

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO DE CIÊNCIAS JURÍDICAS E ECONÔMICAS
PROGRAMA DE PÓS-GRADUAÇÃO EM ECONOMIA

ALBERSON DA SILVA MIRANDA

MÉTODOS DE MACHINE LEARNING PARA
RECONCILIAÇÃO ÓTIMA DE SÉRIES
TEMPORAIS HIERÁRQUICAS E AGRUPADAS
DE INSTITUIÇÕES FINANCEIRAS

VITÓRIA

2024

ALBERSON DA SILVA MIRANDA

MÉTODOS DE MACHINE LEARNING PARA
RECONCILIAÇÃO ÓTIMA DE SÉRIES TEMPORAIS
HIERÁRQUICAS E AGRUPADAS DE INSTITUIÇÕES
FINANCEIRAS

Dissertação apresentada ao Programa de Pós-Graduação em Economia da Universidade Federal do Espírito Santo, como requisito para a obtenção do título de Mestre em Economia.

Orientador: Prof. Dr. Guilherme A. de A. Pereira

VITÓRIA

2024

ALBERSON DA SILVA MIRANDA

MÉTODOS DE MACHINE LEARNING PARA RECONCILIAÇÃO ÓTIMA DE
SÉRIES TEMPORAIS HIERÁRQUICAS E AGRUPADAS DE INSTITUIÇÕES FINAN-
CEIRAS/ ALBERSON DA SILVA MIRANDA. – VITÓRIA, 2024-

67p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Dr. Guilherme A. de A. Pereira

Dissertação (Mestrado) – UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO DE CIÊNCIAS JURÍDICAS E ECONÔMICAS
PROGRAMA DE PÓS-GRADUAÇÃO EM ECONOMIA, 2024.

1. Economia Bancária. 2. Séries Temporais Hierárquicas. 3. Reconciliação Ótima. 4.
Machine Learning. I. Pereira, Guilherme Armando de Almeida. II. Universidade Federal
do Espírito Santo. III. Centro de Ciências Jurídicas e Econômicas. IV. Título

ALBERSON DA SILVA MIRANDA

MÉTODOS DE MACHINE LEARNING PARA
RECONCILIAÇÃO ÓTIMA DE SÉRIES TEMPORAIS
HIERÁRQUICAS E AGRUPADAS DE INSTITUIÇÕES
FINANCEIRAS

Dissertação apresentada ao Programa de Pós-Graduação em Economia da Universidade Federal do Espírito Santo, como requisito para a obtenção do título de Mestre em Economia.

Aprovada em 29 de fevereiro de 2024.

COMISSÃO EXAMINADORA

Prof. Dr. Guilherme A. de A. Pereira
Universidade Federal do Espírito Santo
Orientador

Prof. Dr. Edson Zambon Monte
Universidade Federal do Espírito Santo

Prof. Dr. Fernando Luiz Cyrino Oliveira
Pontifícia Universidade Católica do Rio de Janeiro

RESUMO

Neste estudo, foram conduzidos experimentos de reconciliação ótima de séries temporais e agrupadas para aprimorar a precisão preditiva dos saldos de empréstimos e financiamentos do Banco do Estado do Espírito Santo. Além dos métodos analíticos tradicionais, a investigação explorou abordagens de *machine learning*, incluindo floresta aleatória, *gradient boosting*, *elastic net* e *support vector machines*. A partir da metodologia delineada por [Spiliotis et al. \(2021\)](#), o trabalho propôs duas estratégias alternativas para a reconciliação ótima baseada em *machine learning*. Os resultados revelaram que a escolha do método e estratégia depende do nível hierárquico, onde a combinação correta pode exibir até 91% de ganho de performance nos níveis mais agregados. No entanto, para níveis inferiores, os métodos analíticos, destacadamente o MinT-*Shrink*, foram mais eficazes.

Palavras-chave: Economia Bancária. Séries Temporais Hierárquicas. Reconciliação Ótima. *Machine Learning*.

ABSTRACT

In this research, optimal reconciliation experiments were conducted to enhance the predictive accuracy of loan balances at the Bank of the State of Espírito Santo, focusing on hierarchical and grouped time series. In addition to conventional analytical methods, our investigation delved into machine learning approaches, encompassing random forest, gradient boosting, elastic net, and support vector machines. Following the methodology outlined by [Spiliotis et al. \(2021\)](#), our work proposed two alternative strategies for optimal reconciliation through machine learning methods. The outcomes underscored the significance of tailoring the method and strategy based on the hierarchical level. The right combination exhibited a remarkable performance improvement of up to 91% at the most aggregated levels. Notably, for lower levels, analytical methods, particularly MinT-Shrink, proved to be more effective.

Keywords: Economics of Banking. Hierarchical Time-Series. Optimal Reconciliation. Machine Learning.

LISTA DE FIGURAS

Figura 1 – Séries Hierárquicas	24
Figura 2 – Séries Agrupadas	25
Figura 3 – Séries Hierárquicas Agrupadas (a)	26
Figura 4 – Séries Hierárquicas Agrupadas (b)	26
Figura 5 – Abordagem de reconciliação ótima por ML. Fonte: Spiliotis et al. (2021). . .	32
Figura 6 – O Estado do Espírito Santo e suas meso e microrregiões.	34
Figura 7 – Série temporal do agregado de crédito do Banestes no ES	36
Figura 8 – Séries temporais do agregado de crédito do Banestes por mesorregião do ES	36
Figura 9 – Séries temporais do agregado de crédito do Banestes por microrregião do ES	37
Figura 10 – Séries temporais dos verbetes no agregado do ES	37
Figura 11 – Séries temporais dos verbetes por mesorregião do ES	38
Figura 12 – Previsões base a serem reconciliadas. Em azul, o conjunto de treino e, em verde, as previsões h passos à frente para fora da amostra.	38
Figura 13 – Esquema de modelagem <i>rolling forecast</i>	39
Figura 14 – Crescimento de árvores em algoritmos de <i>boosting</i>	59
Figura 15 – <i>Support vector classifiers</i> . Fonte: Hastie, Tibshirani e Friedman (2009). . . .	61

LISTA DE TABELAS

Tabela 1 – Microrregiões por mesorregião	35
Tabela 2 – Municípios por microrregião	35
Tabela 3 – Conjunto de dados para predição dos modelos de ML	40
Tabela 4 – Resultados Estban: Acurácia dos métodos analíticos de reconciliação	43
Tabela 5 – Resultados Estban: Acurácia dos métodos de ML de reconciliação. Estratégia rolling forecast.	43
Tabela 6 – Resultados Estban: Acurácia dos métodos de ML de reconciliação. Estratégia fitted base.	44
Tabela 7 – Resultados Estban: Acurácia dos métodos de ML de reconciliação. Estratégia reduced fitted base.	44
Tabela 8 – Resultados Estban: Tempo de processamento dos métodos de ML (em horas)	44
Tabela 9 – Intervalos de hiperparâmetros para {xgboost}	63
Tabela 10 – Intervalos de hiperparâmetros para {lightgbm}	63
Tabela 11 – Intervalos de hiperparâmetros para {ranger}	63
Tabela 12 – Intervalos de hiperparâmetros para {e1071} (svm)	64
Tabela 13 – Intervalos de hiperparâmetros para {glmnet}	64
Tabela 14 – Resultados Tourism: Acurácia dos métodos analíticos de reconciliação . . .	65
Tabela 15 – Resultados Tourism: Acurácia dos métodos de ML de reconciliação. Estratégia rolling forecast.	66
Tabela 16 – Resultados Tourism: Acurácia dos métodos de ML de reconciliação. Estratégia fitted base.	66
Tabela 17 – Resultados Tourism: Acurácia dos métodos de ML de reconciliação. Estratégia reduced fitted base.	67
Tabela 18 – Resultados Tourism: Tempo de processamento dos métodos de ML (em horas)	67

LISTA DE ABREVIATURAS E SIGLAS

ETS	<i>Exponential Smoothing</i>
Favar	<i>Factor Augmented Vector Autoregression</i>
Lasso	<i>Least Absolute Shrinkage and Selection Operator</i>
MCRL	Modelo Clássico de Regressão Linear
MASE	<i>Mean Absolute Scaled Error</i>
MinT	<i>Minimum Trace</i>
MQGF	Mínimos Quadrados Generalizados Factíveis
MQO	Mínimos Quadrados Ordinários
MQP	Mínimos Quadrados Ponderados
PIB	Produto Interno Bruto
RMSSE	<i>Root Mean Squared Scaled Error</i>
SFN	Sistema Financeiro Nacional
SVR	<i>Support Vector Regression</i>
XGBoost	<i>Extreme Gradient Boosting</i>

LISTA DE SÍMBOLOS

t	Tempo dentro da amostra
T	Último tempo dentro da amostra
h	Horizonte de previsão, tempo fora da amostra
Ω	Conjunto de dados dentro da amostra
y	Série temporal dentro da amostra
\hat{y}	Série temporal estimada
\tilde{y}	Série temporal reconciliada
n	Número de séries na hierarquia
m	Número de séries no menor nível da hierarquia
k	Número de níveis na hierarquia
\mathbf{S}	Matriz de soma
\mathbf{G}	Matriz de reconciliação
$\{\dots\}$	Conjunto
$ \{\dots\} $	Cardinalidade de um conjunto

SUMÁRIO

1	INTRODUÇÃO	19
1.1	Previsão de saldos de crédito de instituições financeiras	19
1.2	Previsão de séries temporais hierárquicas	20
2	REVISÃO DE LITERATURA	22
2.1	Previsão de saldos de crédito de instituições financeiras	22
2.2	Reconciliação ótima de séries temporais hierárquicas	23
3	RECONCILIAÇÃO DE SÉRIES TEMPORAIS HIERÁRQUICAS E AGRUPADAS	24
3.1	Notação algébrica	24
3.2	Abordagens top-down, bottom-up e middle-out	26
3.3	Coerência e reconciliação	30
4	METODOLOGIA	33
4.1	Dados e variáveis	33
4.2	Análise exploratória dos dados	34
4.3	Previsões base	35
4.4	Design do experimento	39
4.5	Otimização de hiperparâmetros	41
5	RESULTADOS	42
6	CONCLUSÃO	45
	REFERÊNCIAS	47
	APÊNDICES	51
	APÊNDICE A – DEMONSTRAÇÕES	53
	APÊNDICE B – MÉTODOS DE MACHINE LEARNING	57
1	Elastic Net	57
2	Gradient Boosting	58
3	Random Forest	58
4	Support Vector Machines	59
	APÊNDICE C – CONJUNTO DE HIPERPARÂMETROS	63
	APÊNDICE D – RESULTADOS PARA A BASE DE DADOS TOURISM	65

1 INTRODUÇÃO

1.1 Previsão de saldos de crédito de instituições financeiras

Embora no séc. XX ainda houvesse espaço para uma gestão guiada apenas por instinto (WALLANDER, 1999), atualmente é impensável um banco não realizar previsões de seus resultados e comunicar suas expectativas ao mercado. Nesse documento, ou *guidance*, a projeção da carteira de crédito — o total de empréstimos e financiamentos, dentre outros itens — é frequentemente a primeira informação fornecida. Juntamente com as projeções de depósitos, provisões para créditos de liquidação duvidosa, eficiência operacional, entre outros indicadores-chave, essas projeções determinam a temperatura das expectativas da instituição, e isso é essencial para os acionistas e investidores. Essas projeções precisam ser tão precisas quanto possível para que se possa calcular o risco de transacionar com a instituição financeira.

Ainda que não existam penalidades específicas para instituições financeiras que erram (por uma boa margem) em suas projeções, elas podem sofrer consequências negativas em outros aspectos, como na avaliação de seus desempenhos por parte dos investidores e clientes. Estes podem considerar as projeções equivocadas como um sinal de falta de competência ou confiança na instituição financeira, o que pode afetar negativamente a reputação e a imagem da instituição.

Além disso, nos casos em que algum grupo se sentir lesado, os bancos podem enfrentar ações judiciais se suas projeções forem consideradas enganosas ou fraudulentas. Por exemplo, se uma instituição financeira fizer projeções excessivamente otimistas para incentivar os investidores a comprar seus títulos e, posteriormente, as projeções se mostrarem incorretas, ela pode ser acusada de fraude¹ ou, ao menos, gestão temerária² — ambos caracterizados como crime contra o Sistema Financeiro Nacional (SFN).

Por isso, é importante que as instituições financeiras sejam transparentes e precisas em suas projeções, fornecendo informações confiáveis e atualizadas para seus clientes e investidores. No entanto, há também motivações estratégicas para essa atividade. Beccalli et al. (2015) mostraram que, em uma amostra de 55 bancos europeus, a utilização de *guidance* está associada a um aumento de 15% na probabilidade do banco atingir ou superar as expectativas de mercado. Isso, por sua vez, está associado a um incremento de até 5% no retorno por ação em relação aos bancos que não alcançaram ou superaram as expectativas.

¹ Art. 3º: Divulgar informação falsa ou prejudicialmente incompleta sobre instituição financeira. Pena: Reclusão, de 2 (dois) a 6 (seis) anos, e multa. Art. 4º: Gerir fraudulentamente instituição financeira. Pena: Reclusão, de 3 (três) a 12 (doze) anos, e multa (BRASIL, 1986).

² Art. 4º, parágrafo único: Se a gestão é temerária: Pena: Reclusão, de 2 (dois) a 8 (oito) anos, e multa (BRASIL, 1986).

1.2 Previsão de séries temporais hierárquicas

No que concerne a elaboração dessas previsões, os bancos, assim como em diversas outras indústrias, se enquadram em uma categoria de negócio que requer previsões de múltiplas séries temporais correlacionadas que são resultados de agregação. Por exemplo, o total de empréstimos de uma instituição financeira corresponde ao agregado dos empréstimos de cada uma de suas agências; o total de vendas de uma rede nacional de farmácias corresponde ao agregado de vendas de suas unidades em cada estado; o total da produção de uma petrolífrica multinacional corresponde ao total produzido em cada país por cada uma de suas plataformas. A essas estruturas naturais de agregação dá-se o nome de *séries temporais hierárquicas*.

Pode-se realizar previsões individualmente para todos os níveis da estrutura. No caso de uma instituição financeira, isso significa realizar previsões, por exemplo, para cada agência, para o agregado de cada região e para o agregado da instituição. Infelizmente, não há qualquer razão, exceto para métodos de previsão muito simples, para que essas previsões sejam *coerentes* (i.e. que a soma das previsões individuais seja igual à previsão do agregado). Além disso, realizar as previsões individualmente ignoraria os relacionamentos existentes entre as séries temporais na estrutura. Para fazer com que essas previsões se tornem coerentes entre si é que foram desenvolvidos os chamados métodos de *reconciliação*, sendo os mais simples o *top-down*, *bottom-up* e uma combinação das duas, o *middle-out*.

A prática usual em *budgeting*³, principalmente para empresas com muitas filiais, é a *top-down*, ou seja, realizar previsões para o total e então distribuí-las para cada unidade seguindo alguma lógica proporcional. No caso dos bancos de varejo, com muitas agências espalhadas pelo território, esse método pode ser muito útil. Na prática, em um país continental como o Brasil, a quantidade de filiais para as quais se deve realizar previsões pode tornar a tarefa desafiadora mesmo para instituições regionais.

Esse é o caso do Banestes. Com 134 agências distribuídas pelos 78 municípios capixabas, realizar o *budgeting* para R\$ 5,5 bi de faturamento⁴ não é uma tarefa trivial. Além de uma estrutura hierárquica de alta dimensionalidade por conta da quantidade de agências, se tratando de um banco múltiplo⁵ que opera com diversas carteiras, as n modalidades de crédito⁶ expandem a estrutura para um total de $n \times 134$ séries temporais a serem estimadas.

Dada tal complexidade, a abordagem *top-down* se coloca como uma opção viável em termos de tempo de processamento e análise. No entanto, conforme descemos na hierarquia,

³ O orçamento é um documento no qual é definido o planejamento financeiro de um empresa, geralmente para o ano seguinte, estabelecendo metas e objetivos. Nele, são projetadas as expectativas da empresa e é base de comparação para saber como os resultados estão se desviando da performance esperada.

⁴ Conforme demonstrativos publicados referentes ao exercício de 2022 ([BANCO DO ESTADO DO ESPÍRITO SANTO, 2022](#)).

⁵ Para ser classificado como banco múltiplo, a instituição financeira deve operar com, no mínimo, duas carteiras dentre: comercial; investimento ou desenvolvimento; crédito imobiliário; de crédito, financiamento e investimento, e; arrendamento mercantil ([CONSELHO MONETÁRIO NACIONAL, 1994](#)).

⁶ Crédito consignado, rural, imobiliário, pessoal, capital de giro, desconto de títulos etc.

menos precisa ela se torna e, além disso, as características individuais das séries temporais do menor nível hierárquicos são ignoradas.

Tomando o caminho inverso, a abordagem *bottom-up* consiste em realizar previsões para cada série temporal individualmente e, então, agregá-las para obter a previsão para o total. Essa abordagem pode ser mais precisa, pois leva em consideração as características individuais de cada série temporal do nível mais desagregado. No entanto, ela é mais custosa em termos de tempo de processamento e análise. Nesse sentido, cabe ao analista avaliar o *trade-off* entre os ganhos de precisão percebidos com a geração de previsões individuais e a economia de tempo e processamento em realizar o contrário (GROSS; SOHL, 1990).

Além disso, ambas são abordagens de nível único, isto é, são realizadas as previsões para um único nível e então os demais níveis são obtidos agregando ou desagregando. O problema com esses tipos de abordagem é que elas utilizam informação incompleta (HYNDMAN; ATHANASOPOULOS, 2021). Por exemplo, suponha-se que se escolha estimar modelos para cada uma das 134 agências e agregá-las (*bottom-up*). Nesse caso, ignora-se a influência que os níveis mais agregados — aqui a carteira de crédito da região ou de todo o estado — podem ter na estimação do saldo de crédito de cada agência. Por outro lado, se escolher estimar modelos para o nível mais agregado (*top-down*), ignora-se a informação individual de cada agência.

