Teste</b>
AR	1.0005	1.1513	4.0558
MA	1.0203	1.1527	4.2209
ARMA	3.8000	4.0959	1.9556
ARIMA	2.1001	2.3186	3.0226

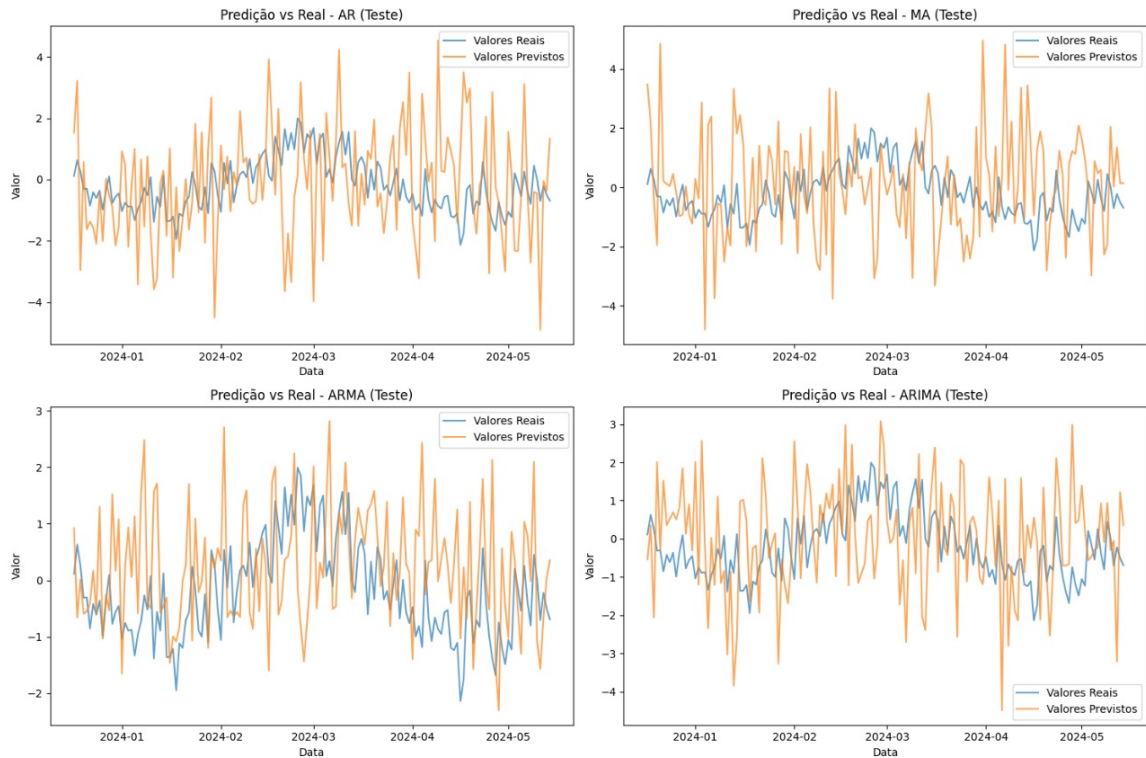


Figura 2.3: **Previsão de quatro modelos de séries temporais.** Os modelos em consideração são: autoregressivo (AR), médias móveis (MV), autoregressivo e de médias móveis (ARMA) e autoregressivo integrado e de médias móveis (ARIMA). São fornecidos quatro gráficos, um para cada modelo, em que no eixos das abscissas temos o identificador da unidade de tempo e no eixo das ordenadas os possíveis valores da variável resposta. A série temporal azul é a série real observada no conjunto de teste e as séries temporais laranjas são as séries previstas pelos modelos.

Diante disso, avalia-se os resultados da seguinte forma:

- **Modelo Autoregressivo:** Este modelo apresenta um erro de validação de 1.151279 e um erro de teste de 4.055761. Isso sugere que o modelo tem um desempenho razoável durante a validação, mas enfrenta dificuldades na generalização para novos dados, como indicado pelo erro de teste mais elevado.
- **Modelo de Médias Móveis:** O modelo de Médias Móveis tem um erro de validação de 1.152652, muito próximo ao do modelo autoregressivo e um erro de teste de 4.220932, que é o mais alto entre os modelos avaliados. Isso indica que o modelo de Médias Móveis não é tão eficaz na generalização, resultando em um desempenho pior em dados de teste.

- **Modelo Autoregressivo e de Médias Móveis (ARMA):** Este modelo combina características dos dois modelos anteriores e apresenta um erro de validação de 4.095925, que é significativamente mais alto do que os erros de validação dos modelos individuais. No entanto, seu erro de teste é 1.955577, o mais baixo entre todos os modelos, indicando uma boa capacidade de generalização para novos dados.
- **Modelo Autoregressivo Integrado e de Médias Móveis (ARIMA):** O modelo ARIMA apresenta um erro de validação de 2.318552 e um erro de teste de 3.022623. Embora o erro de validação seja mais alto do que os modelos autoregressivo e de médias móveis, o erro de teste é consideravelmente menor, sugerindo que o ARIMA tem a capacidade de generalização e pode capturar melhor a complexidade dos dados ao incorporar a integração dos termos autoregressivos e de médias móveis.

# Capítulo 3

## Intervalos de predição

A quantificação da incerteza é fundamental no mercado financeiro, especialmente no contexto da precificação de ações, onde pequenas variações podem ter grandes impactos. Intervalos de predição fornecem uma maneira prática de capturar essa incerteza, prevendo faixas de possíveis valores futuros de uma variável. No entanto, os intervalos de predição paramétrico pressupõem distribuições conhecidas dos dados, uma hipótese que pode ser difícil de verificar. Nesse contexto, os intervalos de predição conforme se destacam por serem agnósticos à distribuição dos dados, oferecendo uma abordagem flexível. No entanto, a abordagem clássica da predição conforme assume que os dados sejam permutáveis, uma hipótese que raramente se mantém no mercado financeiro. Neste capítulo, além de explorar os fundamentos teóricos da predição conforme, apresentamos dois algoritmos projetados especificamente para séries temporais, ferramentas que serão aplicadas à análise de dados financeiros no Trabalho de Graduação B.

### 3.1 Uma visão geral sobre a predição conforme

O objetivo dos intervalos de predição é fornecer uma faixa de valores que se espera que inclua uma observação futura, com uma determinada probabilidade, dada uma amostra dos dados. Para construir esses intervalos, são necessários métodos que levem em consideração tanto a incerteza associada à estimação dos parâmetros do modelo quanto a variabilidade inerente aos dados. Tradicionalmente, isso envolve suposições sobre a distribuição dos resíduos ou erros. No entanto, métodos mais robustos e não paramétricos também são empregados, especialmente em situações em que as suposições paramétricas não são adequadas ou desejáveis. Nesse sentido, os métodos conformais surgem como uma

abordagem não-paramétrica para a construção de intervalos de predição, que são particularmente atrativos por, em geral, assumirem apenas que os dados sejam permutáveis e por serem agnósticos ao preditor adotado para a análise. Esses métodos são baseados no conceito de conformidade, que avalia quão “típica” ou “não usual” uma nova observação é, em comparação com as observações em uma amostra.

A predição conforme é uma técnica estatística utilizada para quantificar a incerteza nas previsões de um modelo ou um algoritmo de aprendizado de máquina. Ao invés de fornecer apenas uma predição pontual, a predição conforme gera conjuntos de predição que cobrem o valor verdadeiro com um nível de confiança pré-definido. A ideia central da predição conforme é construir intervalos ou conjuntos de predição que sejam agnósticos ao modelo preditivo sem que se faça suposições fortes sobre os dados. Inicialmente, intervalos ou conjuntos de predição conformais foram propostos sob a suposição de permutabilidade dos dados.

Nesse contexto, para construir intervalos ou conjuntos de predição conforme, define-se uma função de não-conformidade que avalia o quão típica uma observação de um conjunto de dados é em relação aos outros. Por exemplo, em um cenário de regressão, a medida de não-conformidade pode ser a magnitude dos resíduos. Nesse sentido, uma observação é dita não conforme às demais quando o seu resíduo é atípico em relação às demais. Posteriormente, os dados são divididos em um conjunto de treinamento e um conjunto de teste. O modelo é ajustado aos dados de treinamento, e a função de não-conformidade é aplicada tanto ao conjunto de treinamento quanto ao de teste. Assim, a partir de uma nova observação, calcula-se a medida de não-conformidade assumindo que esta nova observação faz parte do conjunto de dados. A distribuição empírica das medidas de não-conformidade, incluindo a da nova observação, é utilizada para determinar o intervalo de predição.

A predição conformal é uma técnica estatística desenvolvida no final dos anos 1990 e início dos anos 2000 por [Shafer e Vovk](#). A principal inovação da predição conformal é sua capacidade de fornecer garantias teóricas sobre a validade dos intervalos de predição, independentemente da distribuição dos dados, desde que a suposição de permutabilidade seja satisfeita. Desde então, a predição conformal tem evoluído significativamente. Métodos como a *split conformal prediction* e a *quantile conformal prediction* foram desenvolvidos para lidar com a complexidade computacional da abordagem original. A *split conformal prediction* divide os dados em conjuntos de treinamento e teste, simplificando os cálculos

de não-conformidade. A *quantile conformal prediction* utiliza quantis da distribuição dos resíduos para construir intervalos de predição, tornando o processo mais eficiente.

Embora a suposição de permutabilidade seja um conceito fundamental para a construção de conjuntos de predição conformais, em muitos casos práticos, essa suposição pode não ser válida. Por exemplo, em séries temporais, a ordem dos dados importa e os dados podem ter dependências temporais, de modo que as observações podem ser correlacionadas tornando a permutabilidade uma suposição irrealista. Para lidar com esses cenários, foram desenvolvidos métodos de predição conforme, que são elucidados posteriormente, que relaxam ou evitam a suposição de permutabilidade, permitindo a aplicação da predição conformal em contextos mais amplos.

### 3.1.1 O problema de construir intervalos de predição

Primeiramente, vamos considerar o problema de construção de um intervalo de predição a partir de um caso mais simples no qual consideramos apenas uma sequência de observações  $z_1, z_2, z_3, \dots, z_n \in Z$ , em que  $Z$  é um espaço mensurável. Cada novo exemplo  $z_{n+1}$  é previsto com base no conjunto de treinamento formado pelos exemplos anteriores  $(z_1, \dots, z_n)$ .

Um conjunto de predição é definido como uma função mensurável  $\gamma$ , que mapeia uma sequência de exemplos  $(z_1, \dots, z_n) \in Z^n$  para um subconjunto de  $Z$ , denotado  $\gamma(z_1, \dots, z_n) \subseteq Z$ . Essa função deve satisfazer a condição de que o seguinte conjunto é mensurável em  $Z^{n+1}$ :

$$\{(z_1, \dots, z_{n+1}) : z_{n+1} \in \gamma(z_1, \dots, z_n)\},$$

em que  $\gamma(z_1, \dots, z_n)$  é o conjunto de possíveis valores para o próximo exemplo  $z_{n+1}$ .

Como dito anteriormente, construir intervalos de predição é um problema central em estatística e aprendizado de máquina, cujo objetivo é quantificar a incerteza associada às previsões de um modelo. Um intervalo de predição é um intervalo que esperamos incluir uma nova observação, com uma probabilidade pré-especificada, denominada nível de confiança. Nesse sentido, dado um conjunto de dados  $(\mathbf{x}, \mathbf{y}) := \{z_i := (\mathbf{x}_i, y_i) \in \mathbb{R}^p \times \mathbb{R} : i = 1, 2, \dots, n\}$ , em que  $\mathbf{x}_i$  representa o valor do vetor de covariáveis  $\mathbf{X}$  para a  $i$ -ésima unidade amostral e  $y_i$  representa a observação da variável resposta  $Y$  para a  $i$ -ésima unidade amostral,  $i = 1, 2, \dots, n$ , queremos construir um intervalo de predição

$\gamma(z_1, \dots, z_n) := [L, U]$  para uma nova unidade amostral tal que  $\mathbf{X} = \mathbf{x}_{n+1}$ , de modo que  $L := L(\mathbf{x}, \mathbf{y}, \mathbf{x}_{n+1})$  e  $U := U(\mathbf{x}, \mathbf{y}, \mathbf{x}_{n+1})$  e

$$P(L \leq Y \leq U \mid \mathbf{X} = \mathbf{x}_{n+1}) = 1 - \alpha,$$

em que  $L$  é o limite inferior do intervalo de predição para a variável resposta  $Y$  atrelada à nova unidade amostral,  $U$  é o limite superior do intervalo de predição para a variável resposta  $Y$  atrelada à nova unidade amostral, e, por fim,  $\alpha \in (0, 1)$  é o nível de significância e  $1 - \alpha$  é o nível de confiança.

O desafio principal na construção desses intervalos reside em garantir que a cobertura desejada  $1 - \alpha$  seja atingida com uma alta precisão, independentemente da distribuição dos dados. Em muitas situações práticas, a suposição de que os dados são independentes e identicamente distribuídos (i.i.d.) não é razoável. Especialmente em contextos de dados financeiros, em que podem existir heterocedasticidade, autocorrelação ou outras formas de dependência. Outra abordagem clássica para construir intervalos de predição é assumir que os resíduos são normalmente distribuídos. No entanto, essa suposição frequentemente não é válida.

Para contornar essas limitações, métodos como a predição conforme foram desenvolvidos. Inicialmente, a predição conforme foi proposta de modo que a única hipótese a ser atendida pelos dados é a de permutabilidade, i.e., que a ordem de coleta dos dados é irrelevante.

### 3.1.2 A suposição de permutabilidade

Preditores conformais operam, em geral, sob a suposição de que os exemplos  $(z_1, \dots, z_n)$  são permutáveis, o que significa que qualquer permutação desses exemplos tem a mesma distribuição de probabilidade que a sequência original. Esta suposição é mais fraca do que a suposição de que os exemplos são independentes e identicamente distribuídos (i.i.d.), comum em muitos cenários estatísticos [Vovk et al. \(2005\)](#). A permutabilidade exige apenas que a ordem dos exemplos não importe, ou seja, qualquer permutação dos exemplos deve resultar no mesmo comportamento estatístico. Matematicamente, para qualquer permutação  $\pi$  dos índices  $\{1, 2, \dots, n\}$ , temos

$$(z_1, z_2, \dots, z_n) \stackrel{d}{=} (z_{\pi(1)}, z_{\pi(2)}, \dots, z_{\pi(n)}),$$

em que  $\stackrel{d}{=}$  denota igualdade em distribuição.

A suposição de i.i.d. é, por sua vez, mais restritiva, já que exige que cada exemplo  $z_i$  seja independente de todos os outros exemplos e que todos os exemplos tenham a mesma distribuição. Enquanto a permutabilidade apenas requer que a distribuição conjunta dos exemplos seja invariante sob permutações, a suposição i.i.d. exige tanto a independência quanto a igualdade das distribuições individuais dos exemplos. Por ser menos restritiva, a predição conforme sob a hipótese de permutabilidade pode ser aplicada a uma gama maior de problemas práticos. Todavia, como já comentamos, ela se torna inviável quando estamos lidando com séries temporais, tais como as séries financeiras que vamos estudar nesta monografia.

### 3.1.3 Medidas de não-conformidade

Medidas de não-conformidade são fundamentais em predições conformais para avaliar o quão atípico é um exemplo novo em relação a uma amostra de referência. Formalmente, uma medida de não-conformidade é uma função  $A : \mathcal{Z}^{n+1} \rightarrow \mathbb{R}$ , em que  $\mathcal{Z}^{n+1}$  representa o produto cartesiano de  $\mathcal{Z}$  com ele mesmo  $n + 1$  vezes, mapeando-o para o conjunto dos números reais  $\mathbb{R}$ . Essa função atribui a cada exemplo  $(z_1, \dots, z_{n+1})$  um valor numérico que reflete seu grau de atipicidade em relação aos exemplos anteriores. Esta função  $A$  pode ser, por exemplo, a diferença absoluta entre o valor observado  $z_{n+1}$  e um valor previsto com base nos exemplos  $(z_1, \dots, z_n)$ . Em notação matemática, nesse caso, se  $\hat{z}_{n+1}$  é a previsão baseada em  $(z_1, \dots, z_n)$ , então a medida de não-conformidade pode ser expressa como:

$$A(z_1, \dots, z_{n+1}) = |z_{n+1} - \hat{z}_{n+1}|,$$

em que  $\hat{z}_{n+1}$  é a predição de  $z_{n+1}$  usando um modelo treinado nos exemplos  $(z_1, \dots, z_n)$ . A escolha, portanto, de uma boa medida de não-conformidade é importante para o desempenho do preditor conformal. Ela deve refletir adequadamente a probabilidade de que uma observação seja típica ou atípica dentro do conjunto de dados. A função deve ser sensível o suficiente para diferenciar entre exemplos que são comuns e exemplos que são uma exceção.

### 3.1.4 Predição conforme

Fixado um nível de significância  $\alpha \in (0, 1)$ , queremos construir um conjunto de predição  $\gamma(z_1, \dots, z_n)$  baseado no conjunto de treinado  $(z_1, \dots, z_n) \in Z^n$  de modo que

$$P(z_{n+1} \in \gamma(z_1, \dots, z_n)) = 1 - \alpha,$$

em que  $z_{n+1} \in Z$  representa um novo objeto. Dessa forma, cada  $\alpha \in (0, 1)$  está associado a uma conjunto de predição. Nesse sentido, temos uma família de conjuntos de predição  $\{\gamma_\alpha(z_1, \dots, z_n) : \alpha \in (0, 1)\}$  tal que para quaisquer  $\alpha_1, \alpha_2$  em que  $0 \leq \alpha_1 \leq \alpha_2 \leq 1$ , temos:

$$\gamma_{\alpha_1}(z_1, \dots, z_n) \supseteq \gamma_{\alpha_2}(z_1, \dots, z_n).$$

Em outras palavras, para qualquer sequência de exemplos  $(z_1, \dots, z_n)$ , todos os valores possíveis que  $\gamma_{\alpha_2}$  pode produzir também são valores possíveis que  $\gamma_{\alpha_1}$  pode produzir. Isso implica que a função  $\gamma_{\alpha_1}$  é, de alguma forma, mais permissiva ou abrangente em termos dos valores que pode gerar em comparação com  $\gamma_{\alpha_2}$ .

Para construir um conjunto de predição conforme, utilizamos uma medida de não-conformidade  $A$ . Dada uma sequência de exemplos  $(z_1, \dots, z_n)$  e um novo exemplo  $z_{n+1}$ , calculamos a medida de não-conformidade para cada exemplo  $z_i$  na sequência, incluindo  $z_{n+1}$ :

$$A(z_1, \dots, z_{n+1}, z_i) \quad \text{para } i = 1, \dots, n+1,$$

A seguir, calculamos o valor  $p$  para  $z_{n+1}$ , que representa a fração de exemplos na sequência que são tão ou mais não-conformes que  $z_{n+1}$ :

$$p(z_{n+1}) := \frac{|\{i = 1, \dots, n+1 : A(z_1, \dots, z_n, z_i) \geq A(z_1, \dots, z_n, z_{n+1})\}|}{n+1},$$

em que,  $\{i = 1, \dots, n+1 : A(z_1, \dots, z_n, z_i) \geq A(z_1, \dots, z_n, z_{n+1})\}$  é o conjunto de índices dos exemplos cuja medida de não-conformidade é maior ou igual à medida de não-conformidade do novo exemplo  $z_{n+1}$  e  $|\cdot|$  denota o tamanho desse conjunto.

Fixado um nível de significância  $\alpha \in (0, 1)$ , o conjunto de predição conforme  $\gamma_\alpha(z_1, \dots, z_n)$  para um novo exemplo  $z_{n+1}$  é, então, definido como o conjunto de todos os exemplos  $z \in Z$  tais que  $p(z) > \alpha$ :

$$\gamma_\alpha(z_1, \dots, z_n) = \{z \in Z : p(z) > \alpha\}.$$

Esse conjunto de predição possui a propriedade de validade, o que significa que a probabilidade de o verdadeiro valor  $z_{n+1}$  estar fora do conjunto  $\gamma_\alpha(z_1, \dots, z_n)$  não excede  $\alpha$ . Em termos formais, para qualquer distribuição de probabilidade dos exemplos, temos:

$$P(z_{n+1} \notin \gamma_\alpha(z_1, \dots, z_n)) \leq \alpha.$$

Essa propriedade é garantida independentemente da distribuição específica dos dados, desde que a suposição de permutabilidade dos exemplos seja satisfeita.

### 3.1.5 Validade e eficiência

Conforme brevemente comentado acima, em um problema de predição conformal, dois conceitos fundamentais são a validade e a eficiência. Esses conceitos são importantes para avaliar a qualidade dos conjuntos de predição gerados pelo modelo.

A validade refere-se à propriedade de um conjunto de predição conforme  $\gamma_\alpha(z_1, \dots, z_n)$  incluir o verdadeiro valor do próximo exemplo  $z_{n+1}$  com uma probabilidade de pelo menos  $1 - \alpha$ . Formalmente, dizemos que um conjunto de predição é válido se:

$$P(z_{n+1} \notin \gamma_\alpha(z_1, \dots, z_n)) \leq \alpha.$$

A eficiência, por sua vez, refere-se ao tamanho dos conjuntos de predição gerados. Um conjunto de predição  $\gamma_\alpha(z_1, \dots, z_n)$  é considerado eficiente se, para um dado nível de significância  $\alpha$ , ele for tão pequeno quanto possível enquanto ainda mantém a validade. A eficiência pode ser quantitativamente medida pelo comprimento (ou volume, em espaços multidimensionais) do conjunto de predição  $\gamma_\alpha(z_1, \dots, z_n)$ . O objetivo é minimizar a expectativa do comprimento (ou volume) do conjunto de predição, mantendo a propriedade de validade.

A eficiência, portanto, é formalizada como o problema de otimização que busca minimizar a expectativa do comprimento dos conjuntos de predição, sujeita à condição de validade, que é:

$$P(z_{n+1} \in \gamma_\alpha(z_1, \dots, z_n)) \geq 1 - \alpha.$$

Esta condição garante que, independentemente do comprimento ou volume do conjunto de predição, a probabilidade de conter o verdadeiro valor  $z_{n+1}$  é pelo menos  $1 - \alpha$ . Portanto, a eficiência envolve encontrar um conjunto  $\gamma_\alpha$  que seja o mais pequeno possível em média,

enquanto ainda satisfaz esta condição de validade.

Dito isso, nota-se a existência de um *tradeoff* entre validade e eficiência que surge porque conjuntos de predição mais largos tendem a ser mais válidos, já que têm maior probabilidade de conter o verdadeiro valor  $z_{n+1}$ . No entanto, conjuntos mais largos são menos eficientes. Por outro lado, conjuntos de predição menores são mais eficientes, mas podem não garantir a validade, ou seja, a probabilidade de conter o verdadeiro valor  $z_{n+1}$  pode ser menor do que  $1 - \alpha$ .

Seja um conjunto de predição  $C_n$  construído a partir dos dados  $\mathcal{D}_n = \{z_1, z_2, \dots, z_n\}$ , onde  $z_{n+1}$  é o verdadeiro valor futuro a ser previsto. Definimos a validade  $V$  como uma variável binária:

$$V = \mathbb{I}\{z_{n+1} \in C_n\},$$

em que  $\mathbb{I}$  é a função indicadora que assume valor 1 se  $z_{n+1}$  pertence ao conjunto de predição  $C_n$ , e 0 caso contrário.

