UNIVERSIDADE FEDERAL DE SANTA CATARINA
CAMPUS FLORIANÓPOLIS
PROGRAMA DE PÓS-GRADUAÇÃO EM ECONOMIA

Lucas Eduardo Kava

Além da Caixa Preta: Aprendizagem de Máquina Interpretável para Previsão de Séries Temporais Macroeconômicas Brasileiras

Florianópolis
2022

Lucas Eduardo Kava

Além da Caixa Preta: Aprendizagem de Máquina Interpretável para Previsão de Séries Temporais Macroeconômicas Brasileiras

Dissertação submetida ao Programa de Pós-Graduação em Economia da Universidade Federal de Santa Catarina para a obtenção do título de mestre em Economia.
Orientador: Prof. André Alves Portela Santos, Dr.

Florianópolis
2022

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Kava, Lucas Eduardo
Além da Caixa Preta: Aprendizagem de Máquina Interpretável para Previsão de Séries Temporais Macroeconômicas Brasileiras / Lucas Eduardo Kava ; orientador, André Alves Portela Santos, 2022.
107 p.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro Sócio-Econômico, Programa de Pós-Graduação em Economia, Florianópolis, 2022.

Inclui referências.

1. Economia. 2. Aprendizagem de Máquina. 3. Econometria de Séries Temporais. 4. Macroeconomia. I. Alves Portela Santos, André. II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Economia. III. Título.

Lucas Eduardo Kava

Além da Caixa Preta: Aprendizagem de Máquina Interpretável para Previsão de Séries Temporais Macroeconômicas Brasileiras

O presente trabalho em nível de mestrado foi avaliado e aprovado por banca examinadora composta pelos seguintes membros:

Prof. André Alves Portela Santos, Dr.
Universidade Federal de Santa Catarina – UFSC

Prof. Guilherme Valle Moura, Dr.
Universidade Federal de Santa Catarina – UFSC

Prof. João Frois Caldeira, Dr.
Universidade Federal de Santa Catarina – UFSC

Profa. Roseli Basso-Silva, Dra.
Universidade de São Paulo – USP

Certificamos que esta é a **versão original e final** do trabalho de conclusão que foi julgado adequado para obtenção do título de mestre em Economia.

ICP-Edu
Documento assinado digitalmente
Guilherme Valle Moura
Data: 07/04/2022 15:11:26-0300
CPF: 013.234.686-92
Verifique as assinaturas em https://v.ufsc.br

Coordenação do Programa de
Pós-Graduação

ICP-Edu
Documento assinado digitalmente
Andre Alves Portela Santos
Data: 08/04/2022 09:14:06-0300
CPF: 939.975.665-34
Verifique as assinaturas em https://v.ufsc.br

Prof. André Alves Portela Santos, Dr.
Orientador

Florianópolis, 2022.

Dedico à minha mãe, Maria, por sempre ter incentivado
meus estudos.

AGRADECIMENTOS

Se eu tivesse que descrever o mestrado em economia hoje (e isso foi mudando ao longo da minha experiência vivida), eu diria que é o sofrer de felicidade: uma jornada árdua, muitas vezes solitária e exaustiva; mas de grande aprendizado, desenvolvimento de resiliência, amadurecimento intelectual e a sensação de dever cumprido quando acaba. A pandemia foi (e continua sendo) um desafio gigantesco em diversas esferas de nossas vidas – nos trouxe angústias, medos, momentos de profunda introspecção e incontáveis vezes em que pensamos em largar tudo – mas nos fez lembrar que somos queridos e que temos amizades valiosas que são motivos para continuar.

Agradeço ao meu orientador, prof. André, que participou da minha vida acadêmica desde o meu primeiro semestre do mestrado, primeiro como professor e posteriormente como orientador, pois sem ele a realização desta dissertação não seria possível. Foi um grande privilégio poder trabalhar com um pesquisador que admiro e conhecer um ser humano incrível. Serei eternamente grato por todos os momentos em que o senhor foi paciente e solícito comigo, sempre mostrando preocupação e empatia com todos os problemas que eu acabava levando para nossas reuniões de dissertação. Aproveito também para agradecer aos outros professores do PPGEco, que tiveram que se reinventar com a repentina adoção do ensino remoto, todos preocupados com a qualidade do processo de ensino e aprendizagem, sem se esquecer do bem estar da turma. Agradeço também à banca examinadora, composta por economistas que me inspiram, pela avaliação desta dissertação.

Agradeço à família querida que conheci em Florianópolis – Ana Paula, Marcos, Pedrinho e João – que me acolheram desde o primeiro dia em que cheguei assustado com uma vida nova na ilha da magia. Vocês celebraram comigo nos meus momentos felizes, me faziam sentir melhor nos meus momentos de angústia e me davam conforto quando eu sentia saudade de casa. Sinto muito a falta de todos vocês e espero revê-los em breve. Agradeço a minha família por todo o suporte na minha trajetória até aqui: meu pai Eduardo, que mesmo na pandemia, ainda dava um jeitinho de vir me ver; minha mãe Maria, minha base de tudo e maior incentivo do meu esforço; meus irmãos, Rapha e Luís, que são meus companheiros da vida e que a tornam mais leve. Agradeço também aos meus tios, Regina e Éber, que são como segundos pais pra mim; e minha prima Eliara, que é minha irmã e amiga. Muito obrigado a todos vocês por me fazerem quem sou hoje.

Aos meus amigos, peço desculpas pela minha ausência e agradeço muito por tudo que vocês fazem e fizeram por mim. Apesar da singularidade de cada um: Pola, Pedro, Denise, Carol, Gabi, Fran, Rachel, Fer, David, Maria e Gui, obrigado por cada palavra amiga, consolo, abraço e incentivo pra que eu pudesse suportar essa empreitada. Agradeço aos meus amigos virtuais que me aproximei nessa pandemia e me

ajudaram a suportar esse show de horrores: Victor, Renan e Felipe. Vocês todos são incríveis.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001.

“Acho que devemos fazer uma
coisa proibida – senão sufocamos. Mas
sem sentimento de culpa, e sim como
um aviso de que somos livres.”
*(Clarice Lispector, **Um Sopro de Vida**, 1977)*

RESUMO

Previsões de séries temporais são de grande interesse de macroeconomistas, mas encontrar modelos que apresentem resultados mais precisos ainda é um desafio para os pesquisadores da área. Essa dissertação busca avaliar o desempenho preditivo de algoritmos de aprendizagem de máquina como alternativa aos modelos econométricos tradicionais para a previsão de séries macroeconômicas brasileiras, com aplicação de métodos de interpretação agnóstica para melhor compreensão dos resultados fornecidos por aprendizagem de máquina. Com 126 variáveis da economia brasileira entre Fevereiro/2003 até Abril/2021, buscamos prever quatro séries de bastante interesse econômico – o IPCA - Geral, o IBC-Br, a taxa de desemprego da PNAD e a taxa de juros SELIC acumulada no mês – a partir dos modelos de aprendizagem de máquina de regressão penalizada, redes neurais, XGBoost e florestas aleatórias; e de modelos econométricos tradicionais: ARMA, VAR e FAVAR. Nossos resultados mostraram um melhor desempenho dos modelos de aprendizagem de máquina em três das quatro séries avaliadas. Em média para as quatro séries, florestas aleatórias mostraram menores valores para RMSE, já para o MAE os menores resultados foram para regressão penalizada. Por meio da metodologia de Model Confidence Set, também observa-se a presença dos modelos de aprendizagem de máquina entre o conjunto de melhores modelos, em geral com regressão penalizada e florestas aleatórias disputando o pódio de primeiro lugar. A análise de interpretação agnóstica mostrou uma maior importância de preditores referentes aos índices de preços para previsão do IPCA nos modelos de aprendizagem de máquina, ao passo que para o IBC-Br notam-se preditores relacionados ao setor financeiro nacional e internacional. No caso da taxa de desemprego, apesar do melhor desempenho do modelo ARMA na previsão, os modelos de aprendizagem também apontaram uma maior importância para o passado da própria série para a previsão da mesma. A previsão da série da SELIC também mostrou uma importância muito grande da própria série defasada, chamando atenção o resultado de florestas aleatórias que mostrou um comportamento de regra de política monetária para a previsão da série ao incluir variáveis relativas as expectativas sobre o índice de preços IPCA e também de outros índices gerais de preços. Com nossos resultados favorecendo modelos de aprendizagem de máquina, esperamos incentivar o uso dos mesmos juntamente à métodos de interpretabilidade dos resultados em macroeconomia, propiciando melhores previsões que são de grande utilidade aos formuladores de política econômica.

Palavras-chave: Aprendizagem de Máquina. Econometria de Séries Temporais. Macroeconomia.

ABSTRACT

Time series forecasting is of great interest to macroeconomists, however finding models that present more accurate results is still a challenge for researchers in the field. This dissertation looks for evaluating the predictive performance of machine learning algorithms as a alternative for traditional econometric models for the prediction of brazilian macroeconomic time series, with the application of agnostic interpretation methods for a better understanding of the results provided by machine learning. Using 126 variables from the Brazilian economy between February/2003 to April/2021, we seek to predict four series of great economic interest – the inflation rate given by the IPCA - General index, the output growth given by the IBC-Br index, the PNAD unemployment rate and the accumulated SELIC interest rate – using machine learning models of penalized regression, neural networks, XGBoost and random forests; and traditional econometric models: ARMA, VAR and FAVAR. Our results indicated a better performance of machine learning models in three out of the four series evaluated. On average across the four series, random forests showed lower values for RMSE, while for MAE the lowest results were obtained with penalized regression. Through the Model Confidence Set methodology, the presence of machine learning models is also observed among the set of best models, in general with penalized regression and random forests vying for the first place podium. The analysis of agnostic interpretation indicated a greater importance of predictors related to price indexes for forecasting the IPCA in machine learning models, while for the IBC-Br there are predictors related to the national and international financial sector. In the case of the unemployment rate, despite the better performance of the ARMA model in forecasting, the machine learning models also showed greater importance to the lagged value of the series for its prediction. The forecast of the SELIC series also showed a very great importance of the lagged series itself, drawing attention to the result of random forests that showed a behavior of monetary policy rule for the forecasting by including variables related to expectations about the IPCA price index and also other general price indices. With our results favoring machine learning models, we hope to encourage using machine learning methods with methods of agnostic interpretation in macroeconomics, therefore providing better forecasts that are of great importance to economic policymakers.

Keywords: Machine Learning. Time Series Econometrics. Macroeconomics.

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

Anbima	Associação Brasileira das Entidades dos Mercados Financeiro e de Capitais
ANP	Agência Nacional do Petróleo, Gás Natural e Biocombustíveis
Bacen	Banco Central do Brasil
CNI	Confederação Nacional da Indústria
Fecomercio	Federação do Comércio de Bens, Serviços e Turismo do Estado de São Paulo
Fenabrave	Federação Nacional da Distribuição de Veículos Automotores
FGV	Fundação Getulio Vargas
FRED	Dados Econômicos da Reserva Federal dos Estados Unidos (*Federal Reserve Economic Data*)
FUNCEX	Fundação Centro de Estudos do Comércio Exterior
IBGE	Instituto Brasileiro de Geografia e Estatística
IGP	Índice Geral de Preços
INCC	Índice Nacional de Custo de Construção
INPC	Índice Nacional de Preços ao Consumidor
IPA	Índice de Preços por Atacado
IPC	Índice de Preços ao Consumidor
IPCA	Índice Nacional de Preços ao Consumidor Amplo
IPEA	Instituto de Pesquisa Econômica Aplicada
PI	Pesquisa Industrial
Pimes	Pesquisa Industrial Mensal de Emprego e Salário
PIM-PF	Pesquisa Industrial Mensal de Produção Física
PMC	Pesquisa Mensal de Comércio
PNAD	Pesquisa Nacional por Amostra de Domicílios
SECINT	Secretaria Especial de Comércio Exterior e Assuntos Internacionais
SELIC	Sistema Especial de Liquidação e de Custódia
SNIPC	Sistema Nacional de Índices de Preços ao Consumidor

SUMÁRIO

1 INTRODUÇÃO

Presciência sempre foi um assunto que despertou grande interesse e curiosidade humana. Se no passado a humanidade recorria às divindades para descobrir mais sobre o futuro, hoje cientistas utilizam computadores e conceitos estatísticos para predições. Na Economia, é de grande interesse buscarmos metodologias precisas para previsões, uma vez que este tipo de informação é muito relevante, seja para formulação de políticas econômicas ou para melhorar o bem estar da sociedade.

A econometria, um dos pilares básicos na formação de um economista, utiliza conceitos estatísticos para aumentar nossa compreensão sobre problemas econômicos, o que permite testarmos hipóteses sobre a teoria e descobrirmos relações entre variáveis de interesse. Ainda que a econometria tradicional seja uma importante ferramenta, seu uso voltado para previsões é limitado. É notável uma considerável guinada dos economistas na direção do uso de técnicas de aprendizado de máquina (originada do termo em inglês *machine learning*) para este fim.

Diante desta discussão metodológica, é importante pensarmos na macroeconomia. Séries macroeconômicas como desemprego, inflação, crescimento do produto e taxa de juros são indicadores importantes do panorama econômico de um país e vemos cada vez mais esforços de pesquisadores e formuladores de política econômica para perceber tendências, identificar padrões e prever a trajetória dessas variáveis para manter a economia saudável. Por este motivo, diversas questões surgem: quais técnicas de aprendizagem de máquina são boas alternativas para previsões de séries macroeconômicas? Estas técnicas são melhores comparativamente àquelas utilizadas tradicionalmente para previsão? O quão interpretáveis são seus resultados?

Nesta dissertação, buscaremos explorar questões deste nível, estudando a aplicabilidade dos modelos de aprendizagem de máquina de regressão regularizada, árvores de regressão e redes neurais, comparativamente aos resultados do *benchmark* de econometria de séries temporais – ARMA, vetor autorregressivo (VAR) e vetor autorregressivo aumentado por fator (FAVAR). Como se trata de um trabalho empírico, aplicaremos este tipo de metodologia em dados macroeconômicos brasileiros para previsão para quatro variáveis de grande interesse macroeconômico – taxa de crescimento do produto, inflação, taxa de desemprego, taxa de juros nominal – e buscaremos interpretar os resultados de previsão com métodos agnósticos de interpretação de modelos de aprendizagem de máquina: importância de preditor via permutação, efeitos locais acumulados, interação entre preditores e valores de Shapley, que permitem melhor compreensão dos resultados destes modelos que são considerados como "caixas-pretas". Nossa base de dados conta com 126 séries econômicas brasileiras, que compreendem o período de Fevereiro/2003 até Abril/2021, classificadas em nove categorias: crescimento da moeda e política monetária; consumo e vendas; crédito;

emprego, salário e renda; preços; produto e atividade real; financeiro e risco; fiscal e setor externo. Nossos resultados mostraram um bom desempenho dos modelos de aprendizagem de máquina na previsão das séries de interesse, tendo destaque o modelo de florestas aleatórias e regressão penalizada, que apresentaram estatísticas de erro pequenas em todas as séries. A interpretação dos resultados com métodos agnósticos também permitiu uma maior compreensão da importância de séries relacionadas aos índices de preços, financeiras e o do passado da própria variável para a previsão das séries estudadas.

A divisão é feita em seis capítulos: no primeiro capítulo, temos a introdução, revisão da literatura pertinente ao desenvolvimento desta pesquisa e a exposição dos objetivos que buscamos alcançar com este trabalho. O segundo capítulo dedica-se a apresentar a formalização matemática dos modelos de aprendizado de máquina, além dos algoritmos pertinentes a cada um deles que compõem a aplicação computacional. No terceiro capítulo temos os modelos tradicionais utilizados em econometria de séries temporais, com sua formalização matemática. O capítulo quatro apresenta as séries macroeconômicas que compõem nossa base de dados, bem como o tratamento destas séries, seguido da apresentação dos resultados de previsão dos modelos e a análise comparativa da performance entre eles. O capítulo cinco apresenta a teoria dos métodos de interpretação agnóstica dos modelos de aprendizagem de máquina, bem como a aplicação desta técnicas e os resultados obtidos pelos modelos. Por último, o capítulo seis apresenta reflexões e conclusões para esta pesquisa.

1.1 REVISÃO BIBLIOGRÁFICA

Antes mais comumente utilizado pelos cientistas da computação, os modelos de aprendizagem de máquina vem ganhando adeptos em diversas áreas da ciências, como biólogos, estatísticos, cientistas sociais e economistas. Esta crescente adesão ocorre porque nas sociedades contemporâneas temos cada vez mais a oferta de dados, o que permite uma agenda de pesquisa conhecida como *data science* (ou ciência de dados). A revolução empírica na economia da década de 1970 desencadeou uma forte inclinação da produção científica para trabalhos empíricos – e, portanto, utilizando dados – o que fez com que a econometria se consolidasse como base na formação de um economista, o que permite ferramentas necessárias para se aventurarem em outros métodos estatísticos, como o aprendizagem de máquina. Athey e Imbens (2019) buscam sintetizar alguns métodos de aprendizagem de máquina que são úteis aos economistas – visto que, segundo os autores, ainda não há muita adesão deste tipo de metodologia pelos mesmos como ocorre com os estatísticos – uma vez que existe uma variedade considerável de modelos desta classe que diferem entre si e, principalmente, nos oferecem resultados diferentes.

Athey e Imbens (2019) destacam o uso de aprendizagem supervisionada, que

consiste em técnicas de estimação da média condicional de uma variável dado uma quantidade grande de covariáveis. Esta metodologia é dividida em regressão linear regularizada ou penalizada, que inclui modelos *LASSO*, *ridge* e *elastic nets*, que adicionam penalidades à estimação dos parâmetros como forma de reduzir erros de estimação e melhorar o desempenho preditivo; árvores e florestas de regressão, que são comumente utilizadas devido sua flexibilidade em aproximar funções de regressão; aprendizagem profunda e redes neurais, que já aparecem na literatura econométrica na década de 1990s, sendo uma metodologia geral e flexível e que tem bons resultados em ambientes complexos com uma grande quantidade de informações disponíveis; e *boosting*, uma técnica utilizada para melhorar a performance dos métodos de aprendizagem supervisionada. Outra metodologia é o de aprendizagem supervisionada para problemas de classificação, que se divide em árvores e florestas de classificação, máquinas de vetores de suporte e kernel. Os autores também apresentam a aprendizagem não-supervisionada, em que as covariáveis do modelo não possuem um produto, sendo importante dividir os dados em subgrupos. Masini, Medeiros e Mendes (2020) também se dedicam em expor modelos de aprendizagem de máquina pensando em sua aplicação em séries temporais, utilizando modelos lineares e não lineares, além de metodologias híbridas que se mostram vantajosas na aplicação em economia. O artigo traz a formalização matemática dos modelos apresentados; como destaque, apontam um desempenho promissor para o modelo de regressão penalizada e metodologias híbridas.

Seguindo a mesma linha de apresentação, Nosratabadi *et al.* (2020) selecionam um conjunto de metodologias em aprendizagem de máquina – em especial aquelas que são híbridas – úteis em uma gama ampla de aplicações em economia, como mercado de capitais, baseado na previsão do preço de ações; *marketing*, para estudar o comportamento dos consumidores; e criptomoedas, em que busca-se prever o preço das moedas eletrônicas. Modelos híbridos, como apontados pelos autores, são aqueles em que se mesclam dois algoritmos preditivos de aprendizagem de máquina ou um algoritmo de aprendizagem de máquina com um método de otimização que maximize a função de predição. Os autores se dispõem em fazer uma revisão sistemática de metodologias de aprendizagem de máquina e assim mostrar na literatura as aplicações em economia destes modelos.

Pensando em aplicação em séries temporais, Xu (2018) também apresenta aplicações de aprendizagem de máquina que de certa forma propiciam uma melhora dos modelos econométricos tradicionais em economia. Os autores expõem fusão entre modelos de redes neurais com modelos autorregressivos, além de técnicas bayesianas, que propiciem análise causais entre variáveis utilizadas. Babenko *et al.* (2021) também apresentam de forma sintética os métodos “clássicos” de aprendizagem de máquina com aplicações na literatura tanto microeconômica, quanto macroeconômica. De certa

forma, todos os artigos que se propõem a apresentar modelos de aprendizagem de máquina acabam mostrando os mesmos métodos, apesar da riqueza de possibilidades dos diferentes métodos. No Quadro 1 temos a apresentação dos modelos clássicos de aprendizagem de máquina em classes e subclasses.

Quadro 1 – Modelos Clássicos de Aprendizagem de Máquina.

Classe	**Subclasse**	**Método**
Aprendizado Supervisionado	*Regressão*	Regressão Linear
		Regressão Polinomial
		Regressão Penalizada
	Classificação	Regressão logística
		Árvores de Decisão
		Máquina de Suporte de Vetores
		Classificador Naïve Bayes
		k-vizinhos mais próximos
Aprendizado Não-supervisionado	*Agrupamento*	k-médias
		Aglomeração
		Deslocamento médio
		Agrupamento difuso
		DBSCAN
	Motor de Regras	Regras de Associação
		Eclat
		Descoberta Frequente de Padrões (FP)
	Redução de Dimensão	Análise de Componentes Principais
		Mínimos Quadrados Parciais
		Regressão de Componentes Principais
		Vizinhança Estocástica T-distribuída
		Incorporação
		Análise de Discriminante Misto
		Análise de Discriminante Linear

Fonte – Adaptado de Babenko *et al.* (2021) pelo autor.

Ao passo que a abordagem por aprendizagem de máquina é muita vezes vista como oposta à abordagem de econometria, que se sustenta na teoria econômica, Iskhakov, Rust e Schjerning (2020) buscam mostrar situações em que aprendizagem de máquina têm um desempenho melhor em comparação aos modelos tradicionais de econometria. Iskhakov, Rust e Schjerning (2020) destacam que, em se tratando de previsões, aprendizagem de máquina se preocupa em como os computadores fazem estas previsões (sendo mais orientada à prática), enquanto a econometria estrutural se preocupa em como os humanos fazem estas previsões (sendo mais orientada à academia). Na discussão sobre a substituição do conhecimento humano pelas máquinas e também no uso de novas metodologias em econometria estrutural, os autores destacam a perspectiva de Daniel McFadden[1]:

[1] Tradução livre de: *What are the lessons here for econometricians? You should not simply dismiss learning machines and their computer operators as deplorables. You should instead think of them as your worst possible students – ignorant, arrogant, and disinterested. If you can figure out how to*

> Quais são as lições aqui para econometristas? Você não deve simplesmente descartar as máquinas de aprendizagem e seus operadores de computador como deploráveis. Em vez disso, você deve pensar neles como seus piores alunos possíveis - ignorantes, arrogantes e desinteressados. Se você descobrir como avançar e ensiná-los a respeitar e usar o conteúdo científico da economia, eles podem ser seus melhores parceiros de pesquisa. Para manter a econometria estrutural vigorosa e relevante, você precisa continuar a se mover agressivamente para abraçar as inovações na coleta e análise de dados criadas por avanços computacionais. (ISKHAKOV; RUST; SCHJERNING, 2020, p. 37)

1.1.1 Aprendizagem de Máquina e Macroeconomia

D'Orazio (2017), ao entender a ciência como um processo evolucionário, traz uma importante discussão sobre o surgimento de um novo paradigma dentro da macroeconomia: com a grande oferta de dados, os cientistas desta área deveriam adotar novas abordagens epistemológicas e computacionais para lidar com esta agenda de pesquisa. Este novo paradigma traria metodologias que poderiam trazer novos conhecimentos além da nova síntese neoclássica em vigor atualmente na macroeconomia *mainstream*. Chlebus e Świtała (2020) fazem uma revisão da literatura para entender em que ponto artigos científicos econométricos e de aprendizagem de máquina estão competindo entre si, de acordo com resultados esperados. O uso de aprendizagem de máquina em macroeconomia ainda é recente, mas a produção científica da área vem mostrando um crescente uso destas técnicas a fim de previsões. Grande parte desta produção científica ainda é baseada em dados internacionais. Variáveis como taxa de crescimento do produto, inflação e desemprego são as séries macroeconômicas mais utilizadas em aplicações de algoritmos de aprendizagem de máquina.

Yoon (2020) utiliza dados do Japão entre 2001 e 2018 para previsão da taxa de crescimento do PIB, utilizando modelos aprendizagem de máquina de florestas aleatórias (*random forests*). A partir das previsões feitas pelo Fundo Monetário Internacional e o Banco do Japão, que adotam outras metodologias para previsão da variável, o autor percebeu que o uso de aprendizagem de máquina torna as previsões mais precisas, o que encoraja o uso de aprendizagem de máquina para previsão em macroeconomia. Com dados da Tailândia, Chaiboonsri e Wannapan (2019) utilizam modelos de aprendizagem de máquina para previsão de séries macroeconômicas. Eles ressaltam um desempenho melhor em comparação aos modelos paramétricos, justamente porque aprendizagem de máquina consegue capturar informações que estas técnicas mais tradicionais são incapazes de utilizar. Em outro estudo, Chaiboonsri e Wannapan (2020) adotam estimação bayesiana em modelos de aprendizagem de máquina para previsão do produto. Os autores utilizam uma grande quantidade de dados, como

break through and teach them to respect and use the scientific content of economics, they may prove to be your best research associates. To keep structural econometrics vigorous and relevant, you need to continue to move aggressively to embrace the innovations in data collection and analysis created by computational advances.

séries macroeconômicas anuais e informações diárias disponíveis pelo Google Trend, para previsão de ciclos econômicos para um período de três anos. Eles concluem que, apesar das limitações dos modelos de aprendizagem de máquina, houve interessantes resultados para previsão e para a classificação das séries utilizadas. Utilizando redes neurais, Lin *et al.* (2008) também se preocupam com previsão de crises econômicas. Os autores utilizam dados de diversas economias, inclusive a brasileira, e comparam os resultados de aprendizagem de máquina com metodologias tradicionais de previsão, mostrando que o primeiro tem resultados que, além de satisfatórios, também permitiram determinar relações de causalidade entre as variáveis utilizadas.