Uma terceira possibilidade é a *reconciliação ótima*. Ela é uma abordagem que busca resolver esse problema e consiste em realizar previsões para todos os níveis hierárquicos e, então, estimar um modelo para reescrever as previsões do nível mais desagregado como uma combinação linear de todos os elementos da hierarquia. Obtidas as novas previsões no menor nível, elas são então agregadas, gerando previsões coerentes nos níveis superiores. Dessa forma, a informação de todos os níveis é utilizada na estimação dos modelos e na geração das previsões, ao mesmo tempo em que a variância do erro de previsão é minimizada (HYNDMAN; AHMED et al., 2011).

Atualmente, os métodos analíticos baseados na minimização do traço (MinT) da matriz da variância-covariância dos erros, desenvolvidos em Wickramasuriya, Athanasopoulos e Hyndman (2019), são os mais populares na literatura da reconciliação ótima. Esses métodos divergem apenas na forma da qual se dá o relacionamento entre os diferentes elementos da hierarquia, resultando numa estimação por mínimos quadrados ordinários (MQO), mínimos quadrados ponderados (MQP) ou mínimos quadrados generalizados (MQG)⁷.

Entretanto, tais métodos são sujeitos a uma série de restrições, como as do modelo clássico de regressão linear (MCRL), e têm sua capacidade preditiva reduzida quando suas hipóteses são violadas. Além disso, esses métodos requerem que *toda* a informação seja utilizada, mesmo

⁷ Se os erros de previsão são descorrelacionados e homoscedásticos ao longo de toda estrutura (MQO), o que é impossível em séries temporais hierárquicas; se os erros são descorrelacionados e homoscedásticos apenas dentro do mesmo nível hierárquico (MQP estrutural); se os erros são descorrelacionados mas ponderados pela variância da série (MQP); ou se são correlacionados e variantes ao longo de toda a estrutura (MQG, também chamados de *MinT Sample* e *MinT Shrink*).

aquelas que eventualmente não sejam relevantes para aquela previsão. Nesse sentido, métodos de *machine learning* são mais gerais, de modo a permitir parâmetros não lineares e poderem aproximar virtualmente qualquer função, além de incluírem parâmetros de regularização, de forma que não necessariamente utilizam toda a informação disponível. Espera-se, portanto, que esses métodos alcancem melhor performance no problema da reconciliação ótima, justificando a pesquisa e atenção ao tema. Nessa direção, [Spiliotis et al. \(2021\)](#) propuseram uma metodologia para reconciliação ótima de séries temporais hierárquicas utilizando métodos de *machine learning*, especificamente o *gradient boosting* e floresta aleatória, obtendo resultados superiores aos métodos analíticos.

Tomando como ponto de partida a metodologia proposta por [Spiliotis et al. \(2021\)](#), este trabalho estende seu uso para outros métodos de *machine learning* na tarefa de reconciliação ótima, especificamente os métodos de regressão regularizada lasso, ridge e *elastic net*, e *support vector machines* (SVM), além de outra implementação de *gradient boosting*. Paralelamente, propõe e avalia estratégias alternativas na metodologia de [Spiliotis et al. \(2021\)](#), verificando se sua performance se mantém superior a dos métodos analíticos em um contexto de séries temporais financeiras de alta dimensionalidade.

Esta dissertação está organizada da seguinte forma: no Capítulo 2, é realizada uma breve revisão de literatura sobre o tema. No Capítulo 3, é introduzida a formalização algébrica necessária para a compreensão dos métodos de reconciliação. No Capítulo 4, são apresentados os dados e a metodologia do experimento. No Capítulo 5, os resultados são apresentados e discutidos.

2 REVISÃO DE LITERATURA

2.1 Previsão de saldos de crédito de instituições financeiras

A nível macroeconômico, a previsão do agregado de crédito das instituições financeiras é uma preocupação de bancos centrais ao redor do mundo. No Brasil, [Bader, Koyama, Sérgio Mikio e Tsuchida, Marcos Hiroyuki \(2014\)](#) aprimoram o método FAVAR (*Factor Augmented Vector Autoregression*) com uma etapa de análise de correlação canônica para identificar as melhores, em termos de correlação com as variáveis de crédito do SFN, combinações lineares de componentes principais. [Çolak et al. \(2019\)](#) produzem, a partir de séries filtradas do agregado de crédito, indicadores para monitoramento de períodos de expansão e desaceleração de crédito no setor bancário turco.

Já para níveis abaixo do agregado de crédito, poucos trabalhos foram encontrados. Tangenciando o tema da previsão de saldos de crédito, outros tópicos da economia bancária foram objeto de estudo para previsão de séries temporais. [Sezer, Gudelek e Ozbayoglu \(2019\)](#) produziram revisão de literatura de trabalhos que realizaram previsão de séries temporais financeiras

utilizando *deep learning*. [Gorodetskaya, Gobareva e Koroteev \(2021\)](#) propõem o que chamaram de “uma metodologia universal” para aplicação automática de *machine learning* na previsão de séries temporais do setor bancário. Entretanto, nenhum dos trabalhos combinaram estruturas hierárquicas com *machine learning*.

No que diz respeito à previsão de séries temporais em largas hierarquias, [Prayoga, Suhartono e Rahayu \(2017\)](#) trabalharam na previsão do fluxo de caixa do Banco da Indonésia, utilizando uma hierarquia de 3 níveis. Porém, utilizaram apenas a abordagem *top-down* para reconciliação. Inversamente, [Li et al. \(2016\)](#) compararam dois métodos de *machine learning* para previsão da produção de energia solar no estado da Flórida/EUA, redes neurais artificiais e SVM, em uma abordagem *bottom-up*, caracterizando ambos trabalhos como de nível único.

2.2 Reconciliação ótima de séries temporais hierárquicas

Previsões pontuais de séries temporais hierárquicas não é um assunto novo. Ao menos desde a década de 70, pesquisas foram publicadas acerca de abordagens *bottom-up* e *top-down*, suas vantagens e desvantagens, e tentativas de se definir qual é o melhor método⁸. Entretanto, é apenas em [Hyndman, Ahmed et al. \(2011\)](#) que é formalizada uma abordagem prática, via MQO, que utiliza toda a informação disponível.

[Hyndman, Lee e Wang \(2016\)](#) tentam aperfeiçoar o método usando as variâncias das previsões individuais estimadas (dentro da amostra) como estimativa para a matriz de variância-covariância dos erros de reconciliação, de forma a as utilizar como pesos e realizar a reconciliação ótima por mínimos quadrados ponderados (MQP).

[Wickramasuriya, Athanasopoulos e Hyndman \(2019\)](#) argumentam que o que de fato interessa é que as previsões reconciliadas tenham o menor erro. Então, corrigem a abordagem de reconciliação ótima para o objetivo de minimização dos erros das previsões reconciliadas $\tilde{\mathbf{y}}_{t+h}$, ao invés dos erros das previsões individuais $\hat{\mathbf{y}}_{t+h}$. Dado que isso implica na minimização da variância de $\tilde{\mathbf{e}}_{t+h}$, ou seja, na minimização do somatório da diagonal, o traço, da matriz de variância-covariância de $\tilde{\mathbf{e}}_{t+h}$, eles chamaram esse método de Traço Mínimo (MinT, na sigla em inglês). Paralelamente, usam desigualdade triangular para demonstrar que as previsões reconciliadas obtidas por esse método são ao menos tão boas quanto as previsões individuais.

[Panagiotelis et al. \(2021\)](#) reinterpretem a literatura de coerência e reconciliação de previsões pontuais a partir de uma abordagem geométrica, trazendo provas alternativas para conclusões anteriores ao mesmo tempo em que fornece novos teoremas. Além disso, os autores estendem essa interpretação geométrica para o contexto probabilístico, fornecendo métodos paramétricos e não paramétricos (via *bootstrapping*) para reconciliação de previsões probabilísticas.

[Spiliotis et al. \(2021\)](#) propõem a utilização de *machine learning* para a reconciliação

⁸ Uma revisão dessa literatura pode ser encontrada em [Athanasopoulos, Ahmed e Hyndman \(2009\)](#).

ótima de séries temporais, especificamente os métodos de floresta aleatória e *gradient boosting*. Os autores descrevem como vantagens desse método em relação aos anteriores a descrição de relacionamentos não lineares, performance preditiva e a desnecessidade da utilização de todos os elementos da hierarquia na combinação ótima. Para o conjunto de dados utilizados, os autores afirmam que os métodos de *machine learning*, especialmente o *XGBoost*, alcançaram, em média, melhor performance que as abordagens de nível único e o MinT.

3 RECONCILIAÇÃO DE SÉRIES TEMPORAIS HIERÁRQUICAS E AGRUPADAS

3.1 Notação algébrica

Séries temporais hierárquicas são aquelas que podem ser agregadas ou desagregadas naturalmente em uma estrutura aninhada (HYNDMAN; ATHANASOPOULOS, 2021). Para ilustrar, tome a série do PIB de um país fictício com três estados, cada um com dois municípios. Essa série pode ser desagregada por estado que, por sua vez, pode ser desagregada por município (Figura 1).

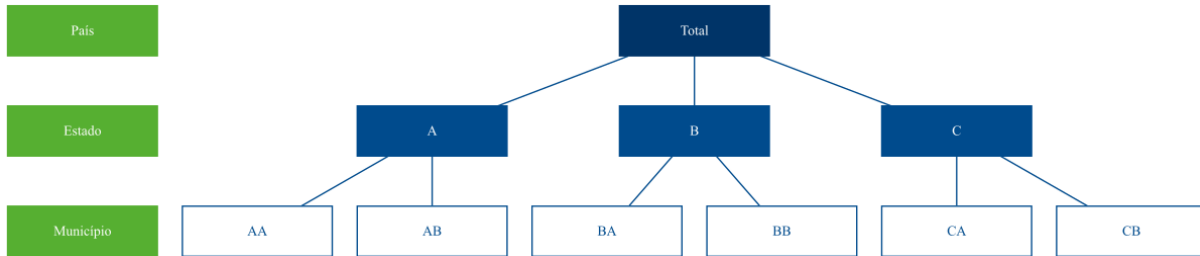


Figura 1 – Séries Hierárquicas

Essa estrutura pode ser representada através de equações para qualquer nível de agregação. Dessa forma, o agregado nacional pode ser descrito pelos agregados dos estados, Equação (1), ou como o agregado dos municípios, Equação (2). Já o agregado para o estado A é representado pela Equação (3).

$$y_t = y_{A,t} + y_{B,t} + y_{C,t} \quad (1)$$

$$y_t = y_{AA,t} + y_{AB,t} + y_{BA,t} + y_{BB,t} + y_{CA,t} + y_{CB,t} \quad (2)$$

$$y_{A,t} = y_{AA,t} + y_{AB,t} \quad (3)$$

Alternativamente, podemos descrever a estrutura completa de forma matricial:

$$\begin{bmatrix} y_t \\ y_{A,t} \\ y_{B,t} \\ y_{C,t} \\ y_{AA,t} \\ y_{AB,t} \\ y_{BA,t} \\ y_{BB,t} \\ y_{CA,t} \\ y_{CB,t} \end{bmatrix}_{10 \times 1} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{10 \times 6} \begin{bmatrix} y_{AA,t} \\ y_{AB,t} \\ y_{BA,t} \\ y_{BB,t} \\ y_{CA,t} \\ y_{CB,t} \end{bmatrix}_{6 \times 1} \quad (4)$$

Uma outra forma de desagregarmos o PIB é por atividade econômica — agricultura, indústrias extrativas, indústria de transformação, eletricidade e gás, construção etc. Essa estrutura não pode ser desagregada naturalmente de uma única maneira, como é a hierarquia de estados e municípios. Não pode ser aninhada por um atributo como a própria geografia. A esse tipo de estrutura dá-se o nome de *séries agrupadas*.

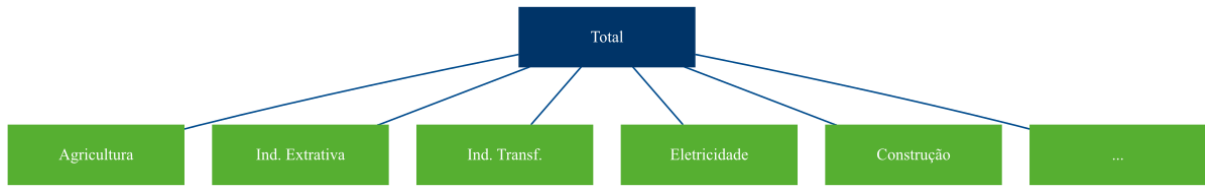


Figura 2 – Séries Agrupadas

Combinando as duas, temos a estrutura de séries hierárquicas agrupadas. Ao contrário da estrutura hierárquica, que só pode ser agregada de uma forma, como com os municípios abaixo dos estados⁹, a adição da estrutura agrupada pode ocorrer tanto acima (Figura 3) quanto abaixo (Figura 4) da hierárquica.

Na notação matricial, a estrutura da Figura 4 é representada como abaixo. Formalmente, o primeiro membro da igualdade é composto pelo vetor y_t n -dimensional com todas as observações no tempo t para todos os níveis da hierarquia. O segundo membro é composto pela matriz de soma S de dimensão $n \times m$ que define as equações para todo nível de agregação, e pelo vetor b_t composta pelas séries no nível mais desagregado.

$$y_t = S b_t \quad (5)$$

⁹ Essa estrutura é única no sentido que o somatório dos municípios totaliza o estado, mas não se pode somar estados para totalizar um município. Outro exemplo de estrutura hierárquica é a série de vendas de uma empresa: pode-se agregar as vendas de cada filial para obter o total, mas não o contrário.

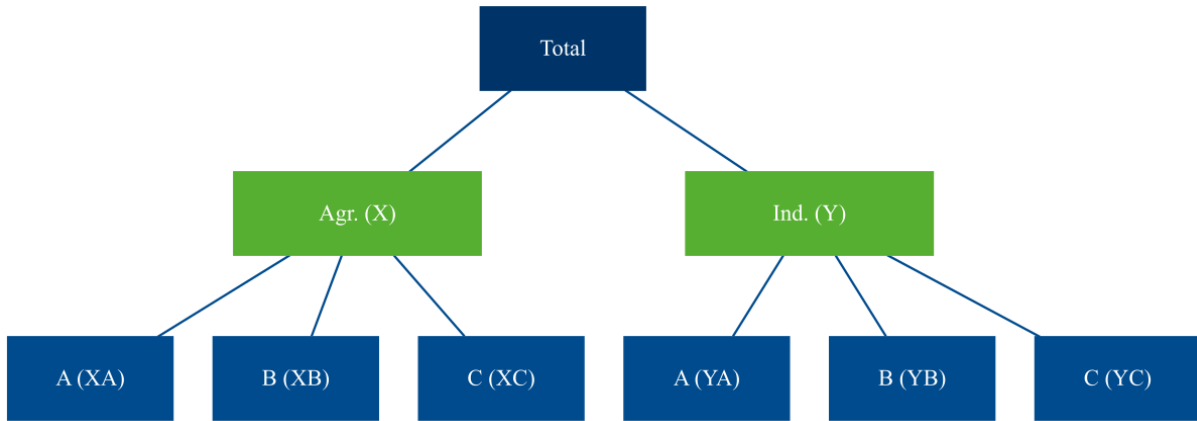


Figura 3 – Séries Hierárquicas Agrupadas (a)

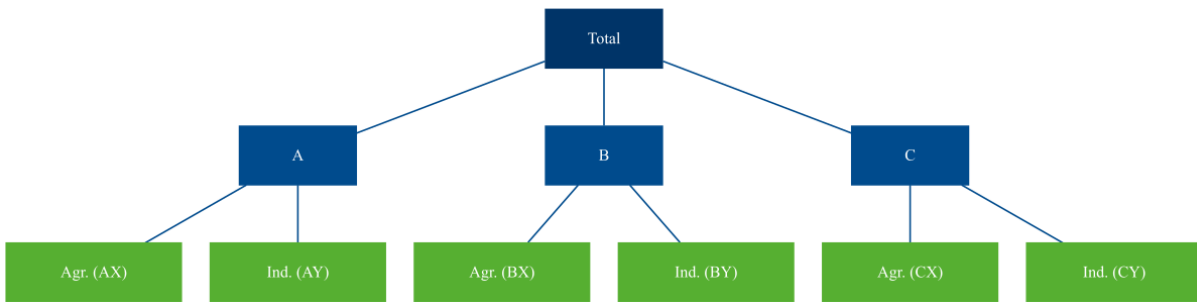


Figura 4 – Séries Hierárquicas Agrupadas (b)

$$\begin{bmatrix} y_t \\ y_{A,t} \\ y_{B,t} \\ y_{C,t} \\ y_{X,t} \\ y_{Y,t} \\ y_{AX,t} \\ y_{AY,t} \\ y_{BX,t} \\ y_{BY,t} \\ y_{CX,t} \\ y_{CY,t} \end{bmatrix}_{12 \times 1} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{12 \times 6} \begin{bmatrix} y_{AX,t} \\ y_{AY,t} \\ y_{BX,t} \\ y_{BY,t} \\ y_{CX,t} \\ y_{CY,t} \end{bmatrix}_{6 \times 1} \quad (6)$$

3.2 Abordagens top-down, bottom-up e middle-out

Talvez as formas mais intuitivas de se pensar em previsões para esses tipos de estrutura sejam as abordagens top-down e bottom-up. Tome a estrutura descrita na Figura 1, por exemplo. Podemos realizar a previsão para o horizonte de tempo h do agregado do PIB, representado no topo da hierarquia por *Total* (Equação 7), e então distribuir os valores previstos proporcionalmente entre os estados e municípios.

$$\hat{\mathbf{y}}_{T+h|T} = E[\mathbf{y}_{T+h}|\Omega_T] \quad (7)$$

Essa é a abordagem top-down. Nela, a previsão para os níveis mais desagregados da hierarquia são determinadas por uma proporção p_i do nível agregado. Por exemplo, as previsões para o município AA são dadas pela Equação 8.