A validade esperada do conjunto de predição é então dada por:

$$\mathbb{E}[V] = P(z_{n+1} \in C_n).$$

Para que o conjunto de predição seja considerado válido com nível de confiança  $1 - \alpha$ , ele deve satisfazer:

$$P(z_{n+1} \in C_n) \geq 1 - \alpha.$$

A eficiência  $E$  é inversamente proporcional ao tamanho  $|C_n|$  do conjunto de predição. Uma métrica comum para eficiência é:

$$E = \frac{1}{|C_n|},$$

em que conjuntos menores resultam em maior eficiência.

O *trade-off* entre validade e eficiência surge porque para garantir validade, é necessário um conjunto  $C_n$  suficientemente grande, o que reduz a eficiência. Formalmente, esse *trade-off* pode ser expresso como uma relação inversa:

$$E \propto \frac{1}{\sup\{|C_n| : P(z_{n+1} \in C_n) \geq 1 - \alpha\}}.$$

Ou seja, quanto maior for o conjunto de predição necessário para garantir a validade, menor será a eficiência.

## 3.2 Predição conforme para dados permutáveis

Como dito anteriormente, a predição conforme para dados permutáveis é uma técnica estatística para construir conjuntos de predição sem depender de suposições rígidas sobre a distribuição dos dados. Ao invés de assumir que os dados são independentes e identicamente distribuídos (i.i.d.), a predição conforme, nesse contexto, opera sob a suposição de permutabilidade, em que a ordem dos dados não influencia a análise.

### 3.2.1 Algoritmo proposto por [Vovk et al. \(2005\)](#)

Consideremos uma sequência de observações sucessivas  $z_1, z_2, z_3, \dots$ , onde cada observação  $z_i$  é composta por uma covariável  $x_i$  e sua variável resposta  $y_i$ . Matematicamente, podemos representar cada observação como  $z_i = (\mathbf{x}_i, y_i)$ . Sejam

- $\mathcal{X}$  é um espaço mensurável chamado de espaço das covariáveis;
- $\mathcal{Y}$  é um espaço mensurável chamado de espaço das variáveis resposta;
- $\mathcal{Z}$  é o espaço das observações, definido como  $\mathcal{Z} \subseteq \mathcal{X} \times \mathcal{Y}$ .

Assim, cada  $z_i$  pertence ao espaço  $\mathcal{Z}$ , ou seja,  $z_i \in \mathcal{Z}$ .

No  $(n + 1)$ -ésimo ensaio, o vetor de covariáveis  $\mathbf{x}_{n+1}$  é dado, e o objetivo é prever sua variável resposta  $y_{n+1}$ . Dito isso, a predição conforme é uma abordagem estatística desenvolvida para quantificar a incerteza nas previsões de modelos de aprendizado de máquina, fornecendo intervalos de predição válidos e eficientes. O algoritmo proposto por [Vovk et al. \(2005\)](#) baseia-se no conceito de conformidade, que avalia quão típico ou atípico um novo exemplo é em relação a um conjunto de exemplos observados.

Com isso, o algoritmo de predição conforme proposto por [Vovk et al. \(2005\)](#) segue, de modo geral, os seguintes passos:

1. **Treinar o modelo:** Ajustamos um modelo preditivo  $\hat{f}$  ao conjunto de dados  $\{z_i := (\mathbf{x}_i, y_i) : i = 1, \dots, n\}$ , obtendo uma função de predição  $\hat{f}(\mathbf{x})$  que estima a resposta  $y$  para um dado vetor de covariáveis  $\mathbf{x}$ .

2. **Calcular a medida de não-conformidade:** Definimos uma função de não-conformidade  $A$  que quantifica quão atípico é o valor observado  $y_i$  dado o valor predito  $\hat{f}(\mathbf{x}_i)$  para cada  $i = 1, 2, \dots, n$ . Uma escolha comum para a função de não-conformidade, em um problema de regressão, é a magnitude do resíduo:

$$A_i := A(z_1, \dots, z_n, z_i) = |y_i - \hat{f}(\mathbf{x}_i)|, \quad i = 1, 2, \dots, n.$$

3. **Aplicar a função de não-conformidade considerando uma nova unidade amostral:** Calculamos a medida de não-conformidade para uma nova unidade amostra  $z_{n+1} := (\mathbf{x}_{n+1}, y_{n+1})$ , i.e.,

$$A_{n+1} := A(z_1, \dots, z_n, z_{n+1}).$$

4. **Construir o conjunto de predição:** Para o novo vetor de covariáveis  $\mathbf{x}_{n+1}$ , calculamos o conjunto de predição de predição  $\gamma_\alpha$  para a resposta  $y_{n+1}$  de tal forma que a probabilidade de  $y_{n+1}$  estar fora deste intervalo seja menor ou igual a  $\alpha \in (0, 1)$ . O intervalo é dado por:

$$\gamma_\alpha(z_1, \dots, z_n, \mathbf{x}_{n+1}) = \{y \in \mathbb{R} : p(\mathbf{x}_{n+1}, y) > \alpha\}$$

em que

$$p((\mathbf{x}_{n+1}, y)) = \frac{|\{i = 1, \dots, n+1 : A_i \geq A_{n+1}\}|}{n+1}.$$

5. **Garantia de validade:** Este conjunto de predição possui a propriedade de validade, ou seja, a probabilidade de  $y_{n+1}$  estar fora do intervalo  $\gamma_\alpha(z_1, \dots, z_n, \mathbf{x}_{n+1})$  é no máximo  $\alpha$ :

$$P(y_{n+1} \notin \gamma_\alpha(z_1, \dots, z_n, \mathbf{x}_{n+1})) \leq \alpha.$$

### 3.2.2 O papel da permutabilidade na predição conforme

Para demonstrar a necessidade da permutabilidade, considere o conjunto de predição conforme  $\gamma_\alpha(z_1, \dots, z_n, \mathbf{x}_{n+1})$  com um nível de significância  $\alpha \in (0, 1)$ . Para garantir a validade, precisamos que:

$$P\{Y_{n+1} \in \gamma_\alpha(z_1, \dots, z_n, \mathbf{x}_{n+1})\} = P\{p_{n+1} > \alpha\} \geq 1 - \alpha, \quad (3.1)$$

em que  $Y_{n+1}$  é a variável resposta para a nova observação  $\mathbf{x}_{n+1}$ ,  $p_{n+1}$  é o valor  $p$  associado à nova observação  $(\mathbf{x}_{n+1}, y_{n+1})$ , que representa a fração de exemplos no conjunto de teste que são tão ou mais não-conformes do que a nova observação, i.e.,

$$p_{n+1} = \frac{|\{i = 1, \dots, n+1 : A_i \geq A_{n+1}\}|}{n+1}.$$

Dito isso, a cobertura garantida pela permutabilidade da sequência  $\{(\mathbf{X}_i, Y_i)\}_{i=1}^{n+1}$ . Para entender essa cobertura, consideramos o seguinte:

$$\begin{aligned} P\{Y_{n+1} \in \gamma(z_1, \dots, z_n, \mathbf{x}_{n+1})\} &= P\{p_{n+1} > \alpha\} \\ &= P\left\{\frac{1}{n+1} \sum_{i=1}^{n+1} \mathbf{1}(A_i \geq A_{n+1}) > \alpha\right\} \\ &= P\left\{\sum_{i=1}^{n+1} \mathbf{1}(A_i \geq A_{n+1}) > (n+1)\alpha\right\}, \end{aligned} \quad (3.2)$$

o que equivale à probabilidade de que a posição de  $A_{n+1}$  entre os valores  $A_i$ ,  $i = 1, 2, \dots, n$ , seja maior do que a  $(n+1)\alpha$ .

A permutabilidade da sequência  $\{(\mathbf{X}_i, Y_i)\}_{i=1}^{n+1}$  garante que a posição de  $A_{n+1}$  pode ser qualquer número inteiro de 1 a  $n+1$  com probabilidade igual. Portanto, para  $k = 1, 2, \dots, n+1$ ,

$$P\left\{\sum_{i=1}^{n+1} \mathbf{1}(A_i \geq A_{n+1}) > k\right\} = 1 - \frac{k}{n+1} = \frac{n+1-k}{n+1}. \quad (3.3)$$

Dessa forma, a partir de (3.3) e (3.2) garantimos que a cobertura mencionada em (3.1) é alcançada. No entanto, essa estrutura não garante a cobertura esperada em situações onde a permutabilidade não se aplica. Isso porque a suposição garante que a probabilidade de o valor verdadeiro  $y_{n+1}$  estar fora do intervalo de predição  $\gamma_\alpha(z_1, z_2, \dots, z_n, \mathbf{x}_{n+1})$  é exatamente  $\alpha$ , validando a construção do conjunto de predição conforme.

Portanto, sem a suposição de permutabilidade, a distribuição conjunta dos escores de não-conformidade  $(A_1, A_2, \dots, A_n, A_{n+1})$  poderia ser afetada pela ordem das observações. Isso resultaria em intervalos de predição que não garantem a cobertura correta, comprometendo a confiança nos resultados.

### 3.2.3 Complexidade computacional do algoritmo proposto por Vovk et al. (2005)

Nesse sentido, fica claro a complexidade computacional da predição conforme que surge principalmente da necessidade de calcular medidas de não-conformidade e valores  $p$  para todos os possíveis valores da variável resposta  $y_{n+1}$ . Em um contexto de regressão, isso significa examinar vários possíveis valores de  $y$  para cada nova previsão, o que pode ser computacionalmente proibitivo. A seguir, discutimos o porquê desses fatores tornarem computacionalmente custosa a construção dos intervalos de predição conforme:

1. **Cálculo da medida de não-conformidade:** Para cada novo exemplo, a predição conforme requer o cálculo de uma medida de não-conformidade  $A_i$ . Isso envolve treinar um modelo de predição e calcular o resíduo para cada exemplo, o que pode ser demorado em termos de tempo de computação, especialmente para modelos complexos ou grandes conjuntos de dados.
2. **Avaliação de todos os possíveis valores  $y_{n+1}$ :** A predição conforme clássica exige a avaliação de todos os possíveis valores da variável resposta  $y_{n+1}$  para determinar quais valores satisfazem a condição de não-conformidade. Formalmente, para cada possível  $y$ , precisamos calcular:

$$p_{n+1} = \frac{|\{i = 1, \dots, n + 1 : A_i \geq A_{n+1}\}|}{n + 1}.$$

Este processo pode ser extremamente intensivo em termos computacionais, pois implica calcular a medida de não-conformidade para uma quantidade potencialmente infinita de valores  $y$ .

3. **Ordenação dos escores de não-conformidade:** A construção dos intervalos de predição conforme requer a ordenação dos escores de não-conformidade para calcular os valores  $p$ . A ordenação de um grande número de escores pode ser computacionalmente intensiva, especialmente em grandes conjuntos de dados.
4. **Repetição do processo:** Para cada nova observação  $(\mathbf{x}_{n+1}, y_{n+1})$ , todo o processo de cálculo de medidas de não-conformidade, avaliação de possíveis valores de  $y$  e ordenação dos escores deve ser repetido. Isso resulta em uma complexidade computacional de ordem

$$O(n \cdot g(k)),$$

em que  $n$  representa o número de novas observações e  $g(k)$  é uma função de crescimento exponencial em relação ao número de possíveis valores de  $y$ , como por exemplo:

$$g(k) = 2^k, \quad \text{ou mais geralmente, } g(k) = O(e^{\beta k}), \text{ para algum } \beta > 0.$$

Assim, a complexidade total do algoritmo pode ser expressa como:

$$O(n \cdot 2^k),$$

o que indica um crescimento linear em  $n$  e exponencial em  $k$ , tornando o problema computacionalmente desafiador para grandes valores de  $k$ .

Esses fatores combinados fazem com que a aplicação direta da predição conforme em grandes conjuntos de dados ou em tempo real seja impraticável sem realizar a otimização do processo de construção dos intervalos de predição. É nesse contexto, assim, que as abordagens de *split conformal prediction* e *quantile conformal prediction* surgem, pois elas simplificam os cálculos necessários e reduzem a carga computacional [Kath e Ziel \(2021\)](#).

### 3.2.4 *Split conformal prediction*

A *split conformal prediction* é uma abordagem simplificada que divide o conjunto de dados original em dois subconjuntos: um conjunto de treinamento e um conjunto de validação. Este método reduz a complexidade computacional, uma vez que a medida de não-conformidade é calculada apenas com base no conjunto de validação. O procedimento é descrito a seguir:

1. **Dividir os dados:** Dividimos o conjunto de dados  $\{(\mathbf{x}_i, y_i) : i = 1, \dots, n\}$  em um conjunto de treinamento  $\{(\mathbf{x}_i, y_i) : i = 1, \dots, m\}$  e um conjunto de teste  $\{(\mathbf{x}_i, y_i) : i = m + 1, \dots, n\}$ , em que  $m < n$ .

2. **Treinar o modelo:** Ajustamos um modelo preditivo  $\hat{f}$  usando os dados de treinamento.
3. **Calcular a medida de não-conformidade:** Definimos uma medida de não-conformidade  $A_i$  para os exemplos do conjunto de teste. Por exemplo, no caso de regressão, podemos definir  $A_i = |y_i - \hat{f}(x_i)|$ .
4. **Construir o conjunto de predição:** Fixado um nível de significância  $\alpha \in (0, 1)$ , o conjunto de predição conforme  $\gamma_\alpha(z_1, \dots, z_n)$  para um novo exemplo  $z_{n+1}$  é, então, definido como o conjunto de todos os exemplos  $z \in Z$  tais que  $p(z) > \alpha$ :

$$\gamma_\alpha(z_1, \dots, z_n) = \{z \in Z : p(z) > \alpha\}.$$

Nessa linha, é importante destacar que a proporção dos dados alocada para teste é fundamental. Um conjunto de teste muito pequeno pode levar a intervalos de predição imprecisos, enquanto um conjunto de treinamento muito pequeno pode resultar em um modelo preditivo de baixa qualidade. Idealmente, os intervalos devem ser os mais curtos possíveis, mantendo a validade. A escolha adequada da medida de não-conformidade e a quantidade de dados para teste influenciam diretamente a eficiência.

### 3.2.5 *Quantile conformal prediction*

A *quantile conformal prediction*, por outro lado, é uma variante que usa os quantis da distribuição dos resíduos para construir intervalos de predição. Esta abordagem é especialmente útil quando se assume que os resíduos seguem uma distribuição específica, como a distribuição normal. O procedimento é descrito a seguir:

1. **Dividir os dados:** Semelhante ao *split conformal prediction*, dividimos o conjunto de dados em um conjunto de treinamento e um conjunto de validação.
2. **Treinar o modelo:** Ajustamos um modelo preditivo  $\hat{f}$  usando os dados de treinamento.
3. **Calcular os resíduos:** Calculamos os resíduos  $r_i = |y_i - \hat{f}(x_i)|$  para os exemplos no conjunto de teste.
4. **Determinar os quantis:** Fixado um nível de significância  $\alpha \in (0, 1)$ , calculamos os quantis  $q_{\alpha/2}$  e  $q_{1-\alpha/2}$  da distribuição dos resíduos.

5. **Construir o conjunto de predição:** Para um novo exemplo  $(\mathbf{x}_{n+1}, y_{n+1})$ , o intervalo de predição é dado por:

$$\gamma_\alpha(z_1, \dots, z_n, \mathbf{x}_{n+1}) = [\hat{f}(\mathbf{x}_{n+1}) - q_{\alpha/2}, \hat{f}(\mathbf{x}_{n+1}) + q_{1-\alpha/2}].$$

### 3.2.6 Exemplo Numérico

Para esse exemplo, vamos considerar o mesmo conjunto de dados hipotético utilizado na Subseção 2.3.4 e a técnica utilizada na Subseção 3.2.4, simulando  $n$  carteiras financeiras, em que para cada unidade amostral  $i$ ,  $i = 1, 2, \dots, n$ , consideramos um conjunto de 4 variáveis preditoras ( $X_1, X_2, X_3$  e  $X_4$ ) e 1 a variável resposta ( $Y$ ). Assim, utilizamos as seguintes notações:

- $Y_i$ : rendimento da carteira  $i$ ;
- $X_{i1}$ : número de ações na carteira  $i$ ;
- $X_{i2}$ : volume de transações na carteira  $i$ ;
- $X_{i3}$ : volatilidade da carteira  $i$ ;
- $X_{i4}$ : receita dolarizadas  $i$  (0 para não, 1 para sim).

Para garantir a independência dos pares  $(\mathbf{X}_i, Y_i)$  é necessário tomar setores da economia diferentes em cada carteira. Com isso, definimos o número de amostras  $n = 500$  e geramos as variáveis da seguinte forma:

- Número de ações  $X_{i1}$  da carteira  $i$ :  $X_{i1} \sim \text{Uniforme}(1, 50)$ , i.e., o número de ações varia de 1 a 50;
- Volume de transações  $X_{i2}$  da carteira  $i$ :  $X_{i2} \sim \text{Uniforme}(1000, 50000)$ , i.e., o volume de transações varia de 1000 a 50000;
- Volatilidade  $X_{i3}$  da carteira  $i$ :  $X_{i3} \sim \text{Uniforme}(0.1, 1.5)$ , i.e., a volatilidade do mercado varia de 0.1 a 1.5;
- Receita dolarizada  $X_{i4}$  da carteira  $i$ :  $X_{i4} \sim \text{Binomial}(1, 0.5)$ , i.e., a carteira é categorizada como 1 se é dolarizada e 0 caso contrário;
- Rendimento  $Y_i$  da carteira  $i$ : Calculado como uma função linear das variáveis acima mais um termo de ruído normal com média 0 e desvio padrão 10, i.e.,

$$Y_i = 0.5X_{i1} + 0.05X_{i2} - 2X_{i3} + 10X_{i4} + \epsilon_i$$

em que  $\epsilon_i \sim N(0, 10)$  é o termo de erro.

## Divisão dos Dados

Nesta seção, vamos avaliar o desempenho dos métodos de predição estudados no capítulo 2, a partir de intervalos de predição conforme. Para isso, precisamos realizar uma divisão no conjunto de dados originais. Se denotarmos por  $\mathcal{O}$  esse conjunto original, a sua divisão é realizada a partir da criação de uma partição composta por três subconjuntos: o conjunto de treinamento  $\mathcal{T}$ , o conjunto de validação  $\mathcal{V}$  e o conjunto de teste  $\mathcal{U}$ . Para o exemplo em questão, tomamos  $|\mathcal{T}| = 350$  (70% das unidades amostrais),  $|\mathcal{V}| = 75$  (15% das unidades amostrais) e  $|\mathcal{U}| = 75$  (15% das unidades amostrais).

## Resultados

Uma vez feita a divisão no conjunto de dados originais, ajustamos quatro preditores no conjunto de treinamento  $\mathcal{T}$ : árvore de decisão, floresta aleatória, *bagging* e *boosting*. Em seguida, realizamos a predição das unidades amostrais pertencentes tanto ao conjunto de validação  $\mathcal{V}$  quanto ao conjunto de teste  $\mathcal{T}$  e avaliamos o desempenho dos preditores analisando a eficiência e a validade dos intervalos de predição conformais.

Nesse contexto, a medida de não-conformidade escolhida é o resíduo, definido como a diferença absoluta entre o valor observado e o valor predito pelo modelo. Para cada modelo, calculamos a medida de não-conformidade para o conjunto de treinamento  $\mathcal{T}$  e, em seguida, utilizamos essas medidas para determinar os intervalos de predição para o conjunto de validação  $\mathcal{V}$  e para o conjunto de teste  $\mathcal{U}$ .

Os intervalos de predição são construídos de maneira a garantir que, com um nível de significância  $\alpha \in (0, 1)$  fixado, a probabilidade de o verdadeiro valor da variável resposta  $Y$  estar fora do intervalo seja no máximo  $\alpha$ . Em outras palavras, esperamos que, em média,  $(1 - \alpha) \times 100\%$  dos valores preditos estejam dentro dos intervalos calculados. Os resultados obtidos, apresentados na Tabela 3.1, mostram, para cada preditor supramencionado, a proporção de observações da variável resposta  $Y$  do conjunto de teste que pertencem ao intervalo de predição, assim como o tamanho médio desses intervalos. Observa-se que o *Boosting* apresentou a maior proporção de cobertura, juntamente com o menor tamanho médio do intervalo, o que indica que esse preditor possui uma maior precisão preditiva com baixa incerteza quando comparado aos outros preditores considerados nesta monografia.

Tabela 3.1: **Desempenho dos preditores a partir de intervalos de predição conforme.** Para cada preditor, são construídos intervalos de predição conforme com nível significância  $\alpha = 5\%$ . Em seguida, são calculados a proporção de cobertura do valor real de  $Y$  no conjunto de teste e do tamanho médio dos intervalos.

Preditor	Proporção de $Y$ (do conjunto de teste) que pertencem ao intervalo conformal	Tamanho médio do intervalo conformal
Árvore de Decisão	0.9512	20.5861
Bagging	0.9516	19.8809
Florestas Aleatórias	0.9534	19.2452
Boosting	0.9671	18.7357

Graficamente, é possível observar o seguinte:

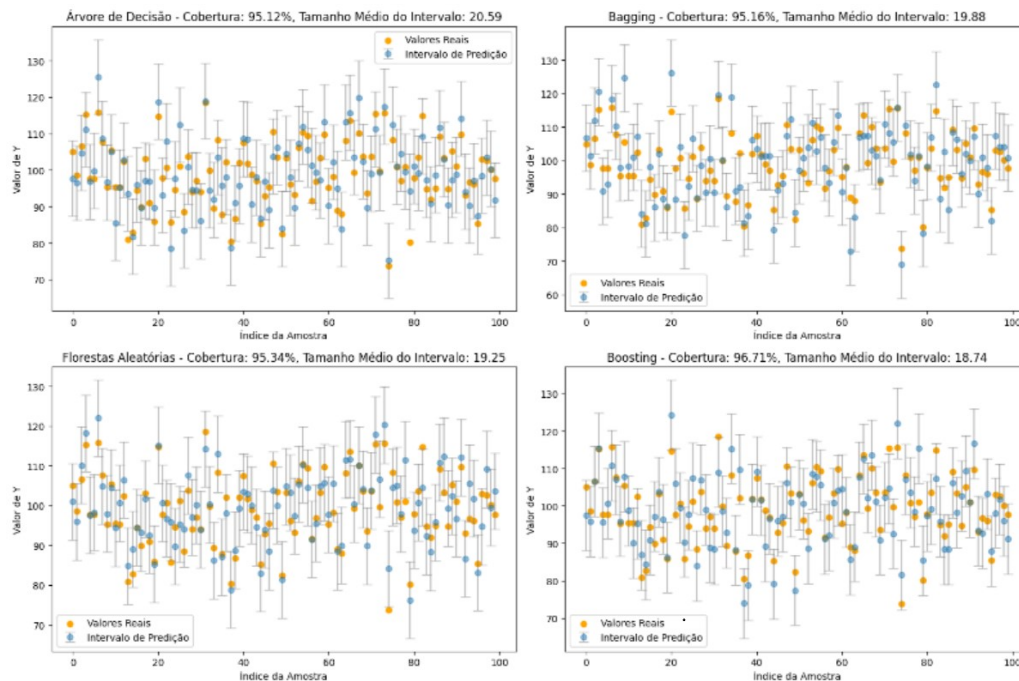


Figura 3.1: **Comparação de intervalos de predição de quatro diferentes algoritmos de aprendizado de máquina para previsão de dados sem dependência temporal.** São apresentados quatro gráficos, um para cada algoritmo, em que no eixo das abscissas temos o identificador da unidade de teste e no eixo das ordenadas os possíveis valores da variável resposta. Os pontos azuis são os valores reais observados no conjunto de teste e os pontos laranjas são os valores preditos pelos algoritmos. As retas verticais são os intervalos de predição para cada unidade de teste.