A taxa de crescimento do produto é alvo de interesse de Paruchuri (2021). O autor utiliza séries macroeconômicas da economia italiana entre 1995 e 2015 e aplicam modelos – regressão de mínimos quadrados ordinários (MQO), auto-regressão não linear (NAR) e com variáveis exógenas (NARX), regressão vetorial de suporte (SVR) e árvores aceleradas (BT). Seus resultados mostram um resultado superior dos modelos de aprendizagem de máquina quando comparados com os resultados dos modelos estatísticos, chamando especial atenção ao modelo NARX que apresentou avisos relevantes sobre crises econômicas do presente estudado pelo autor. Para a taxa de desemprego estadunidense, Hall *et al.* (2018) compara o modelo de regressão penalizada com modelo autorregressivo e de passeio aleatório, mostrando um desempenho melhor do modelo de aprendizagem de máquina. O autor enfatiza os ganhos no uso de aprendizagem para outras séries macroeconômicas, especialmente por conta dos ganhos em precisão de previsão que estes modelos oferecem. O estudo de mestrado de Dimoski e Pettersen (2020) buscam prever o preço das habitações na Noruega a partir de técnicas de aprendizagem de máquina, comparando seus resultados com os dados publicados pelo banco central do país. Seus resultados mostram um bom resultado dos modelos, mas que não foram capazes de ter uma melhor performance que as previsões do banco central norueguês de forma geral, mas sim em períodos específicos do período total da amostra. No doutorado, Almosova (2019) utiliza redes neurais para previsão da inflação dos Estados Unidos em diferentes horizontes, comparando os resultados com técnicas tradicionais, como o modelo autorregressivo e o passeio aleatório, mostrando um desempenho muito superior para o resultado dos modelos de aprendizagem.

Outros trabalhos comuns em aplicações de aprendizagem de máquina em macroeconomia são aqueles em que comparamos esta classe de modelos com metodologias tradicionais em macroeconometria. Fen e Undavia (2021) adotam modelos de redes neurais e modelos automáticos de aprendizagem de máquina e comparam os resultados com técnicas tradicionais, como modelos autorregressivos e vetores autorregressivos, além de modelos estruturais, muito utilizados em macroeconomia: os Modelos Dinâmicos e Estocásticos de Equilíbrio Geral (DSGE). Os autores utilizam

uma grande quantidade de dados referentes ao PIB de diferentes países, mostrando que o uso deste painel de dados permite uma redução significativa das estatísticas de erro para os modelos, melhorando a performance preditiva dos modelos de aprendizagem de máquina comparativamente aos outros modelos. Com dados da economia japonesa, Maehashi e Shintani (2020) comparam resultados de previsão de uma gama variada de modelos de aprendizagem de máquina com os modelos de fatores e os autorregressivos – tendo aprendizagem de máquina e modelos de fatores um resultado superior ao modelo autorregressivo tradicional. Os autores utilizaram uma extensa base de dados, com 219 variáveis mensais de um período de 01/1973 até 06/2018. Em sua dissertação, Martin (2019) compara os resultados de modelos autorregressivos e vetores autorregressivos com modelos de aprendizagem de máquina – regressão penalizada, máquina de suporte de vetores, redes neurais e floresta aleatória – mostrando um desempenho melhor deste último grupo. A autora utiliza prevê o crescimento do produto sul-africano e conclui que o uso de aprendizagem de máquina é benéfico aos formuladores de política econômica. Para séries econômicas da economia italiana entre 1995 e 2019, Cicceri, Inserra e Limosani (2020) também comparam modelos tradicionais com aprendizagem de máquina. Os autores utilizam uma grande quantidade de modelos obtém interessantes resultados preditivos para as crises econômicas italianas para o período, dada a performance superior dos modelos de aprendizagem de máquina na previsão do PIB italiano.

Já o estudo de Medeiros *et al.* (2019) baseia-se dados de inflação da economia estadunidense para fins de previsão com adoção de aprendizagem de máquina, que se mostra muito mais precisa do que as metodologias convencionais, principalmente na presença de muitas covariáveis. Os autores também revelam um melhor desempenho do modelo de florestas aleatórias, o que mostra certo consenso da literatura que estuda previsões de séries macroeconômicas. Isto ocorre porque esta classe de modelos permite utilizar uma relação não linear entre variáveis macroeconômicas e inflação. Garcia, Medeiros e Vasconcelos (2017), com uso de modelos de aprendizagem de máquina, estimam a inflação brasileira em tempo real (isto é, somente com a informação disponível ao econometrista no momento em que a previsão é feita), visto que esta variável apresenta uma grande volatilidade de curto prazo em países emergentes. Os autores utilizam as previsões feitas por especialistas como *benchmark* e como preditor para os modelos de previsão – modelo de fatores, LASSO/adaLASSO, florestas aleatórias e regressão de subconjunto completo. O desempenho dos modelos variou bastante com o horizonte de previsão que se buscou estimar, com destaque do modelo LASSO que teve mesma performance do relatório FOCUS para previsão de t+1 períodos, ao passo que o adaLASSO teve um desempenho melhor para um horizonte t+2. O melhor resultado veio quando construiu-se uma previsão que utiliza a média dos resultados dos modelos, sendo superior a todas as alternativas. No trabalho de

Babii, Ghysels e Striaukas (2021), desta vez com previsão para a taxa de crescimento do PIB estadunidense, os modelos LASSO também mostram um desempenho muito promissor, principalmente pela performance dessa classe de modelos em questões de grandes dimensões.

Usando dados para a economia brasileira, Speranza, Tanscheit e Vellasco (2020) adotam métodos de aprendizagem de máquina para previsão da inflação brasileira. Os autores lidam três relações macroeconômicas: a lei de Okun, que relaciona desemprego e a taxa de crescimento do produto; a curva de Phillips, que relaciona a expectativa de inflação e a taxa de desemprego; e o efeito da liquidez, que relaciona a taxa de juros definida pelo banco central e a taxa de crescimento do produto. Os autores aplicam algoritmos genéticos e regressões de redes neurais e seus resultados mostram que se o Banco Central do Brasil tivesse utilizado previsões baseadas na metodologia utilizada pelos pesquisadores, sob as mesmas condições econômicas, a inflação estaria abaixo daquela alcançada pelos modelos DSGE em 62,8% do tempo, ao passo que o desemprego estaria abaixo do alcançado em 39,69% do tempo, evidenciando o *trade-off* entre as variáveis descritas pela Curva de Phillips. Não só para a política monetária, a aprendizagem de máquina também vem sendo vista como promissora para formuladores de política econômica. Genberg e Karagedikli (2021) também se preocupam com a aplicação de aprendizagem de máquina em bancos centrais. Ao comparar as alternativas disponíveis a partir de uma "fronteira de Pagan", os autores classificam os modelos de aprendizagem mais consistentes com os dados e menos consistentes com a teoria – posição contrária dos modelos DSGE nesta classificação – mas que estão em mesma posição dos modelos VAR, mais aceitos pela macroeconometristas. Os autores concluem que aprendizagem de máquina é capaz de alcançar os mesmos objetivos dos VAR, mas com resultados melhores em termos de previsão. Em estudo do Banco Central Europeu, Hirschbühl, Onorante e Saiz (2021) utilizando aprendizagem de máquina, buscam analisar os ciclos de negócios das economias europeias a partir de quatro casos: o mercado de trabalho durante a pandemia da COVID-19; a atividade econômica semanal para a área euro; a previsão em tempo real do crescimento real do PIB; e impacto de incerteza de políticas econômicas em componentes da demanda. Os autores identificam a aprendizagem de máquina como importantes ferramentas auxiliares para os métodos tradicionais, não podendo ser substitutas por conta da falta de interpretabilidade e de inferência estatística dessas ferramentas, ainda que existam avanços – como é o caso do uso de valores de Shapley, uma ferramenta de interpretação agnóstica.

1.1.2 Interpretabilidade dos Modelos de Aprendizagem de Máquina

Interpretabilidade dos modelos de aprendizagem de máquina também chama atenção de outros pesquisadores de macroeconomia. Buckmann, Joseph e Robertson

(2021), utilizando nove séries de dados da economia estadunidense para previsão taxa de desemprego, comparam os resultados de previsão de modelos de aprendizagem de máquina com metodologias tradicionais de previsão de séries temporais – regressão linear e modelo autorregressivo – além de aplicarem técnicas de interpretabilidade para melhor compreensão dos resultados obtidos. Seus resultados mostram um resultado superior para os modelos de aprendizagem de máquina, em especial dos modelos de floresta aleatórias, que a partir da técnica de regressão de Shapley, permitiram concluir um papel relevante de variáveis como produção industrial e S&P500 para a previsão do desemprego estadunidense. Bluwstein *et al.* (2021) utilizam modelos de aprendizagem de máquina para previsão de crises econômicas. Com dados de 17 países, os autores adotam modelos de redes neurais, florestas aleatórias, máquina de vetores de suporte e árvores extremamente aleatórias, comparando os resultados com a regressão logística. Suas conclusões mostram um resultado superior para os modelos de aprendizagem de máquina comparados ao método estatístico, identificando por meio dos valores de Shapley uma importância maior das séries relacionadas ao mercado de crédito.

Dada a aparente lacuna dos modelos aprendizagem de máquina no sentido de interpretabilidade dos resultados à luz da teoria macroeconômica, Coulombe *et al.* (2020) buscam encontrar o elo entre os métodos de aprendizagem de máquina e as metodologias padrão da macroeconomia, baseando-se em quatro características: não-linearidade, que muito embora seja muito popular utilizar modelos linearizados, a vantagem de utilizarmos funções não-lineares permitem previsões com erros menores caso o processo gerador de dados também seja não linear; regularização, que é o processo de diminuir a dimensão da estimação para os casos em que ocorre a presença de muitas covariáveis; validação cruzada, que serve como um critério de seleção semelhante aos critérios de informação Akaike e Schwarz–Bayesian, mais comumente utilizado em macroeconometria; e funções de perda alternativas, já que é mais comum o uso da função de erro quadrática. Segundo Coulombe *et al.* (2020), utilizar modelos não-lineares têm uma diferença significativa em previsões macroeconômicas, principalmente em cenários de grande incerteza, fricções financeira e bolhas financeiras associadas às hipotecas.

Depois da contribuição sobre porquê o uso de aprendizagem de máquina é interessante para a macroeconomia, Coulombe (2020) propõe um algoritmo batizado como *Macroeconomic Random Forest* – MRF – cujos resultados são interpretáveis à luz da teoria macroeconômica por meio dos parâmetros generalizados variando no tempo – GTVPs – bem como seu forte poder preditivo. O modelo MRF foi capaz de prever a queda no desemprego durante a crise de 2008, bem como previsão da inflação para o mesmo período. Além disso, o autor discute a relação da curva de Phillips e o MRF, em que o modelo foi capaz de identificar tanto mudanças estruturais quanto

mudanças na atividade econômica e da inflação, sendo uma importante ferramenta para testar a veracidade de 'relações controversas' dentro da teoria macroeconômica, como é o caso do *trade-off* inflação e desemprego.

Na Tabela 1, temos a síntese dos artigos empíricos aplicados em Macroeconomia citados nesta revisão da literatura, permitindo uma comparação facilitada para o leitor.

Tabela 1 – Síntese dos Artigos Empíricos em Macroeconomia.

Autor(es)	**Modelos ML**	**Série(s) alvo**	**País(es)**
Yoon (2020)	Florestas Aleatórias, Modelos de Aceleração por Gradiente	Crescimento do PIB	Japão
Chaiboonsri e Wannapan (2019)	Árvores de Decisão, SVM, k-NN	PIB	Tailândia
Chaiboonsri e Wannapan (2020)	Série Temporal Estrutural Bayesiano, Análise discriminante, k-NN	PIB	Tailândia
Lin *et al.* (2008)	Fuzzy Neural, Redes Neurais	PIB	Vários
Paruchuri (2021)	SVR, Árvores Aceleradas, NAR, NARX, Regressão Linear	PIB	Itália
Hall *et al.* (2018)	Regressão Penalizada	Taxa de Desemprego	EUA
Dimoski e Pettersen (2020)	Redes Neurais, Redes Elásticas, Floresta Aleatória	Preços de Habitação	Noruega
Almosova (2019)	Redes Neurais	Inflação	USA
Fen e Undavia (2021)	Redes Neurais, Modelos Automáticos de ML	PIB	Vários
Maehashi e Shintani (2020)	LASSO, Redes Neurais, Árvores Aceleradas, Florestas Aleatórias	Produção Industrial, Taxa de Desemprego, Salário Real, Consumo Real, Índice de Preços	Japão
Martin (2019)	Regressão Penalizada, SVM, redes neurais, florestas aleatórias	Crescimento PIB	África do Sul
Cicceri, Inserra e Limosani (2020)	SVR, NAR, NARX, k-NN, Árvores Aceleradas	Crescimento do Produto	Itália
Medeiros *et al.* (2019)	LASSO, adaLASSO, ElNet, adaElNet, Ridge, BVAR, Floresta Aleatória	Inflação	USA
Garcia, Medeiros e Vasconcelos (2017)	Modelo de Fatores, LASSO, adaLASSO, Florestas aleatórias, regressão de subconjunto completo	Inflação	Brasil
Speranza, Tanscheit e Vellasco (2020)	Algoritmos genéticos, redes neurais	Inflação	Brasil

Continua na próxima página.

Continuação da Tabela 1.

Autor(es)	**Modelos ML**	**Série(s) alvo**	**País(es)**
Hirschbühl, Onorante e Saiz (2021)	Vários	Desemprego, PIB	Europa
Buckmann, Joseph e Robertson (2021)	Florestas Aleatórias, Redes neurais, LASSO, Ridge, SVR	Taxa de Desemprego	EUA
Bluwstein *et al.* (2021)	redes neurais, florestas aleatórias, SVM, árvores extremamente aleatórias	PIB	Vários
Coulombe *et al.* (2020)	SVR, Regressão linear, SVM, Florestas Aleatórias	Taxa de Desemprego, Inflação, Produção Industrial	EUA
Coulombe (2020)	Florestas Aleatória	Inflação, Desemprego	EUA

Fonte: Produzido pelo autor.

2 MODELOS DE APRENDIZAGEM DE MÁQUINA PARA PREVISÃO DE SÉRIES TEMPORAIS

Existem uma quantidade considerável de modelos utilizados para estimação e previsão de séries temporais, sejam eles parte da econometria ou de aprendizagem de máquina. Nesta dissertação buscaremos utilizar métodos de aprendizagem de máquina tanto lineares quanto não lineares. Descreveremos a seguir os modelos utilizados baseados em Friedman, Hastie e Tibshirani (2001), livro de referência para estudo dos algoritmos de aprendizagem de máquina.

2.1 MÉTODOS LINEARES DE REGRESSÃO

Modelos de regressão lineares são aqueles em que assume-se que a função de regressão $E(Y|X)$ é linear nas variáveis independentes $X^T = (X_1, X_2, \ldots, X_p)$. Uma forma generaliza de um modelo de regressão linear é dada pela Equação (1):

$$f(X) = E(Y|X_1, \ldots, X_p) = \beta_0 + \sum_{j=1}^{p} X_j \beta_j. \tag{1}$$

Independente de como for X_j, a linearidade nesta classe de modelos é definida pela linearidade dos parâmetros β. Para a aplicação do modelos, utilizamos um conjunto de dados de treinamento observáveis $(x_1, y_1) \ldots (x_N, y_N)$ de tal forma que é possível estimarmos os parâmetros $\beta = (\beta_0, \beta_1, \ldots, \beta_p)^T$. A forma mais comum de estimarmos o vetor de parâmetros é por meio do método de mínimos quadrados em que desejamos minimizar a soma do quadrado dos resíduos na seguinte forma:

$$\begin{aligned} SQR(\beta) &= \sum_{i=1}^{N} (y_i - f(x_i))^2 \\ &= \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij} \beta_j \right)^2 . \end{aligned} \tag{2}$$

Para minimizarmos a Equação (2), adotamos X como uma matriz $N \times (p+1)$ em que cada linha representa um vetor de *inputs* com o primeiro elemento igual à 1 e y é um vetor N dimensional de produtos, ambos do conjunto de treinamento. Assim, a soma do quadrado dos resíduos pode ser reescrita como a função quadrática para os $p+1$ parâmetros:

$$SQR(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T (\mathbf{y} - \mathbf{X}\beta). \tag{3}$$

Diferenciando em relação à β obtemos que:

$$\begin{aligned} \frac{\partial SQR}{\partial \beta} &= -2\mathbf{X}^T (\mathbf{y} - \mathbf{X}\beta), \\ \frac{\partial^2 SQR}{\partial \beta \partial \beta^T} &= 2\mathbf{X}^T \mathbf{X}. \end{aligned} \tag{4}$$

Assumindo que $\mathbf{X}$ possui posto completo[1] e que a matriz $\mathbf{X}^T\mathbf{X}$ é positiva definida, podemos definir a primeira derivada igual à zero tal que:

$$\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta) = 0. \tag{5}$$

O que nos permite encontrar a solução única $\widehat{\beta}$:

$$\widehat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}. \tag{6}$$

Os valores previstos para a variável dependente dado um vetor de *inputs* x_0 é dado por $\widehat{f}(x_0) = (1 : x_0)^T\widehat{\beta}$, o que significa que para o vetor de *inputs* do conjunto de treinamento teremos:

$$\widehat{\boldsymbol{y}} = \mathbf{X}\widehat{\beta} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}, \tag{7}$$

em que $\widehat{y}_i = \widehat{f}(x_i)$ e $\mathbf{H} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ é chamada matriz chapéu ou matriz de projeção. Ela é utilizada para computar a projeção ortogonal, uma vez que ao minimizarmos a soma do quadrado dos resíduos escolhendo um $\widehat{\beta}$, então o vetor residual $\boldsymbol{y} - \widehat{\boldsymbol{y}}$ é ortogonal ao subespaço gerado pelas variáveis preditoras x_i. A ortogonalidade é mostrada pela Equação (5) e garante que estimando $\widehat{\boldsymbol{y}}$ seja a projeção ortogonal de $\mathbf{y}$ em relação ao seu subespaço.

2.1.1 Métodos de Encolhimento

Métodos de encolhimento são aqueles em que é selecionado um subconjunto de variáveis dentro do conjunto de variáveis explicativas $X^T = (X_1, X_2, \ldots, X_p)$. Existe certa variedade na oferta de modelos que utilizam métodos de encolhimento, mas os mais utilizados são a regressão ridge e o lasso, que serão apresentadas nesta seção.

2.1.1.1 Regressão Ridge

A regressão Ridge impõe encolhimento na quantidade de parâmetros da regressão ao aplicar uma penalidade em relação ao seu tamanho. Os coeficientes $\widehat{\beta}^{ridge}$ são aqueles que minimizam a soma do quadrado dos resíduos, a partir de um parâmetro $\lambda \geq 0$ que controla a proporção de encolhimento da regressão (quanto maior o valor de λ, maior o encolhimento). Os coeficientes $\widehat{\beta}^{ridge}$ são dados pela seguinte expressão:

$$\widehat{\beta}^{ridge} = \arg\min_{\beta} \left\{ \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}. \tag{8}$$

Na Equação (8), o intercepto β_0 foi deixado de lado, pois caso fosse considerado, a penalização dependeria da origem de Y, pois a simples adição de uma constante

[1] Matrizes com posto completo são aquelas em que todas as linhas e/ou colunas são linearmente independentes. Nem sempre conseguimos isso com nosso conjunto de dados de treinamento o que implica em $\widehat{\beta}$ não seja único, sendo comum a exclusão das colunas redundantes em $\mathbf{X}$.

para cada uma das observações y_i não terá um efeito uniforme para a previsão de Y. Dessa forma, adotamos a matriz X como tendo apenas p colunas, ao invés de $p+1$ colunas como mostramos no método de mínimos quadrados ordinários. Reescrevendo o critério da Equação (8) na forma matricial:

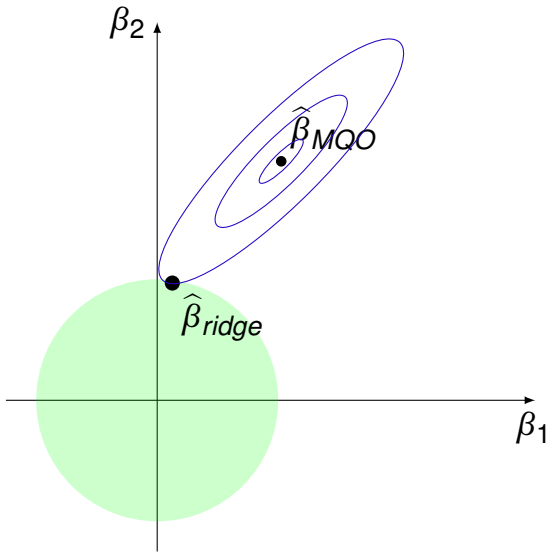
$$SQR(\lambda) = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta) + \lambda\beta^T\beta. \tag{9}$$

O resultado da regressão de ridge será como:

$$\widehat{\beta}^{ridge} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}, \tag{10}$$

em que I é a matriz identidade $p \times p$. A adição de uma constante positiva em $\mathbf{X}^T\mathbf{X}$ antes da inversão desta matriz garante que ela seja não singular[2], mesmo que $\mathbf{X}^T\mathbf{X}$ não possua posto completo – sendo essa a principal motivação de adoção da regressão de ridge para estatísticos, que permitia contornar o problema em que a matriz de covariáveis não possuía posto completo. A Figura 1 mostra a comparação entre os estimadores de ridge e MQO, para o caso bidimensional. A área do círculo, definida como $\beta_1^2 + \beta_2^2 \leq \lambda$, representa a restrição imposta nos parâmetros para o modelo ridge, em que o estimador para o mesmo é encontrado pela interseção com a elipse que representa a soma do quadrado dos resíduos (SQR), conforme ele se distancia do seu ponto de mínimo encontrado em $\widehat{\beta}_{MQO}$.

Figura 1 – Comparação entre os estimadores de ridge e MQO para o caso de duas covariáveis.



Fonte – Adaptado de Friedman, Hastie e Tibshirani (2001) pelo autor.

[2] Uma matriz quadrada é dita singular quando ela não admite inversa.

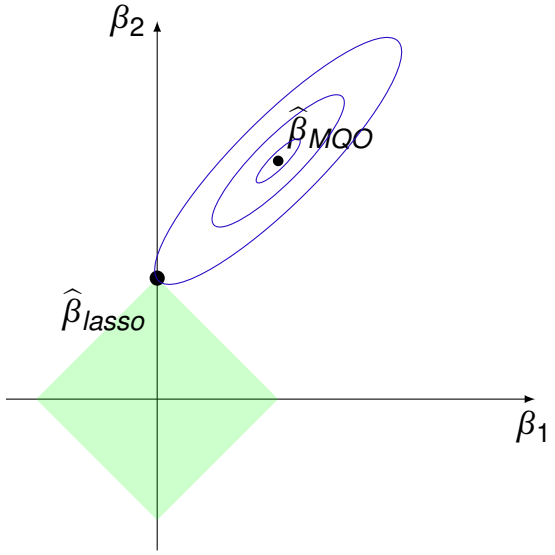
2.1.1.2 Operador de Encolhimento Absoluto Mínimo e Seleção (LASSO)

Também conhecido na literatura como busca de base, o método de encolhimento de lasso tem certa similaridade com a regressão ridge, ainda que seus resultados sejam diferentes devidos a diferentes penalidades aplicadas. Na regressão ridge, a penalidade $\sum_1^p \beta_j^2$ é chamada L_2, ao passo que em lasso temos uma penalidade L_1 na forma $\sum_1^p |\beta_j|$. O problema na forma de Lagrange pode ser escrito como:

$$\widehat{\beta}^{lasso} = \arg\min_{\beta} \left\{ \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}. \tag{11}$$

A penalidade L_1 faz a solução para $\widehat{\beta}^{lasso}$ ser não linear em y_i e, portanto, não há forma fechada para expressarmos os resultados desses coeficientes como fazemos para a regressão ridge, necessitando técnicas de programação quadrática para resolvermos o problema analiticamente. Na Figura 2 podemos enxergar a comparação entre os estimadores para β em lasso e MQO, para o caso com duas covariáveis. A área do losango é a restrição imposta pelo modelo lasso e o estimador de lasso é obtido pela interseção entre as elipses do estimador de MQO e a restrição, tendo portanto uma solução de canto em que o parâmetro β_1 é igual a zero, o que indica que este tipo de método é capaz de selecionar variáveis ao minimizar os erros.

Figura 2 – Comparação entre os estimadores de lasso e MQO para o caso de duas covariáveis.



Fonte – Adaptado de Friedman, Hastie e Tibshirani (2001) pelo autor.