$$\tilde{\mathbf{y}}_{AA,T+h|T} = p_1 \hat{\mathbf{y}}_{T+h|T} \quad (8)$$

Para isso, temos de definir uma matriz com todos esses pesos, que, seguindo a formulação de [Hyndman e Athanasopoulos \(2021\)](#), chamamos de \mathbf{G} :

$$\mathbf{G} = \begin{bmatrix} p_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{6 \times 10} \quad (9)$$

\mathbf{G} é uma matriz $m \times n$ que multiplica o vetor $\hat{\mathbf{y}}_{T+h|T}$ que chamamos de *previsões base*, isto é, as previsões individuais para todos os níveis de agregação. A equação para a abordagem *top-down* será, então:

$$\tilde{\mathbf{y}}_{T+h|T} = \mathbf{S}\mathbf{G}\hat{\mathbf{y}}_{T+h|T} \quad (10)$$

Na notação matricial para a estrutura da Figura 1, temos:

$$\begin{bmatrix} \tilde{\mathbf{y}}_t \\ \tilde{\mathbf{y}}_{A,t} \\ \tilde{\mathbf{y}}_{B,t} \\ \tilde{\mathbf{y}}_{C,t} \\ \tilde{\mathbf{y}}_{AA,t} \\ \tilde{\mathbf{y}}_{AB,t} \\ \tilde{\mathbf{y}}_{BA,t} \\ \tilde{\mathbf{y}}_{BB,t} \\ \tilde{\mathbf{y}}_{CA,t} \\ \tilde{\mathbf{y}}_{CB,t} \end{bmatrix}_{10 \times 1} = \mathbf{S}_{10 \times 6} \begin{bmatrix} p_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{6 \times 10} \begin{bmatrix} \hat{\mathbf{y}}_{T+h|T} \\ \hat{\mathbf{y}}_{A,T+h|T} \\ \hat{\mathbf{y}}_{B,T+h|T} \\ \hat{\mathbf{y}}_{C,T+h|T} \\ \hat{\mathbf{y}}_{AA,T+h|T} \\ \hat{\mathbf{y}}_{AB,T+h|T} \\ \hat{\mathbf{y}}_{BA,T+h|T} \\ \hat{\mathbf{y}}_{BB,T+h|T} \\ \hat{\mathbf{y}}_{CA,T+h|T} \\ \hat{\mathbf{y}}_{CB,T+h|T} \end{bmatrix}_{10 \times 1} \quad (11)$$

Multiplicando as matrizes \mathbf{G} e $\hat{\mathbf{y}}$ obtemos o vetor \mathbf{b}_t ¹⁰, que contém as previsões para os níveis mais desagregados, agora definidas como proporção do total.

$$\begin{bmatrix} \tilde{y}_t \\ \tilde{y}_{A,t} \\ \tilde{y}_{B,t} \\ \tilde{y}_{C,t} \\ \tilde{y}_{AA,t} \\ \tilde{y}_{AB,t} \\ \tilde{y}_{BA,t} \\ \tilde{y}_{BB,t} \\ \tilde{y}_{CA,t} \\ \tilde{y}_{CB,t} \end{bmatrix}_{10 \times 1} = \mathbf{S}_{10 \times 6} \begin{bmatrix} p_1 \hat{y}_{T+h|T} \\ p_2 \hat{y}_{T+h|T} \\ p_3 \hat{y}_{T+h|T} \\ p_4 \hat{y}_{T+h|T} \\ p_5 \hat{y}_{T+h|T} \\ p_6 \hat{y}_{T+h|T} \end{bmatrix}_{6 \times 1} \quad (12)$$

Note que, por se tratar de um método *top-down*, é necessário apenas o primeiro elemento do vetor de previsões base, ou seja, a previsão do nível mais agregado (Equação 12). Sendo essa exatamente uma das vantagens do método *top-down*, na prática, podemos anular os demais elementos de $\hat{\mathbf{y}}$.

$$\begin{bmatrix} \tilde{y}_t \\ \tilde{y}_{A,t} \\ \tilde{y}_{B,t} \\ \tilde{y}_{C,t} \\ \tilde{y}_{AA,t} \\ \tilde{y}_{AB,t} \\ \tilde{y}_{BA,t} \\ \tilde{y}_{BB,t} \\ \tilde{y}_{CA,t} \\ \tilde{y}_{CB,t} \end{bmatrix}_{10 \times 1} = \mathbf{S}_{10 \times 6} \begin{bmatrix} p_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{6 \times 10} \begin{bmatrix} \hat{y}_{T+h|T} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}_{10 \times 1} \quad (13)$$

Substituindo a matriz \mathbf{S} , temos as equações que definem as previsões reconciliadas.

¹⁰ Ver Equação 5.

$$\begin{bmatrix} \tilde{y}_t \\ \tilde{y}_{A,t} \\ \tilde{y}_{B,t} \\ \tilde{y}_{C,t} \\ \tilde{y}_{AA,t} \\ \tilde{y}_{AB,t} \\ \tilde{y}_{BA,t} \\ \tilde{y}_{BB,t} \\ \tilde{y}_{CA,t} \\ \tilde{y}_{CB,t} \end{bmatrix}_{10 \times 1} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{10 \times 6} \begin{bmatrix} p_1 \hat{y}_{T+h|T} \\ p_2 \hat{y}_{T+h|T} \\ p_3 \hat{y}_{T+h|T} \\ p_4 \hat{y}_{T+h|T} \\ p_5 \hat{y}_{T+h|T} \\ p_6 \hat{y}_{T+h|T} \end{bmatrix}_{6 \times 1} \quad (14)$$

Já a abordagem bottom-up parte do raciocínio inverso e define as previsões de cada elemento da estrutura a partir das previsões dos elementos mais desagregados. Para tanto, basta modificar a matriz \mathbf{G} .

$$\mathbf{G} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{6 \times 10} \quad (15)$$

Portanto, \mathbf{G} define a abordagem — se *top-down* ou *bottom-up* —, e \mathbf{S} define a maneira da qual as previsões são somadas para formar as equações de previsão para cada elemento da estrutura. Portanto, chamamos \mathbf{G} de matriz de reconciliação.

$$\begin{bmatrix} \tilde{y}_t \\ \tilde{y}_{A,t} \\ \tilde{y}_{B,t} \\ \tilde{y}_{C,t} \\ \tilde{y}_{AA,t} \\ \tilde{y}_{AB,t} \\ \tilde{y}_{BA,t} \\ \tilde{y}_{BB,t} \\ \tilde{y}_{CA,t} \\ \tilde{y}_{CB,t} \end{bmatrix}_{10 \times 1} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{10 \times 6} \begin{bmatrix} \hat{y}_{AA,T+h|T} \\ \hat{y}_{AB,T+h|T} \\ \hat{y}_{BA,T+h|T} \\ \hat{y}_{BB,T+h|T} \\ \hat{y}_{CA,T+h|T} \\ \hat{y}_{CB,T+h|T} \end{bmatrix}_{6 \times 1} \quad (16)$$

3.3 Coerência e reconciliação

Seja somando as previsões do nível mais desagregado para formar os níveis superiores da hierarquia (*bottom-up*) ou distribuindo proporcionalmente as previsões do nível mais agregado (*top-down*), o vetor $\tilde{\mathbf{y}}_t$ representa as previsões *coerentes*. Isso significa que as previsões são totalizadas corretamente — as previsões de cada elemento agregado corresponde ao somatório das previsões dos níveis inferiores da hierarquia. Isso é garantido pela multiplicação das matrizes \mathbf{SG} .

Não fosse essa pré multiplicação, nada garantiria a coerência das previsões. Tomando a estrutura da Figura 1 como exemplo, seria um acaso improvável que as previsões do agregado para o estado do Espírito Santo fossem exatamente a soma das previsões individuais de seus municípios. Isso porque não há qualquer razão para que cada série siga o mesmo processo estocástico.

Os métodos de gerar previsões coerentes ($\tilde{\mathbf{y}}_t$) a partir de previsões base ($\hat{\mathbf{y}}_t$) são chamados de métodos de *reconciliação*. Os métodos de reconciliação tradicionais apresentados, *top-down* e *bottom-up*, utilizam informação limitada. No método *top-down*, utiliza-se apenas informações do nível mais agregado — por isso, apenas a primeira coluna em (Equação 9) é diferente de zero. Já na abordagem *bottom-up*, utiliza-se apenas as informações dos níveis mais desagregados, o que resulta na submatriz identidade $m \times m$ na Equação 15, enquanto as colunas que representam os níveis mais agregados são nulas.

Alternativamente, podemos pensar numa matriz \mathbf{G} qualquer que utilize toda a informação disponível e tenha algumas propriedades que garantam que as previsões coerentes tenham o menor erro o possível. Esse é o problema de pesquisa trabalhado na *reconciliação ótima*.

Os métodos analíticos de reconciliação ótima são aqueles que estimam a matriz de reconciliação, \mathbf{G} , através de regressão linear. Isso resulta na redefinição das previsões do nível mais desagregado como uma combinação linear¹¹ das previsões de todos os elementos de todos os níveis da hierarquia, utilizando, assim, toda a informação disponível.

O estado-da-arte para esse tipo de método é o *MinT* (WICKRAMASURIYA; ATHANASSOPOULOS; HYNDMAN, 2019). Nele, o objetivo é minimizar o erro das previsões reconciliadas:

$$\tilde{\mathbf{e}}_{t+h|t} = \mathbf{y}_{t+h} - \tilde{\mathbf{y}}_{t+h|t} \quad (17)$$

Essa equação pode ser reescrita como $\tilde{\mathbf{e}}_t = \mathbf{SG}\hat{\mathbf{e}}_t$, que tem variância dada por¹²

$$\text{Var}[\tilde{\mathbf{e}}] = \mathbf{SG}\hat{\mathbf{W}}_{t+h|t}\mathbf{G}'\mathbf{S}' \quad (18)$$

¹¹ Por essa razão, esses métodos são também chamados de métodos de combinação.

¹² Ver demonstração A.3.

em que $\hat{\mathbf{W}}_{t+h|t}$ é a matriz de variância-covariância dos erros de previsão base.

A abordagem consiste então em se obter um valor de \mathbf{G} que minimize o traço de $\text{Var}[\mathbf{y}_{t+h} - \tilde{\mathbf{y}}_{t+h|t}]$. Isso resultaria no melhor (variância mínima) estimador linear não viesado¹³ (WICKRAMASURIYA; ATHANASOPOULOS; HYNDMAN, 2019). Sua solução é dada por

$$\mathbf{G} = (\mathbf{S}' \hat{\mathbf{W}}_{t+h|t}^\dagger \mathbf{S})^{-1} \mathbf{S}' \hat{\mathbf{W}}_{t+h|t}^\dagger \quad (19)$$

em que $\hat{\mathbf{W}}_{t+h|t}^\dagger$ é a inversa generalizada de Moore-Penrose para $\hat{\mathbf{W}}_{t+h|t}$ ¹⁴. Essa formulação corresponde a um problema de regressão por mínimos quadrados generalizados, considerando \mathbf{S} como a matriz de preditores e \mathbf{G} os coeficientes a serem estimados. Consequentemente, as previsões ótimas reconciliadas são dadas por

$$\tilde{\mathbf{y}}_{t+h|t} = \mathbf{S}(\mathbf{S}' \hat{\mathbf{W}}_{t+h|t}^\dagger \mathbf{S})^{-1} \mathbf{S}' \hat{\mathbf{W}}_{t+h|t}^\dagger \hat{\mathbf{y}}_{t+h|t} \quad (20)$$

Spiliotis et al. (2021) propõem estimar essa combinação através de métodos de ML ao invés de MQG. Sua abordagem consiste nas etapas transcritas a seguir, quase *ipsis litteris*:

1. Seja k a quantidade de níveis da hierarquia, m a quantidade de séries e n o tamanho de cada série. As séries são divididas em uma série de conjuntos de treinamento e conjuntos de teste, com cada conjunto de treinamento compreendendo as primeiras $p < n$ observações (para $p = q, q+1, \dots, n-1$) e o conjunto de teste correspondente compreendendo apenas as observações no tempo $p+1$.
2. Um modelo de previsão é ajustado para cada série em cada conjunto de treinamento e previsões um passo à frente são produzidas para cada conjunto de teste.
3. Um modelo de ML separado¹⁵ é construído para prever cada uma das m_k séries inferiores da hierarquia. O conjunto de treinamento de cada modelo consiste em $n-p$ observações e $m+1$ variáveis. As primeiras m variáveis (usadas como preditores ou entradas) são as previsões um passo à frente produzidas durante o processo de *rolling origin* para as m séries da hierarquia, e a última variável (a variável explicada) é o valor real da série de nível inferior nos tempos correspondentes. A função de perda dos modelos é a soma dos erros quadráticos, e os hiperparâmetros dos modelos de ML são determinados arbitrariamente pelo usuário ou por meio de um procedimento de otimização.
4. A amostra completa das séries (todas as n observações) é usada para produzir previsões base de h passos à frente para as m séries da hierarquia, onde h é o horizonte de previsão de interesse.

¹³ A ausência de viés é garantida pela condição de ausência de viés em $\tilde{\mathbf{y}}$.

¹⁴ A necessidade da inversa generalizada aqui é trivial, uma vez que a inversa regular, do tipo \mathbf{A}^{-1} , requer matriz quadrada, o que não acontece no caso de séries temporais hierárquicas, uma vez que, necessariamente, tem-se $n > m$ ($n = m + \text{número de nós de agregação}$). Além disso, $\hat{\mathbf{W}}$ é posto incompleto (ver demonstração A.4).

¹⁵ Os autores trabalharam com *random forest* e *XGBoost*.

5. Os modelos m_k que foram construídos na 3ª etapa são usados para fornecer previsões para as séries do nível inferior da hierarquia, usando as previsões base produzidas na 4ª etapa como entrada. Esse processo é repetido h vezes, cada vez para um horizonte de previsão diferente.
6. As previsões produzidas pelos modelos de ML na 5ª etapa são agregadas (somadas) para que as previsões reconciliadas sejam produzidas para o restante dos níveis hierárquicos.

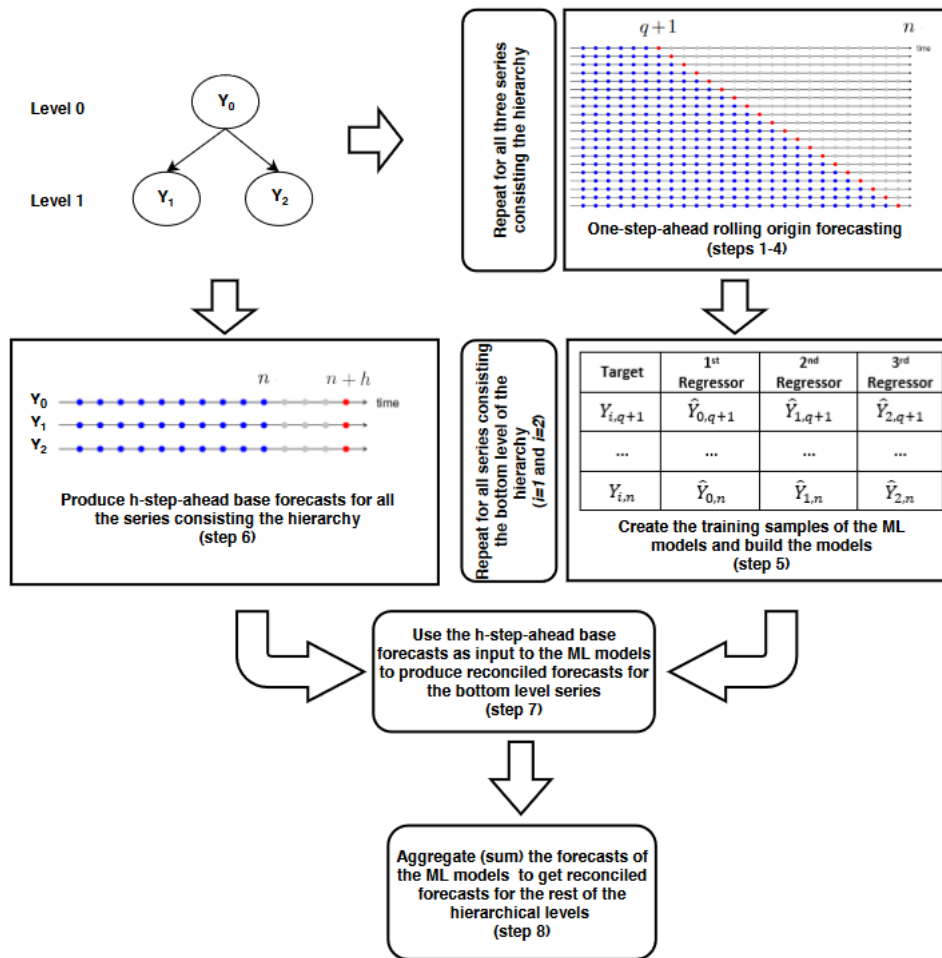


Figura 5 – Abordagem de reconciliação ótima por ML. Fonte: [Spiliotis et al. \(2021\)](#).

Isto posto, a presente dissertação estende a abordagem de [Spiliotis et al. \(2021\)](#) para a reconciliação de séries temporais hierárquicas e agrupadas; avalia a efetividade de mais 4 implementações de métodos de ML (*lasso*, *ridge*, *elastic net*, SVM e *lightGBM*), além dos 2 propostos por eles (*XGBoost* e *random forest*); e propõe variações metodológicas para as 3 primeiras etapas da metodologia, obtendo ganhos significativos de performance no contexto de séries temporais financeiras.

4 METODOLOGIA

Neste capítulo estão contidas explicações sobre os dados e variáveis, sobre o *design* da modelagem e sobre a avaliação dos modelos.