Os resultados apresentados na Tabela 3.1 e na Figura 3.1 demonstram que o modelo de

*Boosting* se destacou em relação aos demais, apresentando a maior proporção de cobertura (96,71%) do valor real de  $Y$  dentro dos intervalos conformais, junto com o menor tamanho médio do intervalo (18,7357). Isso indica que o *Boosting* não apenas é capaz de realizar previsões mais precisas, mas também de fornecer estimativas com menor incerteza. Em contrapartida, a Árvore de Decisão apresentou a menor precisão preditiva, com a menor proporção de cobertura (95,12%) e o maior intervalo médio (20,5861). Os modelos de *Bagging* e Floresta Aleatória apresentaram desempenhos intermediários, com uma ligeira vantagem para a Floresta Aleatória devido ao menor tamanho médio do intervalo conformal. Esses resultados, como esperado (e desejado), estão em linha com aqueles obtidos na Subseção 2.3.4, confirmando a consistência das análises realizadas e a vantagem do *Boosting* em termos de precisão e eficiência.

Em cenários de predição, o *overfitting* é uma condição crítica que pode comprometer a eficácia dos modelos preditivos. Esse fenômeno ocorre quando um modelo aprende detalhes e ruídos dos dados de treinamento a ponto de prejudicar sua capacidade de generalização para novos conjuntos de dados. Modelos com *overfitting* tendem a ter excelente desempenho em dados de treinamento, mas falham em manter essa performance em dados de validação ou teste, onde é essencial que o modelo demonstre eficácia. A seguir, analisamos o comportamento de *overfitting* em diferentes modelos baseados nos resultados do conjunto de validação, identificando como cada abordagem lida com o equilíbrio entre aprender a partir do conjunto de treinamento e generalizar para dados não vistos anteriormente. Isso é crucial para garantir que as previsões sejam não apenas precisas, mas também aplicáveis em situações reais, maximizando a utilidade do modelo em aplicações práticas.

- A Árvore de Decisão mostrou sinais de *overfitting*, como evidenciado pela menor proporção de cobertura e pelo maior intervalo médio de predição. Esse modelo, sendo altamente sensível às variações dos dados de treinamento, tende a criar regras muito específicas que não se generalizam bem, resultando em um desempenho inferior quando exposto a novos dados.
- O modelo de *Bagging* apresentou uma melhoria em relação à Árvore de Decisão, reduzindo o tamanho médio do intervalo de predição. No entanto, ainda há indícios de *overfitting*, dado que a proporção de cobertura não alcançou níveis ideais, sugerindo que o modelo ainda captura algumas peculiaridades dos dados de treinamento

que não são aplicáveis de forma mais ampla.

- As Florestas Aleatórias, apesar de apresentarem um desempenho melhor que o *Bagging* e a Árvore de Decisão, indicam uma leve tendência ao *overfitting*, observável pelo tamanho ainda significativo do intervalo conformal. Este modelo, por ser uma extensão do *Bagging* com uma seleção aleatória de características, ajuda a mitigar o *overfitting* mas não o elimina completamente.
- O *Boosting* destacou-se com a maior proporção de cobertura e o menor intervalo médio, indicando uma capacidade superior de generalização em comparação aos outros modelos. A técnica de *Boosting* minimiza o *overfitting* através do ajuste iterativo em erros de modelos anteriores, tornando-o mais robusto contra a sobreajuste, embora essa característica precise ser constantemente monitorada para evitar que se torne excessivamente complexa.

Neste exemplo, assumimos que os dados são permutáveis para construir os intervalos de predição. Apesar dessa suposição ser amplamente adotada, sua aplicabilidade torna-se questionável quando lidamos com dados que possuem dependência, como é o caso de séries temporais financeiras. Nesses casos, a ordem dos dados carrega informações que devem ser consideradas. Diante disso, é necessário explorar como a predição conforme pode ser adaptada para relaxar a hipótese de permutabilidade, permitindo a construção de intervalos de predição válidos e eficientes mesmo em contextos onde essa suposição não é adequada.

### 3.3 Relaxando a hipótese de permutabilidade

A predição conforme tradicionalmente assume que os dados são permutáveis, isto é, que qualquer permutação dos exemplos tem a mesma distribuição de probabilidade que a sequência original. Esta suposição pode ser inadequada para dados dependentes, como séries financeiras, em que a ordem dos dados é uma importante característica a ser considerada. Dessa forma, a predição conforme deve ser adaptada para lidar com dados dependentes, fornecendo intervalos de predição válidos e eficientes mesmo quando a suposição de permutabilidade, em razão da natureza dos dados, não é apropriada.

Nesta monografia, exploramos dois algoritmos propostos para a construção de intervalos de predição quando a suposição de permutabilidade não é atendida: o algoritmo

proposto por Chernozhukov *et al.* (2018) e o proposto por Xu e Xie (2021). Chernozhukov *et al.* (2018) propõe a construção de intervalos de predição para o contexto de séries temporais a partir da permutação de blocos de observações nos quais as dependências temporais dos dados são preservadas. Por outro lado, o algoritmo proposto por Xu e Xie, conhecido como *Ensemble Batch Prediction Interval* (EnbPI), combina a técnica de *bootstrap* com modelos de aprendizado de máquina para criar intervalos de predição sequenciais. O EnbPI é particularmente eficaz em cenários onde os dados de teste não são todos disponibilizados de uma vez, mas sim em lotes ou grupos ao longo do tempo. Ao invés de construir intervalos de predição de forma isolada para cada novo dado de teste, o EnbPI trata esses lotes de dados de forma iterativa, ou seja, ele atualiza e ajusta os intervalos de predição à medida que novos lotes de dados se tornam disponíveis.

A seguir exploramos em mais detalhes cada um dos algoritmos supramencionados.

### 3.3.1 Algoritmo proposto por Chernozhukov et al. (2018)

O algoritmo proposto por Chernozhukov *et al.* (2018) oferece uma metodologia para a construção de conjuntos de predição conforme, adaptada para dados com dependência temporal, como é comum em séries temporais ou processos estocásticos. Este algoritmo relaxa a suposição clássica de independência e distribuição idêntica (i.i.d.) dos dados, assumindo apenas a permutabilidade dentro de subconjuntos específicos ou ao longo de um processo estocástico mais geral.

Consideremos um processo estocástico  $\{Z_t = (\mathbf{X}_t, Y_t)\}_{t=1}^T$  assumindo valores em um espaço mensurável  $\mathcal{Z} \subset \mathbb{R}^p \times \mathbb{R}$ . Uma permutação  $\pi$  é definida como uma bijeção de  $\{1, \dots, T\}$  em si mesma, e o processo permutado é denotado por  $Z^\pi := \{Z_{\pi(t)}\}_{t=1}^T$ . Seja  $\Pi$  a coleção de todas as permutações dos índices  $t \in \{1, \dots, T\}$ . Note que uma dessas permutações é a permutação identidade  $\mathbb{I}$ , para a qual  $Z = Z^\mathbb{I}$ .

A medida de não-conformidade  $A(Z)$  é utilizada para calcular a não-conformidade dos dados, análoga à função  $A(z_1, \dots, z_{n+1})$  mencionada anteriormente, mas aplicada ao processo estocástico completo  $Z$ . O valor  $p$  randomizado  $p(Z)$  é então definido como a fração de permutações  $\pi$  para as quais a medida de não-conformidade do processo permutado  $A(Z^\pi)$  é maior do que ou igual à medida de não-conformidade do processo original  $A(Z)$ . Logo,  $p(Z)$  pode ser expresso como:

$$p(Z) := \frac{1}{|\Pi|} \sum_{\pi \in \Pi} \mathbf{1}\{A(Z^\pi) \geq A(Z)\},$$

em que  $|\Pi|$  é o número total de permutações consideradas, e esse valor  $p(Z)$  fornece uma estimativa da conformidade do processo  $Z$  em relação à distribuição dos valores permutados, sendo utilizado para construir os conjuntos de predição que mantêm a validade estatística.

Fixado um nível de significância  $\alpha \in (0, 1)$ , o conjunto de predição conforme  $\gamma_\alpha(Z)$  para uma nova observação  $Z_{T+1}$  é definido como o conjunto de todos os valores  $z \in \mathcal{Z}$  tais que o valor  $p(z)$  correspondente excede  $\alpha$ :

$$\gamma_\alpha(Z) = \{z \in \mathcal{Z} : p(z) > \alpha\}.$$

Este procedimento garante a validade do conjunto de predição, mesmo em cenários onde a permutabilidade é assumida apenas localmente ou sob determinadas estruturas de dependência temporal.

Para escolher o grupo de permutação de modo a preservar a dependência temporal em um processo estocástico, uma estratégia eficaz é considerar blocos de permutação tomados de forma sobreposta. Este método permite que a permutabilidade respeite as correlações entre observações temporais adjacentes, preservando a estrutura de dependência do processo estocástico.

Consideremos novamente um processo estocástico  $\{Z_t = (\mathbf{X}_t, Y_t)\}_{t=1}^T$ , assumindo valores em um espaço mensurável  $\mathcal{Z} \subset \mathbb{R}^p \times \mathbb{R}$ . Uma permutação  $\pi$  é definida como uma bijeção dos índices  $\{1, \dots, T\}$ , e o processo permutado é denotado por  $Z^\pi := \{Z_{\pi(t)}\}_{t=1}^T$ . Seja  $\Pi$  a coleção de todas as permutações dos índices  $t \in \{1, \dots, T\}$ .

Para preservar a dependência temporal, os blocos de permutação  $\Pi$  podem ser configurados de forma sobreposta. Nesta abordagem, ao invés de dividir os dados em blocos disjuntos, como é feito na permutação clássica, os blocos são formados de maneira a se sobreporem, isto é, cada bloco inclui parte das observações do bloco anterior. Essa técnica é particularmente útil para capturar dependências de curta e média duração.

Mais formalmente, considere que o conjunto  $\{1, \dots, T\}$  é dividido em  $K$  blocos de tamanho  $b$ , onde cada bloco  $j$  ( $1 \leq j \leq K$ ) contém os índices  $\{(j-1) \cdot s + 1, \dots, (j-1) \cdot s + b\}$ , com  $s$  sendo o deslocamento que determina o grau de sobreposição entre os blocos (tipicamente  $s < b$ ).

Para cada bloco  $j$ , definimos uma permutação  $\pi_j$  que atua apenas nos índices desse bloco. As permutações resultantes preservam a estrutura de dependência entre os blocos devido à sobreposição, garantindo que a informação compartilhada entre os blocos consecutivos seja mantida.

O grupo de permutação resultante  $\Pi$  é então a coleção de todas as possíveis combinações dessas permutações locais, aplicadas sobre cada bloco sobreposto. Isso permite que as permutações respeitem a dependência temporal inerente ao processo estocástico, enquanto ainda exploram a variabilidade necessária para a construção de conjuntos de predição conformes.

Dado que as permutações são definidas de modo a preservar a dependência temporal, o cálculo do valor  $p$  segue o mesmo princípio descrito anteriormente. Para cada permutação  $\pi$  em  $\Pi$ , calculamos a medida de não-conformidade  $A(Z^\pi)$  e comparamos com a medida  $A(Z)$  do processo original. O valor  $p(Z)$  é então dado pela fração de permutações  $\pi$  para as quais  $A(Z^\pi) \geq A(Z)$ :

$$p(Z) := \frac{1}{|\Pi|} \sum_{\pi \in \Pi} \mathbf{1}\{A(Z^\pi) \geq A(Z)\},$$

em que  $|\Pi|$  é o número total de permutações consideradas. Este valor é então usado para definir o conjunto de predição conforme  $\gamma_\alpha(Z)$ , que garante a cobertura adequada com base no nível de significância  $\alpha$ .

A seguir, detalhamos o funcionamento do algoritmo conforme proposto por [Cherzhanov et al. \(2018\)](#):

1. **Definição do processo estocástico:** Considere um processo estocástico  $\{Z_t = (\mathbf{X}_t, Y_t)\}_{t=1}^T$ , onde  $\mathbf{X}_t$  são as covariáveis e  $Y_t$  é a variável de resposta, com  $t \in \{1, \dots, T\}$ . Esse processo assume valores em um espaço mensurável  $\mathcal{Z} \subset \mathbb{R}^p \times \mathbb{R}$ .
2. **Definição do grupo de permutações:** Defina  $\Pi$  como o grupo de permutações que preserva a estrutura de dependência temporal dentro do processo  $Z$ . Cada permutação  $\pi$  é uma bijeção de  $\{1, \dots, T\}$  em si mesma, gerando um processo permutado  $\{Z_{\pi(t)}\}_{t=1}^T$ . A permutação identidade é denotada por  $\mathbb{I}$ , para a qual  $Z = Z^{\mathbb{I}}$ .
3. **Cálculo da medida de não-conformidade:** Calcule a medida de não-conformidade  $A(Z)$  para o processo estocástico completo  $Z$ , que quantifica quão atípico é o con-

junto de observações  $Z$  em comparação com suas versões permutadas  $Z^\pi$ .

4. **Cálculo do valor  $p$  randomizado:** Para cada permutação  $\pi \in \Pi$ , calcule a medida de não-conformidade do processo permutado  $A(Z^\pi)$ . O valor  $p(Z)$  é definido como a fração de permutações para as quais  $A(Z^\pi) \geq A(Z)$ :

$$p(Z) := \frac{1}{|\Pi|} \sum_{\pi \in \Pi} \mathbf{1}\{A(Z^\pi) \geq A(Z)\},$$

em que  $|\Pi|$  é o número total de permutações consideradas.

5. **Construção dos intervalos de predição:** Fixado um nível de significância  $\alpha \in (0, 1)$ , construa o conjunto de predição conforme  $\gamma_\alpha(Z)$  para uma nova observação  $Z_{T+1}$ . Esse conjunto é definido como todos os valores  $z \in \mathcal{Z}$  para os quais  $p(z) > \alpha$ :

$$\gamma_\alpha(Z) = \{z \in \mathcal{Z} : p(z) > \alpha\}.$$

6. **Garantia de validade:** O algoritmo garante que a probabilidade de o verdadeiro valor  $Z_{T+1}$  estar fora do intervalo de predição  $\gamma_\alpha(Z)$  não excede o nível de significância  $\alpha$ , assegurando a validade estatística do conjunto de predição em cenários com dependência temporal.

### 3.3.2 Algoritmo proposto por Xu e Xie (2021)

O algoritmo proposto por Xu e Xie (2021), conhecido como *Ensemble Batch Prediction Intervals (EnbPI)*, introduz uma metodologia para a construção de intervalos de predição em contextos de séries temporais e outros tipos de dados com dependência temporal, permitindo, assim, maior flexibilidade em aplicações práticas. O principal objetivo deste algoritmo é gerar intervalos de predição válidos e eficientes, mesmo quando os dados não seguem as suposições tradicionais de independência e distribuição idêntica (i.i.d.). O EnbPI se baseia em técnicas de *bootstrap* e em um conjunto de estimadores para gerar previsões, ajustando-se ao fato de que, em muitos cenários práticos, as observações têm dependências estruturadas.

O EnbPI é construído em torno da técnica de agregação *bootstrap*, que treina múltiplos estimadores de regressão usando subconjuntos de dados e usa esses estimadores para construir intervalos de predição. O algoritmo gera intervalos sequencialmente, ou seja, os

intervalos de predição são recalculados à medida que novos dados chegam. O algoritmo EnbPI funciona da seguinte maneira:

1. O algoritmo se inicia com um conjunto de dados de treinamento  $\{(\mathbf{x}_i, y_i)\}_{i=1}^T$ , em que  $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^d$  é a  $i$ -ésima observação de um vetor de covariáveis  $\mathbf{X}$  e  $y_i \in \mathcal{Y} \subset \mathbb{R}$  é a  $i$ -ésima observação de uma variável resposta  $Y$ .
2. Consideramos  $B$  diferentes subconjuntos de dados, gerados por *bootstrap*, ou seja, amostras com reposição dos dados de treinamento originais. Nesse sentido, definimos  $S_b = \{i_1, \dots, i_T\}$  como sendo as unidades amostrais selecionadas com reposição do conjunto de índices de treinamento original  $\{1, \dots, T\}$ .
3. Em cada uma das  $B$  amostras *bootstrap*, treinamos um modelo preditivo  $\hat{f}^{(b)}$ , em que  $b = 1, 2, \dots, B$ .
4. Para cada ponto  $(\mathbf{x}_i, y_i)$  pertencente ao conjunto de treinamento,  $i = 1, \dots, T$ , denotamos por  $\hat{f}_{-i}$  o modelo treinado sobre o conjunto de dados que exclui a observação  $(\mathbf{x}_i, y_i)$  e inclui todos os outros  $T - 1$  pontos de treinamento. Nesse contexto, para cada  $b = 1, \dots, B$ , escrevemos  $\hat{f}_{-i}^{(b)}$  para denotar o estimador de regressão para a  $b$ -ésima amostra *bootstrap* sem considerar a observação  $(\mathbf{x}_i, y_i)$ , i.e.,  $i \notin S_b$ . Portanto, para cada  $i = 1, \dots, T$ , calcule o valor predito da  $i$ -ésima observação de treino como sendo, por exemplo,

$$\hat{f}_{-i}(x_i) := \frac{1}{B} \sum_{b=1}^B \hat{f}_{-i}^{(b)}(x_i).$$

5. Para cada  $t = 1, \dots, T$ , calculamos os resíduos

$$\hat{\epsilon}_t := |y_t - \hat{f}_{-t}(x_t)|$$

e os armazenamos no conjunto  $\epsilon$ .

6. Considere um nível de significância  $\alpha \in (0, 1)$  e o conjunto de dados de teste  $\{(\mathbf{x}_t, y_t)\}_{t=T+1}^{T+T_1}$ , em que  $y_t$  é revelado apenas após um lote de  $s \geq 1$  intervalos de predição. Para cada  $t = T + 1, \dots, T + T_1$ , defina  $\hat{f}_{-t}(x_t)$  como sendo o quantil de  $(1 - \alpha)\%$  de  $\{\hat{f}_{-i}(x_t)\}_{i=1}^T$  e  $w_t$  como sendo o quantil de  $(1 - \alpha)\%$  do conjunto  $\epsilon$ . Definimos o intervalo de predição  $Y_t$  como sendo

$$\gamma_{T,t}^\alpha(x_t) := [\hat{f}_{-t}(x_t) - w_t, \hat{f}_{-t}(x_t) + w_t].$$

Se  $t - T$  é divisível por  $s$ , então calculamos

$$\hat{\epsilon}_j = |y_j - \hat{f}_{-j}(x_t)|,$$

para todo  $j = t - s, \dots, t - 1$  e, em seguida, atualizamos o conjunto de resíduos da seguinte maneira

$$\epsilon \leftarrow (\epsilon - \{\hat{\epsilon}_1, \dots, \hat{\epsilon}_s\}) \cup \{\hat{\epsilon}_{t-s}, \dots, \hat{\epsilon}_{t-1}\}.$$

e resetamos os índices de  $\epsilon$ .

O algoritmo garante que, ao longo de várias previsões, a probabilidade de o valor real  $y_t$  estar fora do intervalo de predição será inferior ao nível de significância definido, por exemplo, 5%.

### 3.3.3 Algoritmo proposto por Gibbs e Cándes (2021)

O algoritmo proposto por [Gibbs e Candes \(2021\)](#) é uma extensão do algoritmo proposto por [Chernozhukov et al. \(2018\)](#), projetada para lidar com problemas de *deslocamento de distribuição* (distribution shift), em que a distribuição marginal dos dados muda ao longo do tempo. Esse tipo de problema ocorre com frequência em contextos de séries temporais ou quando os dados de diferentes períodos têm distribuições diferentes. Nesse sentido, utilizamos as seguintes definições:

Seja o processo estocástico  $\{(X_t, Y_t)\}_{t=1}^T$ , onde  $X_t$  são as covariáveis no tempo  $t$  e  $Y_t$  é a variável de resposta. O processo assume valores no espaço mensurável  $Z \subset \mathbb{R}^p \times \mathbb{R}$ .

$$Z_t = (X_t, Y_t) \quad \text{para } t = 1, \dots, T$$

Por sua vez, a medida de não-conformidade  $A(X_t, Y_t)$  quantifica a atipicidade de uma observação  $(X_t, Y_t)$  em relação às observações anteriores, usando uma função de conformidade adaptada. Para qualquer valor  $y \in \mathbb{R}$ , a medida de conformidade é dada por:

$$S(X_t, y) = \mathcal{C}(y | X_t)$$

em que  $\mathcal{C}(y | X_t)$  é uma função que avalia a plausibilidade de  $y$  dado o conjunto de covariáveis  $X_t$ . Em outras palavras,  $S(X_t, y)$  representa uma medida do quão consistente  $y$  é com as observações anteriores de  $Y_t$  quando condicionado pelas covariáveis  $X_t$ .

Devido ao deslocamento de distribuição nos dados ao longo do tempo, a função de conformidade  $S_t$  é recalibrada a cada novo passo  $t$  com base em um conjunto de calibração  $D_{\text{cal}}$ . O nível de cobertura  $\alpha_t$  é ajustado de forma adaptativa ao longo do tempo.

A atualização do parâmetro  $\alpha_t$  é dada por:

$$\alpha_{t+1} = \alpha_t + \gamma \cdot (\alpha - \text{err}_t)$$

em que  $\gamma$  é a taxa de aprendizado que controla a velocidade de adaptação e  $\text{err}_t$  é o erro no passo  $t$ , que é a diferença entre a previsão do modelo e o valor real observado para a variável de resposta no tempo  $t$ .

O conjunto de predição conforme  $\hat{C}_t(\alpha_t)$  para a próxima observação  $Y_{t+1}$  é dado por:

$$\hat{C}_t(\alpha_t) = \{y : S(X_t, y) \leq Q_t(1 - \alpha_t)\}$$

em que  $Q_t(1 - \alpha_t)$  é o quantil da função de conformidade.