2.1.1.3 A relação entre regressão ridge e lasso

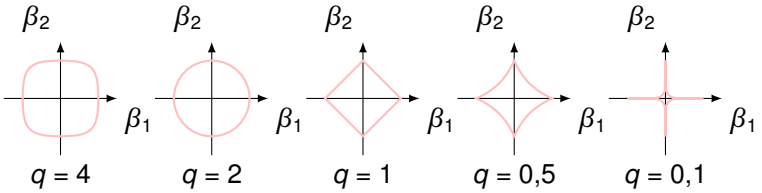
Tanto a regressão ridge quanto lasso aplicam mudanças na estimação de $\widehat{\beta}$ mínimos quadrados. Ao passo que a regressão ridge aplica um encolhimento propor-

cional ao tamanho do parâmetro, lasso submete cada um dos coeficientes ao fator λ, que possibilita que parte deles trunquem em zero. Mesmo com essas diferenças, é possível generalizar estas estimações de tal forma que tenhamos uma expressão que comporta ambos os métodos de encolhimento:

$$\tilde{\beta} = \arg\min_{\beta} \left\{ \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|^q \right\}, \tag{12}$$

em que $q \geq 0$, sendo esta penalização conhecida como L_q. Nesse caso, além da escolha do parâmetro λ, também é preciso escolher o parâmetro q, que terá diferentes formas para a região de restrição, conforme a Figura 3 mostra. O caso em que $q = 1$, em que temos o modelo lasso, é o menor valor para q tal que a função seja convexa. Já para a regressão de ridge, temos $q = 2$. Quando $q \leq 1$ a função tende a se concentrar mais na direção das coordenadas, gerando um trabalho bem maior para o problema de otimização porque as regiões não são convexas, sendo em geral adotado $q \in (1,2)$, que mostra certo compromisso entre ambas as abordagens.

Figura 3 – Comparação de diferentes valores de q na restrição L_q para o caso de duas covariáveis.



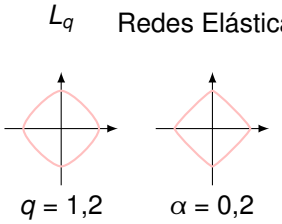
Fonte – Adaptado de Friedman, Hastie e Tibshirani (2001) pelo autor.

Para poupar trabalho computacional, é comum utilizar a penalização conhecida como redes elásticas, que também mostra um comprometimento entre as duas técnicas de encolhimento do modelo. O termo de penalização do modelo de redes elásticas é descrito como:

$$\lambda \sum_{j=1}^{p} (\alpha\beta_j^2 + (1-\alpha)|\beta_j|). \tag{13}$$

A penalidade de redes elásticas é capaz de produzir resultados semelhantes à penalidade L_q por um menor custo computacional, permitindo selecionar variáveis como o modelo lasso e encolher os coeficientes como a regressão de ridge. Na Figura 4, é possível vermos a semelhança entre as duas penalidades, quando $q = 1{,}2$ e $\alpha = 0{,}2$. Entretanto, é importante notar que os vértices para redes elásticas são quinas e, portanto, não diferenciáveis; ao passo que para a restrição L_q os vértices são diferenciáveis.

Figura 4 – Comparação da penalidade L_q e da penalidade de redes elásticas para o caso de duas covariáveis.



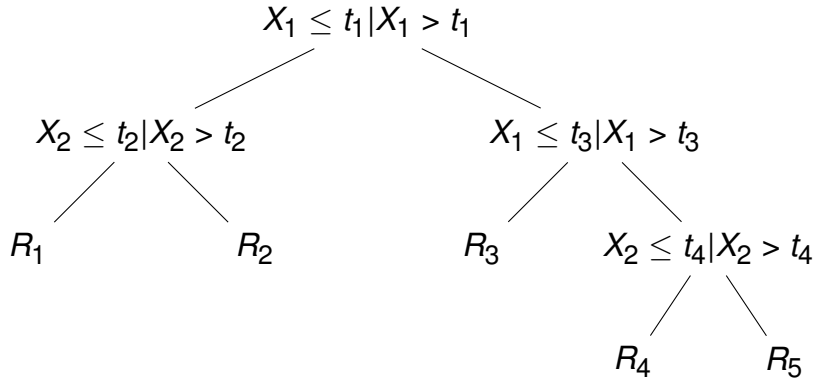
Fonte – Adaptado de Friedman, Hastie e Tibshirani (2001) pelo autor.

2.2 MÉTODOS BASEADOS EM ÁRVORES

Métodos baseados em árvores são técnicas não lineares que envolvem 'segmentar' o espaço de preditores em um número de regiões simples, em que a média ou a moda das observações de treinamento destas regiões são utilizadas para previsão. As regras de divisão usadas para segmentar o espaço de preditores é chamado de árvore, por isto estes métodos também são chamados de árvores de decisão. Os métodos baseados em árvores são de certa forma simples e interpretáveis quando não há uma quantidade muito grande de preditores – em geral, adicionar muitas árvores aumenta a precisão da previsão, ao custo de perdermos capacidade interpretativa.

A Figura 5 demonstra a divisão binária de regiões para o caso em que queremos estudar uma variável Y em relação à outras duas variáveis, X_1 e X_2. A divisão de regiões é feita até que uma regra de parada seja aplicada. No topo da árvore temos todo o conjunto de dados disponíveis, até que é aplicado uma divisão de regiões para $X_1 = t_1$. Se $X_1 \leq t_1$, é seguido para o braço esquerdo, seguindo para o braço direito caso contrário; assim sucessivamente, resultando em cinco regiões $R_1, R_2, \ldots, R_5$.

Figura 5 – Árvore de regressão



Fonte – Adaptado de Friedman, Hastie e Tibshirani (2001) pelo autor.

A regressão que prevê Y sob uma constante c_m – a média ou mediana das

observações na região R_m – é dada pela Equação (14):

$$\widehat{f}(X) = \sum_{m=1}^{5} c_m I\{(X_1, X_2) \in R_m\}. \tag{14}$$

2.2.1 Árvores de Regressão

Para a construção de árvores de regressão para um conjunto de dados na forma (x_i, y_i) para $i = 1,2,\ldots,N$ e $x_i = (x_{i1}, x_{i2}, \ldots, x_{ip})$, o algoritmo para a árvore de regressão precisa decidir quais serão as variáveis de separação e quais serão os pontos de separação (semelhante ao que observamos na Figura 5). Com uma divisão em M regiões $R_1, R_2, \ldots, R_M$, temos o modelo em resposta a uma constante c_m para cada uma das regiões:

$$f(x) = \sum_{m=1}^{M} c_m I(x \in R_m). \tag{15}$$

Se o critério de minimização utilizado é a soma dos quadrados $\sum (y_i - f(x_i))^2$, o melhor $\widehat{c}_m$ é a média de y_i na região R_m, ou seja:

$$\widehat{c}_m = \text{média}(y_i | x_i \in R_m). \tag{16}$$

Para encontrar a melhor partição binária – levando em conta que o caminho pela minimização da soma dos quadrados não é factível, mesmo computacionalmente – é utilizado um algoritmo. Com todo o conjunto de dados disponíveis, a partir da variável de repartição j e o ponto de partição s, é definido o par de planos:

$$\begin{aligned} R_1(j,s) &= \{X | X_j \leq s\}, \\ R_2(j,s) &= \{X | X_j > s\}. \end{aligned} \tag{17}$$

Para isso buscamos a variável j e o ponto de partição s que resolva:

$$\min_{j,s} \left[\min_{c_1} \sum_{x_i \in R_1(j,s)} (y_i - c_1)^2 + \min_{c_2} \sum_{x_i \in R_2(j,s)} (y_i - c_2)^2 \right]. \tag{18}$$

Para qualquer j e s, a minimização é encontrada por:

$$\begin{aligned} \widehat{c}_1 &= \text{média}(y_i | x_i \in R_1(j,s)), \\ \widehat{c}_2 &= \text{média}(y_i | x_i \in R_2(j,s)). \end{aligned} \tag{19}$$

Após encontrar o melhor ponto de partição s, o processo de partição é então repetido para todas as regiões resultantes. Entretanto, o tamanho da árvore é bastante problemático: se for uma árvore muito grande, ela pode se sobreajustar em relação aos dados, ao passo que uma árvore muito pequena pode não capturar de forma adequada a estrutura do modelo. Nesse sentido, o tamanho da árvore é um parâmetro

de ajustamento (*tuning parameter*) – assim como o λ na regressão regularizada – e deverá ser escolhido condicionado à base de dados, escolhendo qual seria o melhor tamanho para a árvore. A estratégia utilizada é de criar uma árvore grande T_0, que pára o processo de partição quando alcança um determinado tamanho mínimo de nó. Esta árvore então é submetida a um processo de poda usando uma relação de *custo de complexidade* do tamanho da árvore.

Seja a sub-árvore $T \subset T_0$ obtida pelo método de poda de T_0 e os nós terminais m, em que o nó m representa a região R_m. Seja $|T|$ a representação do número de nós terminais em T. Assim, temos que:

$$\begin{aligned} N_m &= \#\{x_i \in R_m\}, \\ \widehat{c}_m &= \frac{1}{N_m} \sum_{x_i \in R_m} y_i, \\ Q_m(T) &= \frac{1}{N_m} \sum_{x_i \in R_m} (y_i - \widehat{c}_m)^2. \end{aligned} \tag{20}$$

Em que N_m é o número de variáveis x_i na região R_m, $\widehat{c}_m$ é a média amostral para y_i dentro da região R_m e $Q_m(T)$ é a diferença quadrática média[3] entre y_i e $\widehat{c}_m$. Com o critério de custo de complexidade definido como:

$$C_\alpha(T) = \sum_{m=1}^{|T|} N_m Q_m(T) + \alpha|T|. \tag{21}$$

O problema consiste em encontrar, para cada $\alpha \geq 0$ responsável por mensurar o *tradeoff* entre tamanho da árvore e sua capacidade de se adequar aos dados, a sub-árvore $T_\alpha \subseteq T_0$ que minimize $C_\alpha(T)$. Valores muito altos para α resultam em T_α pequenas, ao passo que $\alpha = 0$ temos a árvore T_0. Assim, para cada α temos uma única sub-árvore T_α que minimiza $C_\alpha(T)$; a árvore T_α é encontrada quando colapsamos sucessivamente os nós internos que produzem o menor acréscimo por nó em $\sum_m N_m Q_m(T)$, até que conseguimos produzir uma árvore com um único nó (raiz). Esse processo gera uma sequência finita de sub-árvores que contém T_α. A estimação de α é obtida por validação cruzada, em que se escolhe $\widehat{\alpha}$ que minimiza a soma dos quadrados de validação cruzada[4], obtendo $T_{\widehat{\alpha}}$.

2.3 MÉTODOS DE ACELERAÇÃO

Métodos de aceleração (*boosting methods*) foram um importante avanço para previsões em modelos de classificação, que posteriormente permitiu também aplicações em regressão. A ideia principal por trás destes métodos é combinar diversos

[3] A equação para $Q_m(T)$ é semelhante ao erro quadrático médio, que é utilizado para comparar resultados de estimadores em estatística, sendo um critério de seleção para o melhor estimador.

[4] Validação cruzada é discutido na Seção 2.5.2.

preditores “fracos” – em que preditor fraco é aquele onde a taxa de erro para previsão é apenas marginalmente melhor do que um palpite aleatório – de tal forma que seja possível criar um “comitê” de preditores fortes, permitindo otimizar uma função de perda e permitindo resultados de previsão melhores.

2.3.1 Aceleração para Árvores de Decisão

Conforme vimos na Seção 2.2, previsões utilizando árvores particionam as variáveis preditoras em regiões disjuntas R_j, $j = 1, 2, \ldots, J$, que representam os nós terminais (ou folhas) da árvore. Uma constante γ_j é definida pela seguinte regra:

$$x \in R_j \Rightarrow f(x) = \gamma_j, \tag{22}$$

ao passo que uma árvore pode ser formalmente expressa em função dos parâmetros $\Theta = \{R_j, \gamma_j\}_1^J$:

$$T(x;\Theta) = \sum_{j=1}^{J} \gamma_j I(x \in R_j). \tag{23}$$

Os parâmetros em Θ podem ser encontrados através da minimização da função de perda:

$$\widehat{\Theta} = \arg\min_{\Theta} \sum_{j=1}^{J} \sum_{x_i \in R_j} L(y_i, \gamma_j). \tag{24}$$

O procedimento de otimização para o problema de minimização é dado em duas etapas: a primeira, é encontrar γ_j para um dado R_j, pois em geral a estimação para este parâmetro é dado pela média das observações de y_i na região R_j; a segunda, mais trabalhosa computacionalmente, é encontrar a região R_j. São necessários algoritmos de reparticionamento recursivo para encontrá-la, mas em geral substitui-se a Equação (24) por uma versão mais suave (embora que, para $\widehat{R}_j = \tilde{R}_j$, γ_j é estimado mais precisamente pelo critério original):

$$\tilde{\Theta} = \arg\min_{\Theta} \sum_{i=1}^{N} \tilde{L}(y_i, T(x_i, \Theta)). \tag{25}$$

Para encontramos o modelo acelerado de árvores, é aplicado um algoritmo de adição progressiva para o problema de minimização. São feitas m interações, cada uma delas buscando otimizar a função de base do argumento multifatorial x e o conjunto de parâmetros γ, que representa as variáveis e os pontos de partição dos nós internos e as previsões dos nós terminais. Em geral a função base de expansão é definida como $f(x) = \sum_{m=1}^{M} \beta_m b(x;\gamma_m)$, em que $\beta = \beta_1, \ldots, \beta_M$ são os coeficientes de expansão e $b(x;\gamma_m) \in \mathbb{R}$. A seguir, é aplicado o seguinte o algoritmo 1 genérico.

Algoritmo 1: Modelo de Adição Progressiva

1. Inicializa $f_0(x) = 0$.
2. Para $m = 1, \ldots, M$:
   a) Computa
   $$(\beta_m, \gamma_m) = \arg\min_{\beta,\gamma} \sum_{i=1}^{N} L(y_i, f_{m-1}(x_i) + \beta b(x_i; \gamma))$$
   b) Define $f_m(x) = f_{m-1}(x) + \beta_m b(x; \gamma_m)$

Assim, o modelo acelerado para as árvores é dado pela soma das árvores:

$$f_M(x) = \sum_{m=1}^{M} T(x; \Theta_m), \tag{26}$$

e, para cada etapa, o algoritmo deverá resolver, para cada conjunto de regiões e de constantes $\Theta_m = \{R_{jm}, \gamma_{jm}\}_1^{J_m}$, dado o modelo corrente $f_{m-1}(x)$:

$$\widehat{\Theta}_m = \arg\min_{\Theta_m} \sum_{i=1}^{N} L(y_i, f_{m-1}(x_i) + T(x_i, \Theta_m)). \tag{27}$$

Dadas as regiões R_{jm}, encontrar as constantes ótimas γ_{jm} para cada região é em geral dada pela solução:

$$\widehat{\gamma}_{jm} = \arg\min_{\gamma_{jm}} \sum_{x_i \in R_{jm}} L(y_i, f_{m-1}(x_i) + \gamma_{jm}). \tag{28}$$

2.3.2 Otimização Numérica via *Gradient Boosting*

A solução da Equação (27) pode ser feita por otimização numérica. Adotando a função de perda na previsão de y a partir de $f(x)$ nos dados de treinamento como:

$$L(f) = \sum_{i=1}^{N} L(y_i, f(x_i)), \tag{29}$$

o objetivo é minimizar $L(f)$ em relação à f, em que $f(x)$ é restringida pela Equação (26). Se ignorarmos essa restrição, a minimização da Equação (29) é vista como a otimização numérica em que os "parâmetros" $\mathbf{f} = \{f(x_1), f(x_2), \ldots, f(x_N)\} \in \mathbb{R}^N$ são valores da função de aproximação $f(x_i)$ para cada N pontos de dados x_i, tendo portanto:

$$\widehat{\mathbf{f}} = \arg\min_{\mathbf{f}} L(\mathbf{f}). \tag{30}$$

Os procedimentos de otimização numérica resolvem esta equação com a soma do componente dos vetores, em que $\mathbf{f}_0 = \mathbf{h}_0$ é um palpite inicial e os sucessivos $\mathbf{f}_m$

são induzidos baseados no vetor de parâmetros correntes $\mathbf{f}_{m-1}$, que é a soma das atualizações induzidas anteriormente. Assim, teremos que:

$$\mathbf{f}_M = \sum_{m=0}^{M} \mathbf{h}_m, \quad \mathbf{h}_m \in \mathbb{R}^N. \tag{31}$$

Os métodos de otimização dinâmica diferem entre si ao definirem diferentes vetores de incremento $\mathbf{h}_m$. Para o método de descida mais íngreme, é escolhido $\mathbf{h}_m = -\rho_m \mathbf{g}_m$, em que ρ_m é um escalar e $\mathbf{g}_m \in \mathbb{R}^N$ é o gradiente de $L(\mathbf{f})$ avaliado em $\mathbf{f} = \mathbf{f}_{m-1}$, apresentando os seguintes componentes:

$$g_{im} = \left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f(x_i)=f_{m-1}(x_i)}, \tag{32}$$

ao passo que ρ_m é a solução de:

$$\rho_m = \arg\min_{\rho} L(\mathbf{f}_{m-1} - \rho \mathbf{g}_m). \tag{33}$$

A solução corrente será dada por $\mathbf{f}_m = \mathbf{f}_{m-1} - \rho_m \mathbf{g}_m$. A Equação (32) tem um papel central para calcular as funções de perda[5] diferenciáveis $L(y,f(x))$, ainda que seja limitado por ser definido apenas nos pontos x_i dos dados de treinamento – e em especial queremos generalizar $f_M(x)$ para pontos em que não estão representados no conjunto de dados de treinamento. Uma possível solução para essa problemática é uma adaptação para estimação de Θ_m: induzir que a árvore $T(x;\Theta_m)$ em cada uma das interações se aproxime o máximo possível do negativo do gradiente a partir de mínimos quadrados – ainda que as regiões obtidas não serão idênticas àquelas encontradas pelo algoritmo anterior, mas que terão o mesmo propósito – resultando em:

$$\tilde{\Theta}_m = \arg\min_{\Theta} \sum_{i=1}^{N} (-g_{im} - T(x_i;\Theta))^2. \tag{34}$$

Após a construção das árvores com a Equação (33), teremos as constantes de cada região calculadas pela Equação (28). O algoritmo 2 apresenta a aplicação genérica da técnica de *gradient boosting* para uma árvore de regressão. Ele se torna específico quando escolhemos uma função de perda específica. Ele inicia com um modelo de constante ótima, que é um nó terminal da árvore, seguindo do cálculo do gradiente negativo conhecido como *pseudo* resíduo (r).

[5] Existem diversos tipos de função de perda, como função erro quadrático, função de Huber, função de entropia cruzada, dentre tantas outras. A decisão sobre qual escolher dependerá do tipo de problema em que se deseja estudar, visto que algumas apresentam uma melhor performance para problemas de classificação, enquanto outras se sobressaem em problemas de regressão.

Algoritmo 2: *Gradiente Boosting* para árvores de regressão

1. Inicializa $f_0(x) = \arg\min_\gamma \sum_{i=1}^{N} L(y_i,\gamma)$.
2. Para $m = 1,\ldots,M$:
   a) Para $n = 1,2,\ldots,N$, computa
   $$r_{im} = -\left[\frac{\partial L(y_i,f(x_i))}{\partial f(x_i)}\right]_{f=f_{m-1}}$$
   b) Ajusta a árvore de regressão para os alvos r_{im} dadas as regiões terminais R_{jm}, $j = 1,2,\ldots,J_m$.
   c) Para $j = 1,2,\ldots,J_m$, computa
   $$\gamma_{jm} = \arg\min_\gamma \sum_{x_i \in R_{jm}} L(y_i, f_{m-1}(x_i) + \gamma).$$
   d) Atualiza $f_m(x) = f_{m-1}(x) + \sum_{j=1}^{J_m} \gamma_{jm} I(x \in R_{jm})$.
3. Produz $\widehat{f}(x) = f_M(x)$.

2.3.2.1 XGBoost

Utilizando as técnicas de aceleração com gradiente, Chen e Guestrin (2016) apresentam um sistema escalonável de aprendizagem de máquina conhecido com *XGBoost*, que utiliza técnicas de *gradient boosting* para aplicação em árvores de regressão. A estrutura deste modelo é muito semelhante com as versões genéricas que vimos até agora, que tem como *core* a minimização de perdas, com o diferencial de que é introduzido um termo de penalização de acordo com a complexidade do modelo (Ω). A versão simplificada da função objetivo para a t-ésima interação é dada como:

$$\tilde{\mathcal{L}}^{(t)} = \sum_{i=1}^{n} [g_i f_t(\mathrm{x}_i) + \frac{1}{2} h_i f_t^2(\mathrm{x}_i)] + \Omega(f_t), \tag{35}$$

em que $g_i = \partial_{\widehat{y}^{(t-1)}} l(y_i, \widehat{y}^{(t-1)})$ e $h_i = \partial^2_{\widehat{y}^{(t-1)}} l(y_i, \widehat{y}^{(t-1)})$ são os gradientes de primeira e segunda ordem da função de custo. Como $\Omega(f) = \gamma T + \frac{1}{2}\lambda ||w||^2$, podemos expandir a Equação (35), definindo o conjunto de instância $I_j = \{i|q(\mathbf{x}_i) = j\}$ para a folha j:

$$\begin{aligned} \tilde{\mathcal{L}}^{(t)} &= \sum_{i=1}^{n} [g_i f_t(\mathrm{x}_i) + \frac{1}{2} h_i f_t^2(\mathrm{x}_i)] + \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2, \\ &= \sum_{j=1}^{T} \left[(\sum_{i \in I_j} g_i) w_j + \frac{1}{2} (\sum_{i \in I_j} h_i + \lambda) w_j^2 \right] + \gamma T, \end{aligned} \tag{36}$$

para uma estrutura fixa $q(\mathbf{x})$, temos que o peso w_j^* para a folha j é dado por:

$$w_j^* = -\frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda}, \tag{37}$$

o que implica em um valor otimizado para o problema que pode ser utilizado como uma medida de performance sobre a qualidade da árvore q, definido como:

$$\tilde{\mathcal{L}}^{(t)}(q) = -\frac{1}{2} \sum_{j=1}^{T} \frac{(\sum_{i \in I_j} g_i)^2}{\sum_{i \in I_j} h_i + \lambda} + \gamma T. \tag{38}$$

Devido a complexidade de enumerar todas as possíveis estruturas q, é aplicado um algoritmo que começa por uma folha e vai adicionando galhos na árvore. Assumindo que I_L é o nó esquerdo após a partição e I_R o nó direito, em que $I = I_L \cup I_R$, a redução de perda após a partição, que é utilizada para encontrar os potenciais candidatos para a repartição, é definido como:

$$\mathcal{L}_{split} = \frac{1}{2} \left[\frac{(\sum_{i \in I_L} g_i)^2}{\sum_{i \in I_L} h_i + \lambda} + \frac{(\sum_{i \in I_R} g_i)^2}{\sum_{i \in I_R} h_i + \lambda} - \frac{(\sum_{i \in I} g_i)^2}{\sum_{i \in I} h_i + \lambda} \right] - \gamma. \tag{39}$$

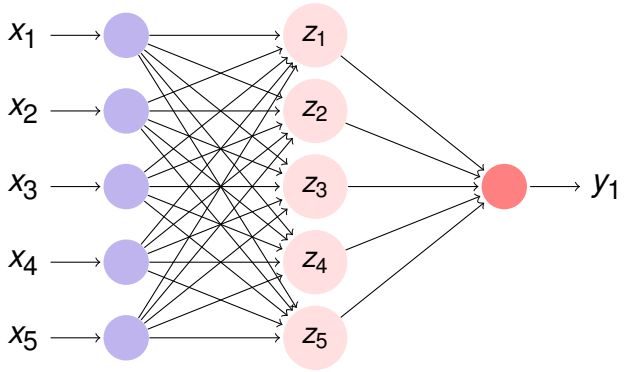
O modelo de XGBoost apresenta três algoritmos para encontrar os pontos de partição: o primeiro, é o algoritmo míope (*greedy algorithm*), que busca encontrar um ponto de ótimo global utilizando os gradientes de primeira e segunda ordens, o que pode ter um custo elevado do ponto de vista computacional; o segundo, um algoritmo de aproximação, que utiliza uma classificação inicial dividindo a base de dados em percentis, fazendo análises comparativas entre os potenciais pontos de partição; e terceiro, um algoritmo capaz de lidar com problemas de esparsidade, em que algumas das covariáveis não apresentam todos os valores comparativamente as outras covariáveis. Sendo assim, o XGBoost é capaz de aplicar algoritmos míopes, permitindo encontrar aproximações de pontos ótimos locais ou globais; permite o processo do modelo de forma *out-of-core*, muito útil para poupar a memória da máquina, especialmente no uso de base de dados muito grandes; permite lidar com problemas de esparsidade na base de dados e também com problemas de paralelização, em que são aplicados um sistema de blocos para evitar um elevado custo computacional para ordenar as variáveis do banco de dados quando o mesmo é demasiado grande.

2.4 REDES NEURAIS

Redes neurais fazem parte de um conjunto de metodologias que se desenvolveram concomitantemente nas áreas de estatística e de inteligência artificial. O objetivo dessa classe de modelos é extrair combinações lineares por meio das covariáveis e então modelar a variável que se deseja prever como uma função não linear destas

covariáveis. Existem diferentes tipos de redes neurais, mas em geral são problemas de regressão em dois estágios ou modelos de classificação, sendo a estruturada em camadas (*layers*): a primeira, chamada de camada de entrada, em que apresentam as covariáveis; a segunda, que podem ser várias ou apenas uma, que são as camadas ocultas com informações não observáveis; e, por último, a camada de saída, que apresenta a variável de interesse. Para o caso de regressão, em geral adota-se apenas y_1, mas para o problemas de classificação podemos ter $y_1, \ldots, y_K$, mas podemos lidar para problemas de regressão com K variáveis alvo. A Figura 6 apresenta o caso para apenas uma camada oculta, cinco covariáveis e uma variável alvo.