O *software* usado foi o R (R CORE TEAM, 2023). As previsões base e os métodos de reconciliação analíticos foram realizados com o pacote {fable} (O'HARA-WILD; HYNDMAN; WANG, 2023) e suas extensões. Já metodologia de *machine learning* (reamostragem, otimização de hiperparâmetros, treino e predição) foi executada com o pacote {mlr3} (LANG et al., 2024) e suas extensões. As implementações do XGBoost e LightGBM foram realizadas com seus pacotes homônimos {xgboost} (CHEN; HE et al., 2024) e {lightgbm} (SHI et al., 2024), do Elastic Net com o pacote {glmnet} (FRIEDMAN et al., 2023), a do Random Forest com o pacote {ranger} (WRIGHT; WAGER; PROBST, 2023) e a do SVM com o pacote {e1071} (R-e1071).

4.1 Dados e variáveis

Os dados usados nesse trabalho são dados terciários obtidos do *datalake* público Base dos Dados (DAHIS et al., 2022). A fonte primária são os bancos comerciais e múltiplos com carteira comercial que disponibilizam mensalmente os saldos dos principais verbetes do balancete via documento 4500¹⁶ ao Banco Central do Brasil, que os compila e publica, agrupados por agência bancária e por município, no relatório ESTBAN — Estatística Bancária Mensal e por Município¹⁷.

O que compõe os verbetes de crédito, ou seja, os valores das séries temporais a serem trabalhadas, são os saldos de crédito ativo (empréstimos e financiamentos), que correspondem ao principal mais os juros calculados até 59 dias de atraso¹⁸. Além das estatísticas bancárias, foram obtidas informações de regiões, mesorregiões e microrregiões dos estados, também a partir *datalake* Base dos Dados, com o objetivo de enriquecer a estrutura hierárquica dos dados do ESTBAN, limitada aos municípios.

Em relação à cobertura temporal, utilizaremos dados entre 2003 e 2022. Por conta da limitação imposta pelo processo de *rolling origin* (1ª etapa da metodologia de Spiliotis et al. (2021)), que requer previsões para dentro do conjunto de treinamento, apesar do Banestes contar com 134 agências, foram mantidas apenas as agências com série completa, ou seja, que já estavam em atividade em 2003.

Por fim, o conjunto de dados conta com séries mensais de saldos de empréstimos e

¹⁶ Esses documentos são relatórios eletrônicos obrigatórios demandados pelo Bacen às instituições financeiras que permitem ao regulador o conhecimento minucioso dos bancos e de seus clientes.

¹⁷ <https://www4.bcb.gov.br/fis/cosif/estban.asp?frame=1>

¹⁸ Não são consideradas crédito ativo as operações de crédito liquidadas ou que tenham sido transferidas para prejuízo. São transferidas para prejuízo as operações de crédito em atraso há mais 6 meses após sua classificação de risco em H, que é a mínima (CONSELHO MONETÁRIO NACIONAL, 1999).

financiamentos de 79 agências bancárias, distribuídas por 62 municípios, entre janeiro de 2003 e dezembro de 2022, totalizando 37.920 observações. Esses dados então são organizados de forma hierárquica por estado, mesorregião, microrregião, município e agência bancária; e, de forma agrupada, por verbete. Com a estrutura hierárquica e agrupada, o conjunto de dados alcança 114 480 observações.

4.2 Análise exploratória dos dados

O estado do Espírito Santo está localizado no sudeste brasileiro e é dividido em 78 municípios, que estão agrupados em 4 mesorregiões e 13 microrregiões.

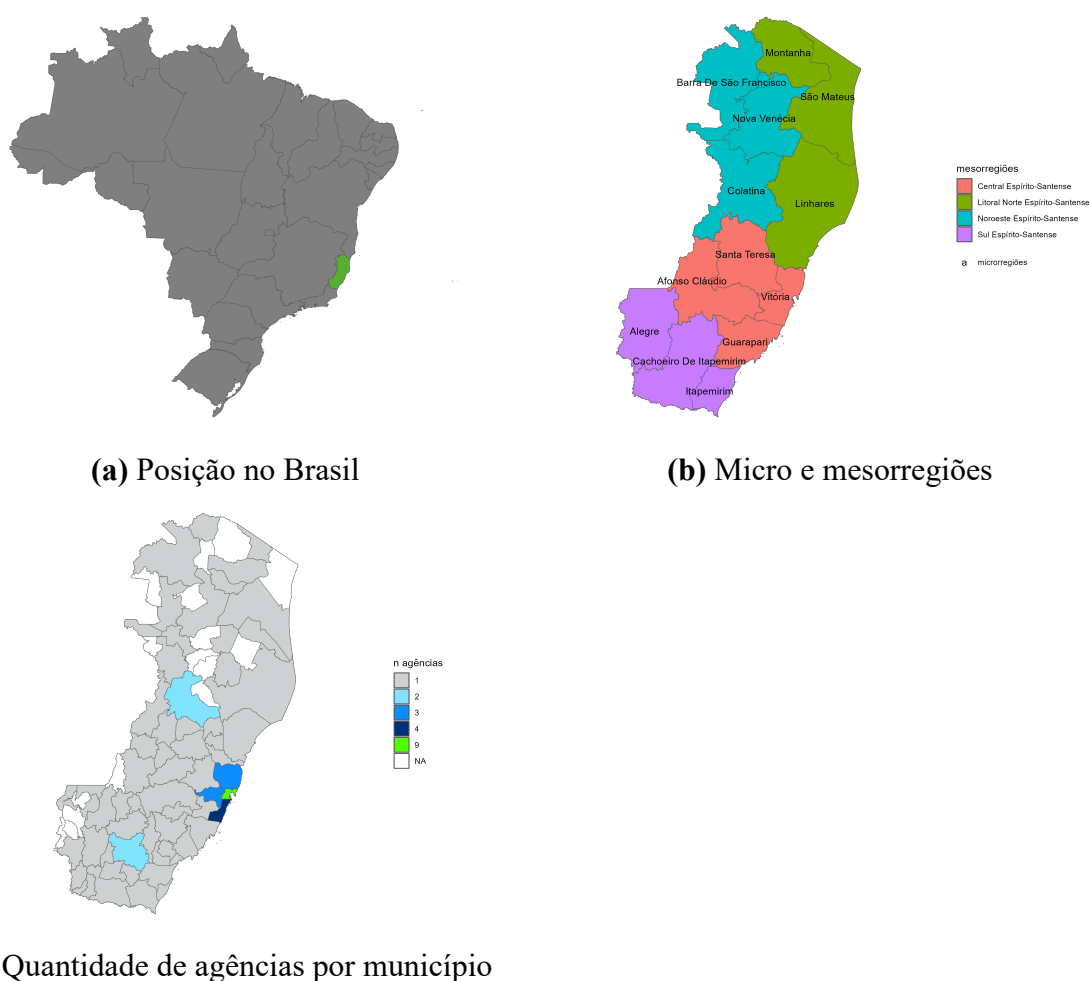


Figura 6 – O Estado do Espírito Santo e suas meso e microrregiões.

As microrregiões que compõem cada mesorregião são apresentadas na Tabela 1 e na Figura 6b. Já os municípios que compõem cada microrregião são apresentados na Tabela 2. A Figura 6c mostra a quantidade de agências por município.

Na amostra selecionada para este trabalho, com exceção dos municípios de Colatina e Cachoeiro de Itapemirim, que contam com 2 agências cada, todos os demais municípios no interior contam com uma única agência. Já na microrregião da capital Vitória, se encontram

Tabela 1 – Microrregiões por mesorregião

Mesorregião	Microrregiões
Central Espírito Santense	Afonso Claudio, Guarapari, Santa Teresa, Vitória
Litoral Norte Espírito Santense	Linhares, Montanha, São Mateus
Noroeste Espírito Santense	Barra de São Francisco, Colatina, Nova Venécia
Sul Espírito Santense	Alegre, Cachoeiro de Itapemirim, Itapemirim

Tabela 2 – Municípios por microrregião

Microrregiões	Municípios
Afonso Claudio	Afonso Claudio, Conceição do Castelo, Domingos Martins, Laranja da Terra, Marechal Floriano, Venda Nova do Imigrante
Alegre	Alegre, Guacui, Ibatiba, Iuna, Muniz Freire
Barra de São Francisco	Barra de São Francisco, Ecoporanga, Mantenópolis
Cachoeiro de Itapemirim	Apiaca, Atilio Vivacqua, Bom Jesus do Norte, Cachoeiro de Itapemirim, Castelo, Jerônimo Monteiro, Mimoso do Sul, Muqui, São José do Calçado, Vargem Alta
Colatina	Baixo Guandu, Colatina, Pancas
Guarapari	Alfredo Chaves, Anchieta, Guarapari, Iconha, Piuma, Rio Novo do Sul
Itapemirim	Itapemirim, Marataizes, Presidente Kennedy
Linhares	Aracruz, Fundão, Ibiracu, João Neiva, Linhares, Rio Bananal
Montanha	Mucurici, Pinheiros
Nova Venécia	Águia Branca, Boa Esperança, Nova Venécia, Vila Valério
Santa Teresa	Itaguacu, Itarana, Santa Leopoldina, Santa Maria de Jetiba, Santa Teresa, São Roque do Canaã
São Mateus	Jaguare, Pedro Canário, São Mateus
Vitória	Cariacica, Serra, Viana, Vila Velha, Vitória

municípios com múltiplas unidades. Considerando que essa microrregião é a mais representativa (Figura 9), é importante a manutenção do nível hierárquico municipal.

A série temporal do agregado de crédito no Banestes no Espírito Santo é apresentada na Figura 7. Em relação à distribuição, a mesorregião Central Espírito-santense concentra mais crédito do que o somatório das demais regiões (Figura 8), sendo a microrregião de Vitória a responsável por essa concentração (Figura 9). Na ótica dos verbetes, o crédito para financiamentos é uma pequena fração do saldo de empréstimos, independentemente da mesorregião (Figura 10 e Figura 11).

4.3 Previsões base

Embora os métodos de ML também sejam utilizados no contexto de previsão de séries temporais, é importante ressaltar que este não é caso deste trabalho. As previsões de séries temporais são realizadas anteriormente, na obtenção das previsões base \hat{y}_t . Nada impede que essas previsões sejam obtidas por um modelo de ML, mas isso não é o foco deste trabalho. A aplicação dos métodos de ML aqui ocorrem na reconciliação, ou seja, na combinação contemporânea, *cross section*, das previsões base que, por sua vez, podem ter sido obtidas através de qualquer método.

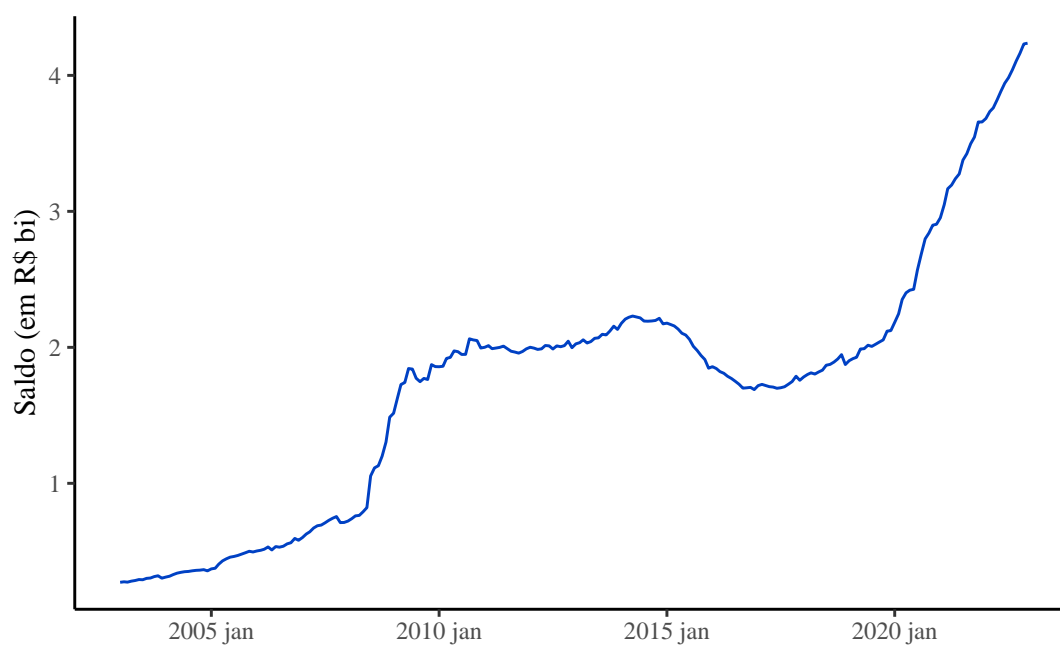


Figura 7 – Série temporal do agregado de crédito do Banestes no ES

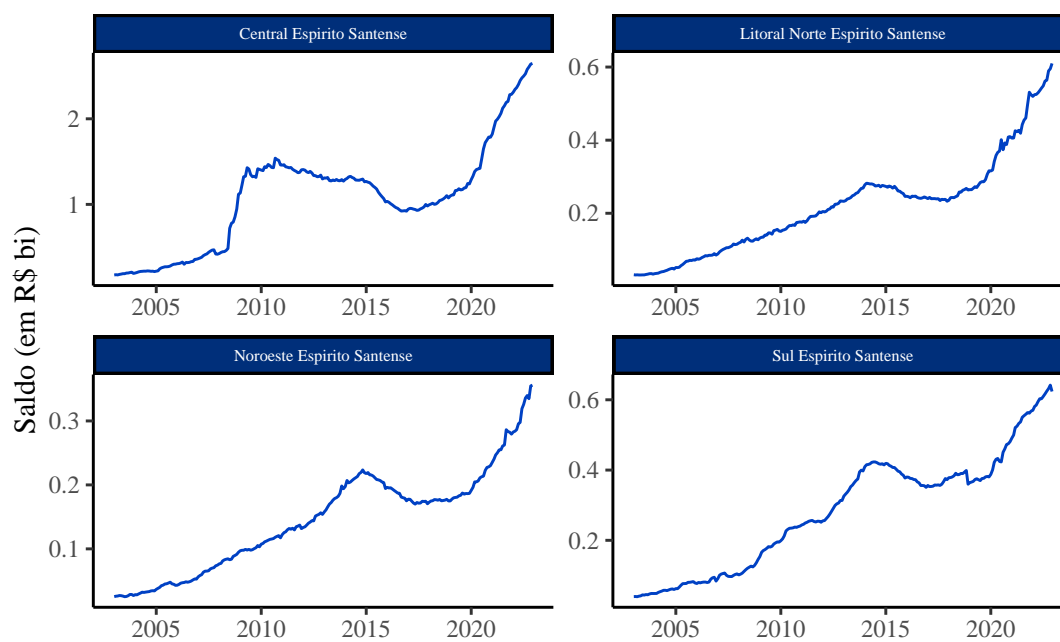


Figura 8 – Séries temporais do agregado de crédito do Banestes por mesorregião do ES

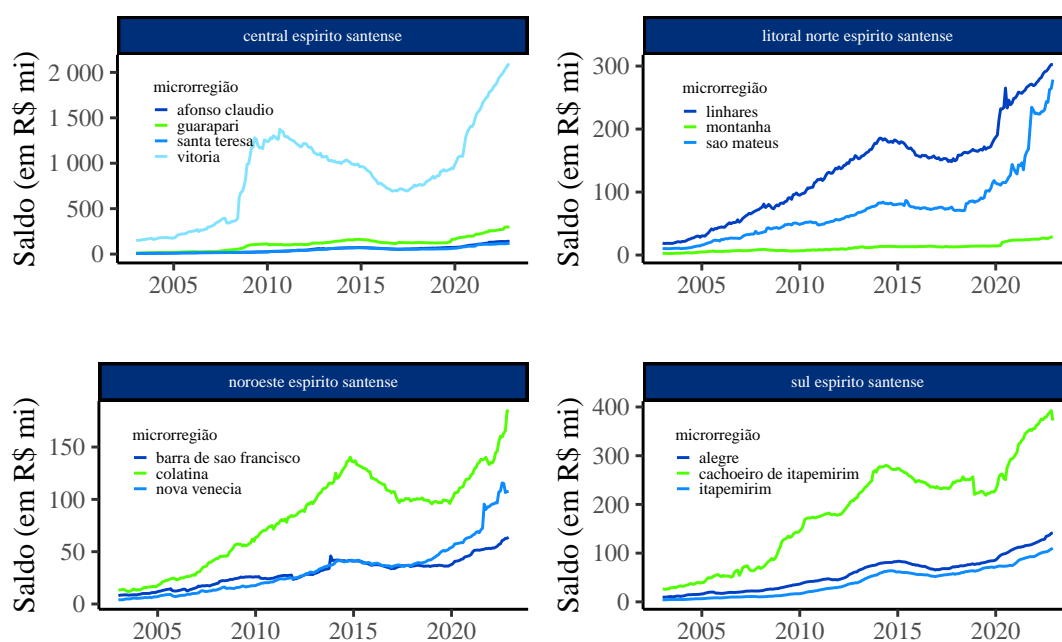


Figura 9 – Séries temporais do agregado de crédito do Banestes por microrregião do ES