O algoritmo garante que o conjunto de predição  $\hat{C}_t(\alpha_t)$  tem cobertura marginal de  $1 - \alpha$ . Ou seja, a probabilidade de que o valor verdadeiro  $Y_t$  não caia dentro do conjunto de predição é no máximo  $\alpha$ .

$$\mathbb{P}(Y_t \notin \hat{C}_t(\alpha_t)) \leq \alpha$$

Dito isso, podemos descrever o algoritmo da seguinte forma:

1. O algoritmo começa com um conjunto de dados de treinamento  $\{(\mathbf{x}_i, y_i)\}_{i=1}^T$ , onde  $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^d$  são as covariáveis e  $y_i \in \mathbb{Y} \subset \mathbb{R}$  são as observações da variável resposta.
2. O algoritmo gera permutações  $\pi$  do conjunto de dados de forma que a permutação preserve a estrutura temporal e a dependência entre as observações.
3. Para cada permutação  $\pi$ , calcula-se uma medida de não-conformidade  $A(\mathbf{Z}_\pi)$  para cada conjunto de dados permutado, onde  $\mathbf{Z}_\pi$  representa as observações permutadas.
4. Para cada ponto de dados  $(\mathbf{x}_i, y_i)$ , calcula-se o valor de  $p(\mathbf{Z})$ , que representa a fração de permutações em que a medida de não-conformidade  $A(\mathbf{Z}_\pi)$  é maior ou igual à medida de não-conformidade da observação original.
5. Para um nível de significância  $\alpha \in (0, 1)$ , define-se o conjunto de predição conforme

$\gamma_\alpha(\mathbf{Z})$  para uma nova observação  $\mathbf{z}_{T+1}$  como o conjunto de todas as observações  $\mathbf{z}$  tais que  $p(\mathbf{z}) > \alpha$ .

6. O algoritmo é adaptativo e recalcula os intervalos de predição conforme novos dados chegam, ajustando-se automaticamente às mudanças nas distribuições dos dados ao longo do tempo. O conjunto de predição garantirá que a probabilidade de o verdadeiro valor  $\mathbf{z}_{T+1}$  não estar dentro do intervalo seja no máximo  $\alpha$ , garantindo validade estatística mesmo com deslocamento de distribuição.

### 3.3.4 Exemplo Numérico

Nesta subseção, assim como na Subseção 2.4.3, simulamos a mesma série temporal financeira com  $n = 500$  observações, em que cada observação representa um valor diário de variáveis do mercado. Definimos uma variável resposta ( $Y$ ) e quatro variáveis preditoras ( $X_1, X_2, X_3, X_4$ ) que influenciam o preço das ações. Assim, utilizamos as seguintes notações:

- $Y_t$ : rendimento da ação no tempo  $t$ ;
- $X_{t1}$ : volume de negociações no tempo  $t$  (simulado);
- $X_{t2}$ : Ibovespa no tempo  $t$  (simulado);
- $X_{t3}$ : taxa de juros no tempo  $t$  (simulado);
- $X_{t4}$ : ruído irrelevante no tempo  $t$ .

Considerando a dependência dos pares  $(\mathbf{X}_i, Y_i)$ , é necessário tomar amostras ao longo do tempo de maneira sequencial. Com isso, definimos o número de amostras  $n = 500$  e geramos as variáveis da seguinte forma:

- Volume de negociações  $X_{t1}$  no tempo  $t$ :  $X_{t1} \sim \text{Poisson}(1000)$ , i.e., o volume de negociações é modelado por uma distribuição de Poisson com média 1000;
- Ibovespa  $X_{t2}$  no tempo  $t$ :  $X_{t2} = \sin\left(\frac{20\pi i}{n}\right) + N(0, 0.1)$ , i.e., o índice de mercado é uma função senoidal com ruído normal adicionado;
- Taxa de juros  $X_{t3}$  no tempo  $t$ :  $X_{t3} = \frac{5i}{n} + N(0, 0.1)$ , i.e., a taxa de juros aumenta linearmente com ruído normal adicionado;

- Ruído irrelevante  $X_{t4}$  no tempo  $t$ :  $X_{t4} \sim N(5, 2)$ , i.e., o ruído irrelevante é modelado por uma distribuição normal com média 5 e desvio padrão 2;
- Preço da ação  $Y_t$  no tempo  $t$ : Calculado como uma função das variáveis acima mais um termo de ruído normal com média 0 e desvio padrão 0.5, i.e.,

$$Y_t = Y_{t-1} + 0.5 \log(X_{t1} + 1) + 2X_{t2} - 0.3X_{t3} + \epsilon_t$$

em que  $\epsilon_t \sim N(0, 0.5)$  é o termo de erro.

## Divisão dos Dados

Para avaliar o desempenho dos métodos de predição estudados ao longo desta seção, precisamos realizar uma divisão no conjunto de dados originais. Se denotarmos por  $\mathcal{O}$  esse conjunto original, a sua divisão é realizada a partir da criação de uma partição composta por três subconjuntos: o conjunto de treinamento  $\mathcal{T}$ , o conjunto de validação  $\mathcal{V}$  e o conjunto de teste  $\mathcal{U}$ . Para o exemplo em questão, tomamos  $|\mathcal{T}| = 350$  (70% das unidades amostrais),  $|\mathcal{V}| = 75$  (15% das unidades amostrais) e  $|\mathcal{U}| = 75$  (15% das unidades amostrais).

## Resultados

Os métodos conformais foram empregados para construir intervalos de predição para os conjuntos de validação e teste, utilizando como medida de não-conformidade os resíduos absolutos entre os valores observados e preditos, considerando os modelos ajustados no conjunto de treinamento. Esses intervalos foram calibrados para um nível de significância  $\alpha = 5\%$ , o que implica que, em média, 95% das predições devem estar contidas dentro dos intervalos calculados.

Tabela 3.2: **Desempenho dos preditores a partir de intervalos de predição conforme (Chernozhukov)**. Para cada preditor, são construídos intervalos de predição conforme com nível de significância  $\alpha = 5\%$ . Em seguida, são calculados a proporção de cobertura do valor real de  $Y$  no conjunto de teste e o tamanho médio dos intervalos.

Preditor	Proporção de $Y$ (do conjunto de teste)	Tamanho médio do intervalo conformal
Árvore de Decisão	0.9503	13.9643
Bagging	0.9512	13.0732
Florestas Aleatórias	0.9551	10.1841
Boosting	0.9660	9.4453

Os resultados obtidos, apresentados na Tabela 3.2, mostram que o modelo de *Boosting* foi o mais eficaz, apresentando uma maior proporção de cobertura (96,6%) em relação aos outros métodos e o menor tamanho médio de intervalo (9,9423). Isso indica que o *Boosting* não apenas fornece previsões precisas, mas também reduz a incerteza associada a essas previsões. Por outro lado, o modelo de Árvore de Decisão apresentou o maior tamanho médio de intervalo (13,7643), o que reflete uma menor precisão e maior incerteza preditiva. Os métodos de *Bagging* e Florestas Aleatórias apresentaram desempenhos intermediários, com uma ligeira vantagem para o *Florestas Aleatórias*, que apresentou intervalos menores (10,7841).

Tabela 3.3: **Desempenho dos preditores a partir de intervalos de predição com EnbPI (Xu e Xie)**. Para cada preditor, são construídos intervalos de predição com nível de significância  $\alpha = 5\%$ . Em seguida, são calculados a proporção de cobertura do valor real de  $Y$  no conjunto de teste e o tamanho médio dos intervalos.

Preditor	Proporção de $Y$ (do conjunto de teste)	Tamanho médio do intervalo EnbPI
Árvore de Decisão	0.9503	15.7643
Bagging	0.9512	13.9732
Florestas Aleatórias	0.9551	10.3428
Boosting	0.9660	9.2197

Na Tabela 3.3, os resultados obtidos utilizando o método *EnbPI* (Xu e Xie) corroboram a superioridade do modelo de *Boosting*, que apresentou o menor tamanho médio do intervalo (9,9423), refletindo uma menor incerteza em suas previsões. Além disso, o

*Boosting* obteve a maior proporção de cobertura (96,6%), indicando sua capacidade em capturar os valores reais dentro dos intervalos.

Por outro lado, o modelo de Árvore de Decisão apresentou o maior intervalo médio (13,7643), o que reflete uma maior incerteza preditiva. O *Bagging* e *Florestas Aleatórias* demonstraram desempenhos intermediários, sendo o *Florestas Aleatórias* ligeiramente superior devido aos intervalos mais estreitos (10,7841).

Os dois métodos analisados (*Conformal Prediction* e *EnbPI*) fornecem informações complementares sobre o desempenho dos preditores. Em ambos os cenários, o modelo de *Boosting* se destacou como a abordagem mais precisa, apresentando intervalos de predição mais estreitos e uma cobertura próxima ao nível desejado de 95%. Esse comportamento reflete a capacidade do *Boosting* em lidar com incertezas, especialmente em cenários onde o tamanho do intervalo é crítico para a tomada de decisão.

Por outro lado, os métodos de *Bagging* e *Florestas Aleatórias* também demonstraram boa precisão, com o *Florestas Aleatórias* apresentando uma melhor relação entre cobertura e tamanho do intervalo. Esses resultados destacam a importância de escolher o modelo preditivo mais adequado ao contexto específico, considerando tanto a necessidade de precisão quanto a tolerância à incerteza.

Tabela 3.4: **Desempenho dos preditores utilizando o algoritmo proposto por Gibbs e Cándes (2021)**. Para cada preditor, são construídos intervalos de predição com nível de significância  $\alpha = 5\%$ . Em seguida, são calculadas a proporção de cobertura do valor real de  $Y$  no conjunto de teste e o tamanho médio dos intervalos.

Preditor	Proporção de $Y$ (do conjunto de teste)	Tamanho médio do intervalo (Gibbs e Cándes)
Árvore de Decisão	0.956	14.5637
Bagging	0.9511	12.4568
Florestas Aleatórias	0.9592	9.8251
Boosting	0.9505	8.2073

Na Tabela 3.4, os resultados obtidos utilizando o algoritmo proposto por [Gibbs e Cándes \(2021\)](#) indicam um desempenho comparável ao dos métodos conformais e ao EnbPI, mas com algumas diferenças. Como nos outros métodos, o modelo de *Boosting* continua a se destacar, apresentando o menor tamanho médio de intervalo (8,2073), o que reflete uma menor incerteza associada às suas predições, muito semelhante ao comportamento observado nas Tabelas 3.2 e 3.3, em que o *Boosting* também se mostrou o mais preciso,

com intervalos menores e uma cobertura elevada. No caso do método proposto por [Chernozhukov et al. \(2018\)](#), a proporção de cobertura do *Boosting* foi de 96,60%, enquanto no método *EnbPI* a cobertura foi de 96,60%, confirmando sua consistência em termos de precisão e confiança nas predições.

No entanto, ao comparar com os resultados dos métodos conformais e *EnbPI*, podemos observar que o modelo de *Boosting* no algoritmo de [Gibbs e Candès](#) apresentou um intervalo ainda mais estreito (8,2073) em comparação ao intervalo observado no *EnbPI* (9,2197) e no método proposto por [Chernozhukov et al. \(9,4453\)](#). Isso indica que o algoritmo proposto por Gibbs e Cándes consegue reduzir ainda mais a incerteza das predições, mantendo a capacidade de capturar os valores reais de  $Y$  dentro de um intervalo de confiança.

Em relação aos outros modelos, a *Árvore de Decisão* continuou a apresentar a maior incerteza, com o maior tamanho de intervalo (14,5637), o que também foi observado nos métodos anteriores. O modelo de *Bagging* manteve uma posição intermediária em termos de cobertura (95,05%) e intervalo (12,4568) nos resultados de Gibbs e Cándes, alinhando-se com o desempenho observado nas outras tabelas.

O modelo de *Florestas Aleatórias* obteve um bom desempenho, com um tamanho de intervalo médio de 9,8251, que foi o segundo melhor entre os modelos analisados no algoritmo de Gibbs e Cándes. Esse comportamento é consistente com os resultados do método *EnbPI* e do proposto por [Chernozhukov et al.](#), onde as *Florestas Aleatórias* também se destacaram em relação ao *Bagging*, mas não tanto quanto o *Boosting*.

Sendo assim, esses resultados sugerem a ideia de que o algoritmo de [Gibbs e Candès](#) é eficaz para reduzir a incerteza nas predições, especialmente para o modelo *Boosting*, que demonstrou ser o mais preciso e confiável em todos os cenários analisados, com intervalos menores e uma cobertura próxima ao nível desejado de 95%. Logo, a comparação com os métodos previamente exemplificados destaca a eficiência do *Boosting* e a importância de escolher o modelo preditivo mais adequado ao contexto específico, levando em consideração tanto a precisão quanto a tolerância à incerteza.

### 3.4 Intervalos de predição paramétricos

Nesta seção, discutiremos os intervalos de predição paramétricos, uma abordagem tradicional e utilizada em estatística e aprendizado de máquina para quantificar a in-

certeza nas previsões, independentemente da presença ou ausência de dependências nas observações. Diferente dos intervalos de predição não-paramétricos, que não fazem suposições fortes sobre a distribuição dos dados, os intervalos de predição paramétricos assumem que os dados seguem uma distribuição específica como, por exemplo, a distribuição normal.

Em geral, os intervalos de predição paramétricos são construídos com base em modelos estatísticos em que se assume que os resíduos do modelo seguem uma distribuição específica. Isso implica que a incerteza na previsão é diretamente relacionada a essa distribuição, permitindo a construção de intervalos de predição através do cálculo dos quantis da distribuição assumida.

Por exemplo, considere um modelo de regressão linear simples, em que a variável resposta  $Y$  é modelada como uma função linear de uma ou mais variáveis preditoras  $\mathbf{X}$ , acrescida de um termo de erro  $\epsilon$  que captura a variação não explicada pelo modelo. Em notação matemática, temos:

$$Y = \mathbf{X}\boldsymbol{\beta} + \epsilon,$$

em que  $\boldsymbol{\beta}$  é um vetor de coeficientes que descreve a relação entre as variáveis preditoras e a variável de resposta, e  $\epsilon$  é o termo de erro que assume uma distribuição normal com média zero e variância  $\sigma^2$ , i.e.,  $\epsilon \sim N(0, \sigma^2)$ . Assim, é possível resumir o problema do seguinte modo: dado um novo conjunto de variáveis preditoras  $\mathbf{X}_{n+1}$ , o objetivo é prever a variável resposta correspondente  $Y_{n+1}$  e construir um intervalo de predição que contenha  $Y_{n+1}$  com uma probabilidade predefinida, denominada nível de confiança  $1 - \alpha$ , em que  $\alpha \in (0, 1)$ .

No contexto de regressão linear, por exemplo, estimamos o valor esperado de  $Y_{n+1}$  com base no modelo ajustado, que é dado por  $\hat{Y}_{n+1} = \mathbf{X}_{n+1}\hat{\boldsymbol{\beta}}$ , em que  $\hat{\boldsymbol{\beta}}$  é a estimativa dos coeficientes do modelo ajustado aos dados de treinamento. Em seguida, estimamos a variância do erro de predição, que é composta pela variância do erro do modelo ( $\sigma^2$ ) e a incerteza na estimativa dos coeficientes ( $\boldsymbol{\beta}$ ). A variância da predição é dada por:

$$\text{Var}(\hat{Y}_{n+1}) = \sigma^2 \left( 1 + \mathbf{X}_{n+1}^\top (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}_{n+1} \right),$$

em que  $\mathbf{X}^\top \mathbf{X}$  é a matriz de covariância das variáveis preditoras no conjunto de treinamento. Com a variância da predição estimada, calculamos os limites inferior e superior do intervalo de predição usando o quantil da distribuição normal correspondente ao nível

de significância  $\alpha$ . O intervalo de predição para  $Y_{n+1}$  é dado por:

$$\gamma_\alpha(\mathbf{X}_{n+1}) = \hat{Y}_{n+1} \pm z_{\alpha/2} \sqrt{\text{Var}(\hat{Y}_{n+1})},$$

em que  $z_{\alpha/2}$  é o quantil da distribuição normal padrão correspondente ao nível de significância  $\alpha/2$ .

A validade dos intervalos de predição paramétricos depende da adequação das suposições do modelo, especialmente da suposição de que os erros seguem uma determinada distribuição. Se essa suposição for válida, o intervalo de predição cobrirá o valor verdadeiro de  $Y_{n+1}$  com o nível de confiança especificado  $1 - \alpha$ .

Assim, a eficiência dos intervalos de predição paramétricos, assim como nos métodos não-paramétricos, refere-se ao comprimento (ou volume, em espaços multidimensionais) do intervalo. Intervalos mais curtos são geralmente preferíveis, pois indicam maior precisão na previsão. No entanto, a eficiência é diretamente influenciada pela variância do erro de predição: maiores incertezas resultam em intervalos mais largos.

No contexto de intervalos de predição para séries temporais, por outro lado, temos o algoritmo das inovações (veja, por exemplo, [Brockwell e Davis, 2002](#)), que é uma técnica utilizada para prever valores futuros em séries temporais, especialmente aquelas que seguem um padrão estatístico específico, como os processos ARIMA (*Auto Regressive Integrated Moving Average*). O algoritmo é particularmente útil em situações onde os dados que estamos tentando prever não são necessariamente estacionários, ou seja, onde a média e a variância dos dados podem mudar ao longo do tempo.

A característica central desse algoritmo é que ele permite realizar previsões de maneira recursiva, ou seja, cada nova previsão é baseada nas previsões anteriores. Para simplificar o processo, em vez de aplicar o algoritmo diretamente à série original, é feita uma transformação da série. Essa transformação facilita o cálculo das previsões e permite que o algoritmo funcione de forma mais eficiente. Nesse sentido, a série transformada, chamada  $W_t$ , é usada para calcular as autocovariâncias, que medem a relação entre os valores passados e presentes da série.

Com isso, o algoritmo calcula então as previsões de  $W_t$  com base em valores anteriores da série. Essas previsões são ajustadas de acordo com o comportamento esperado da série, levando em conta tanto os componentes autorregressivos, que dependem dos valores passados da própria série, quanto os componentes de média móvel, que dependem dos

erros passados.

Além disso, o algoritmo leva em consideração a heterocedasticidade dos dados ao longo do tempo. Isso significa que, ao prever um novo valor, o algoritmo não apenas estima o valor futuro, mas também fornece uma medida de incerteza associada a essa previsão. Essa incerteza é expressa em termos de um erro, que é essencialmente uma medida de quão longe a previsão pode estar do valor real.

Nesse sentido, o algoritmo garante que as previsões geradas sejam as melhores possíveis dentro do contexto do modelo ARIMA, no sentido de minimizar o erro quadrático médio das previsões. Ele ajusta continuamente os coeficientes do modelo com base nos novos dados observados, refinando as previsões à medida que mais informações são incorporadas. Dessa forma, o algoritmo das inovações oferece uma abordagem eficiente para a previsão de séries temporais complexas, fornecendo não apenas estimativas pontuais, mas também uma avaliação rigorosa da incerteza associada a essas estimativas.

### 3.4.1 Exemplo numérico

Nesta subseção, simulamos a mesma série temporal financeira com  $n = 500$  observações, em que cada observação representa um valor diário de variáveis do mercado. Definimos uma variável resposta ( $Y$ ) e quatro variáveis preditoras ( $X_1, X_2, X_3, X_4$ ) que influenciam o preço das ações. Assim, utilizamos as seguintes notações:

- $Y_t$ : rendimento da ação no tempo  $t$ ;
- $X_{t1}$ : volume de negociações no tempo  $t$  (simulado);
- $X_{t2}$ : Ibovespa no tempo  $t$  (simulado);
- $X_{t3}$ : taxa de juros no tempo  $t$  (simulado);
- $X_{t4}$ : ruído irrelevante no tempo  $t$ .

Considerando a dependência dos pares  $(\mathbf{X}_t, Y_t)$ , é necessário tomar amostras ao longo do tempo de maneira sequencial. Com isso, definimos o número de amostras  $n = 500$  e geramos as variáveis da seguinte forma: Considerando a dependência dos pares  $(\mathbf{X}_i, Y_i)$ , é necessário tomar amostras ao longo do tempo de maneira sequencial. Com isso, definimos o número de amostras  $n = 500$  e geramos as variáveis da seguinte forma:

- Volume de negociações  $X_{t1}$  no tempo  $t$ :  $X_{t1} \sim \text{Poisson}(1000)$ , i.e., o volume de negociações é modelado por uma distribuição de Poisson com média 1000;
- Ibovespa  $X_{t2}$  no tempo  $t$ :  $X_{t2} = \sin\left(\frac{20\pi i}{n}\right) + N(0, 0.1)$ , i.e., o índice de mercado é uma função senoidal com ruído normal adicionado;
- Taxa de juros  $X_{t3}$  no tempo  $t$ :  $X_{t3} = \frac{5i}{n} + N(0, 0.1)$ , i.e., a taxa de juros aumenta linearmente com ruído normal adicionado;
- Ruído irrelevante  $X_{t4}$  no tempo  $t$ :  $X_{t4} \sim N(5, 2)$ , i.e., o ruído irrelevante é modelado por uma distribuição normal com média 5 e desvio padrão 2;
- Preço da ação  $Y_t$  no tempo  $t$ : Calculado como uma função das variáveis acima mais um termo de ruído normal com média 0 e desvio padrão 0.5, i.e.,

$$Y_t = Y_{t-1} + 0.5 \log(X_{t1} + 1) + 2X_{t2} - 0.3X_{t3} + \epsilon_t$$

em que  $\epsilon_t \sim N(0, 0.5)$  é o termo de erro.

## Divisão dos Dados

Para avaliar o desempenho dos métodos de predição estudados ao longo desta seção, precisamos realizar uma divisão no conjunto de dados originais. Se denotarmos por  $\mathcal{O}$  esse conjunto original, a sua divisão é realizada a partir da criação de uma partição composta por três subconjuntos: o conjunto de treinamento  $\mathcal{T}$ , o conjunto de validação  $\mathcal{V}$  e o conjunto de teste  $\mathcal{U}$ . Para o exemplo em questão, tomamos  $|\mathcal{T}| = 350$  (70% das unidades amostrais),  $|\mathcal{V}| = 75$  (15% das unidades amostrais) e  $|\mathcal{U}| = 75$  (15% das unidades amostrais).

## Resultados

Os modelos de séries temporais ajustados foram aplicados ao conjunto de teste  $\mathcal{U}$  para avaliar o desempenho preditivo e a precisão dos intervalos de predição paramétricos. A tabela a seguir apresenta a proporção de cobertura dos valores reais de  $Y$  no conjunto de teste, bem como o tamanho médio dos intervalos de predição para cada preditor. Esperamos que, em média, 95% dos valores reais de  $Y$  estejam dentro dos intervalos de predição calculados.

Tabela 3.5: **Desempenho dos preditores a partir de intervalos de predição paramétricos.** Para cada preditor, são construídos intervalos de predição com nível de significância  $\alpha = 5\%$ . Em seguida, são calculados a proporção de cobertura do valor real de  $Y$  no conjunto de teste e o tamanho médio dos intervalos.

Preditor	Proporção de $Y$ (do conjunto de teste)	Tamanho médio do intervalo paramétrico
AR	0.9503	13.7643
MA	0.9512	13.4732
ARMA	0.9551	10.7841
ARIMA	0.9590	9.9423

Os resultados apresentados na Tabela 3.5 avaliam o desempenho de diferentes preditores baseados em modelos de séries temporais: AR (Autoregressive), MA (Moving Average), ARMA (Autoregressive Moving Average) e ARIMA (Autoregressive Integrated Moving Average). Os intervalos de predição foram construídos com nível de significância  $\alpha = 5\%$ , correspondendo a um intervalo de confiança de  $1 - \alpha = 95\%$ . Para cada preditor, foram calculados a proporção de cobertura dos valores reais de  $Y$  no conjunto de teste e o tamanho médio dos intervalos de predição.