Figura 6 – Exemplo de Rede Neural com uma camada oculta.



Fonte – Produzido pelo autor.

As covariáveis derivadas Z_m – também chamadas de unidades ocultas (*hidden units*) pois não são observáveis – são criadas a partir da combinação linear das covariáveis de entrada X. As variáveis alvo Y_k são modeladas como uma função das combinações lineares de Z_m. Para $Z = (Z_1, Z_2, \ldots, Z_M)$ e seja $T = (T_1, T2, \ldots, T_K)$ o vetor de resultados, temos o seguinte modelo:

$$\begin{aligned} Z_m &= \sigma(\alpha_{0m} + \alpha_m^T X), \quad m = 1, \ldots, M, \\ T_k &= \beta_{0k} + \beta_k^T Z, \quad k = 1, \ldots, K, \\ f_k(X) &= g_k(T), \quad k = 1, \ldots, K. \end{aligned} \tag{40}$$

A função de ativação $\sigma(\upsilon)$ é geralmente escolhida como uma *sigmoide*, definida como $\sigma(\upsilon) = 1/(1 + e^{\upsilon})$, ainda que existam outras possibilidades para definirmos esse tipo de função. Para o caso de regressões, a função do produto $g_k(T)$ é dada pelo vetor de resultados T – ainda que para problemas de classificação sejam utilizadas funções mais complexas que aplicam mais uma camada de transformações para este vetor.

2.4.1 Ajuste de Redes Neurais

Como é possível observar no modelo em (40), redes neurais apresentam parâmetros desconhecidos – também chamado de pesos (*weights*) – que devem ser

encontrados de tal forma que apresentem o melhor ajuste para a base de dados utilizada. O conjunto de pesos θ é formado pelos parâmetros α_{0m}, α_m, β_{0k} e β_k tal que:

$$\begin{aligned} \{\alpha_{0m},\alpha_m; m = 1,2,\ldots,M\} \quad M(p+1) \ \textit{pesos}, \\ \{\beta_{0m},\beta_k; k = 1,2,\ldots,K\} \quad K(M+1) \ \textit{pesos}. \end{aligned} \tag{41}$$

Para uma medida de ajuste no caso de regressões, é comum utilizar a soma do quadrado dos erros, definida como:

$$R(\theta) = \sum_{k=1}^{K}\sum_{i=1}^{N}(y_{ik} - f_k(x_i))^2. \tag{42}$$

Um minimizador global para $R(\theta)$ acaba por ser uma solução sobre-ajustada, o que carece de técnicas de regularização. A forma geral de minimizarmos a função de erro, utilizando técnicas de gradiente conhecidas como retro-propagação do erro. Para o caso de uma função de erro quadrado, adotando o modelo descrito em (40) e $z_{mi} = \sigma(\alpha_{0m} + \alpha_m^T x_i)$ para $z_i = (z_{1i},z_{2i},\ldots,z_{Mi}$, temos que:

$$\begin{aligned} R(\theta) &\equiv \sum_{i=1}^{N} R_i, \\ &= \sum_{i=1}^{N}\sum_{k=1}^{K}(y_{ik} - f_k(x_i))^2. \end{aligned} \tag{43}$$

Temos as seguintes derivadas parciais:

$$\begin{aligned} \frac{\partial R_i}{\partial \beta_{km}} &= -2(y_{ik} - f_k(x_i))g_k^{'}(\beta_k^T z_i)z_{mi}, \\ \frac{\partial R_i}{\partial \alpha_{ml}} &= -\sum_{k=1}^{K} 2(y_{ik} - f_k(x_i))g_k^{'}(\beta_k^T z_i)\beta_{km}\sigma^{'}(\alpha_m^T x_i)x_{il}. \end{aligned} \tag{44}$$

A partir dessas derivadas, aplica-se o método de atualização de derivadas para a (r+1)-ésima interação, que terá a forma:

$$\begin{aligned} \beta_{km}^{(r+1)} &= \beta_{km}^{(r)} - \gamma_r \sum_{i=1}^{N} \frac{\partial R_i}{\partial \beta_{km}^r}, \\ \alpha_{ml}^{(r+1)} &= \alpha_{ml}^{(r)} - \gamma_r \sum_{i=1}^{N} \frac{\partial R_i}{\partial \alpha_{ml}^r}, \end{aligned} \tag{45}$$

em que γ_r é chamado de taxa de aprendizado – uma constante utilizada, possível de ser otimizada, que é utilizada para minimizar a função de erro a cada atualização – também conhecida como aproximação estocástica, que garante convergência dentro da estrutura do modelo. As derivadas na Equação (44) podem ser reescritas em função

do erro do modelo em relação aos resultados (δ_{ki}) e do erro em relação as camadas ocultas ($s_m i$), sendo reescritas como:

$$\begin{aligned} \frac{\partial R_i}{\partial \beta_{km}} &= \delta_{ki} z_{mi}, \\ \frac{\partial R_i}{\partial \alpha_{ml}} &= s_{mi} x_{il}. \end{aligned} \tag{46}$$

Estes erros satisfazem a Equação (47), chamada funções de retro-propagação do erro. Elas garantem que as atualizações aplicadas em (45) a partir de um algoritmo de dois passos. Para o "passo pra frente" (*forward pass*) os pesos são fixos e os valores previstos $\widehat{f}_k(x_i)$ são calculados a partir do modelo (40); já o "passo para trás" (*backward pass*), os erros δ_{ki} são computados e então propagados retroativamente via Equação (47), permitindo encontrar s_{mi}. O conjunto de resultados então são usados para encontrar os gradientes para a Equação (45) através da Equação (46). O algoritmo em dois passos é o que se conhece por retro-propagação – e que, por vezes, leva demasiado tempo para ser aplicado.

$$s_{mi} = \sigma'(\alpha_m^T x_i) \sum_{k=1}^{K} \beta_{km} \delta_{ki}. \tag{47}$$

2.5 FLORESTAS ALEATÓRIAS

Criado por Breiman (2001), florestas aleatórias (*random forests*) é um algoritmo que cria um conjunto de árvores não-correlacionadas. Em alguns casos, temos um desempenho bastante semelhante aos métodos de aceleração devido a simplicidade para treinamento e para ajustamento (*tuning*) dos parâmetros/hiper-parâmetros. O conjunto de árvores para essa classe de modelos apresenta diferentes variâncias, a depender das hipóteses sobre elas. Sendo B o conjunto de árvores, cada uma com variância σ^2, se elas forem independentes e identicamente distribuídas, a média de B terá variância σ^2/B. Se por ventura relaxarmos a hipótese de independência, assumindo apenas que elas são identicamente distribuídas e com correlação positiva ρ, teremos que a variância da média de B dada por:

$$\rho\sigma^2 + \frac{1-\rho}{B}\sigma^2. \tag{48}$$

É fácil notar que, ao aumentarmos B, temos uma redução do segundo termo da Equação (48) (no limite, para B muito grande, este termo desapareceria ou seria tão pequeno que seria insignificante). O algoritmo de florestas aleatórias surge então para otimizar a minimização desta variância, agindo de tal forma que diminua a correlação entre as árvores, sem aumentar muito a variância. Este objetivo é alcançado ao selecionarmos aleatoriamente as covariáveis na formação das árvores do conjunto B. Assim, o algoritmo 3 para florestas aleatórias é definido como:

Algoritmo 3: Floresta aleatória para regressão

1. para $b = 1$ até B:
   a) Cria uma amostra por *bootstrap*[1] $\mathbf{Z}^*$ de tamanho N a partir dos dados de treinamento.
   b) Cria uma árvore T_b para a floresta aleatória para os dados da amostra, repetindo recursivamente os seguintes passos para cada nó terminal da árvores, até que o tamanho mínimo de nó n_{min} é alcançado.
      i. Seleciona m variáveis aleatoriamente do conjunto de p variáveis.
      ii. Seleciona a melhor variável/ponto de partição entre as m variáveis.
      iii. Divide o nó em dois nós filhos.
2. Resulta no conjunto de árvores $\{T_b\}_1^B$.

Para fazer uma previsão em um novo ponto x, para o caso de regressão, temos que:

$$\widehat{f}_{rf}^B(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x).$$

[1] *Bootstrap* é uma técnica estatística de reamostragem, que é usada para validação de subconjuntos aleatórios.

No processo de escolha aleatória de $m \leq p$ do conjunto de covariáveis como candidatas de partição, em geral escolhe-se $m = p/3$ ou valores tão pequenos quanto 1, ao passo que $n_{min} = 5$. A redução de m diminui a correlação entre os pares de árvores dentro da amostra e, mesmo sob a Equação (48), é reduzida a variância da média. Ainda assim, quando o número de variáveis é grande, mas aquelas que são relevantes são poucas, florestas aleatórias podem ter uma performance ruim para um m pequeno. Após o crescimento de B árvores em $\{T(x;\Theta_b)\}_1^B$, onde Θ_b caracteriza a b-ésima árvore da floresta aleatória em termos de variáveis de partição, pontos de partição dos nós e valores de nós terminais, temos que o preditor para a floresta aleatória é dada por:

$$\widehat{f}_{rf}^B(x) = \frac{1}{B}\sum_{b=1}^{B} T(x;\Theta_b). \tag{49}$$

2.5.1 Análise de Florestas Aleatórias

No modelo de florestas aleatórias, especialmente no caso de regressão, é interessante fazermos uma análise de viés, variância e correlação. Sendo a forma limitada ($B \to \infty$) do estimador da regressão de floresta aleatória num ponto alvo x, dependente dos dados de treinamento $\mathbf{Z}$, definida como:

$$\widehat{f}_{rf}(x) = E_{\Theta|\mathbf{Z}}\, T(x;\Theta(\mathbf{Z})), \tag{50}$$

a partir da Equação (48), temos que a variância para este estimador será dado por:

$$Var\widehat{f}_{rf}(x) = \rho(x)\sigma^2(x), \tag{51}$$

em que $\rho(x) = \mathrm{corr}[T(x;\Theta_1(\mathbf{Z})), T(x;\Theta_2(\mathbf{Z}))]$ é a correlação de amostras entre qualquer par de árvores usada no cálculo da média, sendo $\Theta_1(\mathbf{Z})$ e $\Theta_2(\mathbf{Z})$ são um par aleatório de árvores da floresta aleatória amostrada nos dados aleatórios de $\mathbf{Z}$; e $\sigma^2(x) = VarT(x;\Theta(\mathbf{Z}))$ é a variância de amostragem para qualquer árvore desenhada aleatoriamente. A variância para única árvore (ao invés da média das árvores) $VarT(x;\Theta(\mathbf{Z}))$ é a variância total e pode ser decomposta em dois argumentos utilizando variância condicional:

$$\begin{aligned} Var_{\Theta,\mathbf{Z}} T(x;\Theta(\mathbf{Z})) &= Var_{\mathbf{Z}} E_{\Theta|\mathbf{Z}} T(x;\Theta(\mathbf{Z})) + E_{\mathbf{Z}} Var_{\Theta|\mathbf{Z}} T(x;\Theta(\mathbf{Z}), \\ \text{Variância Total} &= Var_{\mathbf{Z}} \widehat{f}_{rf}(x) + \text{Variância dentro-de-}\mathbf{Z}. \end{aligned} \tag{52}$$

A variância dentro-de-$\mathbf{Z}$ (*within-*$\mathbf{Z}$ *variance*) é resultada da aleatorização e aumenta quando m diminui. O primeiro termo é a variância de amostragem para o conjunto de floresta aleatória, que diminuí quando m diminui. Nesse sentido, a variância de uma árvore individual não é tão sensível a magnitude de m tal qual a variância do conjunto de floresta aleatória é.

Já o viés para a floresta aleatória é o mesmo que o viés para qualquer árvore individual amostrada $T(x;\Theta(\mathbf{Z}))$:

$$\begin{aligned} \text{Viés}(x) &= \mu(x) - E_{\mathbf{Z}} \widehat{f}_{rf}(x) \\ &= \mu(x) - E_{\mathbf{Z}} E_{\Theta|\mathbf{Z}} T(x;\Theta(\mathbf{Z})) \end{aligned} \tag{53}$$

Uma discussão mais profunda sobre o viés necessitaria de conhecermos a função verdadeira que representa estas árvores. Seja como for, a tendência é de que quando m diminui, o viés aumenta – surgindo um *trade-off* entre viés-variância na escolha de m.

2.5.2 Avaliação dos Modelos: Validação Cruzada

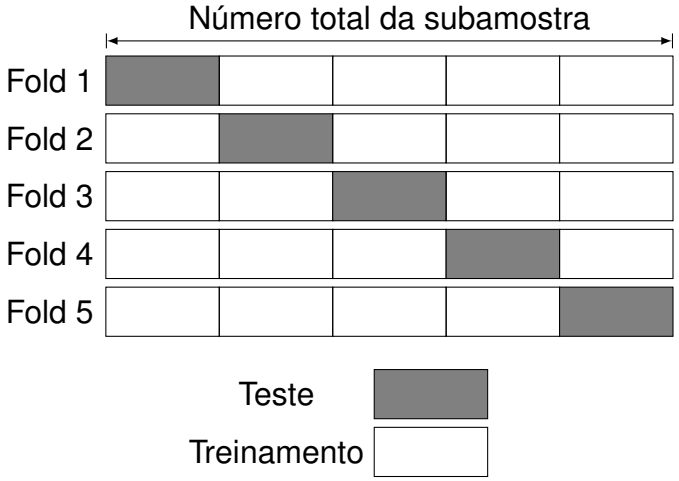
A análise do desempenho dos modelos é necessária do ponto de vista comparativo para que possamos escolher o melhor modelo que se adéque aos dados. Para isso, o conjunto de observações é dividido em dois grupos: um de treinamento, que será usado para o ajuste e aprendizado do modelo; e um de teste, que serve como conjunto de dados independentes para real avaliação do modelo. Friedman, Hastie e Tibshirani (2001) apresentam diversas técnicas para seleção de modelos, com especial atenção a mais utilizada: a validação cruzada. Sendo a função de perda para o erro entre Y e $\widehat{f}(X)$, calculado com os dados de treinamento, definida:

$$L(Y,\widehat{f}(X)) = \begin{cases} (Y - \widehat{f}(X))^2 & \text{erro quadrático} \\ |Y - \widehat{f}(X)| & \text{erro absoluto,} \end{cases}$$

o método de validação cruzada estima o erro de previsão fora da amostra $Err = E[L(Y,\widehat{f}(X))]$ dado um conjunto de treinamento $\mathcal{T}$. O método mais utilizado em validação cruzada é conhecido como *K-fold*, pois seleciona K diferentes subamostras

do conjunto total do treinamento, dividindo esses *folds* em treinamento e teste aleatoriamente. Por exemplo, para $K = 5$, a Figura 7 mostra uma divisão aleatória em 5 subamostras, em que cada uma delas tem uma segunda divisão em treinamento e teste. A proporção de escolha, tanto aqui quanto na divisão inicial da base de dados, em geral respeita uma proporção 70/30 ou 80/20 – como é mostrado pela figura – podendo variar de acordo com a decisão do pesquisador.

Figura 7 – Estrutura de divisão para validação cruzada K-fold para K=5.



Fonte – Produzido pelo autor.

Seja uma função de indexação $k : \{1,\ldots,N\} \mapsto \{1,\ldots,K\}$ que indica a partição em que cada observação i será direcionada pela aleatorização, denotando $\widehat{f}^{-k}(x)$ a função ajustada computada pela k-ésima parte do conjunto de dados. A estimação do erro de previsão por meio de validação cruzada é definido como:

$$CV(\widehat{f}) = \frac{1}{N}\sum_{i=1}^{N} L(y_i,\widehat{f}^{-k(i)}(x_i)). \tag{54}$$

Em geral a escolha de K é 5 ou 10. Para o conjunto de modelos $f(x,\alpha$, com um parâmetro que pode ser otimizado α, temos $\widehat{f}^{-k}(x,\alpha)$ o α-ésimo modelo ajustado a k-ésima parte dos dados removidos. Então, para este conjunto de modelos temos que encontrar $\widehat{\alpha}$ que minimiza:

$$CV(\widehat{f},\alpha) = \frac{1}{N}\sum_{i=1}^{N} L(y_i,\widehat{f}^{-k(i)}(x_i,\alpha)). \tag{55}$$

Dessa forma, utilizamos a validação cruzada para encontrar os melhores valores para os parâmetros/hiperparâmetros que propiciem o melhor ajuste dos modelos aos dados utilizados. Em geral, a forma correta para aplicação de validação cruzada K-fold é a seguinte:

1. Divide a amostra em K *folds* de validação cruzada aleatoriamente.

2. Para cada *fold* k = 1,2,...,K.

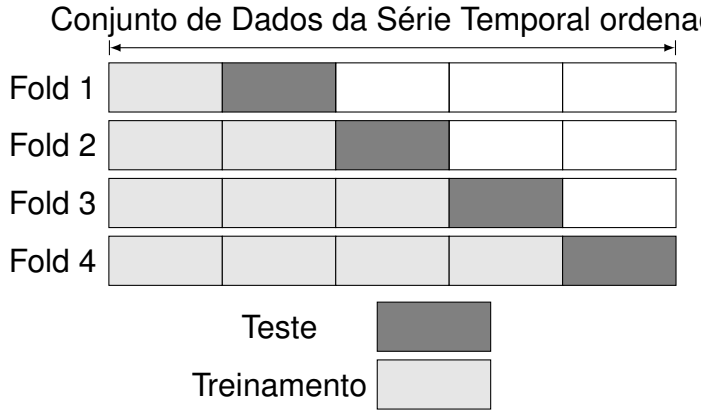
    a) Encontra o subconjunto de "bons"preditores que apresentem forte correlação univariada com a variável de interesse, usando todas as amostras, exceto aqueles que pertencem ao *fold* k.

    b) Usando apenas o subconjunto de preditores, crie um conjunto de preditores multivariado, usando todas as amostras exceto aquelas presentes no *fold* k.

    c) Usa este conjunto de preditores para prever a variável de interesse para as amostras no *fold* k. Os erros estimados nesse passo são acumulados para todos os K folds de tal forma que irá produzir o erro de previsão estimado de validação cruzada.

Devido as características de séries temporais, é comum vermos na literatura destes casos um tipo de validação cruzada conhecida como pseudo avaliação fora da amostra, em que a janela de estimação é partida em treinamento e teste respeitando a sequência das séries temporais dos dados, como é aplicado por Yoon (2020). Na Figura 8 temos a técnica de pseudo avaliação fora da amostra adotada por Yoon (2020), em que a estrutura dos blocos respeita a temporalidade das séries, isto é, não se utilizam dados do futuro para prever o passado, deixando de fora parte do conjunto total de dados. A forma clássica para pseudo avaliação fora da amostra seria representada pelo *fold* 3, técnica essa bastante utilizada na literatura de previsão de séries temporais.

Figura 8 – Estrutura de divisão pseudo avaliação fora da amostra com K=4.



Fonte – Produzido pelo autor.

Ainda que a literatura opte pela pseudo avaliação fora da amostra para previsão de séries temporais em modelos de aprendizagem de máquina, Bergmeir, Hyndman e Koo (2018) trazem importantes resultados ao mostrar que o método de validação cruzada tem um resultado superior ao primeiro, mesmo na aplicação em séries temporais. Os resultados mostram que o método de validação cruzada de K-fold se sobressai

quando os erros de previsão são não correlacionados, resultado que em geral se obtém quando aplicamos modelos de aprendizagem de máquina. Assumindo que a série temporal de interesse segue um modelo de regressão não linear $y_t = g(\mathbf{x}_t,\boldsymbol{\theta})\varepsilon_t$ em que $g(\dots)$ é uma função contínua e diferenciável dos vetor de parâmetros $\boldsymbol{\theta}$ para todo $\mathbf{x}_t = (y_{t-1},y_{t-2},\dots,y_{t-p})'$ e ε_t é o erro de regressão. Supondo que $\{\tilde{y}_t\}_{t=1}^m$ é um outro processo com mesma distribuição que os dados da amostra $\{y_t\}_{t=1}^n$ (por exemplo, os dados futuros para a série) e que $\tilde{\mathbf{x}}_t = (\tilde{y}_{t-1},\tilde{y}_{t-2},\dots,\tilde{y}_{t-p})'$. O erro de previsão (EP) para o modelo não linear é definido como $EP = E\{\tilde{y} - g(\tilde{\mathbf{x}}_t,\widehat{\boldsymbol{\theta}})\}^2$, em que $\widehat{\boldsymbol{\theta}}$ é obtido ao minimizarmos a função de erro. Se consideramos a estimação de EP por validação cruzada, temos o conjunto de treinamento definido como $\{(\mathbf{x}_j,y_j); j = p+1,\dots,n, j \neq t\}$ e o conjunto de teste $\{(\mathbf{x}_t,y_t)\}$. A estimação de EP por validação cruzada será dada por:

$$\widehat{EP} = \frac{1}{n-p}\sum_{t=p+1}^{n}\{y - g(\mathbf{x}_t,\widehat{\boldsymbol{\theta}}_{-t})\}^2,$$

em que $\widehat{\boldsymbol{\theta}}_{-t}$ é a estimação de $\widehat{\boldsymbol{\theta}}$ para o conjunto de treinamento, deixando de fora as informações disponíveis no conjunto de teste. Para que a validação cruzada funcione, $\widehat{EP}$ deve se aproximar de EP. Para isso, são necessários que três requisitos sejam garantidos:

1. Primeiro, que a série que estamos estudando seja um processo estacionário (o conceito de estacionaridade é melhor discutido no Capítulo 3 quando apresentamos modelos autorregressivos).
2. $\widehat{\boldsymbol{\theta}}_{-t}$ é um estimador consistente de $\boldsymbol{\theta}$.
3. Os erros ε_t são sequências de diferenças de martingale, que implica a não existência de correlação serial para esses erros. Essa hipótese é fundamental para que a validação cruzada possa valer.

Se os três requisitos são atendidos, essa estrutura teórica é provada pelos autores que $\widehat{EP} \xrightarrow{p} EP$. A aplicação empírica pelos autores mostrou que o uso de validação cruzada para previsão de séries temporais garantiu erros menores comparativamente ao uso de pseudo avaliação fora da amostra, controlando o problema de sobre-ajuste que pode surgir.

3 MODELOS ECONOMÉTRICOS TRADICIONAIS PARA PREVISÃO DE SÉRIES TEMPORAIS

No Capítulo 2, nosso objetivo era apresentar a formalização e intuição dos modelos de aprendizagem de máquina. Neste capítulo, revisitaremos os modelos tradicionais de econometria de séries temporais, que serão a referência em termos comparativos de desempenho para os modelos anteriormente descritos. Para a definição dos modelos utilizados aqui, adotaremos Tsay (2010) como referência no tratamento de econometria de séries temporais com aplicações financeiras.

O *core* da econometria de séries temporais é o conceito de estacionariedade, que permite que classifiquemos as séries em estritamente estacionárias e fracamente estacionária. Uma série temporal $\{r_t\}$ é dita estritamente estacionária se, e somente se, a distribuição conjunta de $(r_{t_1}, \ldots, r_{t_k})$ é idêntica à distribuição conjunta de $(r_{t_{1+t}}, \ldots, r_{t_{k+t}})$ para qualquer t, em que k é um inteiro positivo arbitrário e $(t_1, \ldots, t_k)$ é uma coleção dos k inteiros positivos, isto é, uma série é estritamente estacionária caso sua distribuição conjunta é invariante no tempo. Já as séries fracamente estacionárias, mais factíveis de serem verificadas empiricamente e que são condição suficiente para a análise dos modelos que iremos estudar, é definida quando a média de r_t e a covariância entre r_t e r_{t-l} não varia ao longo do tempo. Temos que $E(r_t) = \mu$, que é constante, e $Cov(r_t, r_{t-l}) = \gamma_l$, que também é constante e dependente apenas de l.

3.1 MODELO AUTORREGRESSIVO DE MÉDIAS MÓVEIS

Apresentado por Box, Jenkins e Reinsel (1994), modelos autorregressivos de médias móveis (ARMA) são compostos por dois polinômios: um representa o componente autorregressivo, isto é, a relação da variável em relação ao seu passado; o outro representa o componente de médias móveis, que relaciona a variável com o erro e o passado dos erros. A notação é utilizada por ARMA(p,q), em que p denota a ordem da parte autorregressiva e q a ordem do componente de médias móveis. Transformações matemáticas permitem que possamos escrever modelos ARMA somente em termos AR ou somente em termos MA. Para fins de apresentação, analisaremos a propriedades de um modelo ARMA(1,1), que embora simples, permite conclusões importantes que naturalmente se estendem à versões mais complexas desta classe de modelos. Uma série temporal segue um processo ARMA(1,1) caso satisfaça:

$$r_t - \varphi_1 r_{t-1} = \varphi_0 + a_t - \theta_1 a_{t-1}, \tag{56}$$

em que $\{a_t\}$ é um ruído branco com média zero e variância σ_a^2. No lado esquerdo da Equação (56) temos o componente autorregressivo, ao passo que no lado direito temos o componente de médias móveis com um termo constante φ_0.