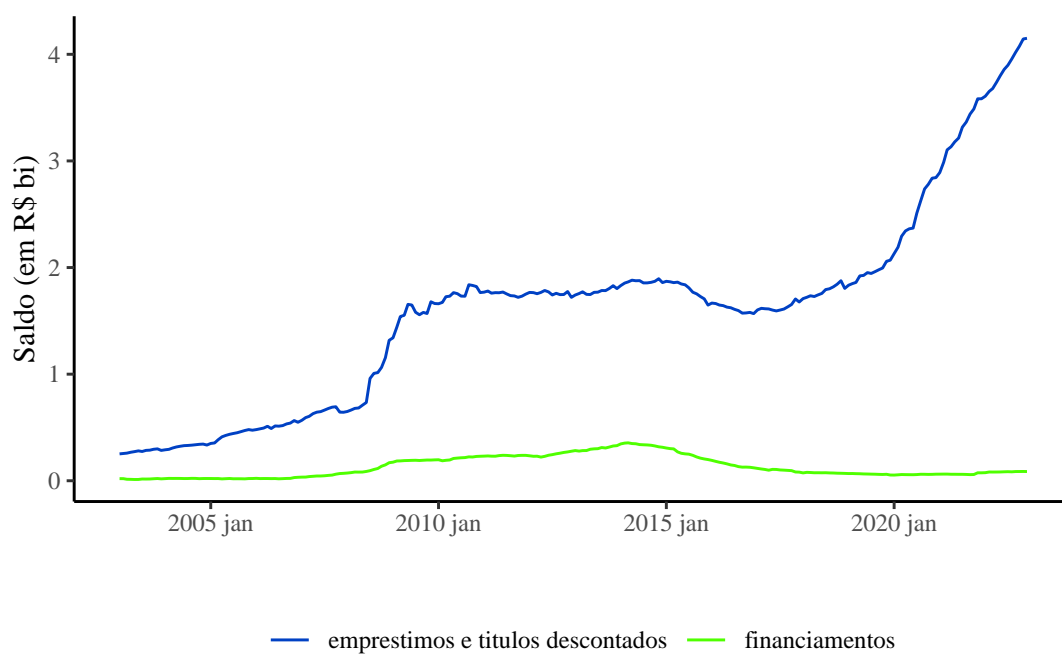


Figura 10 – Séries temporais dos verbetes no agregado do ES

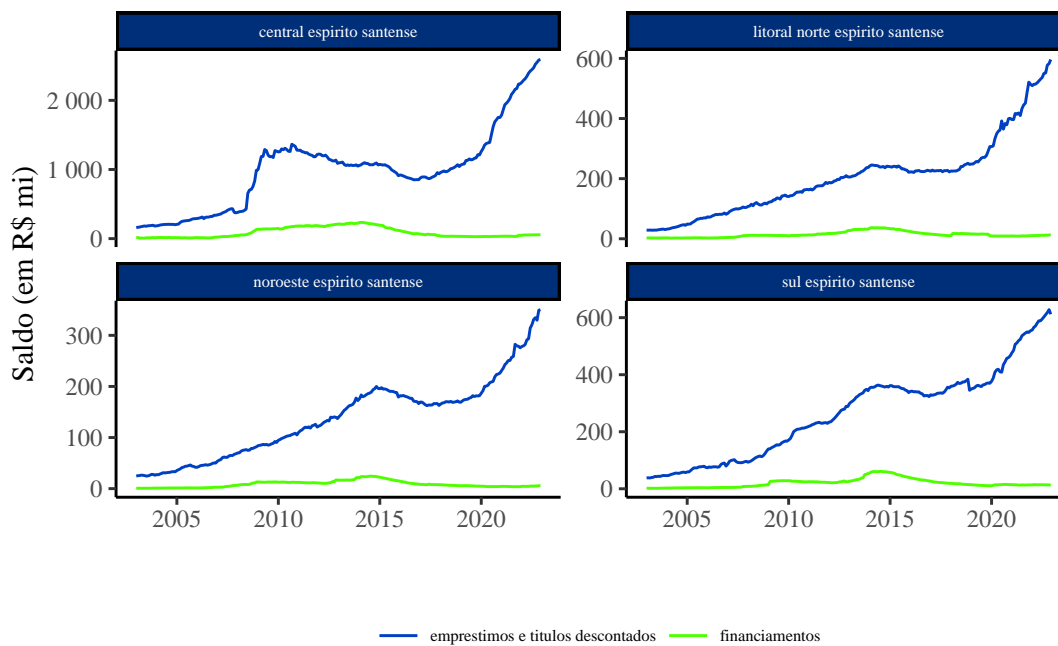


Figura 11 – Séries temporais dos verbetes por mesorregião do ES

Uma vez que o foco deste trabalho está no incremento de performance proporcionado pela reconciliação ótima sobre um conjunto de previsões base e não na qualidade destas, elas foram obtidas por meio de métodos básicos para previsão de séries temporais, especificamente o algoritmo de Hyndman-Koehler-Snyder-Große para suavimento exponencial (ETS) (HYNDMAN; KOEHLER et al., 2002). Esses métodos são amplamente utilizados na literatura de séries temporais e, portanto, servem como *benchmark* para a avaliação dos métodos de reconciliação ótima.

Os modelos foram treinados com dados de 2003 a 2021 e as previsões h passos à frente foram realizadas para o ano de 2022, com $h = 1, 2, 3, \dots, 12$ (Figura 12). Para verificar a qualidade básica do ajuste, foram realizados testes de Ljung-Box para os resíduos das previsões. Considerando nível de significância de $\alpha = 0.05$ e 12 defasagens, cerca de 80% dos modelos não rejeitam a hipótese nula de que os resíduos são decorrelacionados. Isso sugere que os modelos de previsão base são, em sua maior parte, adequados para o propósito deste trabalho.



Figura 12 – Previsões base a serem reconciliadas. Em azul, o conjunto de treino e, em verde, as previsões h passos à frente para fora da amostra.

4.4 Design do experimento

O objetivo do experimento consiste em comparar o ganho de desempenho dos métodos de reconciliação em relação às previsões base, ou seja, dada a performance das previsões individuais, o quanto mais precisas elas se tornam ao aplicar um determinado método de reconciliação. Para isso, serão utilizadas as abordagens trabalhadas em [Spiliotis et al. \(2021\)](#): a de nível único *bottom-up*¹⁹, o método analítico de reconciliação ótima *MinT-Shrink* e os métodos de reconciliação ótima baseados em *machine learning*: *gradient boosting*, especificamente o *XGBoost*, e *Random Forest*. Adicionalmente, serão avaliados os métodos *Elastic Net* (*Lasso*, *Ridge* e *Elastic Net*) e *Support Vector Machines*, além de outra implementação de *gradient boosting*, o *Light-GBM*. Uma breve introdução sobre cada um desses métodos pode ser encontrado no Apêndice B.

Foram avaliadas três estratégias para obtenção das previsões reconciliadas. A primeira é a proposta por [Spiliotis et al. \(2021\)](#), adaptada à base de dados Estban e com as janelas de treino e previsão mais próximas ao caso de uso de uma instituição financeira. Ela consiste em:

1. *rolling forecast*: A amostra treino é separada em p subamostras para todas as séries em todos os níveis de agregação, com a primeira separação abrangendo y_1 até Y_Q , a segunda y_1 até Y_{Q+1} e assim sucessivamente até a última separação abrangendo y_1 até Y_{Q+p-1} . Para cada subamostra foi treinado um modelo (de mesmo algoritmo usado na seção 4.3) e obtida previsão um passo a frente. Para este trabalho, a amostra treino foi dividida em 50%, com $Q = 120$ e $p = 108$.

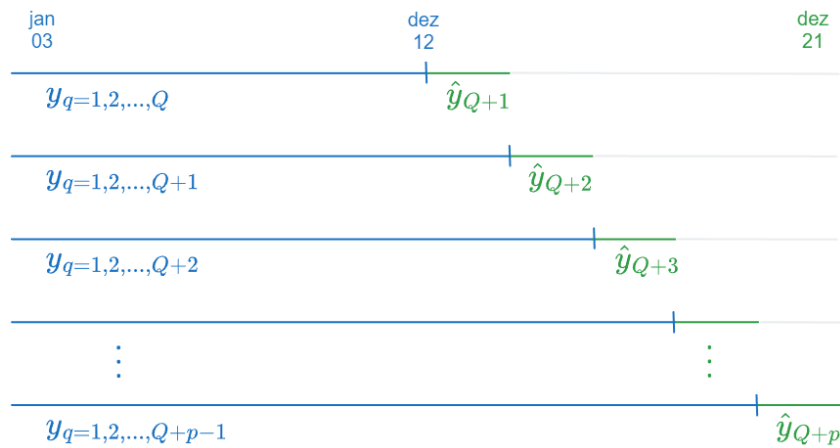


Figura 13 – Esquema de modelagem *rolling forecast*

2. Treino dos modelos de *machine learning*: Para cada série do nível mais desagregado, y_m , é treinado um modelo de ML com $n + 1$ variáveis, compostas pelas n séries previstas —

¹⁹ Como o banco de dados Estban é agrupado, a desagregação não é única e as abordagens *top-down* e *middle-out* não podem ser aplicadas ([ATHANASOPOULOS; HYNDMAN et al., 2023](#)).

que incluem todos os níveis de agregação —, mais a própria y_m como variável explicada (Tabela 3). Cada uma das n séries contam com p previsões obtidas no passo 1, representadas na Figura 13 pela cor verde. Isso resulta em um modelo de reconciliação ótima para cada elemento do menor nível da hierarquia, combinando informações disponíveis de todos os níveis hierárquicos.

3. Reconciliação ótima: Com os modelos treinados no passo 2, passa-se as previsões base obtidas na seção 4.3 como regressores para se obter as previsões reconciliadas h passos à frente das séries do nível mais desagregado \tilde{y}_m .
4. Agregação: Assim como nos métodos analíticos de combinação ótima, a obtenção das previsões reconciliadas para os demais níveis de hierárquicos \tilde{y}_n se dá através da agregação semelhante ao *bottom-up*, mas ao invés de se somar as previsões base \hat{y}_m , somam-se as previsões reconciliadas \tilde{y}_m .

Tabela 3 – Conjunto de dados para predição dos modelos de ML

Var. Explicada	Variável 1	Variável 2	...	Variável n
$y_{1,Q+1}$	$\hat{y}_{1,Q+1}$	$\hat{y}_{2,Q+1}$...	$\hat{y}_{n,Q+1}$
$y_{1,Q+2}$	$\hat{y}_{1,Q+2}$	$\hat{y}_{2,Q+2}$...	$\hat{y}_{n,Q+2}$
...
$y_{m,Q+p}$	$\hat{y}_{1,Q+p}$	$\hat{y}_{2,Q+p}$...	$\hat{y}_{n,Q+p}$

Dessa forma, essa metodologia é semelhante à aplicada na reconciliação ótima analítica, se afastando principalmente em três pontos: (i) a utilização de algoritmos de ML ao invés de MQG, (ii) a não atribuição de peso de forma obrigatória para todos os nós da hierarquia e (iii) no ajuste de um modelo individual para cada série do nível mais desagregado, permitindo maior especialização e sendo capaz de se adaptar melhor aos diferentes padrões de cada série (SPILIOTIS et al., 2021).

Um ponto negativo na metodologia proposta por Spiliotis et al. (2021) é o processo de *rolling origin* (passo 1). Estimar um modelo para cada conjunto de treinamento sequencialmente em um cenário de largas hierarquias e diversos grupos pode ser muito custoso em tempo de processamento. No caso do banco de dados Estban, temos 79 agências e 2 verbetes, totalizando 158 séries a serem estimadas. Para cada uma delas, devem ser ajustados p modelos, somando ao todo 17.064 modelos.

Nesse sentido, uma alternativa é o processo de reestimação, que consiste na reestimação de um modelo para um novo conjunto de dados. Nessa estratégia, que chamremos de *reduced fitted values*, os m modelos são treinados até Q e então reajustados para o conjunto de dados disponível em $Q + p$, se esquivando do processo de *rolling origin*. Uma restrição dessa abordagem é que, fixados os hiperparâmetros anteriores, não necessariamente todos modelos alcançarão convergência no reajuste de seus coeficientes para o novo conjunto de dados.

Porém, as duas estratégias possuem uma fragilidade em comum: a necessidade de realização de previsões para dentro da amostra treino. Isso pode ser um problema para séries temporais com poucas observações ou incompletas. No caso da base de dados Estban, algumas das agências foram criadas após o período escolhido para a separação do conjunto de treino em Q (dezembro/2012), sendo necessário sua exclusão do dataset e invalidando o uso da metodologia para essas unidades.

Nesses casos, uma 3ª estratégia para permitir o uso dos métodos de ML é usar os valores ajustados dos modelos das previsões base \hat{y} (seção 4.3) como input para os modelos de ML. Isso significa usar previsões 1 passo à frente obtidas ao longo de todo o período $q = 1, 2, 3, \dots, Q + p$. Além de permitir a inclusão de séries incompletas — agências criadas durante o período, no caso da base de dados Estban —, essa abordagem também aumenta o tamanho da amostra treino em Q observações, o que pode melhorar a performance dos modelos de ML. Chamaremos essa estratégia de *fitted base*.

Em suma, vamos realizar o experimento através da metodologia de [Spiliotis et al. \(2021\)](#), com a adição de 5 métodos de ML, e também testar duas outras possibilidades não investigadas em seu artigo original, doravante denominadas *fitted base* e *reduced fitted base*.

4.5 Otimização de hiperparâmetros

A maior parte dos métodos de *machine learning* são altamente parametrizáveis, sendo sua performance de generalização (para fora da amostra) sensível à escolha de seus hiperparâmetros. Quando disponível, os hiperparâmetros a serem otimizados e seus espaços de busca seguiram a recomendação em [Bischl et al. \(2021\)](#).

Os conjuntos de hiperparâmetros e seus intervalos são apresentados no Apêndice C. Para a otimização, foram utilizadas dois calibradores: (i) busca em grade (com resolução de 10 combinações), mais custoso em tempo de processamento, para os métodos com menor quantidade de hiperparâmetros a serem otimizados, e (ii) otimização bayesiana (na configuração padrão do pacote {mlr3MBO}), mais eficiente para os métodos com maior quantidade de hiperparâmetros. A estratégia de reamostragem utilizada foi a validação cruzada *k-fold* com $k = 10$.

A otimização bayesiana foi usada em todos os métodos, exceto no *elastic net*, uma vez que apenas um (no caso do *lasso* e *ridge*) ou dois hiperparâmetros foram otimizados. A medida de performance utilizada para a otimização foi a raiz do erro quadrático médio (*root mean squared error* — RMSE).

Por fim, cada modelo foi calibrado individualmente, ou seja, cada agência possui um conjunto de hiperparâmetros otimizados para cada um dos 7 métodos de ML empregados para reconciliação ótima.

5 RESULTADOS

As tabelas a seguir apresentam os resultados obtidos para o experimento. As métricas utilizadas foram a Raiz do Erro Médio Escalado Quadrático (*Root Mean Squared Scaled Error* — RMSSE) e o erro médio escalado absoluto (*Mean Absolute Scaled Error* — MASE). Para fins de comparação, além da base de dados de interesse, o Estban, o experimento foi executado também para a base de dados “Tourism”, disponível no pacote {tsibble} (WANG et al., 2024), e seus resultados reportados no apêndice D.

A Tabela 4 contém as medidas de acurácia RMSSE e MASE para os métodos analíticos de reconciliação ótima BU (*bottom-up*) e MinT, e para as previsões base, ou seja, sem aplicar qualquer método de reconciliação. A primeira coluna especifica o método utilizado, enquanto as demais colunas apresentam a média da performance em cada nível de agregação.

As Tabela 5, Tabela 6 e Tabela 7 reportam as medidas de acurácia para os métodos de reconciliação ótima baseados em *machine learning*. Já a Tabela 8 reporta o tempo de processamento para as etapas de calibragem, treino e predição desses métodos²⁰.

Em geral, os métodos baseados em árvore, além de requererem maior tempo de processamento devido a sua complexidade no espaço de hiperparâmetros, também tenderam a perder qualidade de performance conforme suas previsões são agregadas para formação dos níveis superiores da hierarquia. Contrariamente, os métodos de regressão regularizada e o SVM se mostraram mais robustos à agregação.

Nas tabelas a seguir, **negrito** indica a melhor performance entre os métodos para aquele determinado nível de agregação, e sublinhado indica que aquele método de ML superou o método analítico de melhor performance naquele nível de agregação.

Nos resultados para a base de dados Estban estão incluídas as médias de performance para cada nível hierárquico e agrupado. As colunas “agregado”, “mesorregião”, “microrregião”, “município” e “agência”, fazem referência à estrutura hierárquica, ou seja, tratam o verbete de forma agregada. Já as colunas “verbetes”, “bottom” e “hierarquia”, incluem também a estrutura agrupada, tratando o verbete de forma desagregada. Detalhadamente:

- Agregado: performance do método para a série que representa o total, com os verbetes agregados, (Figura 7).
- Mesorregião: a média das performances do método para as séries do agregado de cada mesorregião, com os verbetes agregados (Figura 8).
- Microrregião: a média das performances do método para as séries do agregado de cada microrregião, com os verbetes agregados (Figura 9).

²⁰ Os métodos analíticos não tiveram seu tempo de processamento medidos porque executam quase que instantaneamente, já sinalizando uma vantagem para esses métodos.

- Município: a média das performances do método para as séries do agregado de cada município, com os verbetes agregados.
- Agência: a média das performances do método para as séries de cada agência, com os verbetes agregados.
- Verbetes: a média das performances do método para as séries de cada verbete, para o total da hierarquia (Figura 10).
- Bottom: a média das performances do método para as séries do nível mais desagregado, ou seja, verbete por agência.
- Hierarquia: a média das performances do método para todas as séries, agregadas e desagregadas.