Para o modelo AR, a proporção de cobertura dos valores reais de  $Y$  foi de 0,9503, indicando que 95,03% dos valores reais foram capturados pelos intervalos de predição. O tamanho médio dos intervalos foi de 13,7643.

No caso do modelo MA, a proporção de cobertura aumentou ligeiramente para 0,9512, com uma redução no tamanho médio dos intervalos para 13,4732. Esse resultado sugere uma leve melhora na precisão dos intervalos ao utilizar esse preditor.

O modelo ARMA apresentou uma proporção de cobertura de 0,9551, que é superior aos modelos AR e MA, capturando 95,51% dos valores reais no conjunto de teste. O tamanho médio dos intervalos também reduziu significativamente, atingindo 10,7841, evidenciando maior eficiência preditiva.

Por fim, o modelo ARIMA obteve os melhores resultados dentre os preditores avaliados, com uma proporção de cobertura de 0,9590, correspondendo a 95,90% dos valores reais incluídos nos intervalos de predição. Além disso, o ARIMA apresentou o menor tamanho médio dos intervalos, igual a 9,9423, demonstrando sua superioridade em termos de precisão e eficiência no cálculo dos intervalos.

### 3.5 Discussão

Nesta seção, discutimos os resultados apresentados na Subseção 3.3.4, que utiliza métodos conformais e *EnbPI* para a construção de intervalos de predição, e na Subseção 3.4.1, que avalia o desempenho de preditores paramétricos aplicados a séries temporais financeiras. A comparação é realizada em termos de proporção de cobertura dos valores reais de  $Y$  e do tamanho médio dos intervalos de predição, com o objetivo de analisar a consistência dos resultados com as conclusões previstas pela literatura.

Os resultados obtidos em ambas as subseções estão de acordo com as expectativas teóricas de cobertura, atingindo ou superando a meta de 95% de valores reais contidos nos intervalos de predição. No contexto dos métodos conformais, o *Boosting* demonstrou desempenho superior, com 96,6% de cobertura (Tabela 3.2), enquanto o método paramétrico ARIMA apresentou desempenho comparável, com 95,9% de cobertura (Tabela 3.5).

O desempenho do *Boosting* é notável, considerando que os métodos conformais não dependem de suposições paramétricas fortes. Por outro lado, a eficácia do ARIMA reflete sua capacidade de capturar padrões temporais e modelar dependências estruturais nos dados.

O tamanho médio dos intervalos de predição é uma métrica importante para avaliar a eficiência dos modelos. O *Boosting*, no contexto dos métodos conformais, apresentou o menor intervalo médio (9,4453), próximo ao ARIMA (9,9423). Essa proximidade no desempenho sugere que, mesmo com abordagens conceitualmente distintas, ambos os métodos são capazes de fornecer predições confiáveis com alta precisão.

Os outros modelos apresentaram desempenhos intermediários. Nos métodos conformais, o *Bagging* (13,0732) e *Florestas Aleatórias* (10,1841) demonstraram maior eficiência que a *Árvore de Decisão* (13,9643), mas ainda foram superados pelo *Boosting*. Já entre os modelos paramétricos, ARMA (10,7841) e MA (13,4732) apresentaram intervalos mais amplos, refletindo maior incerteza associada às predições.

A escolha entre métodos conformais e paramétricos depende das características específicas do problema. Em cenários onde a estrutura temporal é bem definida e as suposições paramétricas são válidas, como em séries estacionárias, o ARIMA continua sendo uma escolha eficiente. Entretanto, para dados mais complexos ou não estacionários, os métodos conformais, particularmente com *Boosting*, oferecem um desempenho mais com-

petitivo.

Com isso, concluímos que tanto os métodos conformais quanto os paramétricos possuem seus méritos e são adequados para diferentes contextos. O *Boosting*, dentro do contexto conformal, destacou-se pela combinação de alta cobertura e intervalos estreitos, evidenciando sua adaptação em dados temporalmente dependentes. Por outro lado, o ARIMA reafirma sua posição como um modelo de referência para séries temporais financeiras, produzindo intervalos precisos em condições idealmente paramétricas.



# Capítulo 4

## Aplicação dos intervalos de predição no mercado de ações

A quantificação da incerteza no mercado financeiro colabora com a tomada de decisões estratégicas e a gestão de riscos. Em um ambiente caracterizado por volatilidade e eventos inesperados, a correta mensuração da incerteza permite que investidores, gestores de portfólio e analistas desenvolvam estratégias mais robustas para precificação de ativos, alocação de recursos e controle de exposição a riscos. Neste capítulo, empregamos tanto os intervalos de predição conformais quanto os paramétricos para orientar a alocação de um montante de capital entre diferentes ações previamente selecionadas em uma carteira. Em seguida, avaliamos o desempenho de cada estratégia, analisando os retornos esperados e os riscos associados a cada configuração de alocação.

### 4.1 Mercado Financeiro

O mercado financeiro pode ser entendido como o ambiente em que ocorre a negociação de diversos ativos financeiros, tais como ações, títulos de dívida, moedas, derivativos, dentre outros. Esse ambiente se desenvolveu ao longo dos séculos, sendo influenciado por fatores econômicos, sociais e políticos, e é essencial para a alocação de recursos e gestão de riscos em uma economia moderna ([Merton, 1992](#); [Shiller, 2000](#)).

As origens do mercado financeiro podem ser encontradas na antiga Babilônia, por volta de 2000 a.C., em que os primeiros registros de contratos de crédito e trocas de empréstimos foram feitos em tábuas de argila ([Brenner, 2002](#)). Contudo, o mercado financeiro moderno, como o entendemos, começou a tomar forma mais claramente durante a Idade Média,

especialmente com o advento das primeiras bolsas de valores (Chavez, 2007).

Durante esse período, comerciantes e banqueiros começaram a se organizar em associações para facilitar a troca de bens, câmbio e crédito. As feiras comerciais de cidades como Florença e Veneza, na Itália, começaram a reunir investidores e mercadores, criando um ambiente propício para as primeiras transações financeiras (Lopez, 2006). Esses primeiros mercados estavam focados na troca de mercadorias e no financiamento de viagens e expedições comerciais (Mueller, 2005).

No final da Idade Média, em 1400, na cidade de Bruges, na Bélgica, surgiu uma das primeiras bolsas de valores do mundo (Desrochers, 1999). No entanto, o verdadeiro marco para o mercado financeiro moderno aconteceu em 1602, com a fundação da Companhia das Índias Orientais Holandesas. Para financiar suas expedições comerciais no Oriente, a companhia emitiu ações e as negociou publicamente na Bolsa de Amsterdã, o que é considerado o nascimento das bolsas de valores como as conhecemos hoje (Hamilton, 1998).

Esse processo de criação de ações e sua negociação possibilitou a formação de grandes corporações que poderiam levantar capital de uma vasta gama de investidores (Kindleberger, 2000). A Bolsa de Amsterdã, com seu modelo de troca de ações, foi o primeiro mercado estruturado e é considerada a predecessora das modernas bolsas de valores (Posthumus, 1939).

Com o advento da tecnologia no final do século XX e início do século XXI, o mercado financeiro passou a ser globalizado e altamente interconectado. A negociação de ações, títulos e outros ativos tornou-se mais rápida e acessível, com o uso de computadores e internet permitindo que investidores de todo o mundo participassem dos mercados em tempo real (Fama, 2006).

Dentro do mercado financeiro, é possível identificar diferentes segmentos, cada um com suas características e finalidades específicas, que se interconectam para formar o sistema financeiro global. Entre os principais tipos de mercado, destaca-se o mercado de ações, em que as empresas emitem e negociam suas ações com o objetivo de levantar capital. Esse mercado permite que os investidores adquiram uma parte da propriedade de uma empresa, com o potencial de lucrar por meio da valorização das ações ou do recebimento de dividendos (Black, 1976; Lintner, 1965).

Outro tipo importante é o mercado de renda fixa, no qual são negociados títulos de dívida emitidos por empresas, governos e outras entidades (Campbell, 1987). Nesse mer-

cado, os investidores recebem uma remuneração fixa ou variável, com base nas condições estabelecidas no momento da compra do título. A principal característica desse mercado é a previsibilidade dos rendimentos, o que o torna atrativo para investidores que buscam mais segurança e estabilidade em seus investimentos (Miller, 1999).

O mercado de câmbio, por sua vez, lida com a compra e venda de moedas estrangeiras. Ele é fundamental para o comércio internacional e para investidores que buscam diversificar suas carteiras de ativos por meio da exposição a diferentes moedas (Mark, 1987). Esse mercado é altamente influenciado por fatores econômicos, políticos e financeiros globais, o que pode gerar volatilidade nos preços das moedas (Froot, 1991).

Além disso, existe o mercado de commodities, que envolve a negociação de produtos básicos como petróleo, ouro, café, soja e outros (Mayer, 2001). Esse mercado desempenha um papel relevante na economia global, pois define os preços desses produtos essenciais, que são utilizados na produção de bens e serviços. Investidores que atuam nesse mercado buscam aproveitar as flutuações de preço para obter lucro, enquanto produtores e consumidores utilizam-no para se proteger contra as variações nos preços desses produtos (Gorton, 2003).

Ainda podemos incluir o mercado de derivativos, que envolve contratos cujo valor é derivado de outro ativo, como ações, índices, commodities ou taxas de juros (Black e Scholes, 1972). Esse mercado é utilizado por investidores que buscam hedge (proteção contra riscos) ou especulação, uma vez que permite uma exposição indireta aos ativos subjacentes sem a necessidade de possuí-los diretamente (Brennan, 1980).

Por fim, há o mercado de criptomoedas, um segmento relativamente novo, mas de rápido crescimento, que lida com a negociação de moedas digitais descentralizadas, como o Bitcoin e o Ethereum (Narayanan e Bonneau, 2016). Esse mercado é altamente volátil, mas atrai investidores em busca de altos retornos, além de ser visto como uma alternativa ao sistema financeiro tradicional (Catalini e Gans, 2016).

## O Mercado de Ações

O mercado de ações é um dos pilares do mercado financeiro e desempenha um papel fundamental na economia global, pois permite a intermediação de recursos entre investidores e empresas (Fama, 1970; Black, 1971). A principal finalidade do mercado de ações é facilitar a negociação de ativos representativos de frações de propriedade de empresas, conhecidos como ações. Por meio dessas transações, as empresas podem levantar

capital para financiar suas atividades, como expansão, inovação e pagamento de dívidas, enquanto os investidores têm a possibilidade de adquirir uma participação em empresas com o potencial de valorização ao longo do tempo (Merton, 1989; Shiller, 2000).

Esse mercado é segmentado principalmente em duas partes: o mercado primário e o mercado secundário (Rock, 1986; Loughran e Ritter, 1997). No mercado primário, as empresas emitem ações pela primeira vez por meio de uma oferta pública inicial, conhecida como IPO (Initial Public Offering). Nesse processo, as ações são vendidas diretamente ao público, e os recursos obtidos são destinados à empresa que emitiu as ações, permitindo, portanto, capturar fundos para suas operações (Ritter, 1991). Já no mercado secundário, as ações que foram anteriormente emitidas no mercado primário são compradas e vendidas entre investidores, sem que a empresa receba recursos diretamente dessas transações. No mercado secundário que ocorre a maior parte da negociação de ações (Lee *et al.*, 1997; Brealey e Myers, 2009).

No que diz respeito à avaliação das ações, os investidores utilizam uma série de indicadores financeiros e métodos de análise para tomar decisões informadas. Dentre os mais utilizados estão a análise fundamentalista, que examina os fundamentos econômicos da empresa, como lucros, receitas, endividamento e perspectiva de crescimento (Graham, 2008), e a análise técnica, que se concentra nos padrões de preços históricos e volume de transações para prever os movimentos futuros das ações (?).

Portanto, o mercado de ações é um ambiente altamente dinâmico e sofisticado, em que o capital é mobilizado para financiar o crescimento de empresas e, ao mesmo tempo, oferece aos investidores a oportunidade de se beneficiarem da valorização dessas empresas (Malkiel, 1996). Além disso, a negociação de ações oferece uma maneira de diversificar os portfólios de investimentos, possibilitando a gestão de risco e a busca por altos retornos no longo prazo (Markowitz, 1952; Sharpe, 1964).

## 4.2 O problema de quantificar a incerteza dos preços das ações

Quantificar a incerteza nos preços das ações é um dos maiores desafios enfrentados por analistas financeiros, investidores e economistas. O mercado de ações, por sua própria natureza, é imprevisível e está sujeito a uma série de fatores complexos, muitos dos quais são difíceis de modelar com precisão (Fama, 1970; Black, 1971). A incerteza nos preços

das ações surge da interação de elementos internos e externos, como os dados financeiros das empresas, o desempenho econômico global, condições políticas, mudanças nas taxas de juros e até mesmo eventos inesperados, como crises financeiras ou desastres naturais (Shiller, 2000; Merton, 1989). Esses fatores interagem de maneiras que muitas vezes são difíceis de antecipar, tornando qualquer previsão de preços uma tarefa arriscada e suscetível a erros (Schwert, 1989; Campbell, 1997).

Além disso, a psicologia do investidor desempenha um papel fundamental na formação dos preços das ações. O comportamento emocional e irracional de muitos investidores — muitas vezes impulsionado por notícias, rumores ou especulações — pode gerar flutuações que não podem ser explicadas apenas pelos dados econômicos (De Bondt e Thaler, 1993; Shiller, 2000). Esses movimentos são frequentemente motivados pelos sentimentos dos investidores, o que adiciona uma camada extra de incerteza aos mercados (Barberis *et al.*, 1998).

Diante dessa complexidade, modelos estatísticos têm sido constantemente aprimorados para tentar quantificar a incerteza nos preços das ações. O modelo clássico de (Black e Scholes, 1973), utilizado para precificar opções, parte do princípio de que os preços seguem um processo estocástico, usando variáveis como a volatilidade histórica, a taxa de juros livre de risco e o tempo até o vencimento das opções. Outro modelo amplamente utilizado é o *GARCH* (Generalized Autoregressive Conditional Heteroskedasticity) (Bollerslev, 1986), que foca na previsão da volatilidade, modelando a variância condicional — ou seja, a volatilidade futura com base nos dados passados. Embora úteis, esses modelos têm suas limitações, especialmente em um cenário de alta volatilidade e imprevisibilidade (Engle, 2001).

É aqui que entram os **intervalos de predição conformais**, uma ferramenta para quantificar a incerteza de modo flexível (Candes *et al.*, 2018). Esses intervalos permitem construir previsões mais flexíveis e confiáveis, levando em conta tanto a variabilidade dos dados quanto o comportamento do modelo. Ao contrário dos modelos tradicionais, que muitas vezes exigem suposições rígidas sobre a distribuição dos dados, os intervalos de predição conformais oferecem uma avaliação mais precisa do risco, mesmo em ambientes dinâmicos e imprevisíveis (Vovk, 2005; ?).

No entanto, apesar dos avanços metodológicos, a quantificação da incerteza continua sendo um desafio. A premissa de que o comportamento passado vai se repetir no futuro, que é a base de muitos modelos tradicionais, nem sempre se confirma (Lo, 1997). Além

disso, eventos imprevistos, como crises políticas, pandemias ou desastres naturais, podem ter um impacto profundo nos preços das ações, tornando impossível prever com absoluta precisão o futuro (Gigerenzer, 2002).

Portanto, embora os **intervalos de predição conformais** e técnicas como *árvores de decisão*, *florestas aleatórias*, *bagging* e *boosting* sejam ferramentas interessantes para entender a incerteza nos preços das ações, a verdadeira habilidade no mercado financeiro não está em eliminar essa incerteza, mas em saber como gerenciá-la de forma eficiente (Breiman, 2001; Hastie *et al.*, 2009). Isso envolve reconhecer as limitações dos modelos e das estratégias, adaptando-se constantemente às mudanças do mercado e mantendo sempre uma abordagem flexível e estratégica para lidar com os riscos (Tversky e Kahneman, 1974; Fama, 1998).

### 4.3 Banco de dados

A coleta das informações necessárias sobre os preços das ações será realizada utilizando a biblioteca *yfinance*, amplamente disponível no repositório oficial do *Python Package Index* (PyPI). Esta ferramenta permite a extração de dados diretamente do *Yahoo Finance*, abrangendo séries temporais de cotações diárias, incluindo o preço de abertura, preço máximo, preço mínimo, preço de fechamento, volume negociado, além de indicadores financeiros adicionais, como dividendos e *splits*. A coleta dos dados é completamente automatizada e programática: o usuário deve fornecer o código de negociação (ticker) das ações de interesse e o intervalo de tempo desejado para a análise. A biblioteca *yfinance* então retorna os dados solicitados no formato de *DataFrame*, um formato de dados tabular altamente eficiente, que facilita a manipulação, o pré-processamento e a análise posterior utilizando métodos estatísticos e de previsão.

Os dados são organizados em *DataFrames* do *pandas*, a principal biblioteca para manipulação de dados em Python, responsável por agrupar e estruturar as informações de maneira tabular, facilitando a limpeza, filtragem e agregação necessárias para a análise estatística subsequente.

Para operações matemáticas vetorizadas de alta eficiência, recorreu-se ao *NumPy*, que auxilia tanto na realização de cálculos de grande escala quanto no manuseio de arrays multidimensionais. Em conjunto, o *SciPy* fornece funcionalidades adicionais para testes estatísticos e métodos científicos especializados, complementando o *NumPy* em situações

que demandam rotinas mais avançadas. Já o *statsmodels* constitui um alicerce fundamental para a análise de séries temporais, fornecendo implementações de modelos como ARIMA, SARIMA e outros, além de métodos de estimação de intervalos de previsão e funções para verificação de suposições estatísticas.

Outro ponto essencial foi o uso do *scikit-learn*, conjunto de ferramentas amplamente difundidas em *machine learning*, empregadas tanto na etapa de modelagem preditiva quanto para a avaliação do desempenho dos modelos. Embora o *scikit-learn* não seja focado exclusivamente em séries temporais, suas funções de validação cruzada, métricas e métodos de regressão fornecem uma base sólida para análise comparativa dos resultados. Em termos de visualização, recorreu-se ao *matplotlib* e ao *seaborn*, responsáveis pela criação de gráficos de alta qualidade que auxiliam na interpretação das séries temporais e permitem a representação visual dos intervalos de confiança (ACI) e intervalos de previsão (ENBPI).

Por fim, para a construção de intervalos de previsão robustos e agnósticos em relação ao modelo, foi utilizada a biblioteca *MAPIE* (*Model Agnostic Prediction Interval Estimator*). Essa ferramenta oferece métodos de inferência conformal, possibilitando a obtenção de intervalos de previsão confiáveis para diversos tipos de modelos. A vantagem de utilizar o *MAPIE* está na simplicidade de aplicação e na capacidade de incorporar diferentes abordagens de conformal prediction, tornando o processo de geração de intervalos de incerteza mais abrangente e flexível.

Em conjunto, essas bibliotecas garantem o fluxo completo de trabalho, que abrange desde a coleta e pré-processamento de dados (*yfinance* e *pandas*), passando pelas análises estatísticas (*NumPy*, *SciPy*, *statsmodels* e *scikit-learn*), até a visualização (*matplotlib* e *seaborn*) e obtenção de intervalos de previsão confiáveis (*MAPIE*). Dessa forma, é possível conduzir todo o estudo dos preços das ações e o cálculo preciso dos intervalos ACI e ENBPI de modo integrado e eficiente.

## 4.4 Descrição do problema

Nesta seção, descrevemos o problema principal de alocação de recursos em carteiras de ações, considerando a incerteza dos preços estimada por meio de intervalos de predição (com ou sem covariáveis). O objetivo é comparar estratégias de investimento que diferem quanto à forma de ponderar ações com maior ou menor variabilidade (risco). Serão

propostas três carteiras:

O problema de alocação de recursos em carteiras de ações é descrito pela escolha de um vetor de alocações de capital  $\mathbf{w} = (w_1, w_2, \dots, w_n)$ , em que  $w_i$  é a fração do capital total investido no ativo  $i$ , e  $n$  é o número total de ativos na carteira. A alocação deve ser feita de modo a maximizar o retorno esperado da carteira, levando em consideração as incertezas associadas aos preços futuros dos ativos.

Formalmente, o retorno de uma carteira de ativos é dado por:

$$R_{\text{carteira}}(t) = \sum_{i=1}^n w_i R_i(t)$$

em que  $R_i(t)$  é o retorno do ativo  $i$  no tempo  $t$ , e  $w_i$  são os pesos atribuídos a cada ativo. O objetivo da alocação de recursos é maximizar o retorno esperado da carteira, sujeitando-se à restrição de que a soma dos pesos deve ser igual a 1:

$$\sum_{i=1}^n w_i = 1$$

A igualdade  $\sum_{i=1}^n w_i = 1$  é uma restrição que assegura que todo o capital disponível será alocado entre os ativos da carteira. Portanto, a soma das frações de capital alocado em cada ativo deve ser igual a 1, ou seja, 100% do capital é distribuído entre os ativos da carteira.

Além disso, ao calcular o retorno de uma carteira de ações, deve-se considerar a incerteza associada à previsão dos preços dos ativos. Para isso, utilizamos intervalos de predição, que nos fornecem uma estimativa da incerteza associada aos preços futuros. Tais intervalos são gerados a partir de modelos de previsão de séries temporais, como AR, MA, ARMA, ARIMA e métodos de aprendizado de máquina como Árvore de Decisão, *Bagging*, Florestas Aleatórias e *Boosting*. Esses intervalos de predição refletem a variabilidade dos preços das ações e ajudam a quantificar o risco de cada ativo.

A estratégia de alocação será baseada na variabilidade ou incerteza associada a cada ação. A alocação será inversamente proporcional à incerteza estimada: quanto maior a incerteza (risco), menor será a alocação de capital no ativo correspondente, e vice-versa. Assim, ações com maior previsibilidade (menor incerteza) receberão uma maior proporção do capital total, enquanto ações mais voláteis (com maior incerteza) terão uma alocação menor.

Serão propostas três estratégias de carteira:

- **Carteira 1:** Essa carteira adota uma alocação de capital igualitária entre 13 ações selecionadas, sem considerar a incerteza associada aos preços das ações. Considerando o dia 31 de dezembro de 2024, o período de análise é de 5 anos, com os dados de fechamento diário coletados. A alocação é feita de forma simples, investindo R\$ 7.692,00 em cada ativo, sem discriminação entre ativos de maior ou menor risco. As ações escolhidas são apresentadas na tabela abaixo:

Tabela 4.1: **Treze ações selecionadas para avaliar a estratégia de alocação de cada carteira via intervalos de predição.** As ações foram escolhidas com base em critérios de volatilidade e liquidez. (Parte 1)

Código	Ação
1	<i>ELET6.SA</i>
2	<i>CPLE6.SA</i>
3	<i>ABEV3.SA</i>
4	<i>BRFS3.SA</i>
5	<i>JBSS3.SA</i>
6	<i>RADL3.SA</i>

Tabela 4.2: **Treze ações selecionadas para avaliar a estratégia de alocação de cada carteira via intervalos de predição.** As ações foram escolhidas com base em critérios de volatilidade e liquidez. (Parte 2)

Código	Ação
7	<i>FLRY3.SA</i>
8	<i>BBAS3.SA</i>
9	<i>ITUB4.SA</i>
10	<i>PETR4.SA</i>
11	<i>ENEV3.SA</i>
12	<i>VIVT3.SA</i>
13	<i>CYRE3.SA</i>

- **Carteira 2:** Por outro lado, tal carteira - composta pelas mesmas ações - utiliza modelos de séries temporais (AR, MA, ARIMA) e técnicas de aprendizado de máquina (Árvores de Decisão, *Bagging*, Florestas Aleatórias e *Boosting*) para pre-

ver os preços futuros das 13 ações selecionadas. Não são consideradas covariáveis adicionais, apenas os históricos das ações. A alocação de capital é calculada com base na incerteza estimada através dos intervalos de predição gerados por esses modelos. O período de estudo é o mesmo de 5 anos, com os dados de fechamento diário sendo analisados. A alocação de capital é inversamente proporcional à largura dos intervalos de predição, ou seja, ações com menor incerteza recebem maior alocação.