3.1.1 Propriedades Estatísticas dos Modelos ARMA(1,1)

Assumindo que a série $\{r_t\}$ é fracamente estacionária, aplicamos o operador esperança na Equação (56), sabendo que $E(a_i) = 0$, para qualquer i:

$$E(r_t) - \varphi_1 E(r_{t-1}) = \varphi_0 + E(a_t) - \theta_1 E(a_{t-1}) \Leftrightarrow E(r_t) = \mu = \frac{\varphi_0}{1-\varphi_1}. \tag{57}$$

Assumindo que $\varphi_0 = 0$, multiplicaremos a Equação (56) por a_t e depois aplicando a esperança:

$$E(r_t a_t) - \varphi_1 E(r_{t-1} a_t) = E(a_t^2) - \theta_1 E(a_{t-1} a_t) \Leftrightarrow E(r_t a_t) = E(a_t^2) = \sigma_a^2. \tag{58}$$

Pelo modelo, temos que $E(r_{t-1} a_t) = 0$ e $E(a_t^2) = Var(a_t) = \sigma_a^2$; por hipótese de que não há autocorrelação serial entre os erros, temos $E(a_{t-1} a_t) = 0$. Com o modelo na forma $r_t = \varphi_1 r_{t-1} + a_t - \theta_1 a_{t-1}$, podemos aplicar o operador variância, sabendo que pela Equação (58) $E(r_{t-1} a_{t-1}) = \sigma_a^2$ e que, por ser fracamente estacionária, $Var(r_t) = Var(r_{t-1})$:

$$\begin{aligned} Var(r_t) &= \varphi_1^2 Var(r_{t-1}) + \sigma_a^2 + \theta_1^2 \sigma_a^2 - 2\varphi_1 \theta_1 E(r_{t-1} a_{t-1}) \Rightarrow \\ (1-\varphi_1^2) Var(r_t) &= (1 - 2\varphi_1 \theta_1)\sigma_a^2 \Leftrightarrow \\ Var(r_t) &= \frac{(1-2\varphi_1\theta_1)\sigma_a^2}{1-\varphi_1^2}. \end{aligned} \tag{59}$$

Como a variância é positiva, é preciso que $\varphi_1^2 < 1$ – condição necessária para estacionariedade da série. Nosso último passo é encontrar a função de autocorrelação, multiplicando a Equação (56) por r_{t-l} assumindo $\varphi_0 = 0$, teremos:

$$r_t r_{t-l} - \varphi_1 r_{t-1} r_{t-l} = a_t r_{t-l} - \theta_1 a_{t-1} r_{t-l}. \tag{60}$$

Para $l = 1$, tomando a esperança da Equação (60) e utilizando a Equação (58) para $t-1$, teremos que:

$$\gamma_1 - \varphi_1 \gamma_0 = -\theta_1 \sigma_a^2, \tag{61}$$

em que $\gamma_l = Cov(r_t, r_{t-l})$ e $\gamma_0 = Var(r_t)$. Para o caso em que $l > 1$, temos que:

$$\gamma_l - \varphi_1 \gamma_{l-1} = 0. \tag{62}$$

A função de autocorrelação (FAC) será dada por:

$$\frac{\gamma_1}{\gamma_0} = \rho_1 = \varphi_1 - \frac{\theta_1 \sigma_a^2}{\gamma_0} \Rightarrow \rho_l = \varphi_1 \rho_{l-1}, \quad l > 1. \tag{63}$$

3.1.2 Modelos ARMA Generalizados

Um modelo ARMA(p,q) assume a seguinte forma, com p e q inteiros não negativos e $\{a_t\}$ é um ruído branco:

$$r_t = \varphi_0 \sum_{i=1}^{p} \varphi_i r_{t-i} + a_t - \sum_{i=1}^{q} \theta_i a_{t-i}. \tag{64}$$

A Equação (64) pode ser reescrita ao utilizarmos o operador lag B, em que $B^p X_t = X_{t-p}$. Podemos reescrever o modelo como:

$$(1 - \varphi_1 B - \cdots - \varphi_p B^p) r_t = \varphi_0 + (1 - \theta_1 B - \cdots - \theta_q B^q) a_t. \tag{65}$$

Nesta equação, $(1-\varphi_1 B-\cdots-\varphi_p B^p)$ é o polinômio para o componente autorregressivo, enquanto $(1-\theta_1 B-\cdots-\theta_q B^q)$ é o polinômio para o componente de médias móveis. É preciso garantir que não existam fatores comuns entre os polinômios AR e MA, caso contrário ele seria de uma ordem menor do que a (p,q). Na prática, a identificação das defasagens p e q são encontradas por meio de critérios de informação, como o Akaike (AIC) e Bayesiano (BIC), prezando sempre pela parcimônia na escolha das defasagens para que não haja tanto impacto nos graus de liberdade para estimação. Caso a série seja fracamente estacionária, então temos a média incondicional dada por $E(r_t) = \varphi_0/(1 - \varphi_1 - \cdots - \varphi_p)$.

3.1.3 Previsão com Modelos ARMA

Adotando h como a origem de previsão e a informação disponível por F_h, a previsão para um período r_{h+1} em um modelo ARMA(p,q) é obtida por:

$$\widehat{r}_h(1) = E(r_{h+1}|F_h) = \varphi_0 + \sum_{i=1}^{p} \varphi_i r_{r+1-i} - \sum_{i=1}^{q} \theta_i a_{h+1-i}. \tag{66}$$

Esta equação é associada com o erro de previsão $e_h(1) = r_{h+1} - \widehat{r}_h(1)$. A variância do erro de previsão para um passo à frente é dada por $Var[e_h(1)] = \sigma_a^2$. Generalizando para l passos à frente, teremos:

$$\widehat{r}_h(l) = E(r_{h+l}|F_h) = \varphi_0 + \sum_{i=1}^{p} \varphi_i \widehat{r}_h(l-i) - \sum_{i=1}^{q} \theta_i a_h(l-i). \tag{67}$$

Nesta equação, $\widehat{r}_h(l-i) = r_{h+l-i}$ caso $l-i \geq 0$ e $a_h(l-i) = 0$ se $l-i > 0$ e $a_h(l-i) = a_{h+l-i}$ caso $l - i \geq 0$. A previsão para os passos à frente é resolvida de forma recursiva, através de substituições. O erro associado para l passos à frente é definido como:

$$e_h(l) = r_{h+l} - \widehat{r}_h(l). \tag{68}$$

Para o cálculo da variância, necessitamos transformar o modelo ARMA(p,q) em sua representação MA. Sejam os polinômios $(1 - \varphi_1 B - \cdots - \varphi_p B^p) = \varphi(B)$ e $(1 - \theta_1 B - \cdots - \theta_q B^q) = \theta(B)$, temos a divisão longa definida como:

$$\frac{\theta(B)}{\varphi(B)} = 1 + \psi_1 B + \psi_2 B^2 + \cdots \equiv \psi(B). \tag{69}$$

O modelo ARMA(p,q), utilizando a Equação (69), e com $\mu = E(r_t) = \varphi_0/(1 - \varphi_1 - \cdots - \varphi_p$ será dado por:

$$r_t = \mu + a_t + \psi_1 a_{t-1} + \psi_2 a_{t-2} + \cdots = \mu + \psi(B) a_t. \tag{70}$$

Esta equação mostra de forma clara os impactos dos choques passados no período corrente da série temporal. Os coeficientes $\{\psi_i\}$ são chamados de função impulso resposta de um modelo ARMA. Eles se aproximam de zero a medida que $i \to \infty$, o que implica que a previsão da série convergirá para a média incondicional a medida que aumentamos os passos à frente – em outras palavras, no longo prazo, as séries convergem para a sua média – conforme mostra a Equação (71). A velocidade em que a previsão para a série se aproxima da média é o que determina a velocidade do que se conhece por reversão à média. Para o modelo na forma MA, a previsão l passos à frente será dada por:

$$\widehat{r}_h(l) = \mu + \psi_l a_h + \psi_{l+1} a_{h-1} + \cdots, \tag{71}$$

ao passo que o erro de previsão será dado por:

$$e_h(l) = a_{h+l} + \psi_1 a_{h+l-1} + \cdots + \psi_{l-1} a_{h+1}, \tag{72}$$

e a variância do erro, que deverá convergir para $Var(r_t)$ é dada por:

$$Var[e_h(l)] = (1 + \psi_1^2 + \cdots + \psi_{l-1}^2)\sigma_a^2. \tag{73}$$

3.2 MODELO DE VETORES AUTORREGRESSIVOS

Como apresentado por Tsay (2010), um modelo de vetores autorregressivos (VAR) é uma forma de lidarmos com múltiplas séries temporais. Sendo $\mathbf{r}_t$ um vetor $k \times 1$, um modelo VAR(1) é definido como:

$$\mathbf{r}_t = \boldsymbol{\varphi}_0 + \boldsymbol{\Phi}\mathbf{r}_{t-1} + \mathbf{a}_t, \tag{74}$$

em que $\boldsymbol{\varphi}_0$ é um vetor k-dimensional, $\boldsymbol{\Phi}$ é uma matriz $k \times k$ e $\{\mathbf{a}_t\}$ é uma sequência de vetores aleatórios serialmente não-correlacionadas, com média zero e matriz de covariância $\boldsymbol{\Sigma}$, que deverá ser positiva definida. Para o caso bi-variado, em que $k = 2$ e, portanto, $\mathbf{r}_t = (r_{1t}, r_{2t})'$ e $\mathbf{a}_t = (a_{1t}, a_{2t})'$, o modelo VAR(1) é definido como:

$$\begin{aligned} r_{1t} &= \varphi_{10} + \Phi_{11} r_{1,t-1} + \Phi_{12} r_{2,t-1} + a_{1t}, \\ r_{2t} &= \varphi_{20} + \Phi_{21} r_{1,t-1} + \Phi_{22} r_{2,t-1} + a_{2t}, \end{aligned} \tag{75}$$

em que Φ_{ij} é o (i,j)-ésimo elemento de $\boldsymbol{\Phi}$ e φ_{i0} é o i-ésimo elemento de $\boldsymbol{\varphi}_0$. Estas equações nos mostram que existem relações linearmente dependentes entre as séries r_1 e r_2, uma em relação ao passado da outra, representadas pelos parâmetros Φ_{12} e Φ_{21}.

3.2.1 Condição de Estacionariedade e Momentos para VAR(1)

Se o modelo VAR(1) for fracamente estacionário, tomando a expectativa da Equação (74) e sabendo que $E(\mathbf{a}_t) = 0$, teremos que:

$$E(\mathbf{r}_t) = \boldsymbol{\varphi}_0 + \boldsymbol{\Phi} E(\mathbf{r}_{t-1}).$$

Como $E(\mathbf{r}_t)$ não varia ao longo do tempo, temos que:

$$\boldsymbol{\mu} \equiv E(\mathbf{r}_t) = (I - \boldsymbol{\Phi})^{-1}\boldsymbol{\varphi}_0,$$

em que $\mathbf{I}$ é a matriz identidade $k \times k$ e a matriz $I - \boldsymbol{\Phi}$ é não singular. Se adotarmos que $\boldsymbol{\varphi}_0 = (I - \boldsymbol{\Phi})\boldsymbol{\mu}$, podemos reescrever o modelo como:

$$(\mathbf{r}_t - \boldsymbol{\mu}) = \boldsymbol{\Phi}(\mathbf{r}_{t-1} - \boldsymbol{\mu}) + \mathbf{a}_t \Leftrightarrow \tilde{\mathbf{r}}_t = \boldsymbol{\Phi}\tilde{\mathbf{r}}_{t-1} + \mathbf{a}_t.$$

Nesta forma, o modelo é conhecido como corrigido pela média. Sob substituições sucessivas para as séries passadas, teremos que:

$$\tilde{\mathbf{r}}_t = \mathbf{a}_t + \boldsymbol{\Phi}\mathbf{a}_{t-1} + \boldsymbol{\Phi}^2\mathbf{a}_{t-2} + \boldsymbol{\Phi}^3\mathbf{a}_{t-3} + \cdots.$$

Como $\mathbf{a}_t$ não possui correlação serial, é possível concluir que $\text{Cov}(\mathbf{a}_t, \mathbf{r}_{t-j}) = \mathbf{0}$. Da mesma forma, temos que $\text{Cov}(\mathbf{r}_t, \mathbf{a}_t) = \boldsymbol{\Sigma}$. É também possível concluir que $\mathbf{r}_t$ depende da inovação passada $\mathbf{a}_{t-j}$ a partir da matriz de coeficientes $\boldsymbol{\Phi}^j$, que deverá convergir para zero quando $j \to \infty$, resultando que os k autovalores de $\boldsymbol{\Phi}$ deverão ser menores que 1 em módulos a fim de garantir que não hajam trajetórias explosivas para a séries – em outras palavras, que haja convergência – condição essa que garante estacionaridade fraca para a série $\mathbf{r}_t$. Aplicando o operador de covariância para o modelo nessa forma, temos:

$$\text{Cov}(\mathbf{r}_t) = \boldsymbol{\Gamma}_0 = \boldsymbol{\Sigma} + \boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Phi}' + \boldsymbol{\Phi}^2\boldsymbol{\Sigma}(\boldsymbol{\Phi}^2)' + \cdots = \sum_{i=0}^{\infty} \boldsymbol{\Phi}^i\boldsymbol{\Sigma}(\boldsymbol{\Phi}^i)'.$$

Como $\text{Cov}(\mathbf{a}_t, \mathbf{r}_j) = E(\mathbf{a}_t\tilde{\mathbf{r}}_j') = \mathbf{0}$ para $j > 0$, então temos que:

$$E(\tilde{\mathbf{r}}_t\tilde{\mathbf{r}}_{t-\ell}') = \boldsymbol{\Phi}E(\tilde{\mathbf{r}}_{t-1}\tilde{\mathbf{r}}_\ell)', \quad \ell > 0,$$

o que nos permite escrever a relação para a matriz de covariância cruzada para $\mathbf{r}_t$:

$$\boldsymbol{\Gamma}_\ell = \boldsymbol{\Phi}^\ell\boldsymbol{\Gamma}_0, \quad \ell > 0.$$

Por substituições recursivas, temos que, de forma genérica, podemos definir a covariância cruzada como:

$$\boldsymbol{\Gamma}_\ell = \boldsymbol{\Phi}^\ell\boldsymbol{\Gamma}_0, \quad \ell > 0. \tag{76}$$

Ao multiplicarmos esta equação pela matriz diagonal $\mathbf{D}^{-1/2}$, obtemos a autocorrelação por:

$$\boldsymbol{\rho}_\ell = \mathbf{D}^{-1/2}\boldsymbol{\Phi}\boldsymbol{\Gamma}_{\ell-1}\mathbf{D}^{-1/2} = \mathbf{D}^{-1/2}\boldsymbol{\Phi}\mathbf{D}^{-1/2}\mathbf{D}^{-1/2}\boldsymbol{\Gamma}_{\ell-1}\mathbf{D}^{-1/2} = \boldsymbol{Y}\boldsymbol{\rho}_{\ell-1},$$

em que $\boldsymbol{Y} = \mathbf{D}^{-1/2}\boldsymbol{\Phi}\mathbf{D}^{-1/2}$, o que satisfaz a seguinte representação genérica para a autocorrelação:

$$\boldsymbol{\rho}_\ell = \boldsymbol{Y}^\ell\boldsymbol{\rho}_0, \quad \ell > 0. \tag{77}$$

3.2.2 Modelo VAR Generalizado

A generalização do modelo VAR(1) para um modelo VAR(p) é feita de forma bastante semelhante com o que vimos nos modelos ARMA. Uma série $\mathbf{r}_t$ é um modelo VAR(p) caso satisfaça:

$$\mathbf{r}_t = \boldsymbol{\varphi}_0 \boldsymbol{\Phi}_1 \mathbf{r}_{t-1} + \cdots + \boldsymbol{\Phi}_p \mathbf{r}_{t-p} + \mathbf{a}_t, \quad p > 0, \tag{78}$$

em que $\boldsymbol{\varphi}_0$ e $\mathbf{a}_t$ foram definidos como em VAR(1) e $\boldsymbol{\Phi}_j$ são matrizes $k \times k$. Com o operador de defasagens B, podemos reescrever o modelo na forma:

$$(\mathbf{I} - \boldsymbol{\Phi}_1 B - \cdots - \boldsymbol{\Phi}_p B^p)\mathbf{r}_t = \boldsymbol{\varphi}_0 + \mathbf{a}_t \Leftrightarrow \boldsymbol{\Phi}(B)\mathbf{r}_t = \boldsymbol{\varphi}_0 + \mathbf{a}_t.$$

Caso $\mathbf{r}_t$ seja fracamente estacionária, então:

$$\boldsymbol{\mu} = E(\mathbf{r}_t) = (\mathbf{I} - \boldsymbol{\Phi}_1 - \cdots - \boldsymbol{\Phi}_p)^{-1}\boldsymbol{\varphi}_0 = [\boldsymbol{\Phi}(1)]^{-1}\boldsymbol{\varphi}_0$$

Utilizando o modelo corrigido pela média, encontramos as mesmas propriedades para o modelo VAR(1):

1. Cov($\mathbf{r}_t$,$\mathbf{a}_t$) = $\boldsymbol{\Sigma}$, a matriz de covariância para $\mathbf{a}_t$.
2. Cov($\mathbf{r}_{t-\ell}$,$\mathbf{a}_t$) = $\mathbf{0}$ para $\ell > 0$.
3. Covariância cruzada definida como $\boldsymbol{\Gamma}_\ell = \boldsymbol{\Phi}_1 \boldsymbol{\Gamma}_{\ell-1} + \cdots + \boldsymbol{\Phi}_p \boldsymbol{\Gamma}_{\ell-p}$ para $\ell > 0$.
4. Autocorrelação definida como $\boldsymbol{\rho}_\ell = \boldsymbol{Y}_1 \boldsymbol{\rho}_{\ell-1} + \cdots + \boldsymbol{Y}_p \boldsymbol{\rho}_{\ell-p}$ para $\ell > 0$.

3.2.3 Previsão com Modelo VAR

A previsão com modelos VAR utiliza as mesmas técnicas utilizados para o caso uni-variado discutidos na seção do modelo ARMA. Partindo de uma origem h, temos que no modelo VAR(p) a previsão para um período a frente é dada por $\mathbf{r}_h(1) = \boldsymbol{\varphi}_0 + \sum_{i=1}^{p} \boldsymbol{\Phi}_i \mathbf{r}_{h+1-i}$, com um erro de previsão definido como $\mathbf{e}_h(1) = \mathbf{a}_{h+1}$ e matriz de covariância $\boldsymbol{\Sigma}$. Para dois passos a frente, temos:

$$\mathbf{r}_h(2) = \boldsymbol{\varphi}_0 + \boldsymbol{\Phi}_1 \mathbf{r}_h(1) + \sum_{i=2}^{p} \boldsymbol{\Phi}_i \mathbf{r}_{h+2-i}.$$

O erro associado para este caso é $\mathbf{e}_h(2) = \mathbf{a}_{h+2} + \boldsymbol{\Phi}_1[\mathbf{r}_t - \mathbf{r}_h(1)] = \mathbf{a}_{h+2} + \boldsymbol{\Phi}_1 \mathbf{a}_{h+1}$, com matriz de covariância $\boldsymbol{\Sigma} + \boldsymbol{\Phi}_1 \boldsymbol{\Sigma} \boldsymbol{\Phi}_1'$. Se $\mathbf{r}_t$ é fracamente estacionária, a previsão para ℓ passos a frente, conforme ℓ aumenta, tende a convergir para a média $\boldsymbol{\mu}$.

3.3 VETOR AUTORREGRESSIVO AUMENTADO POR FATOR

Proposto por Bernanke, Boivin e Eliasz (2005), vetores autorregressivos aumentados por fator (do inglês, *factor-augmented vector autoregressive* ou simplesmente

FAVAR) são alternativas aos tradicionais vetores autorregressivos estruturais, que apresentam um problema de dimensionalidade e em geral suportam no máximo entre seis ou oito variáveis, o que pode ser pouco condizente com os fatos estilizados, especialmente no caso de efeitos de política monetária estudado pelo artigo, que necessita de um conjunto grande de variáveis macroeconômicas. Essa alta precaução em adicionar mais variáveis sob a penalidade na redução dos graus de liberdade também tem efeitos negativos na interpretação da função de impulso resposta, uma vez que ao se estudar a resposta aos choques de determinadas variáveis o modelo acaba sendo pouco preciso por falta de outro conteúdo informacional relevante que não faz parte do modelo. Outro ponto fraco de modelos VARs é que eles não são robustos à contaminações por erros de medida nas variáveis utilizadas no modelo, uma vez que são fortemente dependentes de construções teóricas para representação dos problemas.

3.3.1 Estrutura de um modelo FAVAR

Seja Y_t um vetor $M \times 1$ de variáveis econômicas observáveis que possui efeitos na economia, ao passo que exista um vetor de fatores não observáveis F_t com dimensão $K \times 1$, em que K é "pequeno". Estes fatores não observáveis são tratados como "atividade econômica" ou "condições de crédito" e não são representadas por apenas uma série, mas sim por um conjunto de variáveis econômicas. Assumindo que a dinâmica de (F_t, Y_t) é dado por:

$$\begin{bmatrix} F_t \\ Y_t \end{bmatrix} = \Phi(L) \begin{bmatrix} F_{t-1} \\ Y_{t-1} \end{bmatrix} + \upsilon_t, \tag{79}$$

em que $\Phi(L)$ é um operador de defasagens com os parâmetros que relacionam as variáveis de ordem d e υ_t é um termo de erro com média zero e matriz de covariância $\mathbf{Q}$. A Equação (79) é um VAR em (F_t, Y_t), sendo um VAR tradicional caso os termos de $\Phi(L)$ que relacionam Y_t com F_{t-1} são todos iguais a zero; caso contrário, a Equação (79) é um FAVAR. Como F_t não são observáveis, a Equação (79) não pode ser diretamente estimada – por isso, assumindo que há um vetor de variáveis "informacionais" X_t com $N \times 1$, em que N é o número de séries temporais, com $K + M << N$. Assumimos que X_t se relacionam com os fatores não observáveis F_t e com os fatores observáveis Y_t por:

$$X_t^{'} = \Lambda^f F_t^{'} + \Lambda^y Y_t^{'} + e_t^{'}, \tag{80}$$

em que Λ^f é uma matriz $N \times K$ e Λ^y é uma matriz $N \times M$, ao passo que e_t é um vetor de erros $N \times 1$ com média zero e que pode ser assumido como fracamente correlacionado – para o caso de estimação por componentes principais – ou não correlacionado, para o caso de verossimilhança. A Equação (80) nos mostra que Y_t e F_t, que em geral são correlacionadas, representam forças que afetam a dinâmica de X_t – ao passo que condicionalmente a Y_t, X_t é uma medida ruidosa para os fatores não observáveis F_t.

Podemos estimar o modelo FAVAR utilizando componentes principais ou utilizando verossimilhança. O primeiro método apresenta vantagens por permitir correlação fraca para o vetor de erros e_t, além de ser simples e rápido para a implementação, sendo uma alternativa não paramétrica para estimação. A técnica de estimação com componentes principais é feita em dois passos: no primeiro, são estimados os $K + M$ componentes principais de X_t, denotado por Bernanke, Boivin e Eliasz (2005) como $\hat{C}(F_t, Y_t)$. Como $\hat{C}(F_t, Y_t)$ é uma combinação linear de seus argumentos, isto é, $\hat{C}(F_t, Y_t) = [\hat{C}(F_1, Y_1), \ldots, \hat{C}(F_T, Y_T)]$ encontrar $\hat{F}_t$ é possível quando identificamos qual parte de $\hat{C}(F_t, Y_t)$ não estendido por Y_t. Existem diversas formas para isso, Bernanke, Boivin e Eliasz (2005) adotam $\hat{C} = \sqrt{T}\hat{Z}$, em que $\hat{Z}$ é o autovetor correspondente ao K maior autovalor de XX'. Em seguida, estima-se o FAVAR substituindo F_t por $\hat{F}_t$.

3.3.2 Análise dos Componentes Principais

Como adotaremos a estimação por duas etapas utilizando componentes principais, precisamos resgatar a metodologia desta abordagem. A análise de componentes principais (PCA, do inglês *principal component analysis*) é útil para estudar a matriz de covariância de um vetor de séries temporais. Conforme apresentado por Tsay (2010), dada uma variável aleatória k-dimensional $\mathbf{r} = (r_1, \ldots, r_k)'$ com uma matriz de covariância $\boldsymbol{\Sigma}_r$, a análise de componentes principais utiliza combinações lineares de $\mathbf{r}_t$ para explicar a estrutura de $\boldsymbol{\Sigma}_r$. A análise também pode ser feita para a matriz de correlação $\boldsymbol{\rho}_r$, uma vez que ela é a matriz de covariância do vetor padronizado $\mathbf{r}^* = \mathbf{S}^{-1}\mathbf{r}$, em que $\mathbf{S}^{-1}$ é uma matriz diagonal dos desvios padronizados dos componentes de $\mathbf{r}$.