Tabela 4 – Resultados Estban: Acurácia dos métodos analíticos de reconciliação

.model	agregado	mesorregiao	microrregiao	municipio	agencia	verbetes	bottom	hierarquia
RMSSE								
base	0.197	0.690	0.846	0.671	0.717	0.183	0.656	0.657
bu	0.196	0.561	0.706	0.710	0.739	0.185	0.656	0.663
mint	0.172	0.619	0.722	0.689	0.712	0.140	0.634	0.641
MASE								
base	0.278	0.818	0.998	0.790	0.886	0.250	0.883	0.847
bu	0.240	0.572	0.771	0.820	0.895	0.221	0.883	0.844
mint	0.224	0.692	0.865	0.830	0.891	0.164	0.864	0.837

Tabela 5 – Resultados Estban: Acurácia dos métodos de ML de reconciliação. Estratégia rolling forecast.

modelo	agregado	mesorregiao	microrregiao	municipio	agencia	verbetes	bottom	hierarquia
RMSSE								
elastic net	0.280	0.763	1.178	1.211	1.251	0.770	1.062	1.094
lasso	0.196	0.726	1.054	0.995	1.043	0.501	0.839	0.882
lightgbm	1.407	1.628	1.530	1.294	1.322	0.883	0.972	1.095
random forest	1.227	1.397	1.303	1.118	1.171	0.725	0.858	0.949
ridge	0.416	0.776	1.131	1.511	1.535	0.919	1.357	1.368
svm	0.262	0.745	0.858	0.853	0.911	0.234	0.847	0.838
xgb	1.186	1.405	1.296	1.096	1.139	0.700	0.830	0.924
MASE								
elastic net	0.234	0.726	1.406	1.491	1.582	0.949	1.439	1.428
lasso	<u>0.166</u>	0.714	1.250	1.193	1.298	0.641	1.142	1.147
lightgbm	1.890	1.896	1.874	1.600	1.654	1.234	1.390	1.478
random forest	1.615	1.560	1.501	1.332	1.423	0.996	1.097	1.177
ridge	0.402	0.757	1.315	1.800	1.881	1.090	1.790	1.738
svm	0.306	0.684	0.862	0.991	1.108	0.290	1.251	1.143
xgb	1.542	1.564	1.479	1.297	1.373	0.948	1.022	1.115

Para a base de dados Estban, não houve uma combinação de método e estratégia que fosse consistentemente melhor ao longo de todos os níveis de agregação. Portanto, a escolha do método e da estratégia a serem utilizados dependerá do objetivo do pesquisador²¹.

²¹ Se o objetivo é a elaboração de *guidance*, por exemplo, o pesquisador deve preferir o método e estratégia que

Tabela 6 – Resultados Estban: Acurácia dos métodos de ML de reconciliação. Estratégia fitted base.

modelo	agregado	mesorregiao	microrregiao	municipio	agencia	verbete	bottom	hierarquia
RMSSE								
elastic net	0.777	0.986	1.086	1.038	1.149	0.579	0.924	0.961
lasso	0.661	0.955	1.074	0.909	1.008	0.530	0.826	0.862
lightgbm	1.495	1.649	1.557	1.300	1.342	0.923	0.999	1.116
random forest	1.204	1.397	1.294	1.098	1.150	0.694	0.839	0.930
ridge	1.001	1.146	1.247	1.208	1.327	0.689	1.125	1.147
svm	0.395	0.929	0.928	0.934	0.961	0.319	0.905	0.898
xgb	1.196	1.373	1.282	1.084	1.133	0.699	0.824	0.916
MASE								
elastic net	1.049	1.143	1.317	1.268	1.451	0.795	1.290	1.284
lasso	0.894	1.087	1.319	1.114	1.276	0.728	1.167	1.162
lightgbm	2.027	1.931	1.906	1.610	1.683	1.302	1.433	1.512
random forest	1.576	1.557	1.488	1.311	1.397	0.947	1.043	1.132
ridge	1.338	1.350	1.549	1.501	1.696	0.935	1.584	1.551
svm	0.445	0.947	1.080	1.163	1.226	0.341	1.282	1.217
xgb	1.545	1.509	1.476	1.293	1.375	0.942	1.017	1.109

Tabela 7 – Resultados Estban: Acurácia dos métodos de ML de reconciliação. Estratégia reduced fitted base.

modelo	agregado	mesorregiao	microrregiao	municipio	agencia	verbete	bottom	hierarquia
RMSSE								
elastic net	0.090	0.582	0.730	0.819	0.838	0.164	0.730	0.736
lasso	<u>0.132</u>	0.523	0.766	0.757	0.774	0.187	0.681	0.693
lightgbm	1.406	1.588	1.520	1.281	1.323	0.889	0.971	1.091
random forest	1.248	1.409	1.319	1.119	1.167	0.692	0.857	0.949
ridge	<u>0.141</u>	0.635	0.784	0.902	0.922	0.207	0.841	0.829
svm	0.187	0.743	0.767	0.792	0.834	0.295	0.807	0.792
xgb	1.218	1.347	1.253	1.084	1.140	0.708	0.844	0.927
MASE								
elastic net	0.086	0.584	0.834	0.973	1.008	0.208	0.944	0.922
lasso	<u>0.138</u>	0.520	0.883	0.907	0.933	0.216	0.891	0.878
lightgbm	1.879	1.831	1.832	1.580	1.652	1.236	1.388	1.470
random forest	1.636	1.576	1.526	1.333	1.418	0.947	1.065	1.155
ridge	<u>0.159</u>	0.630	0.894	1.047	1.087	0.231	1.073	1.021
svm	0.225	0.764	0.850	0.949	1.022	0.395	1.176	1.083
xgb	1.593	1.491	1.427	1.293	1.380	0.965	1.064	1.137

Tabela 8 – Resultados Estban: Tempo de processamento dos métodos de ML (em horas)

	xgb	random forest	elastic net	lasso	ridge	svm	lightgbm
reduced fitted base	19.235	5.483	1.428	0.879	0.993	1.279	3.290
fitted base	21.758	5.521	1.363	0.829	0.924	1.273	3.341
rolling forecast	20.908	5.429	1.345	0.838	0.929	1.285	3.377

Para os níveis ao topo da hierarquia, os métodos de ML se mostraram a melhor opção para estimação. No nível agregado, o *elastic net* na estratégia *reduced fitted base* (Tabela 7) se mostrou a melhor opção para a estimação do agregado, com 91% de ganho de performance sobre o MinT (Tabela 4), em termos de RMSSE, ou seja, o MinT obteve quase o dobro do erro do *elastic net*. Da mesma forma, para o nível de mesorregião, o *elastic net* na configuração *lasso* teve a melhor performance, sendo 7% melhor que BU. Por outro lado, tanto nos níveis hierárquicos abaixo, quanto nos níveis agrupados, os métodos de ML não foram capazes de superar os métodos analíticos.

Os resultados se mostraram bastante sensíveis à estratégia utilizada. Na métrica RMSS, nenhum método utilizando as estratégias *rolling forecast* e *fitted base* foi capaz de superar o MinT em qualquer nível de agregação, enquanto na estratégia *reduced fitted base* os métodos de ML mostraram ganhos de performance com os métodos SVM e nas três configurações do *elastic net*.

Em geral, as medidas RMSSE e MASE se mostraram bastante correlacionadas, com os métodos que obtiveram melhor performance em uma métrica também obtendo melhor performance na outra. A exceção foi para o método *lasso* na estratégia *rolling forecast*, que obteve performance melhor que o MinT, em termos de MASE, para o nível agregado, sendo o único resultado positivo para a estratégia *rolling forecast*.

6 CONCLUSÃO

Neste trabalho, foram apresentados experimentos de reconciliação ótima para séries temporais hierárquicas e agrupadas, utilizando métodos analíticos e de *machine learning* com o objetivo de obter coerência e ganhos de acurácia nas previsões de saldos de empréstimos e financiamentos do Banco do Estado do Espírito Santo. Pesquisas anteriores já haviam mostrado que a reconciliação ótima pode trazer ganhos de acurácia, e que métodos de *machine learning* podem ser competitivos em relação aos métodos analíticos.

Este trabalho trouxe, além dos métodos de floresta aleatória e *gradient boosting* já trabalhados em Spiliotis et al. (2021), o método de regressão regularizada *elastic net* e o *support vector machines*, além de avaliar outro método de *gradient boosting*, o *lightGBM*. Paralelamente, este trabalho propôs duas estratégias alternativas para a metodologia de reconciliação ótima baseada em *machine learning* proposta originalmente em Spiliotis et al. (2021).

Os resultados obtidos para a base de dados Estban mostraram, primeiramente, que não houve uma combinação de método e estratégia que obtivesse melhor performance de maneira consistente ao longo de todos os níveis hierárquicos. Dessa forma, a escolha do método e da estratégia a serem utilizados dependerá do objetivo do pesquisador. Para os níveis ao topo da

geram as previsões mais precisas para o nível agregado. Já para elaboração de metas individuais, os níveis individuais ou regionais podem ser preferíveis.

hierarquia, a combinação correta de método e estratégia de estimação (*elastic net* + *reduced fitted base*) gerou até 91% de ganho de performance no nível mais agregado, permitindo à instituição financeira maior precisão para tomada de decisão e planejamento estratégico, além de sinalizar maior confiança nas estimativas comunicadas ao mercado e aos investidores. Por outro lado, para os níveis inferiores na estrutura hierárquica, os métodos analíticos se mostraram a melhor escolha. Isso sugere que, com o objetivo de elaboração de metas individuais — seja para as agências ou superintendências regionais —, os métodos analíticos ainda são preferíveis.

Os resultados também mostram que o resultado da reconciliação ótima é sensível à estratégia utilizada. Apenas na estratégia *reduced fitted base* que os ganhos de performance foram observados, enquanto nas estratégias *rolling forecast* e *fitted base* os métodos de ML não foram capazes de superar os métodos analíticos²².

Contrariamente aos resultados de [Spiliotis et al. \(2021\)](#), os métodos de ML baseados em árvore de decisão (*XGBoost*, *ranger* e *lightGBM*) não foram capazes de superar os métodos analíticos em nenhum nível de agregação.

Para pesquisas futuras, pode-se investigar se a performance dos diferentes métodos e estratégias estão relacionadas às características das séries temporais, por exemplo:

- Os efeitos do ruído de previsão: se os diferentes métodos e estratégias exibem aumento ou deterioração de performance quando as previsões individuais são mais ou menos ruidosas (i.e. se a variância do erro das previsões individuais é maior ou menor).
- Os efeitos de correlação entre as séries: se os métodos e estratégias exibem aumento ou deterioração de performance quando as séries temporais no menor nível hierárquico são mais ou menos correlacionadas.
- Os efeitos de componentes sazonais: se os métodos e estratégias exibem aumento ou deterioração de performance quando as séries temporais do menor nível hierárquico possuem ou não componentes sazonais.
- Os efeitos do tamanho da hierarquia: verificar se os métodos e estratégias exibem aumento ou deterioração de performance quando a hierarquia é mais ou menos profunda (i.e. possui mais ou menos níveis hierárquicos).

²² O mesmo padrão em relação à performance ao longo dos níveis de agregação pôde ser observado na base de dados “Tourism” (Apêndice D). Foi possível encontrar uma combinação de método de ML e estratégia capaz de superar os métodos analíticos para os níveis mais agregados, mas não para os níveis mais desagregados. Naquela base de dados, os métodos de ML superaram os analíticos em todos os níveis de agregação, exceto no mais desagregado, com os métodos *support vector machines* e *elastic net* liderando a performance tanto nas estratégias *rolling forecast* quanto *fitted base*.

REFERÊNCIAS

- ATHANASOPOULOS, G.; AHMED, R. A.; HYNDMAN, R. J. Hierarchical forecasts for Australian domestic tourism. en. **International Journal of Forecasting**, v. 25, n. 1, p. 146–166, jan. 2009. ISSN 0169-2070. DOI: [10.1016/j.ijforecast.2008.07.004](https://doi.org/10.1016/j.ijforecast.2008.07.004). Disponível em: <https://www.sciencedirect.com/science/article/pii/S0169207008000691>>. Acesso em: 11 jan. 2023. Citado na p. 23.
- ATHANASOPOULOS, G.; HYNDMAN, R. J. et al. Forecast reconciliation: A review. en, 2023. Citado na p. 39.
- BADER, F. L. C.; KOYAMA, SÉRGIO MIKIO; TSUCHIDA, MARCOS HIROYUKI. Modelo favar canônico para previsão do mercado de crédito. pt. **Banco Central do Brasil**, v. 369, p. 38, nov. 2014. ISSN 1519-1028. Citado na p. 22.
- BANCO DO ESTADO DO ESPÍRITO SANTO. **Demonstrações Financeiras**. Vitória, 2022. Disponível em: https://www.banestes.com.br/ri/arquivos/informacoes/demoFinanceiras/2022_2S_demonFinanceiras.pdf>. Acesso em: 4 jun. 2023. Citado na p. 20.
- BECCALLI, E. et al. Earnings management, forecast guidance and the banking crisis. **The European Journal of Finance**, v. 21, n. 3, p. 242–268, fev. 2015. ISSN 1351-847X. DOI: [10.1080/1351847X.2013.809548](https://doi.org/10.1080/1351847X.2013.809548). Disponível em: <https://doi.org/10.1080/1351847X.2013.809548>>. Acesso em: 7 mai. 2023. Citado na p. 19.
- BISCHL, B. et al. **Hyperparameter Optimization: Foundations, Algorithms, Best Practices and Open Challenges**. [S.l.]: arXiv, nov. 2021. arXiv:2107.05847 [cs, stat]. DOI: [10.48550/arXiv.2107.05847](https://doi.org/10.48550/arXiv.2107.05847). Disponível em: <http://arxiv.org/abs/2107.05847>>. Acesso em: 4 set. 2023. Citado na p. 41.
- BRASIL. **Lei nº 7.492, de 16 de junho de 1986**. Brasília, DF: Presidência da República, jun. 1986. Disponível em: https://www.planalto.gov.br/ccivil_03/leis/17492.htm>. Citado na p. 19.
- CHEN, T.; GUESTRIN, C. XGBoost: A Scalable Tree Boosting System. In: PROCEEDINGS of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, NY, USA: Association for Computing Machinery, ago. 2016. (KDD '16), p. 785–794. ISBN 978-1-4503-4232-2. DOI: [10.1145/2939672.2939785](https://doi.org/10.1145/2939672.2939785). Disponível em: <https://dl.acm.org/doi/10.1145/2939672.2939785>>. Acesso em: 23 jan. 2024. Citado na p. 58.
- CHEN, T.; HE, T. et al. **xgboost: Extreme Gradient Boosting**. [S.l.], 2024. R package version 1.7.7.1. Disponível em: <https://github.com/dmlc/xgboost>>. Citado na p. 33.

ÇOLAK, M. S. et al. **TCMB - Monitoring and Forecasting Cyclical Dynamics in Bank Credits: Evidence from Turkish Banking Sector**. en. Ankara, Turquia, 2019. Disponível em: <<https://www.tcmb.gov.tr/wps/wcm/connect/EN/TCMB+EN/Main+Menu/Publications/Research/Working+Papers/2019/19-29>>. Acesso em: 6 mar. 2023. Citado na p. 22.

CONSELHO MONETÁRIO NACIONAL. **Resolução nº 2.099, de 17 de agosto de 1994**. Brasília, DF: Banco Central do Brasil, ago. 1994. Disponível em: <https://www.bcb.gov.br/pre/normativos/res/1994/pdf/res_2099_v1_O.pdf>. Citado na p. 20.

_____. **Resolução nº 2.682, de 21 de dezembro de 1999**. Brasília, DF: Banco Central do Brasil, dez. 1999. Disponível em: <https://www.bcb.gov.br/pre/normativos/res/1999/pdf/res_2682_v2_L.pdf>. Citado na p. 33.

DAHIS, R. et al. Data Basis (Base Dos Dados): Universalizing Access to High-Quality Data. en. **SSRN Electronic Journal**, 2022. ISSN 1556-5068. DOI: [10.2139/ssrn.4157813](https://doi.org/10.2139/ssrn.4157813). Disponível em: <<https://www.ssrn.com/abstract=4157813>>. Acesso em: 18 jun. 2023. Citado na p. 33.

FRIEDMAN, J. et al. **glmnet: Lasso and Elastic-Net Regularized Generalized Linear Models**. [S.l.], 2023. R package version 4.1-8. Disponível em: <<https://glmnet.stanford.edu>>. Citado na p. 33.

GORODETSKAYA, O.; GOBAREVA, Y.; KOROTEEV, M. A Machine Learning Pipeline for Forecasting Time Series in the Banking Sector. en. **Economies**, v. 9, n. 4, p. 205, dez. 2021. Number: 4 Publisher: Multidisciplinary Digital Publishing Institute. ISSN 2227-7099. DOI: [10.3390/economies9040205](https://doi.org/10.3390/economies9040205). Disponível em: <<https://www.mdpi.com/2227-7099/9/4/205>>. Acesso em: 27 fev. 2023. Citado na p. 23.

GROSS, C. W.; SOHL, J. E. Disaggregation methods to expedite product line forecasting. en. **Journal of Forecasting**, v. 9, n. 3, p. 233–254, 1990. ISSN 1099-131X. DOI: [10.1002/for.3980090304](https://doi.org/10.1002/for.3980090304). Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1002/for.3980090304>>. Acesso em: 18 jun. 2023. Citado na p. 21.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. **The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition**. 2nd 2009, Corr. 9th Printing 2017 ed. edição. New York, NY: Springer, fev. 2009. ISBN 978-0-387-84857-0. Citado nas pp. 58, 60, 61.

HYNDMAN, R.; ATHANASOPOULOS, G. **Forecasting: principles and practice**. 3. ed. Melbourne, Austrália: OTexts, 2021. Disponível em: <<https://otexts.com/fpp3/>>. Citado nas pp. 21, 24, 27.

HYNDMAN, R. J.; KOEHLER, A. B. et al. A state space framework for automatic forecasting using exponential smoothing methods. **International Journal of Forecasting**, v. 18, n. 3, p. 439–454, jul. 2002. ISSN 0169-2070. DOI: [10.1016/S0169-2070\(01\)00110-8](https://doi.org/10.1016/S0169-2070(01)00110-8). Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0169207001001108>>. Acesso em: 26 nov. 2023. Citado na p. 38.

HYNDMAN, R. J.; AHMED, R. A. et al. Optimal combination forecasts for hierarchical time series. en. **Computational Statistics & Data Analysis**, v. 55, n. 9, p. 2579–2589, set. 2011. ISSN 0167-9473. DOI: [10.1016/j.csda.2011.03.006](https://doi.org/10.1016/j.csda.2011.03.006). Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0167947311000971>>. Acesso em: 11 jan. 2023. Citado nas pp. 21, 23.

HYNDMAN, R. J.; LEE, A. J.; WANG, E. Fast computation of reconciled forecasts for hierarchical and grouped time series. en. **Computational Statistics & Data Analysis**, v. 97, p. 16–32, mai. 2016. ISSN 0167-9473. DOI: [10.1016/j.csda.2015.11.007](https://doi.org/10.1016/j.csda.2015.11.007). Disponível em: <<https://www.sciencedirect.com/science/article/pii/S016794731500290X>>. Acesso em: 11 jan. 2023. Citado na p. 23.