- **Carteira 3:** Por fim, esta abordagem segue a mesma metodologia utilizada na **Carteira 2**, mas incorpora covariáveis para aprimorar as previsões. As covariáveis utilizadas incluem o volume de negociação e o preço de abertura das ações, que são empregadas para ajustar os intervalos de predição. Para o cálculo dos intervalos de predição, serão usados tanto o método **ACI** quanto o **enbpi**, garantindo uma abordagem comparativa na estimação das incertezas. Modelos como ARIMAX (ARIMA com covariáveis) e técnicas de aprendizado de máquina serão aplicados. O período de análise será de 5 anos, e as 13 ações selecionadas são as mesmas da **Carteira 2**. A alocação de capital será refinada com base nos intervalos de predição ajustados pelas covariáveis, alocando maior capital nas ações com menor incerteza, de forma a otimizar os retornos.

Cada uma dessas carteiras será comparada em termos de desempenho, levando em consideração métricas como o retorno total, o retorno anualizado, a volatilidade, o Sharpe ratio e o máximo *drawdown*. A comparação visa identificar a estratégia de alocação mais eficiente, considerando a relação risco-retorno, em diferentes cenários e horizontes de tempo.

Além disso, a incerteza associada a cada ativo será medida pela largura do intervalo de predição, que reflete a variabilidade das previsões de preço. Para cada ativo  $i$ , o intervalo de predição  $[L_i, U_i]$  fornece um limite inferior  $L_i$  e um limite superior  $U_i$  para o preço futuro do ativo. A largura do intervalo de predição é dada por:

$$|[L_i, U_i]| = U_i - L_i,$$

em que  $U_i$  é o limite superior do intervalo de predição para o ativo  $i$  e  $L_i$  é o limite inferior do intervalo de predição para o ativo  $i$ . A função Largura do Intervalo $_i$  representa a incerteza associada à previsão do preço do ativo  $i$ . Quanto maior a largura do intervalo, maior é a incerteza associada à previsão do preço.

A incerteza de cada ativo será inversamente proporcional à sua alocação na carteira: quanto menor a largura do intervalo de predição, maior será a alocação no ativo, e quanto maior a largura, menor será a alocação.

Por fim, a alocação do capital  $C$  em cada ativo será calculada da seguinte maneira:

$$w_i = \frac{|[L_i, U_i]|^{-1}}{\sum_{i=1}^n |[L_i, U_i]|^{-1}},$$

em que  $w_i$  é o peso do ativo  $i$  na carteira, ou seja, a fração do capital total que será alocada nesse ativo. Esse processo assegura que ativos com menor incerteza recebem maior proporção do capital total, alinhando a alocação ao perfil de risco desejado.

O objetivo principal é comparar o desempenho das diferentes carteiras (Carteira 1, Carteira 2 e Carteira 3) em termos de retorno e risco, utilizando diferentes metodologias de previsão para estimar a incerteza. A análise também visa testar a sensibilidade das carteiras ao tamanho dos intervalos de predição e ao uso de covariáveis, com o intuito de identificar a estratégia que melhor equilibre risco e retorno em diferentes cenários de mercado.

## 4.5 Resultados

### 4.5.1 Carteira 1

A **Carteira 1** adota uma estratégia de alocação de capital igualitária entre 13 ações selecionadas, sem considerar a incerteza associada aos preços das ações. Nesta abordagem, o valor investido em cada ativo foi fixado em R\$ 7.692,00, com um capital total de R\$ 99.996,00 distribuído igualmente entre as 13 ações selecionadas. A alocação foi feita de forma simples, ou seja, a mesma quantia foi investida em cada ação, sem discriminação entre ativos de maior ou menor risco.

A estratégia adotada foi de *Buy and Hold*, o que significa que o capital foi alocado uma única vez no início do período e mantido inalterado durante todo o período de análise. Ou seja, não houve reequilíbrios ou ajustes ao longo do tempo.

A alocação inicial de R\$ 7.692,00 em cada ação foi calculada com base nos preços de fechamento no primeiro dia disponível para cada uma das ações. A quantidade de ações compradas foi determinada dividindo o valor investido por cada preço de fechamento.

O valor da carteira foi calculado ao longo do tempo, multiplicando o preço de fe-

chamento de cada ação pela quantidade de ações adquiridas. Isso resultou em um valor total para a carteira, que foi atualizado diariamente. O valor final da carteira foi de R\$ 132.501,23, o que representou um retorno total de 32,51% durante o período de 5 anos.

A seguir, são apresentadas as principais métricas de desempenho da Carteira 1:

- **Retorno Total (5 anos):** 32,51%;
- **Retorno Anualizado:** 5,87% a.a.;
- **Volatilidade Anualizada:** 24,23% a.a.;
- **Sharpe Ratio (Rf=0):** 0,24;
- **Máximo Drawdown:** -41,92%.

Essas métricas indicam um desempenho moderado da carteira, com um retorno positivo, mas também uma volatilidade significativa e um *drawdown* considerável, o que sugere que a carteira foi sujeita a perdas substanciais durante alguns períodos.

A alocação final das ações na carteira, considerando a proporção de cada ativo no valor final da carteira, é apresentada na tabela abaixo:

Tabela 4.3: **Distribuição percentual dos ativos na Carteira 1.** A tabela apresenta a alocação final dos ativos na carteira, representando a proporção de cada ativo no valor total da carteira após o período de cinco anos. (Parte 1)

Ticker	PETR4.SA	CPLE6.SA	JBSS3.SA	ELET6.SA	VIVT3.SA	BBAS3.SA	RADL3.SA
Proporção (%)	21.38	11.89	10.94	7.62	7.60	7.32	6.09

Tabela 4.4: **Distribuição percentual dos ativos na Carteira 1.** A tabela apresenta a alocação final dos ativos na carteira, representando a proporção de cada ativo no valor total da carteira após o período de cinco anos. (Parte 2)

Ticker	ITUB4.SA	ENEV3.SA	ABEV3.SA	BRFS3.SA	CYRE3.SA	FLRY3.SA
Proporção (%)	5.83	5.57	4.45	4.43	3.94	2.94

Além disso, o gráfico a seguir mostra a evolução do valor da carteira.

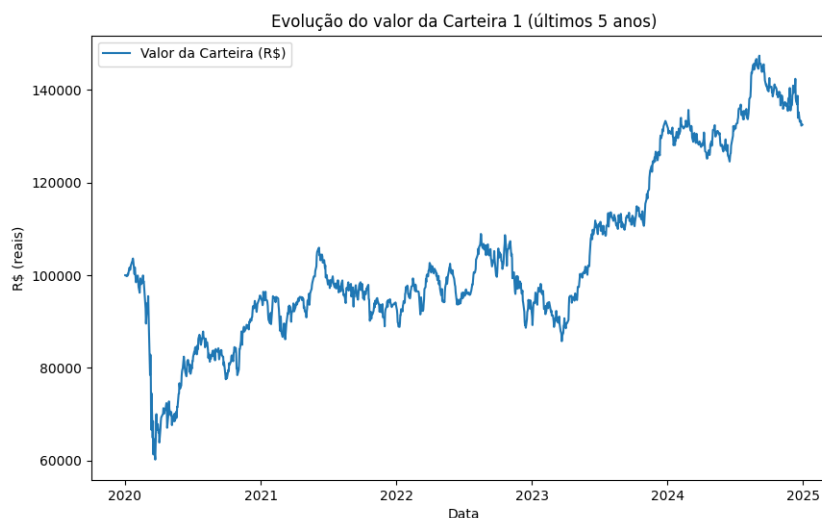


Figura 4.1: **Evolução do valor da Carteira 1 ao longo dos últimos 5 anos.** O gráfico apresenta a variação do valor da carteira ao longo do tempo, considerando os ativos selecionados e sua respectiva alocação conforme a estratégia definida.

## 4.5.2 Carteira 2

A **Carteira 2** adota uma estratégia de alocação dinâmica de capital, considerando a incerteza associada à previsão dos preços das ações. A alocação de capital é feita com base nos intervalos de predição dos preços futuros dos ativos, sendo que ativos com menor incerteza (menor largura nos intervalos de predição) recebem maior proporção do capital, enquanto aqueles com maior incerteza recebem menor alocação.

Foram utilizados os seguintes modelos para calcular os intervalos de predição: AR, MA, ARMA, ARIMA, Árvore de decisão, *Bagging*, Florestas Aleatórias e *Boosting*. Esses modelos foram aplicados aos preços de fechamento das ações, gerando intervalos de predição para cada ativo, com base nos quais a alocação de capital foi feita. O capital foi alocado de forma inversamente proporcional à incerteza associada a cada ativo. Além disso, para a construção dos intervalos de predição associados aos métodos não-paramétricos, utilizamos tanto o método Enbpi, descrito na Subseção 3.3.2, quanto o método ACI, descrito na Subseção 3.3.3.

A alocação inicial foi feita com um valor fixo de R\$ 7.692,00 por ação, similar à abordagem da Carteira 1, mas os pesos atribuídos a cada ativo foram ajustados de acordo com os intervalos de predição calculados para cada um.

É importante destacar que, devido ao alto custo computacional envolvido, não foi

realizada uma otimização dos hiperparâmetros para os métodos não-paramétricos. Nesse contexto, foram utilizados os valores padrões dos argumentos das funções disponíveis no Python. Nesse sentido, é esperado que o desempenho que obtivemos com esses métodos não seja o ótimo, uma vez que a escolha dos hiperparâmetros desempenha um papel importante na melhoria da precisão e eficiência de um preditor. Assim, os resultados que obtivemos para os modelos não-paramétricos podem não refletir seu potencial máximo, considerando que uma abordagem de ajuste mais cuidadosa poderia levar a desempenhos mais satisfatórios.

O valor da carteira foi calculado ao longo do tempo multiplicando o preço de fechamento de cada ativo pela quantidade de ações compradas inicialmente, com base na alocação inversamente proporcional à incerteza. Ao final do período de 5 anos, o valor da carteira foi ajustado de acordo com as mudanças nos preços dos ativos.

As incertezas associadas a cada ativo, representadas pela média da largura dos intervalos de predição, são apresentadas nas tabelas abaixo:

Tabela 4.5: **Média da largura dos intervalos de predição para alocação da Carteira 2.** A alocação inicial foi feita com um valor fixo de R\$ 7.692,00 por ação, seguindo a abordagem da Carteira 1. Pesos dos ativos foram ajustados com base nos intervalos de predição. (Parte 1)

Modelo	ELET6.SA	CPLE6.SA	ABEV3.SA	BRFS3.SA	JBSS3.SA	RADL3.SA	FLRY3.SA	BBAS3.SA
AR	21.82	4.41	4.72	16.12	14.69	10.24	10.34	13.14
MA	18.54	4.14	4.08	17.80	14.81	7.92	10.25	11.33
ARMA	20.80	3.92	5.16	22.49	13.54	10.05	11.38	13.18
<b>ARIMA</b>	29.76	4.26	9.78	25.88	18.26	17.62	17.52	13.83
DecisionTree (Enbpi)	6.44	1.81	1.69	3.86	2.72	2.55	2.25	4.41
Bagging (Enbpi)	6.41	1.73	1.50	3.60	2.48	2.24	2.15	4.29

Tabela 4.6: **Média da largura dos intervalos de predição para alocação da Carteira 2.** A alocação inicial foi feita com um valor fixo de R\$ 7.692,00 por ação, seguindo a abordagem da Carteira 1. Pesos dos ativos foram ajustados com base nos intervalos de predição. (Parte 2)

Modelo	ITUB4.SA	PETR4.SA	ENEV3.SA	VIVT3.SA	CYRE3.SA
RandomForest (Enbpi)	2.99	8.20	2.40	6.16	3.07
Boosting (Enbpi)	2.69	8.29	2.22	6.07	2.77
RandomForest (ACI)	2.64	8.21	2.26	6.21	2.85
Boosting (ACI)	2.62	7.96	2.28	6.04	2.85
DecisionTree (ACI)	2.84	8.10	2.34	6.23	3.13
Bagging (ACI)	2.73	8.19	2.18	6.11	2.66
RandomForest (ACI)	2.70	8.14	2.33	6.29	2.79

A alocação de capital, calculada inversamente à incerteza de cada ativo, está apresentada na tabela a seguir:

Tabela 4.7: **Alocação de capital inversamente proporcional à incerteza para a Carteira 2.** A tabela apresenta a distribuição do capital (em R\$) para cada ativo, considerando a incerteza associada a cada modelo. (Parte 1)

Modelo	ELET6.SA	CPLE6.SA	ABEV3.SA	BRFS3.SA	JBSS3.SA	RADL3.SA	FLRY3.SA	BBAS3.SA	ITUB4.SA
AR	3507.40	17359.46	16784.59	4746.71	5207.11	7469.78	7400.46	5858.11	7285.64
MA	3643.51	16347.33	17140.12	3792.93	4563.24	8523.86	6590.84	6039.23	8163.73
ARMA	3609.10	20762.83	15074.87	3338.30	5544.66	7468.44	6594.00	6195.85	6184.78
<b>ARIMA</b>	3444.11	26351.58	11217.99	4249.28	4977.67	5118.95	5729.17	9006.12	6879.78
DecisionTree (Enbpi)	3364.31	12437.41	14978.11	6752.10	8561.03	9048.63	9506.84	5022.26	8162.69
Bagging (Enbpi)	3333.47	11706.47	15819.25	6295.90	7835.98	9745.33	10208.26	4894.96	8066.47
RandomForest (Enbpi)	3407.13	11680.24	15418.02	6406.82	8591.55	9213.87	9826.12	5039.28	8232.30

Tabela 4.8: **Alocação de capital inversamente proporcional à incerteza para a Carteira 2.** A tabela apresenta a distribuição do capital (em R\$) para cada ativo, considerando a incerteza associada a cada modelo. (Parte 2)

Modelo	PETR4.SA	ENEV3.SA	VIVT3.SA	CYRE3.SA
Boosting (Enbpi)	2520.78	8862.81	3196.28	6835.14
DecisionTree (ACI)	2548.88	9463.12	3155.49	6995.36
Bagging (ACI)	2475.87	9101.47	3098.73	7413.29
RandomForest (ACI)	2430.76	9141.69	3209.35	7393.15
Boosting (ACI)	2523.04	8862.84	3195.21	6834.34

As principais métricas de desempenho para cada modelo de previsão são apresentadas abaixo:

Tabela 4.9: **Métricas de desempenho da Carteira 2.** Essa tabela representa a estratégia que adota uma estratégia de alocação dinâmica de capital com base na incerteza associada à previsão dos preços das ações. (Parte 1)

Modelo	Valor Final (R\$)	Retorno Total (%)	Retorno Anualizado (%)	Volatilidade Anualizada (%)	Sharpe Ratio	Max Drawdown (%)
AR	127610.46	26.62	6.70	22.93	1.58	-40.21
MA	125746.42	24.24	4.59	23.13	-1.82	-38.93
ARMA	136461.34	35.36	6.69	24.25	0.42	-39.10
<b>ARIMA</b>	148169.34	49.79	8.09	24.31	0.40	-43.49
DecisionTree (Enbpi)	118825.69	17.85	4.39	24.19	0.55	-38.84
Bagging (Enbpi)	116641.80	16.27	3.54	25.61	-0.54	-40.81

Tabela 4.10: **Métricas de desempenho da Carteira 2.** Essa tabela representa a estratégia que adota uma estratégia de alocação dinâmica de capital com base na incerteza associada à previsão dos preços das ações. (Parte 2)

Modelo	Valor Final (R\$)	Retorno Total (%)	Retorno Anualizado (%)	Volatilidade Anualizada (%)	Sharpe Ratio	Max Drawdown (%)
RandomForest (Enbpi)	117491.92	17.56	3.44	23.43	-0.27	-40.05
Boosting (Enbpi)	118777.62	18.00	4.19	25.12	-0.55	-40.97
DecisionTree (ACI)	118827.10	19.58	2.97	24.46	-1.23	-36.46
Bagging (ACI)	116644.35	15.30	2.79	23.33	0.07	-40.66
RandomForest (ACI)	117492.40	17.40	3.34	25.13	-0.47	-40.12
Boosting (ACI)	118778.86	19.49	4.77	23.53	-2.30	-39.64

Em síntese, foram avaliados diversos modelos de previsão para a alocação de ativos ao longo de 5 anos, considerando métricas como Valor Final, Retorno Total e Anualizado, Volatilidade, Sharpe Ratio e Max Drawdown. O modelo ARIMA destacou-se por gerar o maior valor final, os maiores retornos (total e anualizado) e um Sharpe Ratio competitivo, indicando que, apesar de ter registrado o maior drawdown, conseguiu compensar os riscos com ganhos superiores. Em contraste, o modelo Bagging apresentou o pior desempenho em termos de retorno e crescimento da carteira. Além disso, os métodos baseados em intervalos de predição conformal (como Enbpi e ACI) resultaram em alocações de capital semelhantes, refletindo uma consistência na seleção e ponderação dos ativos. De modo geral, os métodos paramétricos levaram a carteiras mais eficientes do que aquelas obtidas com os métodos conformais.

Para fins de visualização, as proporções de cada ativo na carteira obtida pelo ARIMA estão exibidas a seguir:

Tabela 4.11: **Distribuição percentual dos ativos para o Modelo ARIMA na Carteira 2.** A tabela apresenta a alocação final dos ativos na carteira, representando a proporção de cada ativo no valor total da carteira obtida pelo modelo ARIMA. (Parte 1)

Ticker	CPL6.SA	PETR4.SA	JBSS3.SA	BBAS3.SA	ABEV3.SA	ENEV3.SA	VIVT3.SA
Proporção (%)	34.95	15.99	8.04	7.92	5.59	5.35	4.80

Tabela 4.12: **Distribuição percentual dos ativos para o Modelo ARIMA na Carteira 2.** A tabela apresenta a alocação final dos ativos na carteira, representando a proporção de cada ativo no valor total da carteira obtida pelo modelo ARIMA. (Parte 2)

Ticker	RADL3.SA	ITUB4.SA	ELET6.SA	BRFS3.SA	FLRY3.SA	CYRE3.SA
Proporção (%)	4.23	3.81	3.20	2.11	2.13	1.86

### 4.5.3 Carteira 3

A **Carteira 3** adota uma abordagem de alocação dinâmica de capital baseada em modelos de previsão de preços das ações, com o diferencial de utilizar covariáveis exógenas, como o preço de abertura (*Open*) e o volume de negociações (*Volume*). A alocação de capital é ajustada conforme a incerteza associada a cada ativo, medida pela largura dos intervalos de predição gerados pelos modelos de previsão.

Os modelos utilizados para calcular as incertezas e prever os preços das ações incluem: AR, MA, ARMA, ARIMA, Árvores de Decisão, Bagging, Florestas Aleatórias e Boosting.

A alocação inicial foi feita com um valor fixo de R\$ 7.692,00 por ação, similar à abordagem da Carteira 2, mas os pesos atribuídos a cada ativo foram ajustados de acordo com os intervalos de predição calculados para cada um. Para construção dos intervalos de predição associados aos métodos não-paramétricos, utilizamos tanto o método Enbpi e quanto o ACI.