3.3.2.1 Teoria do PCA

Seja $\mathbf{w}_i = (w_{i1}, \ldots, w_{ik})'$ um vetor k-dimensional para $i = 1, \ldots, k$. Então podemos escrever uma combinação linear para o vetor $\mathbf{r}$ de tal forma que o vetor $\mathbf{w}_i$ seja padronizado, isto é, $\mathbf{w}_i'\mathbf{w}_i = \sum_{j=1}^{k} w_{ij}^2 = 1$:

$$y_i = \mathbf{w}_i'\mathbf{r} = \sum_{j=1}^{k} w_{ij} r_j. \tag{81}$$

Pelas propriedades de combinações lineares de variáveis aleatórias, temos as seguintes propriedades estatísticas para y_i:

$$\begin{aligned} Var(y_i) &= \mathbf{w}_i'\boldsymbol{\Sigma}_r\mathbf{w}_i, \quad i = 1, \ldots, k; \\ Cov(y_i, y_j) &= \mathbf{w}_i'\boldsymbol{\Sigma}_r\mathbf{w}_j, \quad i, j = 1, \ldots, k. \end{aligned} \tag{82}$$

O objetivo de PCA é encontrar combinações lineares $\mathbf{w}_i$ tal que y_i e y_j são não correlacionados para $i \neq j$ e a variância de y_i seja a maior possível. Em outras palavras, o i-ésimo componente principal de $\mathbf{r}$ é a combinação linear $y_i = \mathbf{w}_i'\mathbf{r}$ que maximiza

$Var(y_i)$ sujeito as restrições $\mathbf{w}_i^{'}\mathbf{w}_i = 1$ e $Cov(y_i,y_j) = 0$ para $j = 1,\ldots,i-1$. Por exemplo, o segundo componente principal de $\mathbf{r}$ seria a combinação linear $y_2 = \mathbf{w}_2^{'}\mathbf{r}$ que maximiza $Var(y_2)$ sujeito as restrições $\mathbf{w}_2^{'}\mathbf{w}_2 = 1$ e $Cov(y_2,y_1) = 0$.

Como a matriz de covariância $\boldsymbol{\Sigma}_r$ é não-negativa definida e portanto pode ser decomposta[1]. Seja $(\lambda_1,\mathbf{e}_1),\ldots,(\lambda_k,\mathbf{e}_k)$ os pares de autovalores-autovetores da matriz $\boldsymbol{\Sigma}$, em que os temos os autovalores na ordem $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_k \geq 0$ e os autovetores definidos como $\mathbf{e}_i = (e_{i1},\ldots,e_{ik})'$, propriamente normalizados. O i-ésimo componente principal de $\mathbf{r}$, $y_i = \mathbf{e}_i^{'}\mathbf{r} = \sum_{j=1}^{k} e_{ij}r_j$ para $i = 1,\ldots,k$, apresenta as seguintes propriedades estatísticas:

$$\begin{aligned} Var(y_i) &= \mathbf{e}_i^{'}\boldsymbol{\Sigma}_r\mathbf{e}_i = \lambda_i, \quad i = 1,\ldots,k; \\ Cov(y_i,y_j) &= \mathbf{e}_i^{'}\boldsymbol{\Sigma}_r\mathbf{e}_j = 0, \quad i \neq j. \end{aligned} \tag{83}$$

Caso alguns autovalores λ_i sejam iguais, as opções para os autovetores e mesmo para os componentes principais não são únicas. Teremos que:

$$\sum_{i=1}^{k} Var(r_i) = \text{traço}(\boldsymbol{\Sigma}_r) = \sum_{i=1}^{k}\lambda_i = \sum_{i=1}^{k} Var(y_i). \tag{84}$$

Como resultado da Equação (84), temos que:

$$\frac{Var(y_i)}{\sum_{i=1}^{k} Var(r_i)} = \frac{\lambda_1}{\lambda_1 + \cdots + \lambda_k}. \tag{85}$$

Esta equação significa que a proporção da variação de $\mathbf{r}$ explicada pelo i-ésimo componente principal é a razão entre o i-ésimo autovalor e a soma de todos os autovalores de $\boldsymbol{\Sigma}_r$. Em aplicações empíricas, a matriz de covariância $\boldsymbol{\Sigma}_r$ pode ser estimada a partir da matriz de covariância da amostra, sob algumas restrições: assumindo que a série temporal $\{\mathbf{r}_t | t = 1,\ldots,T\}$ seja fracamente estacionária, temos a seguinte estimativa:

$$\widehat{\boldsymbol{\Sigma}}_r \equiv [\widehat{\sigma}_{ij,r}] = \frac{1}{T-1}\sum_{t=1}^{T}(\mathbf{r}_t - \bar{\mathbf{r}})(\mathbf{r}_t - \bar{\mathbf{r}})', \quad \bar{\mathbf{r}} = \frac{1}{T}\sum_{t=1}^{T}\mathbf{r}_t. \tag{86}$$

3.4 CRITÉRIOS DE INFORMAÇÃO

Como apresentado por Tsay (2010), critérios de informação são calculadas a partir da estimação de modelos que auxiliam o econometrista na escolha do número de defasagens ARMA, VAR e FAVAR. Existem diversos critérios de informação disponíveis, sendo o mais comum o critério de informação Akaike (AIC) e Schwarz–Bayesian (BIC).

[1] Matrizes não-negativas definidas apresentam propriedades como: todos os seus autovalores são reais, positivos ou iguais a zero; elas podem ser decompostas na forma $\mathbf{A} = \mathbf{P}\boldsymbol{\Lambda}\mathbf{P}'$, em que $\boldsymbol{\Lambda}$ é a matriz diagonal com todos os autovalores de $\mathbf{A}(m \times m)$ e $\mathbf{P}(m \times m)$ é uma matriz que apresenta os m autovetores de $\mathbf{A}$.

Para uma amostra com tamanho T, a estatística AIC é calculada como:

$$\text{AIC} = \frac{-2}{T} ln(\text{verossimilhança}) + \frac{2}{T} \times (\text{número de parâmetros}).$$

A função de verossimilhança é calculada a partir de um processo de estimação por máxima-verossimilhança do modelo. No caso de um AR(ℓ) gaussiano, temos o seguinte AIC:

$$\text{AIC}(\ell) = ln(\tilde{\sigma}_\ell^2) + \frac{2\ell}{T},$$

em que $\tilde{\sigma}_\ell^2$ é a estimativa por máxima-verossimilhança da variância de a_t. O primeiro termo desta equação mostra o quanto o modelo AR(ℓ) se ajusta aos dados, ao passo que o segundo termo desta equação é uma função de penalização para que haja parcimônia na escolha das defasagens (ou parâmetros) do modelo. Diferentes funções de penalização definem diferentes critérios de informação, como é o caso do BIC para um modelo AR(ℓ):

$$\text{BIC} = ln(\tilde{\sigma}_\ell^2) + \frac{\ell ln(T)}{T}.$$

No caso de um modelo VAR(i) gaussiano , temos que:

$$\text{AIC}(i) = ln(|\tilde{\boldsymbol{\Sigma}}_i|) + \frac{k^2 i}{T},$$

em que $\tilde{\boldsymbol{\Sigma}}_i = \frac{1}{T}\sum_{t=i+1}^{T} \hat{\mathbf{a}}_t^{(i)}[\hat{\mathbf{a}}_t^{(i)}]'$ é a estimação por máxima-verossimilhança da matriz de covariâncias e k é a dimensão da matriz de parâmetros do modelo VAR. Já o BIC para o modelo VAR(i) é definido como:

$$\text{BIC}(i) = ln(|\tilde{\boldsymbol{\Sigma}}_i|) + \frac{k^2 i ln(T)}{T}.$$

A seleção da ordem para um modelo dentro das possíveis opções $\ell = 1, \ldots, P$ é feita a partir da análise comparativa dos valores para AIC(ℓ) e/ou BIC(ℓ): escolhe-se a ordem daquele que apresentar o menor valor comparativamente para estes critérios de informação.

4 ANÁLISE EMPÍRICA

Neste capítulo, apresentaremos os resultados da aplicação empírica dos modelos discutidos anteriormente, bem como a análise comparativa do desempenho destes modelos, utilizando estatísticas de erro de previsão. Iniciamos com a descrição da base de dados utilizadas, seguido da apresentação da metodologia aplicada para a seleção dos hiperparâmetros dos modelos de aprendizagem de máquina e por último a análise comparativa dos modelos.

4.1 SÉRIES TEMPORAIS UTILIZADAS

Para estimação e previsão dos métodos e modelos vistos nos capítulos 2 e 3, utilizamos 126 variáveis macroeconômicas, disponíveis na Tabela 2. A decisão de escolha das variáveis foi baseada na literatura, como em Maehashi e Shintani (2020) e Schnorrenberger (2017), que utilizam variáveis macroeconômicas, bem como algumas séries expectacionais. A janela temporal das séries é de Fevereiro/2003 até Abril/2021, totalizando 219 observações.

Tabela 2 – Variáveis Macroeconômicas Utilizadas.

Descrição	**Unidade**	Δ	**Fonte**
Crescimento da moeda e política monetária			
M0 - Base Monetária - média	R$ - milhões	1	Bacen
M0 - Base Monetária Expandida - fim do período	R$ - milhões	1	Bacen
M0 - Base Monetária - Moeda Emitida - média	R$ - milhões	1	Bacen
M0 - Base Monetária - Reservas Bancárias - média	R$ - milhões	1	Bacen
Depósitos de Poupança - fim do período	R$ - milhões	1	Bacen
M1 - fim do período	R$ - milhões	2	Bacen
M2 - fim do período - novo conceito	R$ - milhões	2	Bacen
M3 - fim do período - novo conceito	R$ - milhões	2	Bacen
M4 - fim do período - novo conceito	R$ - milhões	2	Bacen
SELIC acumulada no mês em termos anuais (base 252)	% (a.m.)	1	Bacen
Consumo e vendas			
Vendas industriais reais	Índ. (2016=100)	1	CNI
Energia elétrica - consumo	Gwh	1	Eletrobras
Energia elétrica - consumo - comércio	Gwh	1	Eletrobras
Energia elétrica - consumo - indústria	Gwh	1	Eletrobras
Energia elétrica - consumo - residência	Gwh	1	Eletrobras
Energia elétrica - consumo - outros setores	Gwh	1	Eletrobras
Consumo aparente - derivados de petróleo - média - qde./dia	Barril - milhares	1	ANP
Consumo aparente - gasolina - média - qde./dia	Barril - milhares	1	ANP
Consumo aparente - derivados de petróleo - média - qde./dia	Barril - milhares	1	ANP
Consumo aparente - óleo combustível - média - qde./dia	Barril - milhares	1	ANP
Consumo aparente - óleo diesel - média - qde./dia	Barril - milhares	1	ANP

Continua na próxima página.

Continuação da Tabela 2.

Descrição	**Unidade**	Δ	**Fonte**
Vendas Reais - varejo	Índ. (2011=100)	1	IBGE/PMC
Vendas Domésticas - caminhões	Unidades	1	Fenabrave
Vendas Domésticas - ônibus	Unidades	1	Fenabrave
Vendas Domésticas - carros leves	Unidades	1	Fenabrave
Vendas Domésticas de Automóveis	Unidades	1	Fenabrave
Vendas Domésticas - veículos automotivos	Unidades	1	Fenabrave
Crédito			
Operações de crédito - saldo total da carteira de crédito	R$ - milhões	1	Bacen
Emprego, salário e renda			
Taxa de desemprego	Porcentagem (%)	1	IBGE/PNAD
Horas trabalhadas - indústria	Índ. (2006=100)	2	CNI
Salário mínimo real	R$	2	IPEA
Salário mínimo - paridade do poder de compra (PPP)	US$	1	IPEA
Folha de pagamentos - setor geral	Índ. (Jan/2001 = 100)	1	IBGE/Pimes
Preços			
IPCA - geral	Índ. (1993=100)	1	IBGE/SNIPC
IPCA - comidas e bebidas	Var. % (a.m.)	1	IBGE/SNIPC
IPCA - habitação	Var. % (a.m.)	0	IBGE/SNIPC
IPCA - cuidados de saúde pessoal	Var. % (a.m.)	0	IBGE/SNIPC
IPCA - transporte	Var. % (a.m.)	0	IBGE/SNIPC
IPCA - preços de mercado	Var. % (a.m.)	0	IBGE/SNIPC
IPCA - despesas pessoais	Var. % (a.m.)	0	IBGE/SNIPC
IPCA - vestuário	Var. % (a.m.)	0	IBGE/SNIPC
IPCA - preços de mercado - negociáveis	Var. % (a.m.)	1	IBGE/SNIPC
IPCA - preços de mercado - inegociáveis	Var. % (a.m.)	1	IBGE/SNIPC
INPC - geral	Índ. (1993=100)	0	IBGE/SNIPC
IPA - produtos agrícolas	Índ. (Ago/1994=100)	0	FGV/IGP
IPA-EP - geral	Índ. (Ago/1994=100)	0	FGV/IGP
IGP-DI - geral	Índ. (Ago/1994=100)	0	FGV/IGP
INCC - geral	Índ. (Ago/1994=100)	1	FGV/IGP
IPC - geral	Índ. (Ago/1994=100)	1	FGV/ IGP
Índice de Commodities Brasil	Índ. (Dez/2005=100)	1	Bacen
IPCA - expect. de mercado 1 ano à frente - média	Var. % (a.a.)	0	Focus
IPCA - expect. de mercado 2 anos à frente - média	Var. % (a.a.)	0	Focus
IPCA - expect. de mercado 3 anos à frente - média	Var. % (a.a.)	1	Focus
Produto e atividade real			
Produto Interno Bruto (PIB)	R$ - milhões	1	Bacen
Índice de Atividade Econômica do Banco Central (IBC-Br)	Index (2002=100)	1	Bacen
PI - indústria de processamento - quantum	Índ. (2012=100)	1	IBGE/PIM-PF
PI - bens intermediários - quantum	Índ. (2012=100)	1	IBGE/PIM-PF
PI - bens de consumo - quantum	Índ. (2012=100)	1	IBGE/PIM-PF
PI - bens duráveis - quantum	Índ. (2012=100)	1	IBGE/PIM-PF
PI - bens de consumo semi e não duráveis - quantum	Índ. (2012=100)	0	IBGE/PIM-PF
PI - bens de capital - quantum	Índ. (2012=100)	1	IBGE/PIM-PF

Continua na próxima página.

Continuação da Tabela 2.

Descrição	**Unidade**	Δ	**Fonte**
Utilização da capacidade instalada - indústria	Percentual (%)	1	CNI
Índice de Confiança do Consumidor	Índice	1	Fecomercio
Índice de Condições Econômicas	Índice	1	Fecomercio
Índice de Expectativas Futuras	Índice	1	Fecomercio
PIB - Agricultura - expect. de mercado para 1 ano à frente	Var. % (a.a.)	1	Focus
PIB - Agricultura - expect. de mercado para 2 anos à frente	Var. % (a.a.)	1	Focus
PIB - Agricultura - expect. de mercado para 3 anos à frente	Var. % (a.a.)	1	Focus
PIB - Indústria - expect. de mercado para 1 ano à frente	Var. % (a.a.)	1	Focus
PIB - Indústria - expect. de mercado para 2 anos à frente	Var. % (a.a.)	1	Focus
PIB - Indústria - expect. de mercado para 3 anos à frente	Var. % (a.a.)	1	Focus
PIB - Serviços - expect. de mercado para 1 ano à frente	Var. % (a.a.)	1	Focus
PIB - Serviços - expect. de mercado para 2 anos à frente	Var. % (a.a.)	1	Focus
PIB - Serviços - expect. de mercado para 3 anos à frente	Var. % (a.a.)	1	Focus
PIB - expect. de mercado para 1 ano à frente	Var. % (a.a.)	1	Focus
PIB - expect. de mercado para 2 anos à frente	Var. % (a.a.)	1	Focus
PIB - expect. de mercado para 3 anos à frente	Var. % (a.a.)	1	Focus
Produção Industrial - expect. de mercado para 1 ano à frente	Var. % (a.a.)	1	Focus
Produção Industrial - expect. de mercado para 2 anos à frente	Var. % (a.a.)	1	Focus
Produção Industrial - expect. de mercado para 3 anos à frente	Var. % (a.a.)	1	Focus
Financeiro e risco			
Treasury Bill - 3 meses	% (a.a.)	1	FRED
Treasury Bill - 6 meses	% (a.a.)	1	FRED
LIBOR - baseado em US dólar - 1 mês	% (a.a.)	1	FRED
LIBOR - baseado em US dólar - 3 meses	% (a.a.)	1	FRED
LIBOR - baseado em US dólar - 12 meses	% (a.a.)	1	FRED
EMBI+ - risco brasileiro	Índice	1	JP Morgan
Índice de Ações - Ibovespa - fechamento	Índ. (Jan/1999=100)	2	Anbima
Fiscal			
Dívida bruta - governo geral	% PIB	0	Bacen
Dívida externa líquida do Setor Público	% PIB	2	Bacen
Dívida interna líquida do Setor Público	% PIB	1	Bacen
Resultado Primário - expect. de mercado para 1 ano à frente	% PIB	2	Focus
Resultado Primário - expect. de mercado para 2 anos à frente	% PIB	1	Focus
Resultado Primário - expect. de mercado para 3 anos à frente	% PIB	1	Focus
Resultado Nominal - expect. de mercado para 1 ano à frente	% PIB	1	Focus
Resultado Nominal - expect. de mercado para 2 anos à frente	% PIB	1	Focus
Resultado Nominal - expect. de mercado para 3 anos à frente	% PIB	1	Focus
Dívida Líquida - expect. de mercado para 1 ano à frente	% PIB	1	Focus
Dívida Líquida - expect. de mercado para 2 anos à frente	% PIB	2	Focus
Dívida Líquida - expect. de mercado para 3 anos à frente	% PIB	2	Focus
Setor externo			
Taxa de câmbio - fim do período	R$/US$	2	Bacen
Reservas Internacionais	US$ - milhões	2	Bacen
Exportações - FOB	US$ - milhões	2	SECINT

Continua na próxima página.

Continuação da Tabela 2.

Descrição	**Unidade**	Δ	**Fonte**
Exportações - bens de capital - FOB	US$ - milhões	1	SECINT
Exportações - bens de consumo - FOB	US$ - milhões	1	SECINT
Exportações - bens intermediários - FOB	US$ - milhões	1	SECINT
Importações - FOB	US$ - milhões	1	SECINT
Importações - bens de capital - FOB	US$ - milhões	1	SECINT
Importações - bens de consumo - FOB	US$ - milhões	1	SECINT
Importações - bens intermediários - FOB	US$ - milhões	1	SECINT
Balança comercial - valor	US$ - milhões	1	Bacen
Termos de Troca	Índ. (2006=100)	1	FUNCEX
Contas correntes - valor	U$ - milhões	1	Bacen
Exportações - expect. de mercado 1 ano à frente - média	US$ - milhões	1	Focus
Exportações - expect. de mercado 2 anos à frente - média	US$ - milhões	1	Focus
Exportações - expect. de mercado 3 anos à frente - média	US$ - milhões	2	Focus
Importações - expect. de mercado 1 ano à frente - média	US$ - milhões	2	Focus
Importações - expect. de mercado 2 anos à frente - média	US$ - milhões	2	Focus
Importações - expect. de mercado 3 anos à frente - média	US$ - milhões	2	Focus
Contas Correntes - expect. de mercado 1 ano à frente - média	US$ - milhões	2	Focus
Contas Correntes - expect. de mercado 2 anos à frente - média	US$ - milhões	2	Focus
Contas Correntes - expect. de mercado 3 anos à frente - média	US$ - milhões	2	Focus
Investimento Direto - expect. de mercado 1 ano à frente - média	US$ - milhões	1	Focus
Investimento Direto - expect. de mercado 2 anos à frente - média	US$ - milhões	1	Focus
Investimento Direto - expect. de mercado 3 anos à frente - média	US$ - milhões	1	Focus
Taxa de Câmbio - expect. de mercado 1 ano à frente - média	R$/US$	1	Focus
Taxa de Câmbio - expect. de mercado 2 anos à frente - média	R$/US$	1	Focus
Taxa de Câmbio - expect. de mercado 3 anos à frente - média	R$/US$	1	Focus

Fonte: Produzido pelo autor.

As variáveis são classificadas em nove categorias: crescimento da moeda e política monetária (7,9% do total das variáveis); consumo e vendas (13,4%); crédito (>1%)[1]; emprego, salário e renda (3,9%); preços (15,8%); produto e atividade real (21,4%); financeiro e risco (5,5%); fiscal (9,5%) e setor externo (22,2%). Devido a autocorrelação serial geralmente presente em séries macroeconômicas, adicionamos também as variáveis defasadas para todas as séries dentro do conjunto de preditores para os modelos de aprendizagem de máquina.

4.1.1 Tratamento dos dados

Todas as séries foram estacionarizadas, a partir da diferenciação das séries, seguido de teste de raiz unitária. O número de diferenciações (Δ) para tornar as séries estacionárias está disponível na Tabela 2. Garantir estacionariedade das séries

[1] Devido a grande variação na periodicidade de séries referentes ao mercado de crédito, uma vez que parte delas são descontinuadas ou só estão disponíveis a partir de um período recente, optamos por manter apenas o saldo total da carteira de crédito.

é fundamental para os modelos de previsão tradicionais; estas transformações também foram aplicadas na utilização dos modelos de aprendizagem de máquina, o que permite que todos os modelos tenham o mesmo conjunto de treinamento e seus resultados comparados para a previsão do mesmo conjunto de teste. Para a estimação dos métodos de aprendizagem de máquina, é comum utilizarmos procedimentos que aplicam transformações nos preditores.

4.2 SELEÇÃO DOS HIPERPARÂMETROS

O Quadro 2 apresenta a relação dos modelos de aprendizagem de máquina adotados com seus hiperparâmetros, que foram otimizados a partir de validação cruzada K-fold para $K = 10$ utilizando o conjunto de treinamento, a partir da técnica de busca em grade (do inglês, *grid search*).

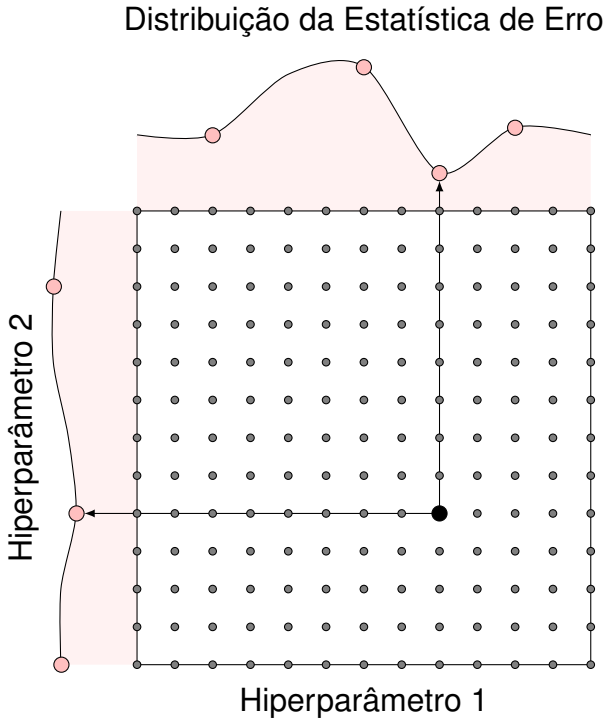
Quadro 2 – Descrição dos Hiperparâmetros otimizados nos Modelos de Aprendizagem de Máquina.

Modelo de Aprendizagem de Máquina	Hiperparâmetro
Regressão Penalizada	Quantidade de regularização/penalização
	Proporção de penalidade lasso
Aceleração de Árvores (XGBoost)	Espessura da árvore
	Número de árvores
	Taxa de aprendizagem
	Número de preditores selecionados aleatoriamente
	Tamanho mínimo dos nós
	Redução de perda mínima
	Proporção de observações amostradas
	Número de iterações antes de parar
Rede neural de camada única	Número de unidades ocultas
	Quantidade de regularização/penalização
	Número de ciclos de treinamento
Floresta aleatória	Número de preditores selecionados aleatoriamente
	Número de árvores
	Tamanho mínimo dos nós

Fonte – Produzido pelo autor.

Petro Liashchynskyi e Pavlo Liashchynskyi (2019) fazem uma análise comparativa de diferentes métodos de otimização de hiperparâmetros. Como definido pelos autores, a busca em grade faz um mapeamento completo dos possíveis valores dentro de um subconjunto de espaço de hiperparâmetros no algoritmo de treinamento. A escolha dos valores para os parâmetros é definida quando comparamos as estatísticas de erro para eles, escolhendo aquele que minimizá-las. Na Figura 9 podemos identificar o processo de mapeamento para o processo de busca em grade. O ponto que otimiza os hiperparâmetros é colocado em negrito e seus valores são aqueles que apresentam a menor estatística de erro, com distribuição representada pela área rosa, propiciando

Figura 9 – Exemplo de busca em grade para o caso de dois hiperparâmetros.



Fonte – Adaptado de Bergstra e Bengio (2012) pelo autor.

um modelo com maior precisão preditiva. É importante ressaltar que o processo de otimização por busca em grade é demorado, uma vez que ele dá pesos iguais para todos os possíveis candidatos durante o mapeamento dos valores dos hiperparâmetros.