KE, G. et al. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In: **ADVANCES in Neural Information Processing Systems**. [S.l.]: Curran Associates, Inc., 2017. v. 30. Disponível em: <<https://proceedings.neurips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html>>. Acesso em: 23 jan. 2024. Citado na p. 58.

LANG, M. et al. **mlr3: Machine Learning in R - Next Generation**. [S.l.], 2024. R package version 0.17.2. Disponível em: <<https://mlr3.mlr-org.com>>. Citado na p. 33.

LI, Z. et al. A Hierarchical Approach Using Machine Learning Methods in Solar Photovoltaic Energy Production Forecasting. en. **Energies**, v. 9, n. 1, p. 55, jan. 2016. Number: 1 Publisher: Multidisciplinary Digital Publishing Institute. ISSN 1996-1073. DOI: [10.3390/en9010055](https://doi.org/10.3390/en9010055). Disponível em: <<https://www.mdpi.com/1996-1073/9/1/55>>. Acesso em: 8 abr. 2023. Citado na p. 23.

O'HARA-WILD, M.; HYNDMAN, R.; WANG, E. **fable: Forecasting Models for Tidy Time Series**. [S.l.], 2023. R package version 0.3.3. Disponível em: <<https://fable.tidyverts.org>>. Citado na p. 33.

PANAGIOTELIS, A. et al. Forecast reconciliation: A geometric view with new insights on bias correction. en. **International Journal of Forecasting**, v. 37, n. 1, p. 343–359, jan. 2021. ISSN 0169-2070. DOI: [10.1016/j.ijforecast.2020.06.004](https://doi.org/10.1016/j.ijforecast.2020.06.004). Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0169207020300911>>. Acesso em: 15 jan. 2023. Citado na p. 23.

PRAYOGA, I.; SUHARTONO, S.; RAHAYU, S. Top-down forecasting for high dimensional currency circulation data of Bank Indonesia. **International Journal of Advances in Soft Computing and its Applications**, v. 9, p. 62–74, jan. 2017. Citado na p. 23.

R CORE TEAM. **R: A Language and Environment for Statistical Computing**. Vienna, Austria, 2023. Disponível em: <<https://www.R-project.org/>>. Citado na p. 33.

SEZER, O. B.; GUDELEK, M. U.; OZBAYOGLU, A. M. **Financial Time Series Forecasting with Deep Learning : A Systematic Literature Review: 2005-2019**. Turquia: arXiv, nov. 2019. arXiv:1911.13288 [cs, q-fin, stat]. Disponível em: <<http://arxiv.org/abs/1911.13288>>. Acesso em: 7 mar. 2023. Citado na p. 22.

SHI, Y. et al. **lightgbm: Light Gradient Boosting Machine**. [S.l.], 2024. R package version 4.3.0. Disponível em: <<https://github.com/Microsoft/LightGBM>>. Citado na p. 33.

SPILOTIS, E. et al. Hierarchical forecast reconciliation with machine learning. en. **Applied Soft Computing**, v. 112, p. 107756, nov. 2021. ISSN 1568-4946. DOI: [10.1016/j.asoc.2021.107756](https://doi.org/10.1016/j.asoc.2021.107756). Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1568494621006773>>. Acesso em: 11 jan. 2023. Citado nas pp. 5, 7, 22, 23, 31–33, 39–41, 45, 46.

WALLANDER, J. Budgeting — an unnecessary evil. en. **Scandinavian Journal of Management**, v. 15, n. 4, p. 405–421, dez. 1999. ISSN 0956-5221. DOI: [10.1016/S0956-5221\(98\)00032-3](https://doi.org/10.1016/S0956-5221(98)00032-3). Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0956522198000323>>. Acesso em: 8 mai. 2023. Citado na p. 19.

WANG, E. et al. **tsibble: Tidy Temporal Data Frames and Tools**. [S.l.], 2024. R package version 1.1.4. Disponível em: <<https://tsibble.tidyverts.org>>. Citado na p. 42.

WICKRAMASURIYA, S. L.; ATHANASOPOULOS, G.; HYNDMAN, R. J. Optimal Forecast Reconciliation for Hierarchical and Grouped Time Series Through Trace Minimization. en. **Journal of the American Statistical Association**, v. 114, n. 526, p. 804–819, abr. 2019. ISSN 0162-1459, 1537-274X. DOI: [10.1080/01621459.2018.1448825](https://doi.org/10.1080/01621459.2018.1448825). Disponível em: <<https://www.tandfonline.com/doi/full/10.1080/01621459.2018.1448825>>. Acesso em: 18 jun. 2023. Citado nas pp. 21, 23, 30, 31.

WRIGHT, M. N.; WAGER, S.; PROBST, P. **ranger: A Fast Implementation of Random Forests**. [S.l.], 2023. R package version 0.16.0. Disponível em: <<http://imbs-hl.github.io/ranger/>>. Citado na p. 33.

ZOU, H.; HASTIE, T. Regularization and variable selection via the elastic net. en. **Journal of the Royal Statistical Society: Series B (Statistical Methodology)**, v. 67, n. 2, p. 301–320, 2005. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-9868.2005.00503.x>. ISSN 1467-9868. DOI: [10.1111/j.1467-9868.2005.00503.x](https://doi.org/10.1111/j.1467-9868.2005.00503.x). Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2005.00503.x>>. Acesso em: 5 jun. 2023. Citado na p. 57.

Apêndices

APÊNDICE A – DEMONSTRAÇÕES

Proposição A.1 (condição de ausência de viés em $\tilde{\mathbf{y}}$). *Se as previsões reconciliadas são não viesadas, então $\mathbf{SGS} = \mathbf{S}$, ou seja, \mathbf{G} é inversa generalizada de \mathbf{S} .*

Demonstração.

$$\tilde{\mathbf{y}}_{t+h|t} = \mathbf{SG}\hat{\mathbf{y}}_{t+h|t} \quad (\text{A.1})$$

Se $\hat{\mathbf{y}}_{t+h|t}$ é não viesado, então

$$\mathbb{E}[\hat{\mathbf{y}}_{t+h|t}] = \mathbb{E}[\mathbf{y}_{t+h|t}] = \mathbf{S}\mathbf{b}_t \quad (\text{A.2})$$

Da mesma forma, se espera-se que as previsões reconciliadas não sejam viesadas,

$$\mathbb{E}[\tilde{\mathbf{y}}_{t+h|t}] = \mathbb{E}[\mathbf{y}_{t+h|t}] = \mathbf{S}\mathbf{b}_t \quad (\text{A.3})$$

Substituindo (A.2) em (A.1), temos

$$\tilde{\mathbf{y}}_{t+h|t} = \mathbf{SGS}\mathbf{b}_t \quad (\text{A.4})$$

Logo, para manter a igualdade entre (A.1) e (A.4), $\mathbf{SGS} = \mathbf{S}$

□

Proposição A.2. $\tilde{\mathbf{e}}_t = \mathbf{SG}\hat{\mathbf{e}}_t$.

Demonstração.

$$\tilde{\mathbf{e}}_{t+h|t} = \mathbf{y}_{t+h} - \tilde{\mathbf{y}}_{t+h|t} \quad (\text{A.5})$$

Substituindo (A.1) em (A.5),

$$\tilde{\mathbf{e}}_{t+h|t} = \mathbf{y}_{t+h} - \mathbf{SG}\hat{\mathbf{y}}_{t+h|t} \quad (\text{A.6})$$

Lembrando que, por definição, $\mathbf{y}_{t+h} = \hat{\mathbf{y}}_{t+h|t} + \hat{\mathbf{e}}_{t+h|t}$, então

$$\tilde{\mathbf{e}}_{t+h|t} = \hat{\mathbf{y}}_{t+h|t} + \hat{\mathbf{e}}_{t+h|t} - \mathbf{SG}\hat{\mathbf{y}}_{t+h|t} \quad (\text{A.7})$$

$$= \hat{\mathbf{e}}_{t+h|t} + \hat{\mathbf{y}}_{t+h|t}(\mathbf{I} - \mathbf{SG}) \quad (\text{A.8})$$

Usando a definição novamente, temos que

$$\tilde{e}_{t+h|t} = \hat{e}_{t+h|t} + (\mathbf{y}_{t+h} - \hat{e}_{t+h|t})(\mathbf{I} - \mathbf{SG}) \quad (\text{A.9})$$

$$= \mathbf{y}_{t+h} - \mathbf{SG}\mathbf{y}_{t+h|t} + \mathbf{SG}\hat{e}_{t+h|t} \quad (\text{A.10})$$

$$= \mathbf{y}_{t+h}(\mathbf{I} - \mathbf{SG}) + \mathbf{SG}\hat{e}_{t+h|t} \quad (\text{A.11})$$

Substituindo (5) em (A.11), temos

$$\tilde{e}_{t+h|t} = \mathbf{S}\mathbf{b}_{t+h}(\mathbf{I} - \mathbf{SG}) + \mathbf{SG}\hat{e}_{t+h|t} \quad (\text{A.12})$$

$$= \mathbf{S}\mathbf{b}_{t+h} - \mathbf{S}\mathbf{b}_{t+h}\mathbf{SG} + \mathbf{SG}\hat{e}_{t+h|t} \quad (\text{A.13})$$

$$= \mathbf{S}\mathbf{b}_{t+h} - (\mathbf{G}'\mathbf{S}')(\mathbf{b}'_{t+h}\mathbf{S}') + \mathbf{SG}\hat{e}_{t+h|t} \quad (\text{A.14})$$

$$= \mathbf{S}\mathbf{b}_{t+h} - \mathbf{SG}\mathbf{S}\mathbf{b}_{t+h} + \mathbf{SG}\hat{e}_{t+h|t} \quad (\text{A.15})$$

Finalmente, pela condição de ausência de viés em $\tilde{\mathbf{y}}$, temos que

$$\tilde{e}_{t+h|t} = \mathbf{S}\mathbf{b}_{t+h} - \mathbf{S}\mathbf{b}_{t+h} + \mathbf{SG}\hat{e}_{t+h|t} \quad (\text{A.16})$$

$$= \mathbf{SG}\hat{e}_{t+h|t} \quad (\text{A.17})$$

□

Proposição A.3. $\text{Var}[\tilde{e}_t] = \mathbf{SG}\hat{\mathbf{W}}\mathbf{G}'\mathbf{S}'$.

Demonstração. Por A.2, temos que

$$\text{Var}[\tilde{e}] = \mathbb{E}[(\mathbf{SG}\hat{e})(\mathbf{SG}\hat{e})'] \quad (\text{A.18})$$

$$= \mathbb{E}[\mathbf{SG}\hat{e}\hat{e}'\mathbf{G}'\mathbf{S}'] \quad (\text{A.19})$$

$$= \mathbf{SG}\hat{\mathbf{W}}\mathbf{G}'\mathbf{S}' \quad (\text{A.20})$$

Em que $\hat{\mathbf{W}}$ é a matriz de variância-covariância dos erros de previsão base. □

Proposição A.4. $\hat{\mathbf{W}}$ é posto incompleto.

Demonstração. Pela propriedade do vínculo do posto do produto de matrizes, ou seja, $\text{pos}(\mathbf{AB}) \leq \min(\text{pos}(\mathbf{A}), \text{pos}(\mathbf{B}))$, temos que

$$\text{pos}(\mathbf{SG}\hat{e}_{t+h|t}) \leq \min(\text{pos}(\mathbf{S}), \text{pos}(\mathbf{G}), \text{pos}(\hat{e}_{t+h|t})) \quad (\text{A.21})$$

Como S é a representação matricial de uma estrutura hierárquica, em que os nós pais totalizam os nós filhos, S apresenta, por hipótese, dependência linear e, conseqüentemente, posto incompleto.

Pela equação (A.21), segue que \tilde{e} é posto incompleto. Da mesma forma, $pos(\tilde{e}\tilde{e}') \leq \min(pos(\tilde{e}), pos(\tilde{e}'))$. Portanto, \hat{W} é posto incompleto. \square

APÊNDICE B – MÉTODOS DE MACHINE LEARNING

1 ELASTIC NET

O *elastic net* (ZOU; HASTIE, 2005) é um método de regressão regularizada que combina as normas L_1 e L_2 , as penalidades do *lasso* e do *ridge*, respectivamente. A função objetivo a ser minimizada é dada por

$$L(\lambda_1, \lambda_2, \beta) = |\mathbf{y} - \mathbf{X}\beta|^2 + \lambda_2|\beta|^2 + \lambda_1|\beta|_1 \quad (\text{B.1})$$

em que λ_1 e λ_2 são os parâmetros de regularização e β é o vetor de coeficientes a serem estimados. A solução para essa função objetivo é dada por¹

$$\hat{\beta} = \arg \min_{\beta} |\mathbf{y} - \mathbf{X}\beta|^2, \text{ sujeito a } (1 - \alpha)|\beta|_1 + \alpha|\beta|^2 \leq t \quad (\text{B.2})$$

com $\alpha = \frac{\lambda_2}{\lambda_1 + \lambda_2}$ e $t \in \mathbb{R}^+$.

A função $(1 - \alpha)|\beta|_1 + \alpha|\beta|^2$ é a penalidade *elastic net*, uma combinação das penalidades *lasso* e *ridge*. O parâmetro α controla a mistura das duas penalidades, incluindo os casos extremos. Note que $\alpha = 0 \implies \lambda_2 = 0$, resultando em uma penalidade exclusivamente *lasso*, enquanto $\alpha = 1 \implies \lambda_1 = 0$, e a penalidade é apenas do tipo *ridge*.

Portanto o *elastic net* é um método de *shrinkage*, uma vez que a penalidade *ridge* reduz o tamanho dos coeficientes, e de *seleção de variáveis*, uma vez que a penalidade *lasso* tende a anular os coeficientes de variáveis irrelevantes. Essas propriedades são desejáveis para a reconciliação de séries temporais, uma vez que a estrutura hierárquica pode conter séries insignificantes para a previsão de outras séries.

Diferentemente dos métodos analíticos estudados, o *elastic net* não possui uma solução fechada. Portanto, é necessário utilizar métodos iterativos para encontrar o valor ótimo de $\hat{\beta}$ e Zou e Hastie (2005) utilizam validação cruzada k -fold para encontrar quais os valores de λ_1 e λ_2 que minimizam o resíduo. Nesse sentido, dado a metodologia de processo iterativo envolvendo calibragem de hiperparâmetros e reamostragem, podemos classificar o *elastic net* como um método de *machine learning*.

¹ Sob o valor otimizado ainda é aplicada correção de escala na forma $(1 + \lambda_2)\hat{\beta}$. Ver Zou e Hastie (2005).

2 GRADIENT BOOSTING

Como cada uma das diversas implementações de *gradient boosting* possui sua teoria adjacente, e não é de objetivo deste trabalho detalhar o funcionamento de cada uma delas, trabalharemos apenas sua intuição.

Assim como os métodos de floresta aleatória, os métodos de *gradient boosting* também são métodos de conjuntos de árvores. A diferença se dá na forma como os modelos são treinados. *Gradient boosting* são métodos que combinam as predições de vários modelos fracos para formar um conjunto — *ensemble*, na definição mais usual, ou comitê (*committee*), na definição de [Hastie, Tibshirani e Friedman \(2009\)](#) quando usado para classificação — mais complexo e preciso (forte).

Um estimador fraco é aquele que tem desempenho apenas ligeiramente melhor que o acaso. O propósito do *boosting* é produzir uma sequência de estimadores fracos, cada um deles focado nos erros dos estimadores anteriores. A cada iteração, as observações classificadas incorretamente (no caso de uma tarefa de classificação) ou de maior variância (no caso de uma tarefa de regressão) na iteração anterior têm seu peso aumentado, e vice-versa. Dessa forma, o modelo subsequente formado na próxima iteração é obrigado a se concentrar nas observações onde as iterações anteriores falharam. Isso que significa transformar um conjunto de estimadores fracos em um conjunto forte.

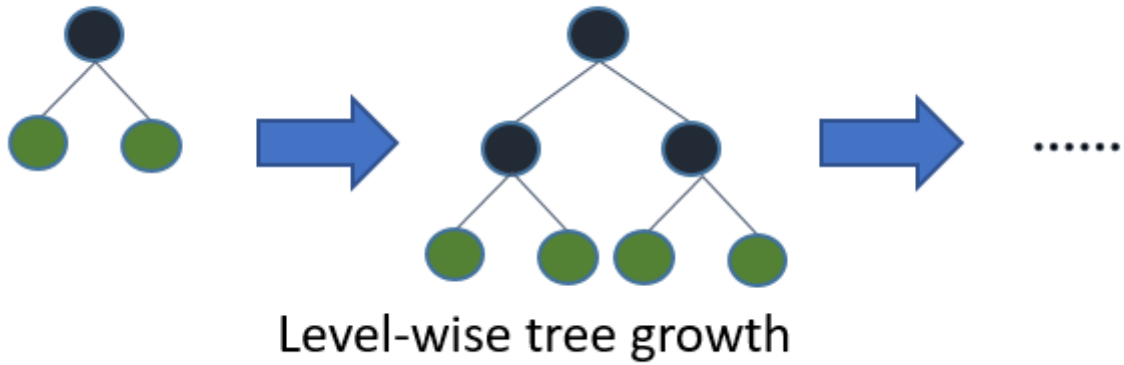
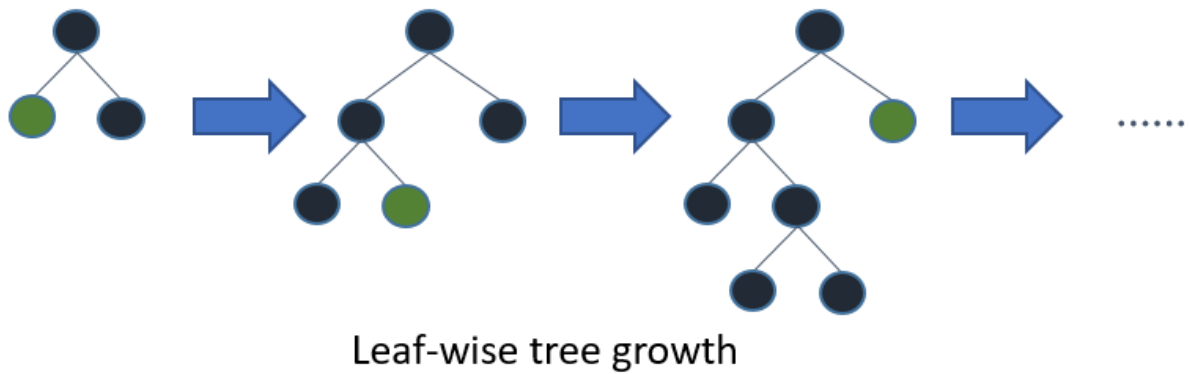
O *gradient boosting* é uma extensão do *boosting* que utiliza o gradiente da função de perda como critério de otimização, de forma que esta se dá na direção em que a função de perda decresce mais rapidamente a cada iteração. Os métodos utilizados neste trabalho são o *XGBoost* ([CHEN; GUESTRIN, 2016](#)) e o *LightGBM* ([KE et al., 2017](#)).