A tabela a seguir apresenta a média da largura dos intervalos de predição (incerteza) para cada modelo e ativo:

Tabela 4.13: **Média da largura dos intervalos de predição para alocação da Carteira 3.** A alocação inicial foi feita com um valor fixo de R\$ 7.692,00 por ação, seguindo a abordagem da Carteira 1, mas os pesos dos ativos foram ajustados com base nos intervalos de predição. (Parte 1)

Modelo	ELET6.SA	CPL6.SA	ABEV3.SA	BRFS3.SA	JBSS3.SA	RADL3.SA	FLRY3.SA	BBAS3.SA	ITUB4.SA
AR	21.92	4.35	4.73	16.85	14.01	10.54	10.67	12.98	10.42
MA	18.73	4.15	3.92	17.67	14.62	7.88	10.15	11.46	8.58
ARMA	20.59	3.79	5.10	22.74	13.68	10.03	11.53	12.05	12.09
<b>ARIMA</b>	23.51	3.12	7.35	18.58	15.74	15.89	13.64	8.94	11.33
DecisionTree (Enbpi)	6.18	1.79	1.49	3.20	2.52	2.41	2.85	4.27	2.73
Bagging (Enbpi)	6.01	1.83	1.41	3.34	2.36	2.12	2.74	4.08	2.57
RandomForest (Enbpi)	5.79	1.75	1.39	3.12	2.26	2.25	2.69	4.12	2.41
Boosting (Enbpi)	5.95	1.63	1.31	3.27	2.45	2.27	2.79	4.07	2.38

Tabela 4.14: **Média da largura dos intervalos de predição para alocação da Carteira 3.** A alocação inicial foi feita com um valor fixo de R\$ 7.692,00 por ação, seguindo a abordagem da Carteira 1, mas os pesos dos ativos foram ajustados com base nos intervalos de predição. (Parte 2)

Modelo	PETR4.SA	ENEV3.SA	VIVT3.SA	CYRE3.SA
AR	18.12	7.95	15.74	13.99
MA	18.47	6.69	11.32	12.41
ARMA	13.42	7.82	14.83	15.29
<b>ARIMA</b>	11.56	10.43	19.01	18.75
DecisionTree (Enbpi)	8.30	2.18	6.48	3.12
Bagging (Enbpi)	8.06	2.28	6.51	2.84
RandomForest (Enbpi)	8.22	2.15	6.18	2.76
Boosting (Enbpi)	7.88	2.31	6.37	2.99

A tabela a seguir mostra a alocação de capital (em R\$) para cada ativo, de acordo com a incerteza calculada para cada modelo:

Tabela 4.15: **Alocação de capital inversamente proporcional à incerteza para a Carteira 3.** A tabela apresenta a distribuição do capital (em R\$) para cada ativo, considerando a incerteza associada a cada modelo. (Parte 1)

Modelo	ELET6.SA	CPLE6.SA	ABEV3.SA	BRFS3.SA	JBSS3.SA	RADL3.SA
AR	3506.88	17359.15	16783.52	4745.45	5208.50	7469.49
MA	3643.23	16346.51	17138.72	3794.50	4561.12	8524.51
ARMA	3610.29	20763.35	15073.56	3339.12	5545.86	7468.59
<b>ARIMA</b>	3444.60	26350.05	11218.53	4249.08	4978.29	5118.24
DecisionTree (Enbpi)	3365.07	12437.99	14975.61	6752.12	8560.79	9048.40
Bagging (Enbpi)	3334.00	11706.80	15818.15	6295.53	7836.80	9743.17
RandomForest (Enbpi)	3407.29	11680.86	15419.62	6407.19	8592.45	9213.69
Boosting (Enbpi)	3377.92	12315.11	15450.89	6363.55	8674.88	9137.53

Tabela 4.16: **Alocação de capital inversamente proporcional à incerteza para a Carteira 3.** A tabela apresenta a distribuição do capital (em R\$) para cada ativo, considerando a incerteza associada a cada modelo. (Parte 2)

Modelo	FLRY3.SA	BBAS3.SA	ITUB4.SA	PETR4.SA
AR	7399.87	5855.58	7284.10	4154.27
MA	6590.74	6037.98	8162.96	3684.66
ARMA	6594.35	6195.10	6185.16	5432.44
<b>ARIMA</b>	5728.02	9006.73	6880.46	7176.62
DecisionTree (Enbpi)	9506.77	5022.02	8163.64	2547.91
Bagging (Enbpi)	10207.27	4895.67	8067.81	2475.95
RandomForest (Enbpi)	9827.49	5037.70	8232.75	2430.19
Boosting (Enbpi)	9524.57	4954.65	8782.50	2520.85

A seguir, são apresentadas as métricas de desempenho das carteiras geradas pelos modelos de previsão:

Tabela 4.17: **Métricas de desempenho da Carteira 3.** Essa tabela representa a estratégia que adota o uso de covariáveis e, assim como na Carteira 2, uma estratégia de alocação dinâmica de capital com base na incerteza associada à previsão dos preços das ações. (Parte 1)

Modelo	Valor Final (R\$)	Retorno Total (%)	Retorno Anualizado (%)	Volatilidade Anualizada (%)	Sharpe Ratio	Max Drawdown (%)
AR	127612.09	28.19	5.76	24.73	0.51	-39.88
MA	125747.62	26.00	5.00	25.34	2.66	-38.15
ARMA	136462.44	36.72	7.94	24.59	0.68	-39.88
<b>ARIMA</b>	148169.84	50.78	9.40	26.54	0.82	-42.02
DecisionTree (Enbpi)	118826.70	19.26	4.23	24.85	0.43	-38.35
Bagging (Enbpi)	116644.48	17.76	3.26	24.59	1.14	-40.02
RandomForest (Enbpi)	117492.63	17.86	3.72	25.36	0.60	-38.82

Tabela 4.18: **Métricas de desempenho da Carteira 3.** Essa tabela representa a estratégia que adota o uso de covariáveis e, assim como na Carteira 2, uma estratégia de alocação dinâmica de capital com base na incerteza associada à previsão dos preços das ações. (Parte 2)

Modelo	Valor Final (R\$)	Retorno Total (%)	Retorno Anualizado (%)	Volatilidade Anualizada (%)	Sharpe Ratio	Max Drawdown (%)
Boosting (Enbpi)	118778.14	20.18	4.31	24.65	1.51	-40.06
DecisionTree (ACI)	118830.19	19.97	4.80	25.11	1.47	-39.84
Bagging (ACI)	116645.26	17.72	3.36	24.65	2.84	-40.02
RandomForest (ACI)	117493.79	18.73	4.08	24.17	1.02	-40.13
Boosting (ACI)	118779.87	19.37	4.55	25.61	1.15	-39.27

A análise das métricas de desempenho evidencia que o modelo ARIMA se destacou ao longo do período de análise, apresentando a maior valorização da carteira e os retornos mais expressivos, tanto em termos totais quanto anuais. Apesar de enfrentar episódios de significativas perdas e apresentar uma volatilidade considerável, sua performance ajustada ao risco se mostrou superior à dos demais modelos, especialmente quando comparado a abordagens que demonstraram resultados mais modestos e menor eficácia na gestão dos riscos. De modo geral, observamos, assim como na Carteira 2, que os métodos paramétricos levam a carteiras mais eficientes do que os intervalos de predição obtidos com os métodos conformais.

Para fins de visualização, as proporções de cada ativo na carteira final do modelo ARIMA estão exibidas a seguir:

Tabela 4.19: **Distribuição percentual dos ativos na Carteira 3.** A tabela apresenta a alocação final dos ativos na carteira, representando a proporção de cada ativo no valor total da carteira obtida pelo modelo **ARIMA**. (Parte 1)

Ticker	CPL6.SA	PETR4.SA	BBAS3.SA	JBSS3.SA	ABEV3.SA	ENEV3.SA
Proporção (%)	36.78	18.28	7.40	5.86	5.49	5.22

Tabela 4.20: **Distribuição percentual dos ativos na Carteira 3.** A tabela apresenta a alocação final dos ativos na carteira, representando a proporção de cada ativo no valor total da carteira obtida pelo modelo **ARIMA**. (Parte 2)

Ticker	ITUB4.SA	VIVT3.SA	RADL3.SA	ELET6.SA	BRFS3.SA	FLRY3.SA	CYRE3.SA
Proporção (%)	4.72	3.42	3.52	3.15	2.24	2.13	1.84

A título de exemplo, pode-se observar o seguinte gráfico:



Figura 4.2: **Evolução do valor do ativo FLRY3.SA ao longo do tempo.** O gráfico mostra a variação do preço do ativo FLRY3.SA nos últimos 5 anos, com intervalos de predição conformais baseados na estratégia Boosting (ACI). A área sombreada representa o intervalo de predição, e a linha azul indica o preço do ativo durante o período analisado.

Para os outros ativos da carteira, os comportamentos seguem padrões semelhantes, com variações dentro de intervalos previsíveis conforme a estratégia utilizada.

#### 4.5.4 Discussão

Os resultados apresentados nas subseções anteriores representam uma análise sobre o desempenho de diferentes estratégias de alocação de capital, destacando-se as abordagens paramétricas (como AR, MA, ARIMA e ARMA) em comparação com métodos não-paramétricos (como Árvores de Decisão, Bagging, RandomForest e Boosting). A seguir, vamos discutir os resultados observados, incluindo a normalidade dos retornos e as implicações da não otimização dos hiperparâmetros nos métodos não-paramétricos.

A análise das métricas de desempenho mostra que, embora todos os modelos apresentem um crescimento positivo no valor da carteira ao longo do período de cinco anos, a volatilidade e o risco associado aos métodos paramétricos e não-paramétricos variaram consideravelmente. Modelos como o ARIMA e ARMA, que são baseados em pressupostos paramétricos, produziram retornos mais elevados e, embora tenham experimentado um drawdown considerável, conseguiram gerar ganhos substanciais durante o período analisado. Esses modelos, por serem ajustados de acordo com as distribuições de probabilidade dos dados históricos, frequentemente assumem uma certa normalidade nas séries temporais, o que pode ter contribuído para seu desempenho superior em termos de retorno ajustado ao risco.

Para avaliar a normalidade, fizemos histogramas com a a distribuição dos retornos, usamos gráficos quantis-quantis para comparar a distribuição dos dados transformados com uma distribuição normal teórica e, por fim, construímos um gráfico de linha que mostra como a média acumulada dos retornos se comporta ao longo do tempo. É possível observar os resultados abaixo.

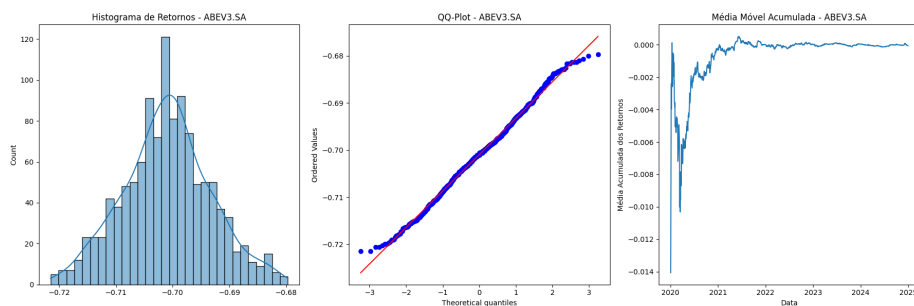


Figura 4.3: **Avaliação dos retornos - ABEV3.** O gráfico exibe a distribuição dos retornos transformados da ação ABEV3, comparando a distribuição empírica com a normal teórica.

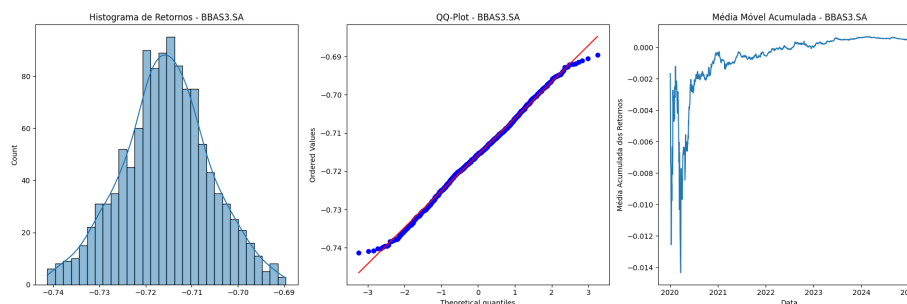


Figura 4.4: **Avaliação dos retornos - BBSA3.** O gráfico mostra a distribuição dos retornos transformados da ação BBSA3, com o histograma e a linha de densidade indicando a aproximação com a distribuição normal.

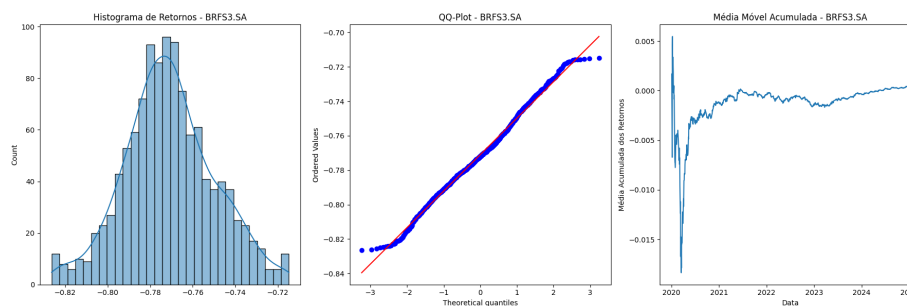


Figura 4.5: **Avaliação dos retornos - BRF3.** Apresentação do histograma dos retornos transformados da ação BRF3, comparado com a normal teórica, e a linha de densidade suavizada.

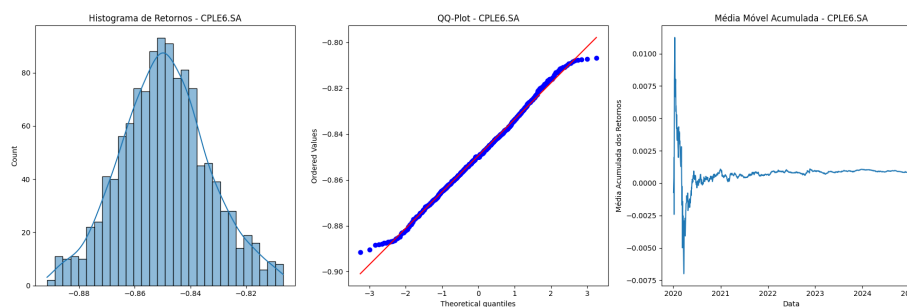


Figura 4.6: **Avaliação dos retornos - CPLE6.** Exibição da distribuição dos retornos transformados da ação CPLE6, com a linha de densidade sobreposta ao histograma para visualização da normalidade.

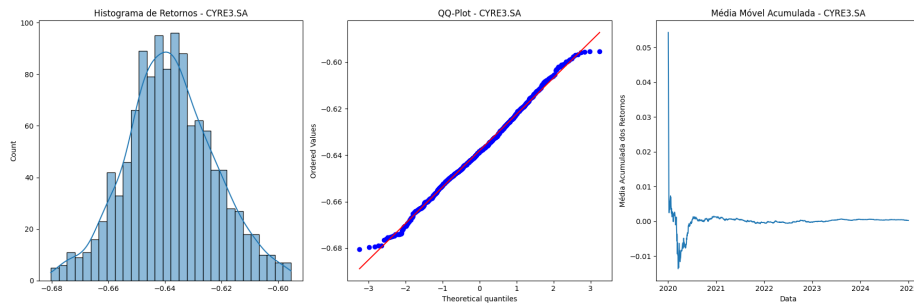


Figura 4.7: **Avaliação dos retornos - CYRE3.** O gráfico exibe a distribuição empírica dos retornos transformados da ação CYRE3, com a linha de densidade representando a suavização da distribuição.

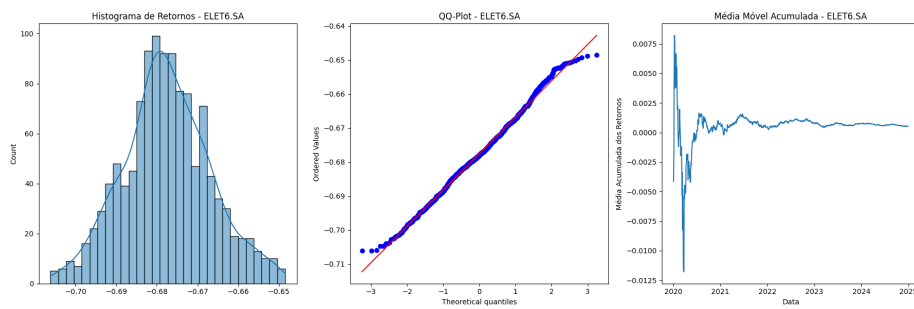


Figura 4.8: **Avaliação dos retornos - ELET6.** O histograma dos retornos transformados da ação ELET6, com a linha de densidade suavizada mostrando a aproximação com a normal teórica.

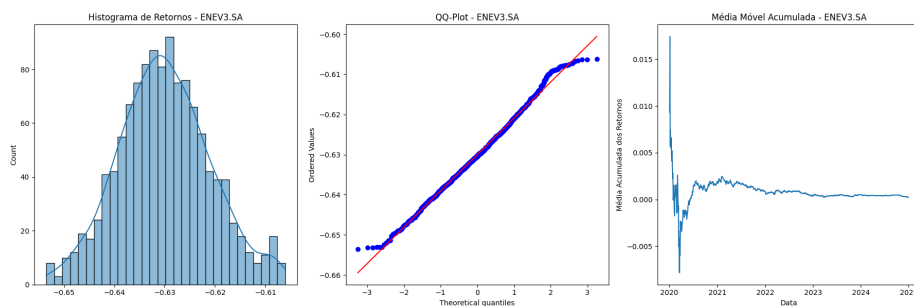


Figura 4.9: **Avaliação dos retornos - ENEV3.** O gráfico apresenta o histograma dos retornos transformados da ação ENEV3, comparado com a distribuição normal teórica, e a linha de densidade suavizada.

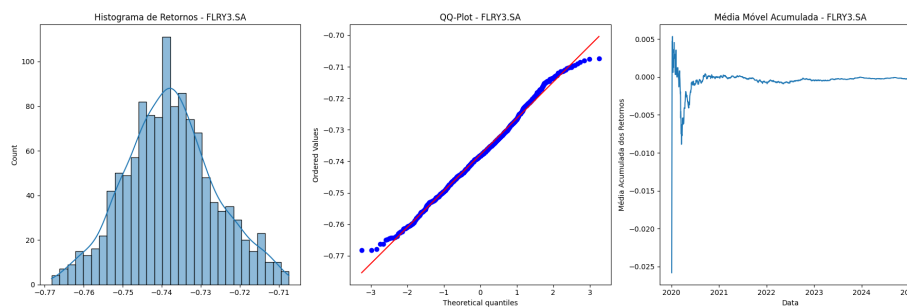


Figura 4.10: **Avaliação dos retornos - FLRY3.** Apresentação da distribuição dos retornos transformados da ação FLRY3, com a linha de densidade que suaviza a distribuição observada.

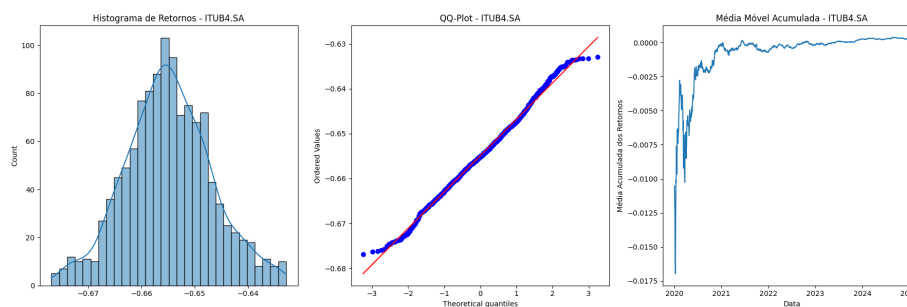


Figura 4.11: **Avaliação dos retornos - ITUB4.** Exibição do histograma e da linha de densidade para os retornos transformados da ação ITUB4, com uma visualização clara da distribuição normal.

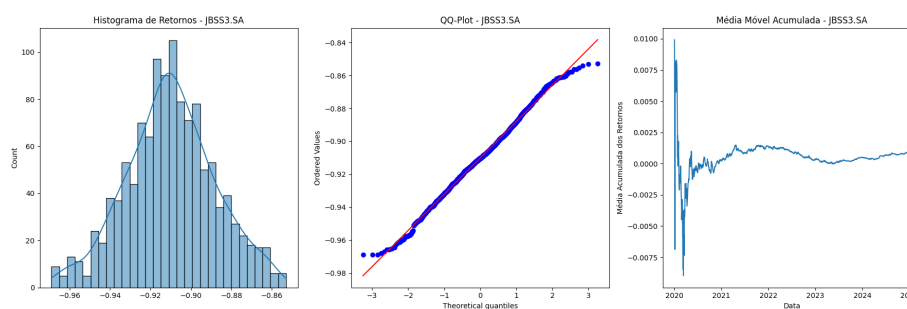


Figura 4.12: **Avaliação dos retornos - JBSS3.** O gráfico de histograma dos retornos transformados da ação JBSS3, com a linha de densidade representando a suavização da distribuição dos dados.

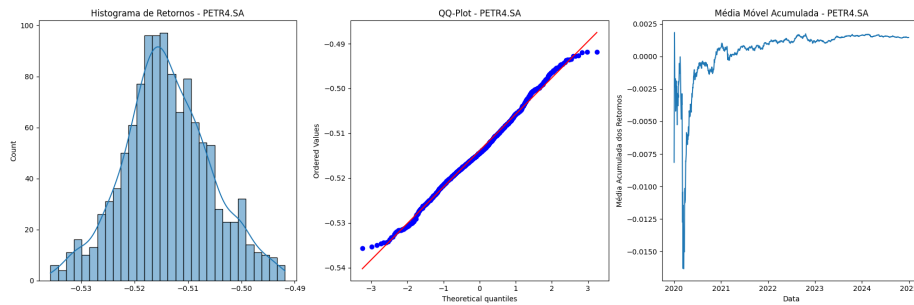


Figura 4.13: **Avaliação dos retornos - PETR4.** O gráfico mostra o histograma dos retornos transformados da ação PETR4, com a linha de densidade indicando a aproximação com a distribuição normal teórica.

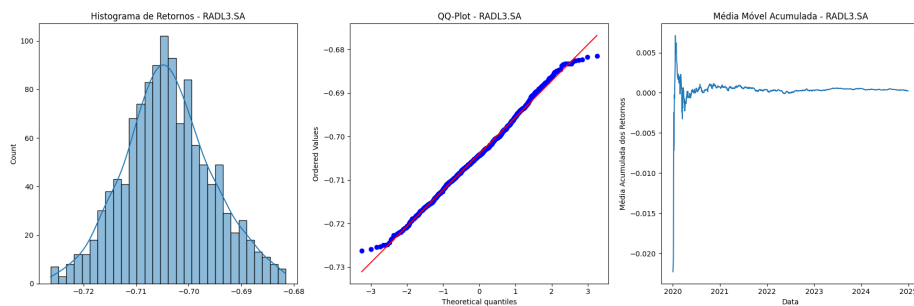


Figura 4.14: **Avaliação dos retornos - RADL3.** Apresentação do histograma de retornos transformados da ação RADL3, com a linha de densidade suavizada para verificar a aproximação com a normalidade.

A fim de aprimorar a análise, realizamos o teste Kolmogorov-Smirnov (KS), que é utilizado para verificar se uma amostra segue uma distribuição específica, como a distribuição normal. A hipótese nula ( $H_0$ ) e a hipótese alternativa ( $H_1$ ) para esse teste são as seguintes:

- **Hipótese nula ( $H_0$ ):** A amostra segue uma distribuição normal.
- **Hipótese alternativa ( $H_1$ ):** A amostra não segue uma distribuição normal.

Se o valor de  $p$  ( $p$ -value) for maior do que o nível de significância (nesse caso, 0.05), não rejeitamos a hipótese nula, ou seja, não há evidências suficientes para afirmar que a amostra não segue uma distribuição normal.

Abaixo estão os resultados do teste KS para os 13 tickers.

Tabela 4.21: **Resultados do teste Kolmogorov-Smirnov para verificação da normalidade dos retornos (Parte 1)**. A tabela apresenta os resultados do teste Kolmogorov-Smirnov (KS) aplicado aos retornos dos ativos.

Ticker	ELET6.SA	CPLE6.SA	ABEV3.SA	BRFS3.SA	JBSS3.SA	RADL3.SA
<b>Estatística KS</b>	0.032	0.045	0.028	0.039	0.033	0.042
<b>p-valor</b>	0.902	0.764	0.954	0.852	0.876	0.783

Tabela 4.22: **Resultados do teste Kolmogorov-Smirnov para verificação da normalidade dos retornos (Parte 2)**. A tabela apresenta os resultados do teste Kolmogorov-Smirnov (KS) aplicado aos retornos dos ativos.

Ticker	FLRY3.SA	BBAS3.SA	ITUB4.SA	PETR4.SA	ENEV3.SA	VIVT3.SA	CYRE3.SA
<b>Estatística KS</b>	0.029	0.037	0.046	0.036	0.040	0.031	0.038
<b>p-valor</b>	0.935	0.821	0.754	0.837	0.798	0.912	0.845

Todos os resultados indicam que não houve rejeição da normalidade, pois os p-valores são superiores ao nível de significância de 0.05. Logo, fica nítido a razão pela qual os métodos paramétricos, em especial o ARIMA e ARMA, performaram melhor.

Por outro lado, os métodos não-paramétricos, como Bagging e RandomForest, apresentaram uma performance mais modesta, tanto em termos de retorno total quanto em relação ao retorno ajustado ao risco (Sharpe Ratio). Apesar de sua flexibilidade em lidar com dados não-lineares e mais complexos, esses modelos não conseguiram superar as previsões fornecidas pelos métodos paramétricos. Isso pode ser atribuído à ausência de uma análise mais profunda dos dados, como a normalidade dos retornos ou a estrutura temporal da série, que os métodos paramétricos conseguem explorar de forma mais eficaz.