Como apresentado por Bergstra e Bengio (2012), um algoritmo de aprendizagem $\mathcal{A}$ tem como objetivo encontrar uma função f que minimize o erro esperado $\mathcal{L}(x; f)$ sob amostras independentes e identicamente distribuídas de x de uma distribuição $\mathcal{G}_X$. O algoritmo $\mathcal{A}$ mapeia os dados encontrados em $\mathcal{X}^{(treinamento)}$ – um conjunto finito de amostras de $\mathcal{G}_X$ – para uma função f, que será produzida de acordo com um conjunto de hiperparâmetros λ, tendo portando o algoritmo $\mathcal{A}_\lambda$, a função $f = \mathcal{A}_\lambda(\mathcal{X}^{(treinamento)}$ para o conjunto de treinamento $\mathcal{X}^{(treinamento)}$. Dessa forma, precisamos escolher λ de tal forma que ele minimize o erro generalizado $E_{x\sim\mathcal{G}_x}\left[\mathcal{L}\left(x; \mathcal{A}_\lambda(\mathcal{X}^{(treinamento)})\right)\right]$. Em outras palavras, o conjunto de hiperparâmetros ótimos λ será dado por:

$$\lambda^{(*)} = \arg\min_{\lambda\in\Lambda} E_{x\sim\mathcal{G}_x}\left[\mathcal{L}\left(x; \mathcal{A}_\lambda(\mathcal{X}^{(treinamento)})\right)\right]. \tag{87}$$

Como a distribuição $\mathcal{G}_X$ é desconhecida, utilizamos a técnica de validação cruzada para estimá-la. A expectativa é portanto substituída pela média pelo conjunto de testes $\mathcal{X}^{(teste)}$ em que seus elementos são desenhos independentes e identicamente

distribuídos de $x \sim \mathcal{G}_x$. Dessa forma, temos:

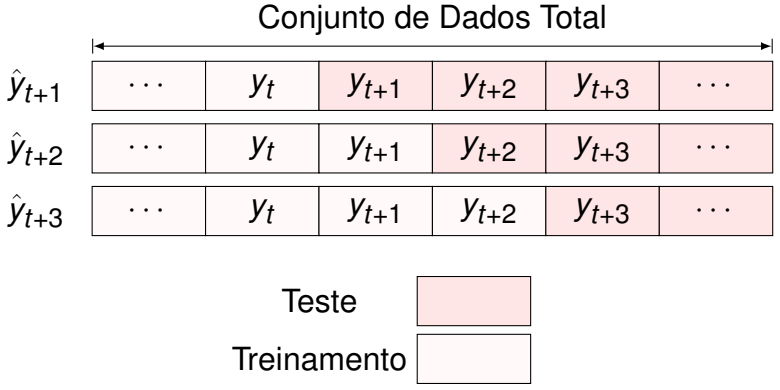
$$\begin{aligned} \lambda^{(*)} &\approx \underset{\lambda \in \Lambda}{\arg\min} \underset{x \in \mathcal{X}^{(teste)}}{\text{média}} \mathcal{L}\left(x; \mathcal{A}_\lambda(\mathcal{X}^{(treinamento)})\right) \\ &\equiv \underset{\lambda \in \Lambda}{\arg\min}\, \boldsymbol{\Psi}(\lambda) \\ &\approx \underset{\lambda \in \{\lambda^1 \ldots \lambda^S\}}{\arg\min}\, \boldsymbol{\Psi}(\lambda) \equiv \widehat{\lambda}. \end{aligned} \tag{88}$$

A função $\boldsymbol{\Psi}(\lambda)$ é chamada de função de resposta do hiperparâmetro. Nosso objetivo principal para encontrar um bom λ é encontrarmos os pontos de tentativa $\{\lambda^1 \ldots \lambda^S\}$ e avaliarmos $\boldsymbol{\Psi}(\lambda)$ para cada um deles de tal forma que temos como resposta o λ^i que melhor se comporta como $\widehat{\lambda}$. Para a escolha de $\{\lambda^1 \ldots \lambda^S\}$ utilizamos a busca em grade, que apesar de sofrer com o problema de dimensionalidade devido o custo computacional, se sobressai em espaços com dimensionalidade pequena.

4.3 ESTIMAÇÃO E RESULTADOS DE PREVISÃO DOS MODELOS PREDITIVOS.

Para a análise comparativa de desempenho dos modelos, separamos a amostra em dois conjuntos de dados: treinamento, que contou com as primeiras 151 observações (70% da amostra) e teste, que contou com as 66 observações finais (30% da amostra). O conjunto de treinamento foi utilizado para aprendizado, no caso dos modelos de aprendizagem de máquina, e para estimação no caso dos modelos autorregressivos. Depois, utilizamos estes resultados para previsão do conjunto de teste, com 66 observações. Optamos por prever 4 séries temporais: IPCA - geral, IBC-Br, taxa de desemprego (PNAD) e SELIC Acumulada com base 252. Estas quatro séries representam as taxas de inflação, crescimento do produto, desemprego e juros básico, importantes variáveis macroeconômicas em uma economia.

Figura 10 – Esquema de previsão iterativa.



Fonte – Produzido pelo autor.

Como vimos no Capítulo 3, os modelos autorregressivos como ARMA, VAR e FAVAR tendem a convergir para a média amostral quando são utilizados para prever

horizontes muito longos. Para contornar este problema e torná-los mais competitivos em relação aos outros modelos, adotamos previsões iterativas: prevemos 1 período a frente (uma vez) por vez. No período seguinte, 1 observação é retirada do conjunto de teste e adicionada no conjunto de treinamento, fazendo uma nova previsão (única) com o modelo. É importante notar que nós não re-estimamos o modelo: a estimação é feita somente uma vez utilizando os dados de treinamento com 151 observações. Entretanto, as previsões um passo a frente passam a contar com a informação do conjunto de teste gradativamente. Esta metodologia tem uma lógica que pode ser descrita na Figura 10, em que $t = 151$, última observação do conjunto de treinamento. Quando vamos fazer a previsão $\hat{y}_{t+2}$, uma observação passa para o conjunto de treinamento e é utilizada para a previsão num modelo que já é estimado.

Para os modelos de aprendizagem de máquina, também adotamos um ajuste que consistiu em não adotar informações contemporâneas para previsão das séries. Por definição, quando vamos prever algo, só temos informação disponível até o momento t para a previsão de $t + 1$. Nesse sentido, as séries que são utilizadas como preditores são defasadas em um período, inclusive sendo incluído a própria série que se deseja prever como preditor, mas defasada. Para a série do IBC-Br, por exemplo, temos todas as 125 séries apresentadas no capítulo anterior, defasadas, além da própria série do IBC-Br defasada.

Toda programação foi feita em linguagem `R`, utilizando os pacotes `tidyverse` e `tidymodels` nos modelos de aprendizagem de máquina; e os pacotes `forecast` e `vars` para os modelos econométricos.

4.3.1 Análise Comparativa dos Erros de Previsão.

Para a análise comparativa do desempenho dos modelos, utilizaremos duas estatísticas: o erro absoluto médio/MAE (do inglês, *mean absolute error*) e a raiz do erro quadrático médio/RMSE (do inglês, *root mean squared error*). Primeiro, comparamos o resultado entre os modelos de aprendizagem de máquina, seguido da comparação destes modelos com os modelos econométricos. O MAE e a RMSE são definidos como:

$$\begin{aligned} \text{MAE} &= \frac{1}{n}\sum_{i=1}^{n}|y_i - \widehat{y}_i|, \\ \text{RMSE} &= \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2}. \end{aligned} \tag{89}$$

Na Tabela 3 temos os resultados destas estatísticas para os modelos de aprendizagem de máquina para cada uma das séries utilizadas, considerando o ajuste para o conjunto de dados de treinamento, teste e de reamostragem, quando utilizamos todos os *folds* de validação cruzada, o que garante maior robustez ao definirmos qual

modelo teve um desempenho melhor. Quando observamos os resultados de teste e reamostragem, uma vez que os resultados para o conjunto de treinamento não são tão informativos do ponto de vista de previsão, temos resultados bastante interessante para os modelos. No IPCA-Geral teve como o modelo de florestas aleatórias com menor RMSE, ao passo que regressão penalizada alcançou o menor MAE. Na série do IBC-Br, o desempenho de redes neurais é substancialmente superior para RMSE, ao passo que XGBoost e regressão penalizada apresentam os menores valores para MAE. No caso da taxa de desemprego, redes neurais tiveram um um desempenho bastante inferior aos outros modelos, ao passo que o desempenho de floresta aleatória é um pouco melhor que o de regressão penalizada. Na série da SELIC, redes neurais voltam a ter o pior desempenho ao passo que os outros modelos apresentam um desempenho mais homogêneo para RMSE e MAE. De toda forma, os modelos apresentaram um desempenho relativamente parecido para boa parte das séries, com exceção de redes neurais que se sobressaíram no caso do IBC-Br.

Tabela 3 – Estatísticas de Erro de Previsão para os modelos de aprendizagem de máquina. Valores em negritos são os menores comparativamente.

Modelo	**Treinamento**		**Teste**		**Reamostragem**	
Série: IPCA - geral	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**
Regressão Penalizada	0.559	0.441	1.020	**0.782**	**1.020**	**0.782**
Xgboost	**0.013**	**0.008**	1.100	0.862	1.120	0.880
Rede Neural	0.030	0.021	1.280	0.997	1.240	1.000
Floresta Aleatória	0.273	0.207	**1.010**	0.800	**1.020**	0.801
Série: IBC-Br	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**
Regressão Penalizada	0.061	0.049	0.170	**0.098**	0.170	0.098
Xgboost	0.047	0.035	0.175	**0.098**	0.174	**0.097**
Rede Neural	0.045	0.036	**0.164**	0.100	**0.164**	0.100
Floresta Aleatória	**0.032**	**0.024**	0.176	0.101	0.176	0.101
Série: Tx. Desemprego	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**
Regressão Penalizada	0.146	0.102	0.158	0.127	0.158	0.127
Xgboost	0.106	0.068	0.173	0.144	0.170	0.140
Rede Neural	**0.068**	**0.051**	0.222	0.166	0.222	0.166
Floresta Aleatória	0.087	0.057	**0.152**	**0.124**	**0.152**	**0.125**
Série: SELIC	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**
Regressão Penalizada	0.047	0.038	**0.039**	**0.023**	**0.039**	**0.023**
Xgboost	0.042	0.022	0.041	0.029	0.038	0.029
Rede Neural	0.039	0.027	0.064	0.050	0.064	0.050
Floresta Aleatória	**0.021**	**0.012**	0.040	0.027	0.040	0.028

Fonte – Produzido pelo autor.

Já na Tabela 4, comparamos os resultados das estatísticas de previsão dos modelos de aprendizagem de máquina com os modelos econométricos de previsão de séries temporais. O uso Regressão Linear estimado por Mínimos Quadrados Ordinários (MQO), que na verdade é uma simples regressão linear estimada por este método, serve como controle, apesar de ter sido capaz de superar os resultados do

modelo VAR para algumas séries. Nos modelos ARMA, utilizamos o pacote `forecast` com a função `auto.arima`, que faz a seleção das defasagens (p,q) por meio de critérios de informação AIC. Para a série do IPCA - geral, o modelo identificou um processo ARMA(3,1); para IBC-Br, um processo AR(2); para a taxa de desemprego (PNAD), um processo ARMA(1,2) e para a SELIC base 252 um processo AR(1). No modelo VAR, utilizamos o pacote `vars` com a função `VARselect`, que com as quatro séries temporais utilizadas, identificou com AIC um processo VAR(4). Para o FAVAR, também utilizamos o AIC para as defasagens do modelo, que identificou para o IPCA - geral um VAR(2); para IBC-Br um VAR(4); para taxa de desemprego um VAR(9) e para a taxa de juros um VAR(10).

A análise comparativa das estatísticas dos modelos na Tabela 4 apresenta o resultado de RMSE e MAE para todas as séries, sendo que a última coluna apresenta a média de RMSE e MAE para todas as séries. Esta coluna mostra um desempenho superior dos modelos de aprendizagem de máquina, especialmente o modelo de florestas aleatórias com RMSE e regressão penalizada com o MAE. Na análise das séries, florestas aleatórias tem um desempenho muito bom para o IPCA, seguido de regressão penalizada; já no caso da SELIC, o modelo ARMA e FAVAR tiveram um desempenho bastante próximo dos modelos de aprendizagem de máquina. Estes modelos, juntos ao XGBoost e Florestas Aleatórias, tiveram um resultado muito próximo de regressão penalizada. No caso da série do IBC, temos um melhor desempenho para o modelo de redes neurais no caso do RMSE, ainda que o menor MAE tenha sido do modelo FAVAR. Já a taxa de desemprego mostrou desempenho melhor para o modelo ARMA, seguido do modelo FAVAR.

Tabela 4 – Estatísticas de Erro de Previsão para os modelos de aprendizagem de máquina e Modelos Tradicionais para o Conjunto de Teste. Valores em negritos são os menores comparativamente.

Modelo	**IPCA - Geral**		**IBC-Br**		**Tx. Desemprego**		**SELIC**		**Média**	
	RMSE	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**
Reg. Penalizada	1.020	**0.782**	0.170	0.098	0.158	0.127	**0.039**	**0.023**	0.347	**0.257**
XGBoost	1.100	0.862	0.175	0.098	0.173	0.144	0.041	0.029	0.372	0.283
Redes Neurais	1.280	0.997	**0.164**	0.100	0.222	0.166	0.064	0.050	0.432	0.328
Fl. aleatórias	**1.010**	0.800	0.176	0.101	0.152	0.124	0.040	0.027	**0.344**	0.263
Reg. Linear	2.417	1.855	2.644	1.999	0.431	0.327	0.204	0.156	1.424	1.084
ARMA	1.084	0.850	0.184	0.096	**0.123**	**0.103**	0.041	0.024	0.358	0.268
VAR	1.931	1.532	0.479	0.349	0.521	0.402	0.947	0.733	0.969	0.754
FAVAR	1.067	0.829	0.181	**0.093**	0.145	0.117	0.046	0.032	0.360	0.268

Fonte – Produzido pelo autor.

4.3.1.1 Model Confidence Set (MCS)

Hansen, Lunde e Nason (2011) apresentam uma metodologia conhecida por *Model Confidence Set* (MCS), que seleciona um conjunto de modelos que apresentam a melhor performance preditiva para um dado nível de confiança, sendo semelhante ao intervalo de confiança utilizado na estimação de parâmetros em uma regressão. A adoção deste é interessante quando se deseja comparar uma quantidade significativa de modelos simultaneamente.

Dado um conjunto de modelos $\mathcal{M}_0$ que se deseja comparar, um 'teste de equivalência' $\delta_{\mathcal{M}}$ e uma regra de eliminação $e_{\mathcal{M}}$; o teste de equivalência é aplicado ao conjunto de modelos $\mathcal{M} = \mathcal{M}_0$. Se rejeitamos $\delta_{\mathcal{M}}$, concluímos que os modelos presentes em $\mathcal{M}$ não possuem a mesma performance preditiva e utilizamos $e_{\mathcal{M}}$ para remover os modelos que não apresentaram uma boa performance comparativa. Esse passo é adotado até que $\delta_{\mathcal{M}}$ não seja mais rejeitada, sobrando apenas os modelos presentes no conjunto $\mathcal{M}^*$ que será o *model confidence set*.

A análise comparativa dos modelos é feita pela diferença entre as funções de perda ($d_{ij,t} \equiv L_{i,t} - L_{j_t}$, em que i, j são modelos pertencentes ao conjunto $\mathcal{M}_0$). A hipótese nula será:

$$H_{0,\mathcal{M}} : E(d_{ij,t}) = 0, \forall i,j \in \mathcal{M} \subset \mathcal{M}_0.$$

O conjunto de melhores modelos $\mathcal{M}^*$ será definido como:

$$\mathcal{M}^* \equiv \{i \in \mathcal{M}_0 : E(d_{ij,t}) \leq 0, \forall j \in \mathcal{M}_0\}.$$

Os procedimentos para adoção do MCS é dado pelo seguinte algoritmo:

Algoritmo 4: Construção do Model Confidence Set

1. Inicia definindo $\mathcal{M} = \mathcal{M}_0$.
2. Teste para a hipótese de igual capacidade preditiva (EPA) para os modelos em $\mathcal{M}$:
   a) Se a hipótese de igual capacidade preditiva não é rejeitada, define $\widehat{\mathcal{M}}^*_\alpha \equiv \mathcal{M}$ e reporta o $(1-\alpha)$ conjunto de confiança.
   b) Se a hipótese de igual capacidade preditiva é rejeitada:
      i. Define-se
      $$d_i \equiv \frac{1}{m} \sum_{j \in \mathcal{M}} d_{ij},$$
      em que m é o número de modelos em $\mathcal{M}$. Esta estatística define a performance do modelo i comparativamente a média de todos os outros modelos.
      ii. Determina o modelo com pior performance em $\mathcal{M}$, definido como
      $$i^* \equiv \arg\max_{i \in \mathcal{M}} \frac{d_{ij}}{\sqrt{\widehat{var}(d_{ij})}},$$
      em que $\widehat{var}(d_{ij})$ é a estimação da variância de d_i. Este termo maximizado é chamado de estatística t_{ij}.
      iii. Remove o modelo i^* de $\mathcal{M}$ e repete o passo 2.

Na Tabela 5 temos os resultados dos melhores modelos pelos métodos descritos, para um α de 0.2. Quanto maior o número de modelos eliminados, maior a heterogeneidade entre os modelos. Em geral, os resultados do ranqueamento são semelhantes aqueles observados quando fazemos a análise comparativa das estatísticas RMSE e MAE entre os modelos. Quanto menor o p-valor MCS, menor sua probabilidade de fazer parte do conjunto de melhores modelos. Os modelos tradicionais em geral participam do conjunto de melhores modelos, mas com probabilidade muito menor que os modelos de aprendizagem de máquina – o pódio é disputado entre os modelos de floresta aleatória e regressão penalizada – salvo o caso da taxa de desemprego, que apresentou como melhor modelo o modelo ARMA.

Tabela 5 – Conjunto de Melhores Modelos para as séries. Modelos com X foram excluídos.

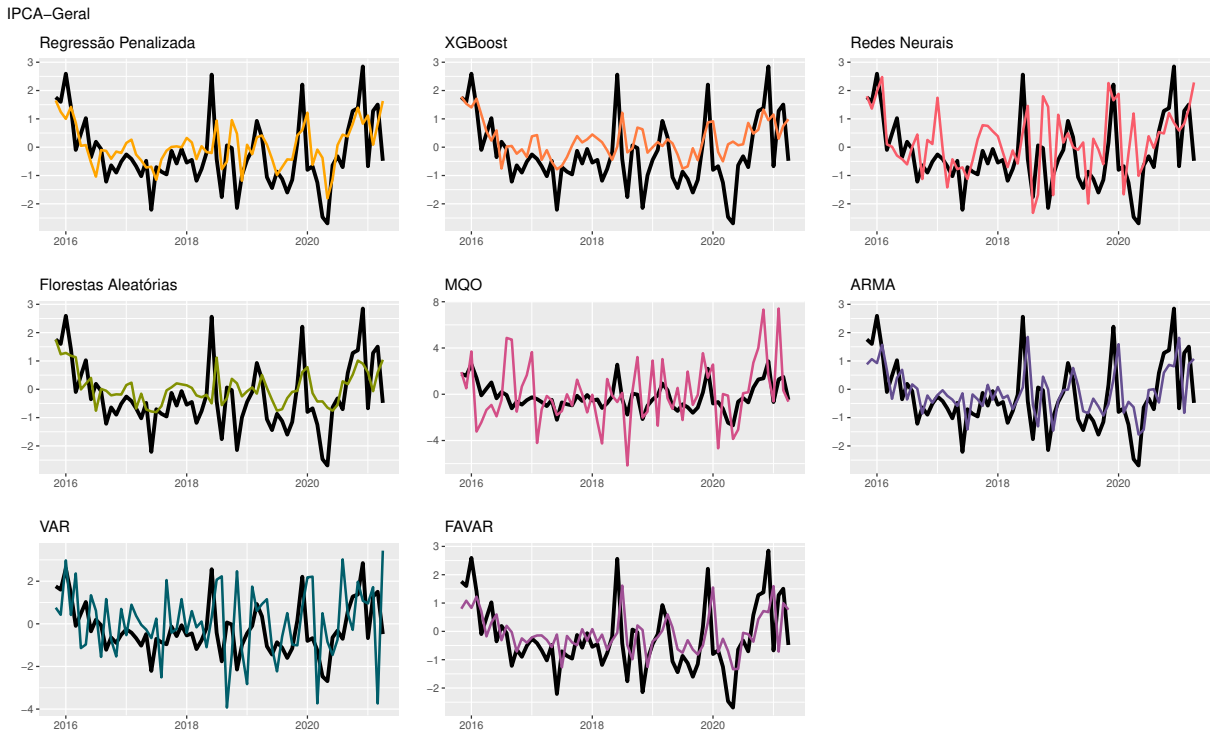
Modelo	**Rank**	t_{ij}	**p-valor MCS**
IPCA - geral: *3 eliminações*			
Regressão Penalizada	2	0.04667257	1.0000
Xgboost	4	1.74823851	0.2774
Floresta Aleatória	1	-0.04667257	1.0000
ARMA	5	1.77164041	0.2670
FAVAR	3	1.49311648	0.4348
IBC-Br: *2 eliminações*			
Regressão Penalizada	2	0.6938866	0.9040
Xgboost	3	0.9946154	0.7706
Rede Neural	1	-0.6938866	1.0000
Floresta Aleatória	6	1.7603473	0.2928
ARMA	5	1.4620005	0.4874
FAVAR	4	1.4037857	0.5254
Taxa de Desemprego: *7 eliminações*			
ARMA	1	-2.511263	1
SELIC Base 252: *4 eliminações*			
Regressão Penalizada	1	-0.1520710	1.0000
Xgboost	3	0.2143146	0.9930
Floresta Aleatória	2	0.1520710	0.9976
ARMA	4	1.4844880	0.3402

Fonte – Produzido pelo autor.

4.3.2 Análise Gráfica das Séries.

A Figura 11 mostra a previsão dos modelos para a série do IPCA. Como vimos nas estatísticas de erro da Tabela 4, o modelo de florestas aleatórias teve um desempenho muito semelhante com a regressão penalizada, o que se comprova no desenho de suas trajetórias. A maior diferença é que a previsão para o modelo de florestas aleatórias aparenta ser mais suave do que a trajetória de previsão do modelo de regressão penalizada. O modelo VAR teve a pior trajetória comparativa para esses modelos, sendo superior apenas ao modelo de regressão linear (MQO), mal desempenho este que se estende por todas as outras séries.

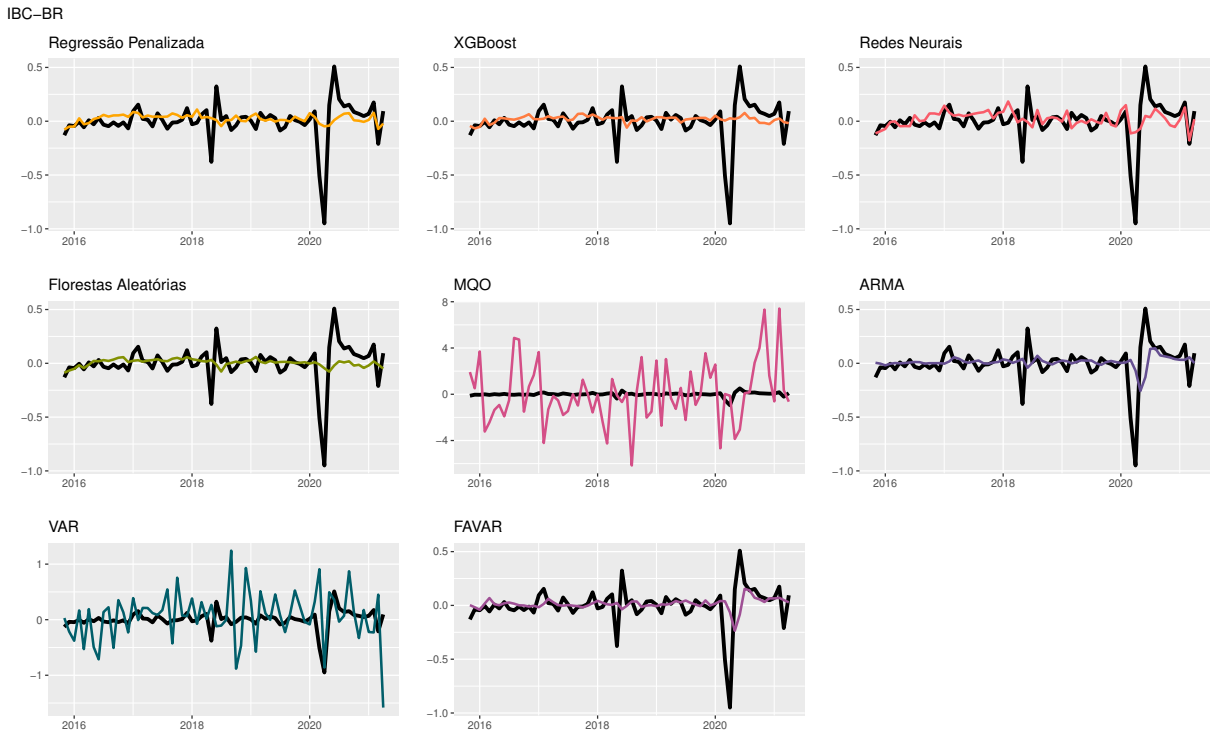
Figura 11 – Previsão do Conjunto de Teste da Série IPCA-Geral.



Fonte: Produzido pelo autor.

A Figura 12 apresenta os resultados dos modelos para a série IBC-BR. O modelo de redes neurais, que apresentou o menor valor para a estatística de RMSE, foi o único capaz de prever minimamente o vale causado pela crise econômica da pandemia da COVID-19. O componente autorregressivo dos modelos ARMA e FAVAR previu o vale do período com atraso, como observamos na figura.