Um das principais diferenças entre os dois métodos é a forma como as árvores são construídas. O *XGBoost* cresce suas árvores de forma *level-wise*, ou seja, cresce todas as folhas do último nível de uma árvore de uma vez, adicionando mais um nível de profundidade completo a cada iteração (Figura 14a). Já o *LightGBM* cresce suas árvores de forma *leaf-wise*, ou seja, cresce uma folha por vez, aprofundando a árvore apenas no nó que resulta na maior variação negativa na função de perda (Figura 14b).

3 RANDOM FOREST

Floresta aleatória é um método de aprendizado de máquina que utiliza conjuntos de árvores de decisão descorrelatadas² para classificação ou regressão. O método consiste em treinar várias árvores de decisão em subconjuntos aleatórios dos dados de treinamento e, então, combinar suas predições ([HASTIE; TIBSHIRANI; FRIEDMAN, 2009](#)). A aleatoriedade é introduzida

² As árvores de decisão são descorrelatadas no sentido que, ao contrário os métodos de *boosting trees*, a próxima árvore não é construída com base na iteração anterior (processo de fortalecimento).

(a) *level-wise*(b) *leaf-wise***Figura 14** – Crescimento de árvores em algoritmos de *boosting*.

de duas formas: na seleção das observações e na seleção das variáveis preditoras.

A intuição para seu algoritmo para regressão é simples e a ideia geral é, para cada árvore de decisão, particionar recursivamente nós de tamanho N em dois nós filhos de forma a aumentar a complexidade do modelo e minimizar a função de custo. Se a próxima partição de um nó resultar em um ou ambos nós filhos de tamanho menor que um mínimo estabelecido via hiperparâmetro, a partição é interrompida (Algoritmo 1).

4 SUPPORT VECTOR MACHINES

A intuição dos métodos de SVMs é mais facilmente compreendida a partir de uma tarefa de classificação de duas classes. Nesse caso, o objetivo é encontrar um hiperplano (i.e. um sub-espço de dimensão $n - 1$) que separe as classes de forma que a margem entre o hiperplano e os pontos de cada classe seja a maior possível. Então, dado um conjunto de N pares de observações e suas respectivas classes, $\{(x_i, y_i)\}_{i=1}^N$, em que $x_i \in \mathbb{R}^p$ e $y_i \in \{-1, 1\}$, queremos encontrar o hiperplano definido por

Algoritmo 1: Floresta aleatória para regressão

T_1, T_2, \dots, T_b é uma floresta aleatória de B árvores de decisão.

$i \in \mathbb{N}$ é a quantidade de observações em um nó.

$n_{min} \in \mathbb{N}$ é o número mínimo de observações em um nó.

for $b = 1 \rightarrow B$ **do**

1. Toma uma amostra de tamanho N dos dados de treino.

while $i < n_{min}$ **do**

a. Seleciona j variáveis aleatoriamente.

b. Escolhe a melhor variável para *split* dentre j .

c. Divide o nó em dois nós filhos.

if *Nenhum dos nós filhos tem* $i < n_{min}$ **then**

| d. Repete os passos a-c para cada nó filho.

end

end

end

2. Produz um conjunto de árvores $\{T_b\}_1^B$.

3. Realiza a predição para cada árvore T_b em um ponto x e calcula a média das predições:

$$\hat{f}(x) = \frac{1}{B} \sum_{b=1}^B T_b(x)$$

$$\{x : f(x) = x^T \beta + \beta_0 = 0\}$$

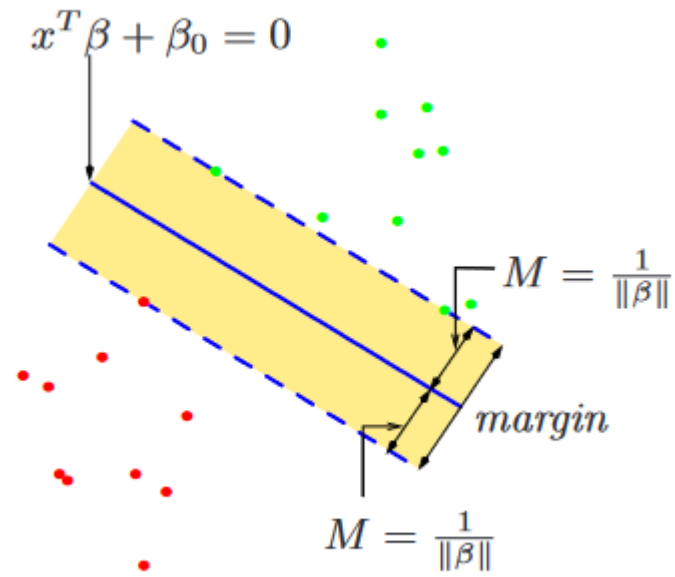
sendo $f(x)$ a distância ortogonal entre x e o hiperplano $f(x) = x^T \beta + \beta_0 = 0$, e M a margem entre as classes, definida como a distância ortogonal entre os pontos mais próximos de cada classe e o hiperplano (HASTIE; TIBSHIRANI; FRIEDMAN, 2009). Portanto, o problema de otimização é dado por

$$\begin{aligned} \max_{\beta, \beta_0} \quad & M \\ \text{sujeito a} \quad & y_i(x_i^T \beta + \beta_0) \geq M, \quad i = 1, \dots, N \end{aligned} \tag{B.3}$$

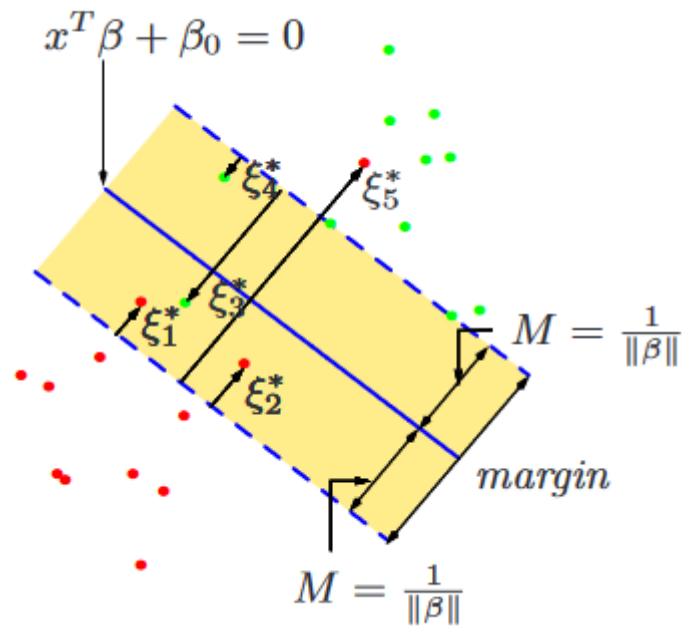
Quando pode-se encontrar um hiperplano com $y_i f(x_i) > 0 \quad \forall i$, tem-se o caso da construção de uma solução única para um hiperplano entre duas classes perfeitamente separadas pela maior margem possível (Figura 15a).

Entretanto, no mundo real, dificilmente um problema de classificação será linearmente separável. Para contornar esse obstáculo, é possível introduzir variáveis de folga $\xi_i \geq 0$ para cada observação (Figura 15b), de forma que a restrição se torne

$$y_i(x_i^T \beta + \beta_0) \geq M(1 - \xi_i), \quad i = 1, \dots, N \tag{B.4}$$



(a) Hiperplano entre duas classes perfeitamente separáveis.



(b) Hiperplano entre duas classes não separáveis.

Figura 15 – *Support vector classifiers*. Fonte: [Hastie, Tibshirani e Friedman \(2009\)](#).

e a função objetivo se torne

$$\min ||\beta|| \quad \text{sujeito a} \quad \left\{ \begin{array}{l} y_i(x_i^T \beta + \beta_0) \geq M(1 - \xi_i) \quad \forall i, \\ \xi_i \geq 0, \sum \xi_i \leq \text{constante} \end{array} \right\} \quad (\text{B.5})$$

Para estender essa ideia para problemas de regressão, é necessário introduzir uma função de perda ϵ -insensível, que penaliza apenas os erros maiores que ϵ , de forma que os erros menores são ignorados durante a otimização, assim como as observações localizadas no lado correto do hiperplano são ignoradas no problema de classificação. A função objetivo se torna

$$\min ||\beta|| \quad \text{sujeito a} \quad \left\{ \begin{array}{l} |y_i - x_i^T \beta - \beta_0| \leq \epsilon \quad \forall i, \\ \sum |\beta_j| \leq \text{constante} \end{array} \right\} \quad (\text{B.6})$$

APÊNDICE C – CONJUNTO DE HIPERPARÂMETROS

Tabela 9 – Intervalos de hiperparâmetros para {xgboost}

Hiperparâmetro	Descrição	Intervalo	Trafo
nrounds	Número de iterações	[1, 5000]	NULL
eta	Taxa de aprendizado	$[-4, 0]$	10^x
max_depth	Profundidade máxima	[1, 20]	NULL
subsample	Subamostra	[0.1, 1]	NULL
colsample_bytree	Subamostra de colunas para uma árvore	[0.1, 1]	NULL
colsample_bylevel	Subamostra de colunas por nível de profundidade	[0.1, 1]	NULL
lambda	Regularização L2	$[-10, 10]$	2^x
alpha	Regularização L1	$[-10, 10]$	2^x

Tabela 10 – Intervalos de hiperparâmetros para {lightgbm}

Hiperparâmetro	Descrição	Intervalo	Trafo
num_iterations	Número de iterações	[1, 1000]	NULL
boosting	Algoritmo de boosting	{gbdt, dart, goss}	NULL
learning_rate	Taxa de aprendizado	$[-4, 0]$	10^x
num_leaves	Número de folhas	[2, 20]	NULL
lambda_l1	Regularização L1	$[-12, 12]$	2^x
lambda_l2	Regularização L2	$[-12, 12]$	2^x
feature_fraction	Subamostra de colunas	[0.1, 1]	NULL
bagging_fraction	Subamostra de linhas	[0.1, 1]	NULL
bagging_freq	Frequência de amostragem	[1, 10]	NULL

Tabela 11 – Intervalos de hiperparâmetros para {ranger}

Hiperparâmetro	Descrição	Intervalo	Trafo
min.node.size	Número mínimo de observações em um nó terminal	[1, 7]	2^x
mtry	Número de variáveis candidatas para split	[1,)	NULL
replace	Amostragem com reposição	{TRUE, FALSE}	NULL
sample.fraction	Fração de observações a serem amostradas	[0.1, 1]	NULL
num.trees	Número de árvores	[1, 2000]	NULL

Tabela 12 – Intervalos de hiperparâmetros para {e1071} (svm)

Hiperparâmetro	Descrição	Intervalo	Trafo
cost	Custo de ξ	$[0, 1]$	2^x
kernel	Kernel	{linear, polynomial, radial, sigmoid}	NULL
degree	Grau do polinômio	$[1, 5]$	NULL
gamma	Influência amostral	$[-12, 12]$	2^x
type	Tipo de SVM	{eps-regression}	NULL

Tabela 13 – Intervalos de hiperparâmetros para {glmnet}

Hiperparâmetro	Descrição	Intervalo	Trafo
alpha	Mix entre lasso e ridge	$[0, 1]$	NULL
lambda	Regularização	$[-12, 12]$	2^x

APÊNDICE D – RESULTADOS PARA A BASE DE DADOS TOURISM

A base de dados "Tourism" consiste na quantidade trimestral de pernoites em visitas na Austrália entre 1998 e 2016. A estrutura é hierárquica e agrupada, composta por 3 níveis hierárquicos — *State* (Estados), *Region* (Regiões) e total —, e agrupado por *Purpose* (Propósito).

Assim como na base de dados Estban, aqui também não houve uma combinação de método e estratégia que fosse consistentemente melhor ao longo de todos os níveis de agregação. Entretanto, uma tendência pode ser observada em ambas bases de dados, com os métodos de ML se mostrando a melhor opção para estimação nos níveis ao topo da hierarquia, enquanto os métodos analíticos se mostram melhor opção para os níveis ao fundo da hierarquia.

Nessa base de dados, o método SVM na estratégia *rolling forecast* se mostrou a melhor combinação para os níveis mais agregados, alcançando 93% de incremento de performance em relação ao MinT, em termos de RMSSE, ou seja, o MinT teve quase o dobro do erro que o SVM. Já a estratégia *reduced fitted base* teve performance muito baixa, com problemas de estimação e necessidade de imputação de dados. Além da perda de performance nas medidas de acurácia, a estratégia *reduced fitted base* também teve tempo de processamento muito longos em relação às outras estratégias. Nessa estratégia também foi observada presença de valores negativos para os métodos *elastic net* e SVM, de forma que foram excluídos do *benchmark*.

Tabela 14 – Resultados Tourism: Acurácia dos métodos analíticos de reconciliação

.model	agregado	state	region	purpose	bottom	hierarquia
RMSSE						
base	1.446	1.260	1.068	1.265	0.925	0.976
bu	2.580	1.634	1.113	2.004	0.925	1.011
mint	1.813	1.296	0.978	1.420	0.876	0.923
MASE						
base	1.533	1.399	1.132	1.330	0.979	1.036
bu	3.164	1.877	1.176	2.323	0.979	1.078
mint	2.086	1.449	1.021	1.512	0.937	0.984

Tabela 15 – Resultados Tourism: Acurácia dos métodos de ML de reconciliação. Estratégia rolling forecast.

modelo	agregado	State	Region	Purpose	bottom	hierarquia
RMSSE						
elastic net	1.990	1.386	1.086	1.541	0.988	1.041
lasso	1.929	1.373	1.100	1.523	1.026	1.069
lightgbm	4.330	2.762	1.651	3.456	1.141	1.354
ranger	2.135	1.365	1.033	1.709	0.908	0.966
ridge	<u>1.256</u>	<u>1.185</u>	1.013	<u>1.202</u>	0.919	0.959
svm	0.940	1.010	1.076	1.011	1.100	1.097
xgb	2.340	1.451	1.114	1.892	0.964	1.031
MASE						
elastic net	2.360	1.572	1.145	1.653	1.058	1.115
lasso	2.264	1.557	1.168	1.593	1.110	1.155
lightgbm	5.505	3.214	1.763	4.060	1.200	1.448
ranger	2.579	1.528	1.073	1.816	0.961	1.020
ridge	<u>1.343</u>	<u>1.309</u>	1.058	<u>1.192</u>	0.981	1.020
svm	1.070	1.096	1.140	1.033	1.178	1.174
xgb	2.888	1.650	1.162	2.118	1.013	1.087

Tabela 16 – Resultados Tourism: Acurácia dos métodos de ML de reconciliação. Estratégia fitted base.

modelo	agregado	State	Region	Purpose	bottom	hierarquia
RMSSE						
elastic net	2.17	1.40	1.10	1.77	0.97	1.03
lasso	1.90	1.45	1.09	1.61	0.97	1.03
lightgbm	4.33	2.76	1.65	3.46	1.14	1.35
ranger	2.12	1.36	1.03	1.72	0.91	0.96
ridge	1.57	<u>1.16</u>	0.97	1.29	0.90	0.93
svm	1.50	<u>1.19</u>	1.05	1.38	1.04	1.05
xgb	2.27	1.42	1.10	1.83	0.96	1.02
MASE						
elastic net	2.59	1.57	1.16	1.95	1.04	1.10
lasso	2.21	1.67	1.16	1.73	1.04	1.11
lightgbm	5.50	3.21	1.76	4.06	1.20	1.45
ranger	2.54	1.54	1.07	1.84	0.97	1.02
ridge	1.77	<u>1.29</u>	1.01	<u>1.31</u>	0.96	0.99
svm	1.75	<u>1.30</u>	1.09	1.36	1.10	1.11
xgb	2.79	1.62	1.15	2.04	1.01	1.08

Tabela 17 – Resultados Tourism: Acurácia dos métodos de ML de reconciliação. Estratégia reduced fitted base.

modelo	agregado	State	Region	Purpose	bottom	hierarquia
RMSSE						
lightgbm	4.33	2.76	1.65	3.46	1.14	1.35
ranger	2.58	1.55	1.11	2.01	0.92	1.01
xgb	3.14	1.93	1.28	2.44	1.02	1.14
MASE						
lightgbm	5.50	3.21	1.76	4.06	1.20	1.45
ranger	3.18	1.76	1.17	2.24	0.98	1.07
xgb	3.93	2.27	1.38	2.86	1.07	1.21

Tabela 18 – Resultados Tourism: Tempo de processamento dos métodos de ML (em horas)

	xgb	ranger	elastic net	lasso	ridge	svm	lightgbm
fitted base	15.767	3.687	1.540	1.282	1.371	1.977	3.535
rolling forecast	12.087	2.987	0.957	0.796	1.073	2.035	3.596
reduced fitted base	24.627	15.386	6.006	4.968	5.323	10.474	19.362