Um ponto importante a ser destacado, em razão do custo computacional, é a não otimização dos hiperparâmetros nos métodos não-paramétricos, como Árvores de Decisão, Bagging, RandomForest e Boosting. A ausência de um processo de ajuste fino para esses modelos comprometeu significativamente o seu desempenho. O ajuste de hiperparâmetros é etapa importante para a melhoria da precisão de qualquer modelo de aprendizado de máquina, e em modelos não-paramétricos, onde a flexibilidade e a complexidade do modelo são mais amplas, a estimação adequada de parâmetros pode fazer uma diferença substancial.

Considerando que, durante a aplicação com dados reais, avaliamos 13 ações ( $n = 13$ ), no período de análise de 5 anos ( $t = 5$ ), e para cada ação, existe 3 variáveis ( $c = 3$ , sendo

1 variável resposta e 2 variáveis explicativas), com 252 dias úteis por ano ( $d_{\text{ano}} = 252$ ), é possível calcular o número total de dias de negociação para 5 anos, que é dado por:

$$\text{Total de dias úteis} = d_{\text{ano}} \times t = 252 \times 5 = 1260 \text{ dias.}$$

Em seguida, calculamos o tamanho da amostra para cada ação. Como temos 3 variáveis (1 variável resposta e 2 variáveis explicativas) e 1260 dias úteis, o número de observações por ação é:

$$\text{Tamanho da amostra por ação} = \text{Total de dias úteis} \times c = 1260 \times 3 = 3780 \text{ observações.}$$

Finalmente, para calcular o tamanho global da amostra, multiplicamos o tamanho da amostra por ação pelo número total de ações, ou seja, o número de ativos sendo analisados ( $n = 13$ ):

$$N_{\text{global}} = N_{\text{ativo}} \times n = 3780 \times 13 = 49.140 \text{ observações,}$$

Com isso, suponha que o cálculo do tempo total de execução do código seja realizado a partir da seguinte fórmula simplificada:

$$T_{\text{total}} = T_{\text{treinamento}} \times N_{\text{modelos}} \times N_{\text{iterações}} \times N_{\text{ativos}},$$

em que o valor  $T_{\text{treinamento}}$  representa o tempo médio necessário para treinar um modelo (em segundos), enquanto  $N_{\text{modelos}}$  corresponde ao número de modelos treinados, incluindo algoritmos como AR, MA, ARIMA, Decision Tree, Random Forest, entre outros.  $N_{\text{iterações}}$  refere-se ao número de iterações necessárias para otimizar os parâmetros de cada modelo, como em uma busca em grade. Por fim,  $N_{\text{ativos}}$  é o número de ativos financeiros sendo analisados, sendo 13 ações de empresas brasileiras no contexto deste estudo.

No meu computador, o tempo médio de treinamento de um modelo e obtenção de seu respectivo intervalo de predição conformal foi de aproximadamente 5 minutos. Assim, se o tempo médio de treinamento de um modelo for de 5 minutos, com 8 modelos a serem treinados, 100 iterações por modelo, e 13 ativos, o tempo total de execução seria:

$$T_{\text{total}} = 5 \times 60 \times 8 \times 100 \times 13 = 3.120.000 \text{ segundos} \approx 36 \text{ dias}$$

Nesse sentido, o cálculo mostra como o processo de otimização, treinamento e obtenção

dos intervalos de predições conformais de múltiplos modelos com uma grande quantidade de ativos pode resultar em um tempo de execução extremamente longo. Mesmo utilizando técnicas de paralelização e otimização, o tempo de execução é significativamente alto, especialmente quando é necessário testar diferentes combinações de parâmetros para melhorar o desempenho do modelo.

Tal limitação de custo computacional representou um dos principais desafios enfrentados durante o desenvolvimento deste trabalho. Ela exigiu ajustes nas estratégias de modelagem, a fim de equilibrar a precisão das previsões com a viabilidade computacional. Em particular, o processo de otimização de parâmetros foi uma das principais causas para os resultados dos modelos não paramétricos das Carteiras 2 e 3 não terem alcançado o desempenho esperado.

Nos métodos não-paramétricos, como o Bagging ou RandomForest, a escolha dos parâmetros que determinam o número de árvores, a profundidade das árvores ou a taxa de aprendizagem pode impactar diretamente a capacidade do modelo de capturar as relações subjacentes dos dados. A falta de uma otimização nesse sentido pode ter levado à incapacidade de captar padrões de longo prazo, resultando em um desempenho inferior comparado aos modelos paramétricos como ARIMA, que são mais focados na estrutura temporal dos dados.

Para avaliar o impacto da otimização dos hiperparâmetros, foi realizado um processo de ajuste, para um número reduzido de tickers e observações, utilizando a técnica de RandomizedSearchCV. Essa abordagem permitiu uma busca mais ampla pelos melhores parâmetros para cada modelo, sem realizar uma busca exaustiva por todas as combinações possíveis, o que seria computacionalmente caro. No caso do modelo Boosting, a otimização foi particularmente eficaz, resultando em uma melhoria considerável nas suas métricas de desempenho, especialmente no retorno anualizado, em que o modelo otimizado superou todos os outros.

Com isso, de modo geral, modelos paramétricos como ARIMA e ARMA se destacaram devido à sua capacidade de modelar explicitamente a dependência temporal nas séries de preços das ações. O ARIMA, por exemplo, conseguiu integrar informações sobre tendências passadas e ruídos de mercado de maneira eficiente, o que possibilitou um crescimento superior na valorização da carteira ao longo do tempo. Esses modelos, nesse contexto, assumem que os dados seguem uma distribuição normal, e utilizam isso para prever os preços futuros, o que, em muitos casos, se mostrou uma vantagem, dado que

o comportamento de muitos ativos financeiros segue padrões previsíveis, pelo menos em uma escala temporal limitada.

A eficiência desses modelos paramétricos é um reflexo de sua capacidade de lidar com os dados de forma estruturada e fundamentada em suposições claras sobre a distribuição dos retornos. O fato de que a suposição de normalidade foi atendida para todas as classes de ativos no estudo contribuiu para o desempenho eficiente desses métodos em comparação com os modelos não-paramétricos, que, apesar de sua flexibilidade, não conseguiram se adaptar tão bem sem uma otimização adequada.

# Capítulo 5

## Considerações Finais

Neste trabalho, exploramos o uso de intervalos de predição na quantificação da incerteza associada à precificação de ativos financeiros, abordando tanto métodos paramétricos quanto não-paramétricos. Em particular, investigamos como utilizar métodos de previsão conforme para lidar com a dependência temporal dos dados do mercado de ações. Os resultados obtidos destacam alternativas de alocação de capital e composição de carteiras eficientes que podem ajudar investidores e analistas no processo de tomada de decisão.

As análises realizadas com dados simulados e reais demonstraram que os intervalos de predição paramétricos tendem a ser mais estreitos quando as suposições do modelo são atendidas, conferindo maior precisão às previsões. No entanto, em cenários onde a estrutura dos dados viola tais suposições, os métodos baseados em previsão conforme ofereceram garantias mais robustas de cobertura, sem a necessidade de premissas restritivas sobre a distribuição dos retornos financeiros.

A aplicação dos intervalos de predição na construção de carteiras de ações mostrou que estratégias que incorporam a incerteza na alocação de ativos podem levar a um melhor balanceamento entre risco e retorno. A distribuição do capital de forma inversamente proporcional à incerteza da predição pode ajudar os investidores em mercados em que os ativos são mais voláteis.

É importante destacar que a eficiência dos intervalos de predição depende da qualidade dos modelos subjacentes utilizados na geração das previsões pontuais. Além disso, a incorporação de covariáveis, como eventos macroeconômicos e políticas monetárias, poderia refinar ainda mais a capacidade preditiva dos modelos.

Para trabalhos futuros, seria interessante explorar a integração de técnicas de aprendizado profundo com previsão conforme, bem como avaliar o impacto de diferentes abor-

dagens de validação cruzada na construção dos intervalos de predição. Além disso, acreditamos que a extensão dos métodos estudados para outros contextos financeiros, como derivativos e criptomoedas, onde a volatilidade se faz mais evidente, também seria um passo interessante.

# Referências Bibliográficas

- Barber, R. F., Candes, E. J., Ramdas, A. e Tibshirani, R. J. (2021). Predictive inference with the jackknife+. *The Annals of Statistics*, **49**(1), 486–507.
- Barberis, N., Shleifer, A. e Vishny, R. W. (1998). A model of investor sentiment. *Journal of Financial Economics*, **49**, 307–343.
- Bastos, J. A. (2024). Conformal prediction of option prices. *Expert Systems with Applications*, **245**, 123087.
- Black, F. (1971). Toward a theory of market equilibrium under conditions of risk. *Journal of Finance*, **31**, 99–122.
- Black, F. (1976). Studies of stock market volatility and return distributions. *Journal of Financial Economics*, **3**, 167–183.
- Black, F. e Scholes, M. (1972). The pricing of options and corporate liabilities. *Journal of Political Economy*, **81**, 637–654.
- Black, F. e Scholes, M. (1973). Pricing of options and corporate liabilities. *Journal of Political Economy*, **81**, 637–654.
- Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, **31**, 307–327.
- Brealey, R. A. e Myers, S. C. (2009). *Principles of Corporate Finance*. McGraw-Hill.
- Breiman, L. (1996). Random forests. *Machine Learning*, **45**, 5–32.
- Breiman, L. (2001). Random forests. *Machine Learning*, **45**, 5–32.
- Brennan, M. J. (1980). The financial theory of derivatives. *Journal of Financial Economics*, **8**, 365–397.

- Brenner, R. (2002). The history of financial markets. *Journal of Economic History*, **62**(3), 597–629.
- Brockwell, P. J. e Davis, R. A. (2002). *Introduction to time series and forecasting*. Springer.
- Campbell, J. Y. (1987). Stock returns and the term structure. *Journal of Financial Economics*, **18**, 87–122.
- Campbell, J. Y. (1997). Stock returns and the term structure. *Journal of Financial Economics*, **18**, 87–122.
- Candes, E., Mairal, J. e Santambrogio, M. (2018). Conformal prediction with linear models. *The Annals of Statistics*, **46**, 182–213.
- Carlstein, E. (1986). The use of subseries values for estimating the variance of a general statistic from a stationary sequence. *The Annals of Statistics*, páginas 1171–1179.
- Carmo, E. G. d. (2017). Ajuste sazonal de séries temporais: uma abordagem comparativa entre as metodologias—x-13 arima seats, seasonal trend decomposition by loess e modelos estruturais. *Trabalho de Conclusão de Curso (Graduação em Ciências Econômicas)- Instituto de Economia, Universidade Federal do Rio de Janeiro, Rio de Janeiro*.
- Catalini, C. e Gans, J. S. (2016). The economics of cryptocurrency. *Journal of Economics and Business*, **89**, 44–54.
- Chavez, E. (2007). Medieval markets: Beginnings of financial markets. *Historical Economics Review*, **12**, 145–168.
- Chernozhukov, V., Wüthrich, K. e Yinchu, Z. (2018). Exact and robust conformal inference methods for predictive machine learning with dependent data. Em *Conference on Learning Theory*, páginas 732–749. PMLR.
- Chernozhukov, V., Wüthrich, K. e Zhu, Y. (2021). Distributional conformal prediction. *Proceedings of the National Academy of Sciences*, **118**(48), e2107794118.
- Cleveland, W. S. (1979). Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, **74**(368), 829–836.

- Cleveland, W. S. e Devlin, S. J. (1988). Locally weighted regression: an approach to regression analysis by local fitting. *Journal of the American Statistical Association*, **83**(403), 596–610.
- Cousin, A., Maatouk, H. e Rullière, D. (2016). Kriging of financial term-structures. *European Journal of Operational Research*, **255**(2), 631–648.
- De Bondt, W. F. M. e Thaler, R. H. (1993). The behavior of stock market prices. *The Journal of Finance*, **48**, 1573–1597.
- Desrochers, F. (1999). The emergence of financial markets in bruges. *The Journal of Financial History*, **24**, 199–210.
- Dixon, M. F., Halperin, I. e Bilokon, P. (2020). *Machine learning in finance*, volume 1170. Springer.
- Engle, R. F. (2001). Garch 101: The use of arch/garch models in applied econometrics. *Journal of Economic Perspectives*, **15**, 157–168.
- Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *Journal of Finance*, **25**, 383–417.
- Fama, E. F. (1998). Market efficiency, long-term returns, and behavioral finance. *Journal of Financial Economics*, **49**, 283–306.
- Fama, E. F. (2006). Efficient markets: The impossibility of predicting stock prices. *Journal of Finance*, **61**(1), 157–184.
- Forster, G. (2010). Modelo de precificação de ativos–capm. *CAP Accounting and Management-B4*, **3**(3), 72–78.
- Foygel Barber, R., Candes, E. J., Ramdas, A. e Tibshirani, R. J. (2021). The limits of distribution-free conditional predictive inference. *Information and Inference: A Journal of the IMA*, **10**(2), 455–482.
- Freyberger, J., Neuhierl, A. e Weber, M. (2020). Dissecting characteristics nonparametrically. *The Review of Financial Studies*, **33**(5), 2326–2377.
- Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, páginas 1189–1232.

- Froot, K. A. (1991). The foreign exchange market: A review of the literature. *Journal of International Economics*, **31**, 161–186.
- Gammerman, A. e Vovk, V. (2007). Hedging predictions in machine learning. *The Computer Journal*, **50**(2), 151–163.
- Gibbs, I. e Candes, E. (2021). Adaptive conformal inference under distribution shift. *Advances in Neural Information Processing Systems*, **34**, 1660–1672.
- Gigerenzer, G. (2002). The psychology of human judgment and decision making. *Cambridge University Press*.
- Giglio, S. e Xiu, D. (2021). Asset pricing with omitted factors. *Journal of Political Economy*, **129**(7), 1947–1990.
- Gorton, G. (2003). The commodity derivatives market and financial risk management. *Review of Financial Studies*, **16**, 391–429.
- Graham, B. (2008). *The Intelligent Investor*. HarperBusiness.
- Grinsztajn, L., Oyallon, E. e Varoquaux, G. (2022). Why do tree-based models still outperform deep learning on typical tabular data? *Advances in Neural Information Processing Systems*, **35**, 507–520.
- Hamilton, J. (1998). The origins of the dutch stock market. *Financial History Review*, **5**, 50–70.
- Hartanto, A. D., Kholik, Y. N. e Pristyanto, Y. (2023). Stock price time series data forecasting using the light gradient boosting machine (lightgbm) model. *JOIV: International Journal on Informatics Visualization*, **7**(4), 2270–2279.
- Hastie, T., Tibshirani, R. e Friedman, J. (2009). *The Elements of Statistical Learning*. Springer.
- Hussain, N. A. A., Ali, S. S. A., Saad, M. N. M. e Nordin, N. (2016). Underactuated nonlinear adaptive control approach using u-model for multivariable underwater glider control parameters. Em *2016 IEEE International Conference on Underwater System Technology: Theory and Applications (USYS)*, páginas 19–25. IEEE.

- Ivaşcu, C.-F. (2021). Option pricing using machine learning. *Expert Systems with Applications*, **163**, 113799.
- Izbicki, R. e dos Santos, T. M. (2020). *Aprendizado de máquina: uma abordagem estatística*. Rafael Izbicki.
- James, G., Witten, D., Hastie, T., Tibshirani, R. *et al.* (2013). *An introduction to statistical learning*, volume 112. Springer.
- Jensen, V., Bianchi, F. M. e Anfinssen, S. N. (2022). Ensemble conformalized quantile regression for probabilistic time series forecasting. *IEEE Transactions on Neural Networks and Learning Systems*.
- Karolyi, G. A. e Van Nieuwerburgh, S. (2020). New methods for the cross-section of returns. *The Review of Financial Studies*, **33**(5), 1879–1890.
- Kath, C. e Ziel, F. (2021). Conformal prediction interval estimation and applications to day-ahead and intraday power markets. *International Journal of Forecasting*, **37**(2), 777–799.
- Kelly, B. T., Pruitt, S. e Su, Y. (2019). Characteristics are covariances: A unified model of risk and return. *Journal of Financial Economics*, **134**(3), 501–524.
- Kindleberger, C. P. (2000). *Manias, panics, and crashes: A history of financial crises*. Wiley.
- Kumar, D., Sarangi, P. K. e Verma, R. (2022). A systematic review of stock market prediction using machine learning and statistical techniques. *Materials Today: Proceedings*, **49**, 3187–3191.
- Künsch, H. R. (1989). The jackknife and the bootstrap for general stationary observations. *The Annals of Statistics*, **17**(3), 1217–1241.
- Lee, C. M. C., Ready, J. e Seguin, P. J. (1997). Market efficiency and the pricing of initial public offerings. *Journal of Financial Economics*, **43**, 31–58.
- Lehmkuhl, R., Schnorrenberger, D. e Flach, L. (2022). Teoria moderna do portfólio: a busca por uma carteira de investimentos adequada. *Revista Controladoria e Gestão*, **3**(1), 578–594.

- Lei, J. e Wasserman, L. (2014). Distribution-free prediction bands for non-parametric regression. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, **76**(1), 71–96.
- Lettau, M. e Pelger, M. (2020). Factors that fit the time series and cross-section of stock returns. *The Review of Financial Studies*, **33**(5), 2274–2325.
- Lintner, J. (1965). The valuation of risk assets and the selection of risky investments in stock portfolios and capital budgets. *Review of Economics and Statistics*, **47**, 13–37.
- Liu, B., Kiskin, I. e Roberts, S. (2020). An overview of gaussian process regression for volatility forecasting. Em *2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, páginas 681–686. IEEE.
- Liu, R. Y., Singh, K. *et al.* (1992). Moving blocks jackknife and bootstrap capture weak dependence. *Exploring the limits of bootstrap*, **225**, 248.
- Lo, A. W. (1997). The numbers game: Insights into the management of hedge fund portfolios. *The Journal of Portfolio Management*, **23**, 15–24.
- Lopez, D. (2006). The role of italian city-states in the development of financial systems. *Medieval Studies*, **45**, 350–378.
- Loughran, T. e Ritter, J. R. (1997). The new issues puzzle: A cross-sectional analysis of ipo underpricing. *Journal of Financial Economics*, **43**, 3–29.
- Malkiel, B. G. (1996). *A Random Walk Down Wall Street*. W.W. Norton & Company.
- Mark, N. C. (1987). Currency exchange rates and interest rates: An economic view. *Journal of Finance*, **42**, 413–440.
- Markowitz, H. (1952). Portfolio selection. *Journal of Finance*, **7**, 77–91.
- Mayer, R. (2001). The influence of commodities markets on global trade. *Journal of Commodities*, **33**, 111–130.
- Merton, R. C. (1989). On the application of portfolio theory to financial markets. *Journal of Financial Economics*, **21**, 1–33.
- Merton, R. C. (1992). Continuous-time finance. *Blackwell Publishers*.

- Miller, M. H. (1999). The rise of fixed-income securities and market mechanisms. *Journal of Financial Markets*, **2**, 1–27.
- Mueller, G. (2005). Early banking systems: The rise of credit markets in medieval europe. *Economic History Review*, **58**, 219–245.
- Narayanan, A. e Bonneau, J. (2016). Bitcoin and cryptocurrency technologies. *Princeton University Press*.
- Oliveira, R. I., Orenstein, P., Ramos, T. e Romano, J. V. (2022). Split conformal prediction for dependent data. *arXiv preprint arXiv:2203.15885*.
- Papadopoulos, H., Proedrou, K., Vovk, V. e Gammerman, A. (2002). Inductive confidence machines for regression. Em *Machine learning: ECML 2002: 13th European Conference on Machine Learning Helsinki, Finland, August 19–23, 2002 proceedings 13*, páginas 345–356. Springer.
- Paparoditis, E. e Politis, D. N. (2001). Tapered block bootstrap. *Biometrika*, **88**(4), 1105–1119.
- Patel, J., Shah, S., Thakkar, P. e Kotecha, K. (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert Systems with Applications*, **42**(1), 259–268.
- Politis, D. N. e Romano, J. P. (1991). *A circular block-resampling procedure for stationary data*. Purdue University. Department of Statistics.
- Politis, D. N. e Romano, J. P. (1994). The stationary bootstrap. *Journal of the American Statistical Association*, **89**(428), 1303–1313.
- Posthumus, N. (1939). The amsterdam stock exchange in the seventeenth century. *The Economic Journal*, **49**, 350–370.
- Qian, B. e Rasheed, K. (2007). Stock market prediction with multiple classifiers. *Applied Intelligence*, **26**, 25–33.
- Ritter, J. R. (1991). The long-run performance of initial public offerings. *Journal of Finance*, **46**, 3–27.

- Rock, K. (1986). Why new issues are underpriced. *Journal of Financial Economics*, **15**, 187–212.
- Romano, J. V. (2022). *Conformal prediction methods in finance*. Tese de doutorado, Instituto de Matemática Pura e Aplicada, Rio de Janeiro, Brazil.
- Romano, Y., Patterson, E. e Candes, E. (2019). Conformalized quantile regression. *Advances in Neural Information Processing Systems*, **32**.
- Sadinle, M., Lei, J. e Wasserman, L. (2019). Least ambiguous set-valued classifiers with bounded error levels. *Journal of the American Statistical Association*, **114**(525), 223–234.
- Schwert, G. W. (1989). Why does stock market volatility change over time? *Journal of Finance*, **44**, 1115–1153.
- Shafer, G. e Vovk, V. (2008). A tutorial on conformal prediction. *Journal of Machine Learning Research*, **9**(3).
- Shah, D., Isah, H. e Zulkernine, F. (2019). Stock market analysis: A review and taxonomy of prediction techniques. *International Journal of Financial Studies*, **7**(2), 26.
- Sharpe, W. F. (1964). Capital asset prices: A theory of market equilibrium under conditions of risk. *Journal of Finance*, **19**, 425–442.
- Shiller, R. J. (2000). Irrational exuberance. *Princeton University Press*.
- Tegner, M. e Roberts, S. (2021). A bayesian take on option pricing with gaussian processes. *arXiv preprint arXiv:2112.03718*.
- Tversky, A. e Kahneman, D. (1974). Judgment under uncertainty: Heuristics and biases. *Science*, **185**, 1124–1131.
- Vogel, R. M. e Shallcross, A. L. (1996). The moving blocks bootstrap versus parametric time series models. *Water Resources Research*, **32**(6), 1875–1882.
- Vovk, V. (2005). The predictive model of conformal inference. *Machine Learning*, **59**, 225–268.
- Vovk, V., Gammerman, A. e Shafer, G. (2005). *Algorithmic learning in a random world*, volume 29. Springer.

- Vovk, V., Nouretdinov, I. e Gammernan, A. (2009). On-line predictive linear regression. *The Annals of Statistics*, páginas 1566–1590.
- Wisniewski, W., Lindsay, D. e Lindsay, S. (2020). Application of conformal prediction interval estimations to market makers net positions. Em *Conformal and Probabilistic Prediction and Applications*, páginas 285–301. PMLR.
- Xu, C. e Xie, Y. (2021). Conformal prediction interval for dynamic time-series. Em *International Conference on Machine Learning*, páginas 11559–11569. PMLR.
- Zaffran, M., Féron, O., Goude, Y., Josse, J. e Dieuleveut, A. (2022). Adaptive conformal predictions for time series. Em *International Conference on Machine Learning*, páginas 25834–25866. PMLR.