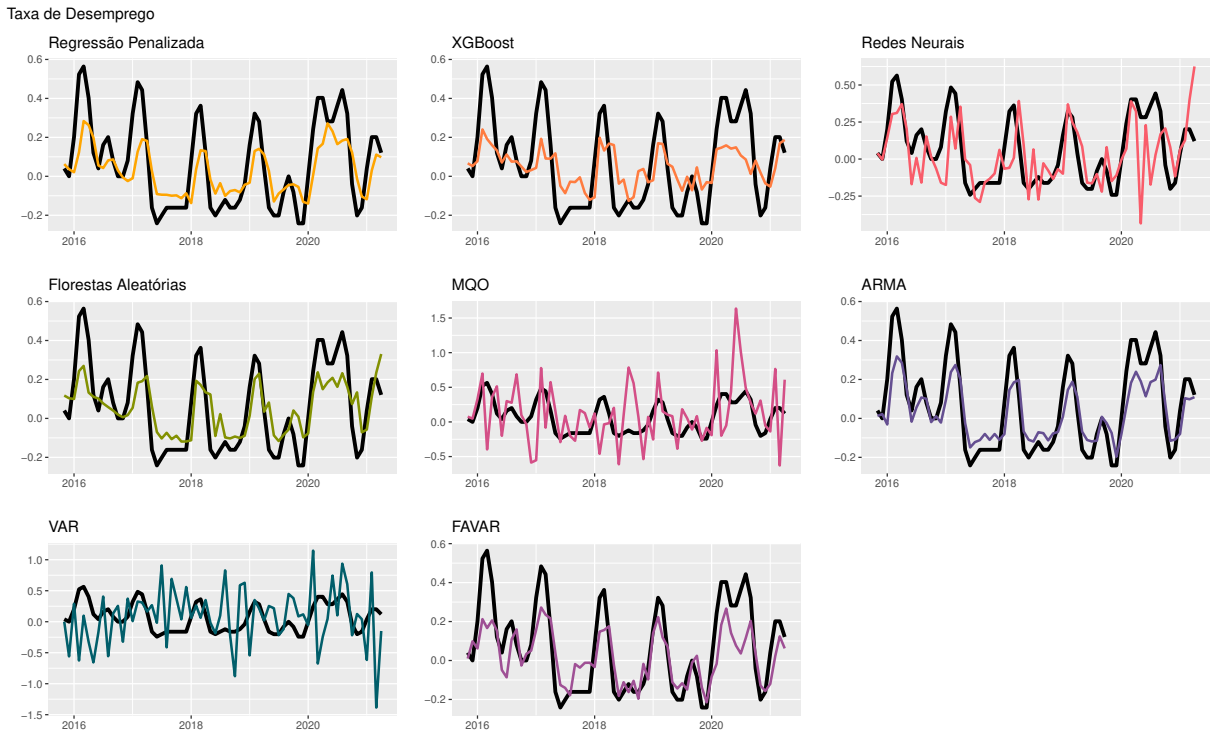
Figura 12 – Previsão do Conjunto de Teste da Série IBC-Br.



Fonte: Produzido pelo autor.

Na Figura 13, temos os resultados para a taxa de desemprego. Nesta série, o modelo ARMA teve o melhor desempenho preditivo. De fato, quando observamos a trajetória, este modelo que embora simples, teve o melhor ajuste comparativo com a série original. O modelo FAVAR também teve um resultado interessante, segundo o segundo colocado na análise comparativa de RMSE e MAE entre os modelos. Ainda que não tem alcançado a mesma magnitude, os modelos de regressão penalizada e florestas aleatórias previram uma trajetória semelhante para a série temporal do desemprego. Redes neurais teve um resultado muito ruim, só não sendo pior que regressão linear e VAR.

Figura 13 – Previsão do Conjunto de Teste da Série Taxa de Desemprego.

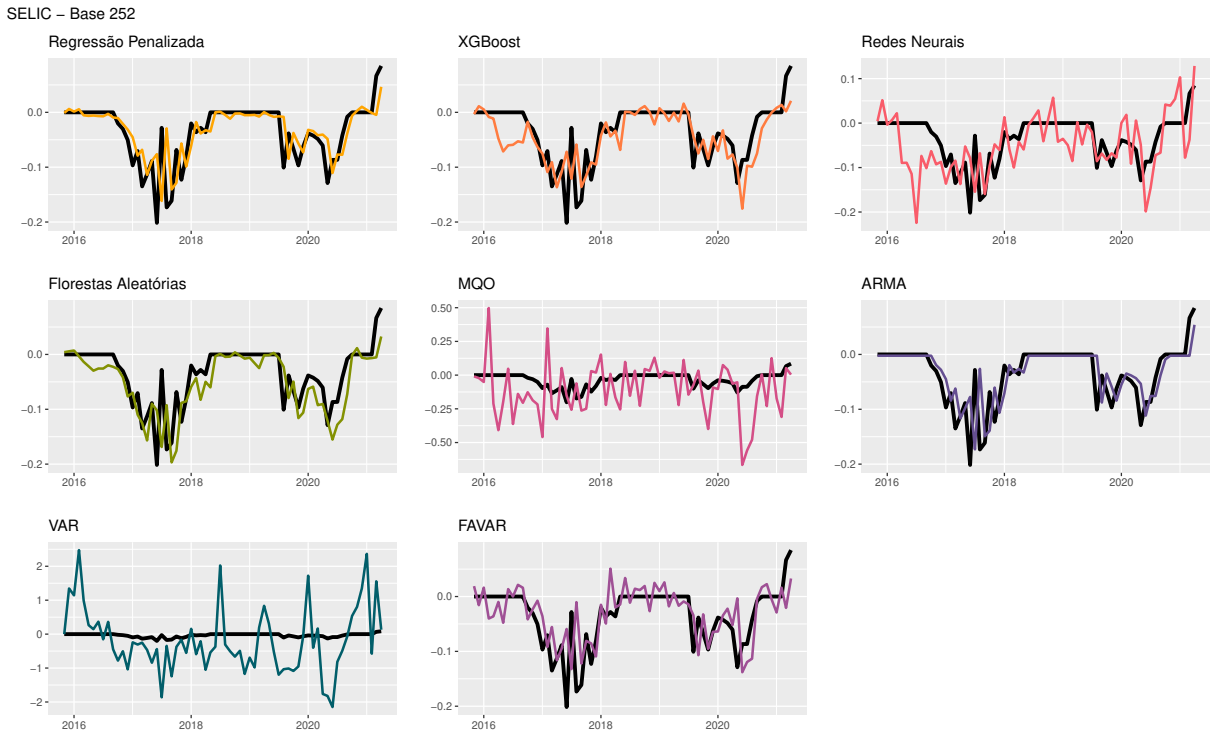


Fonte: Produzido pelo autor.

Por último, a Figura 14, que apresenta os resultados para a taxa de juros básica da economia brasileira, mostrou desempenhos interessantes para os modelos de aprendizagem de máquina e modelos autorregressivos, com exceção do VAR. Ainda que regressão penalizada tenha obtido as menores estatísticas de erro, a análise gráfica nos mostra que sua previsão é pouco informativa em parte das observações: em geral, ele mimetizava o comportamento da série, mas com atraso. O mesmo se observa para o caso do modelo ARMA, que ao identificar um processo AR(1) com um coeficiente muito próximo de 1, teve a trajetória de previsão muito semelhante da série original, com atraso. Modelos XGBoost e Florestas aleatórias, que tiveram estatísticas ligeiramente muito próximas de regressão penalizada, acabam sendo mais interessantes do ponto de vista de previsão, especialmente florestas aleatórias, que aparenta um ajuste visual bastante interessante para a série. O comportamento dos resultados de previsão para o modelo FAVAR também mostrou-se interessante, embora com variância maior. De fato, a estatística RMSE e MAE para este modelo é relativamente próxima daquelas obtidas pelas previsões de XGBoost e florestas aleatórias. É importante notar que, assim como ocorreu na Figura 13, o gráfico para regressão linear e VAR estão em diferentes escalas dos demais devido à magnitude da variação dos resultados da previsão destes modelos, entretanto os dados das curvas em preto, para todos os gráficos, são as mesmas, pois são as séries originais estacionarizadas

para o conjunto de teste.

Figura 14 – Previsão do Conjunto de Teste da Série SELIC-Base 252.



Fonte: Produzido pelo autor.

5 INTERPRETABILIDADE DOS MODELOS DE APRENDIZAGEM DE MÁQUINA

No Capítulo 2 nos dedicamos em apresentar os modelos de aprendizagem de máquina, identificando os processos e etapas que estes algoritmos utilizam para fins de previsão. Como vimos, alguns dos modelos apresentam algum mecanismo de interpretabilidade inerente ao próprio modelo: o modelo de regressão penaliza possui seus parâmetros; o modelo de florestas aleatórias e métodos de aceleração em árvore possuem a estrutura da árvore de decisão para análise de resultados. Entretanto, na prática e principalmente num contexto com muitas variáveis explicativas – como é o caso do conjunto de séries temporais que adotamos – a interpretabilidade inerente destes modelos (com ressalva à regressão penalizada, que estima os coeficientes e é o único método linear que adotamos) é prejudicada. Neste capítulo, nossos esforços se voltarão em estudar e apresentar métodos de interpretabilidade agnósticos, universalmente aplicáveis aos modelos de aprendizagem de máquina e que nos fornecerão importantes pistas pelos mecanismos utilizados por estes modelos na previsão das séries temporais.

5.1 POR QUE INTERPRETABILIDADE IMPORTA?

Miller (2019) define interpretabilidade como o grau de capacidade em que humanos são aptos a entender a causa da decisão ou prever consistentemente o resultado de um modelo. Quanto mais interpretável um modelo de aprendizagem de máquina é, mais fácil para um humano interpretar seus resultados e, consequentemente, mais fácil para este humano explicar aos seus pares os resultados obtidos pelo modelo. Como apontado pelo autor, interpretabilidade é significante do ponto de vista de explicação científica, já que existe uma estrutura lógica para tal fim composto por cinco estruturas: a teoria, que é um conjunto de princípios que justifica o modelo; o modelo, que é a abstração da teoria que representa relações entre "tipos"; os "tipos", que representam uma classe universal abstrata que suporta raciocínio contrafactual; as entidades, uma instanciação dos "tipos"; e os dados, que mensuram as entidades e são observações[1].

Uma discussão mais aprofundada é dada por Molnar (2019), uma vez que em alguns casos não estamos interessados apenas no que será previsto, mas também no porquê dessa previsão ser feita da forma que foi. Neste caso, uma boa previsão só resolve parcialmente o problema original que se deseja entender, pois parte da curiosidade humana é encontrar significado no mundo, buscando um equilíbrio entre as contradições e inconsistências na nossa estrutura de conhecimento. Essa estrutura definida pelo autor pode ser sintetizada pela Figura 15. A interpretação do modelo de aprendizagem de máquina é significante porque queremos saber se uma variável é

[1] Como exemplo do autor, um artrópode (entidade) possui 8 patas (dados). Entidades desse tipo são aranhas, de acordo com o modelo da teoria sobre artrópodes.

Figura 15 – Esquema de Interpretação dos Métodos de Aprendizagem de Máquina (ML)

Fonte: Adaptada pelo autor em base de Molnar (2019).

importante para previsão de outra (testando se existe um relação entre elas), se mudanças nos valores das mesmas afetam o desempenho do resultado final (comparando períodos com e sem crises econômicas, por exemplo) ou o que aconteceria se tivéssemos observações diferentes (comparando as mesmas variáveis em diferentes países, por exemplo). Nas palavras de Molnar (2019), "(...) quanto mais a decisão de uma máquina afeta a vida de uma pessoa, maior a importância para a máquina de explicar seu comportamento."[2] Ainda assim, existem problemas resolvidos por aprendizagem de máquina que não necessariamente precisam ser interpretáveis, pois a previsão da variável de interesse já é suficiente. Entretanto, em economia – em especial em macroeconomia – a interpretabilidade é importante porque auxilia os formuladores de política econômica em decisões de gestão e ajuda os cientistas no desenvolvimento de sua própria teoria.

5.2 MÉTODOS AGNÓSTICOS DE INTERPRETAÇÃO

Nessa seção, apresentaremos um conjunto de métodos de interpretação agnósticos para modelos de aprendizagem de máquina. Molnar (2019) é referência nesta discussão e por isso será referência para as técnicas apresentadas aqui, que posteriormente serão aplicadas nos modelos de aprendizagem de máquina para as séries

[2] Tradução livre de: *The more a machine's decision affects a person's life, the more important it is for the machine to explain its behavior.*

temporais brasileiras.

5.2.1 Importância de Variáveis via Permutação

A importância de variáveis via permutação calcula a importância de uma variável explicativa a partir do cálculo do acréscimo no erro de previsão do modelo após permutar esta variável: caso "embaralhar"os valores da variável aumentem o erro do modelo, dizemos que a variável é importante; se o erro não é alterado, então ela é chamada não importante. O algoritmo para aplicação da Importância do Recurso de Permutação é o seguinte:

Algoritmo 5: Importância do Recurso de Permutação

Entrada: Modelo treinado $\widehat{f}$, matriz de preditores $\mathbf{X}$, vetor alvo $\mathbf{y}$, erro de previsão $L(\mathbf{y},\widehat{f})$.

1. Estima o erro original do modelo $e_{orig} = L(\mathbf{y},\widehat{f}(\mathbf{X}))$.
2. Para cada preditor $j \in \{1,\ldots,p\}$ faz:
   a) Cria uma matriz de preditores $\mathbf{X}_{perm}$ ao permutar o preditor j nos dados em $\mathbf{X}$, causando a quebra na associação entre o preditor j e o valor verdadeiro de $\mathbf{y}$.
   b) Estima o erro $e_{perm} = L(\mathbf{Y},\widehat{f}(\mathbf{X}_{perm}))$ baseado nas previsões dos dados permutados.
   c) Calcula a importância do recurso de permutação como o quociente $FI_j = e_{perm}/e_{orig}$ ou pela diferença $FI_j = e_{perm} - e_{orig}$.
3. Ranqueia as importâncias por FI em ordem decrescente.

Em geral, a importância de variáveis via permutação garante uma boa interpretação do modelo, dando importantes pistas sobre o comportamento do modelo, o que também permite que comparemos estes resultados entre diferentes modelos. Ainda assim, apresenta uma limitação: a análise dos resultados pode ser enviesada quando preditores são muito correlacionados, o que pode, inclusive, causar uma queda em sua importância relativa.

5.2.2 Efeitos Locais Acumulados

Efeitos locais acumulados (ALE, do inglês *Accumulated Local Effects*) calculam a influência dos preditores na média da previsão dos modelos de aprendizagem de máquina. Esta técnica é especialmente boa porque não é enviesada nos casos em que os preditores são fortemente correlacionados, porque cria pequenos blocos com os preditores, vendo as mudanças nas previsões a partir deles. Nesse sentido, o cálculo dos ALEs diminui a complexidade da função de previsão ao utilizar apenas um ou dois preditores, ao invés do conjunto total de preditores, a partir da função de distribuição condicional.

O cálculo do ALE se baseia na média das mudanças nas previsões e acumulam estes resultados em um intervalo, a partir da função de distribuição condicional:

$$\begin{aligned}\widehat{f}_{S,ALE}(x_S) &= \int_{z_{0,S}}^{x_S} E_{\mathbf{X}_C|\mathbf{X}_S=x_S}\left[\widehat{f}^S(\mathbf{X}_S,\mathbf{X}_C)|\mathbf{X}_S = z_S\right] dz_S - \text{constante} \\ &= \int_{z_{0,S}}^{x_S} \left[\widehat{f}^S(z_S,\mathbf{X}_C) d\mathbb{P}(\mathbf{X}_C|\mathbf{X}_S = z_S)\right] dz_S - \text{constante},\end{aligned} \tag{90}$$

em que f é a função de previsão, x_S é o valor de um preditor calculado pela média de preditores em $\mathbf{X}_C$ (tratadas como variáveis aleatórias), em que o cálculo da média é obtido pela expectativa marginal sob os preditores no conjunto C, isto é, a integral dos preditores com o peso calculado pela distribuição de probabilidade. Nesse sentido, são comparados os dois conjuntos S e C, a partir de um intervalo de valores z, para encontrar o efeito do preditor na previsão. O termo $\widehat{f}^S(\mathbf{X}_S,\mathbf{X}_C)$ é a derivada parcial de $\widehat{f}(x_S,x_C)$ em relação à x_S. É estimado os efeitos locais a partir de uma fórmula não centralizada:

$$\tilde{f}_{j,ALE}(x) = \sum_{k=1}^{k_j(x)} \frac{1}{n_k(k)} \sum_{i:x_j \in N_j(k)} \left[\widehat{f}(z_{k,j},x_j^{(i)}) - \widehat{f}(z_{k-1,j},x_j^{(i)})\right]. \tag{91}$$

Pela fórmula, é possível notar que o ALE calcula a diferença nas previsões (efeito), substituindo o preditor de interesse pelos valores de grade z. A soma deste efeitos aparece na fórmula como uma vizinhança $N_j(k)$, que ao dividirmos pelo número de instâncias nos dá a noção de localidade. A soma de todos estes efeitos nos dá a ideia de acumulação. Quando utilizamos a fórmula centralizada, o efeito médio é zero e temos o que segue:

$$\widehat{f}_{j,ALE}(x) = \tilde{f}_{j,ALE}(x) - \frac{1}{n}\sum_{i=1}^{n} \tilde{f}_{j,ALE}(x_j^{(i)}. \tag{92}$$

O valor obtido pelo ALE é interpretado como o efeito principal de um preditor em certo valor comparado com a média prevista dos dados. Os quantis da distribuição do preditor são usados como as grades que definem os intervalos utilizados. As vantagens de usar ALE é que seus resultados não são enviesados quando os preditores são fortemente correlacionados, o tempo de computação é relativamente rápido e a interpretação dos resultados são claros.

5.2.3 Interação entre Preditores

Quando preditores interagem entre si, a previsão da variável de interesse não pode ser vista como a soma dos efeitos dos preditores, uma vez que o efeito de um preditor depende do valor de outro preditor. Nesse sentido, é importante avaliarmos a interação entre os preditores. Existem diversas formas de estimar a força de interação

entre preditores, sendo a estatística-H bastante utilizada, que mensura o quanto a variação de um preditor depende da interação de outros preditores. Sendo o função de dependência parcial (PD, do inglês *Partial Dependence*) uma função que mostra o efeito de marginal de um ou dois preditores no resultado final de predição, se os dois preditores não interagem entre si, ela pode ser escrita como a soma das funções individuais $PD_{jk}(x_j,x_k) = PD_j(x_j) + PD_k(x_k)$. Isso implica que, sem interação de um preditor com nenhum outro, a função de previsão do modelo de aprendizagem de máquina pode ser escrita como:

$$\widehat{f}(x) = PD_j(x_j) + PD_{-j}(x_{-j}). \tag{93}$$

Esta decomposição representa a dependência parcial sem interação entre o preditor j e todos os outros preditores. É então comparado a função de dependência parcial observada e a função de dependência parcial decomposta, calculando a variância do resultado, que serve como forma de análise para a interação. Para isso, calculamos a estatística-H: se esta estatística é zero, não existe interação entre o preditor j e os outros preditores; caso seja um, toda a variança de PD_{jk} ou $\widehat{f}$ é explicada pela soma das funções de dependência parcial – isto é, cada ponto da função de dependência parcial é constante e o efeito na previsão só acontece com a interação das variáveis. Matematicamente, a estatística-H é calculada da seguinte forma quando desejamos entender a interação do preditor j com todos os outros preditores:

$$H_j^2 = \frac{\sum_{i=1}^{n}[\widehat{f}(x^{(i)}) - PD_j(x_j^{(i)}) - PD_{-j}(x_{-j}^{(i)})]^2}{\sum_{i=1}^{n}\widehat{f}^2(x^{(i)})}. \tag{94}$$

A estatística-H tem um custo computacional bastante grande, porque é necessário avaliar a função de dependência parcial para cada ponto da amostra. Ainda assim, seus resultados são interessantes porque nos mostram a parcela da variância resultado da interação entre os preditores, o que nos permite uma análise inclusive através de modelos.

5.2.4 Valores de Shapley

Valor de Shapley é uma técnica que utiliza teoria dos jogos para interpretação de modelos de aprendizagem de máquina. Em um jogo cooperativo, cada valor de um preditor é um jogador e o *payout* destes jogadores é a previsão da variável de interesse. A intuição do valor de Shapley, como aponta Molnar (2019), é simples: o valor de um preditor entra em uma sala em ordem aleatória para participar de um jogo, isto é, contribuir para a previsão. O valor de Shapley do valor de um preditor é a mudança média na previsão que a cooperação dentro da sala recebe quando um valor do preditor se junta a eles.

Dado um subconjunto S de preditores utilizados no modelo, x um vetor de valores de um preditor para ser explicado e p o número total de preditores, o valor de Shapley é a contribuição para o *payout*, ponderada e somada por todas as possíveis combinações de valores do preditor:

$$\varphi_j(val) = \sum_{S \subseteq \{1,\ldots,p\} \setminus \{j\}} \frac{|S|!(p-|S|-1)!}{p!}(val(S \cup \{j\}) - val(S)). \tag{95}$$

A função $val_x(S)$ é a previsão para cada valor dos preditores no conjunto S que são marginalizadas em relação a preditores que não são incluídas no conjunto S, representando a cooperação entre os jogadores.

$$val_x(S) = \int \widehat{f}(x_1, \ldots, x_p) d\mathbb{P}_x - E_X(\widehat{f}(X)). \tag{96}$$

Para avaliar a Equação (96), são feitas múltiplas integrações, uma para cada valor que se deseja mensurar o cooperação para previsão. O valor de Shapley apresenta quatro propriedades importantes: eficiência, simetria, *dummy* e aditividade, em que, quando satisfeitas, garantem que o *payout* entre os jogadores seja justo. Por eficiência, entende-se que a contribuição do preditor deve ser igual a diferença entre a previsão para x e a média:

$$\sum_{j=1}^{p} \varphi_j = \widehat{f}(x) - E_X(\widehat{f}(X)).$$

Já simetria nos mostra que a contribuição de dois valores j e k de um preditor devem ser a mesma caso eles contribuem igualmente para todas as possíveis cooperações:

$$val(S \cup \{j\}) = val(S \cup \{k\}) \forall S \subseteq \{1, \ldots, p\} \setminus \{j,k\} \implies \varphi_j = \varphi_k.$$

Se um preditor j não altera o valor previsto, independente de qual cooperação os seus valores são sujeitos, temos que o seu valor de Shapley é zero, sendo esta a propriedade chamada de *dummy*. Se:

$$val(S \cup \{j\}) = val(S) \forall S \subseteq \{1, \ldots, p\} \implies \varphi_j = 0.$$

Por último, a aditividade nos diz que para um jogo com os *payouts* combinados $val + val^+$, os respectivos valores de Shapley são:

$$\varphi_j + \varphi_j^+.$$

Esta última propriedade nos diz que, para uma floresta aleatória, por exemplo, podemos calcular o valor de Shapley para cada árvore individual, encontrar a média de cada uma, e então encontrar o valor de Shapley para o valor do preditor da floresta aleatória.

Como o cálculo do valor de Shapley exato depende de todas as possíveis cooperações (conjuntos) de valores de um preditor, o cálculo do mesmo pode ser problemático quando tempos muitos preditores. Adotamos então técnicas de aproximação por Monte-Carlo:

$$\widehat{\varphi}_j = \frac{1}{M} \sum_{m=1}^{M} \left(\widehat{f}(x_{+j}^m) - \widehat{f}(x_{-j}^m) \right), \tag{97}$$

em que $\widehat{f}(x_{+j}^m)$ é a previsão para x, mas com um número aleatório de valores do preditor substituídos por valores do preditor de um ponto aleatório dos dados z, com exceção do valor específico j. O vetor x_{-j}^m é praticamente idêntico ao vetor x_{+j}^m, embora o valor x_j^m seja retirado da amostra z. M representa diferente interações, que juntas criam um "Monstro de Frankenstein", nas palavras de Molnar (2019), porque a estimação final do valor de Shapley será a união de diferentes retalhos construídos de forma aleatória. Temos o seguinte algoritmo:

Algoritmo 6: Estimação do Valor de Shapley para um único valor de um preditor.

Requisito: Número de interações M, instância de interesse x, índice j do preditor, matriz de dados X e modelo de aprendizagem de máquina f.

- Para todo $m = 1, \ldots, M$:
  1. Desenha instâncias aleatórias z para a matriz de dados X;
  2. Escolhe uma permutação aleatória o dos valores do preditor;
  3. Ordena a instância x: $x_o = (x_{(1)}, \ldots, x_{(j)}, \ldots, x_{(p)})$;
  4. Ordena a instância z: $z_o = (z_{(1)}, \ldots, z_{(j)}, \ldots, z_{(p)})$;
  5. Constrói duas novas instâncias:
     a) Com j: $x_{+j} = (x_{(1)}, \ldots, x_{(j-1)}, x_{(j)}, z_{(j+1)} \ldots, z_{(p)})$;
     b) Sem j: $x_{-j} = (x_{(1)}, \ldots, x_{(j-1)}, z_{(j)}, z_{(j+1)} \ldots, z_{(p)})$.
  6. Computa a contribuição marginal $\varphi_j^m = \widehat{f}(x_{+j}) - \widehat{f}(x_{-j})$;
- Computa o valor de Shapley como a média: $\varphi_j(x) = \frac{1}{M} \sum_{m=1}^{M} \varphi_j^m$.

Uma das vantagens do valor de Shapley é que seus valores são distribuídos de forma justa entre as instâncias, garantidos pela propriedade de eficiência discutida anteriormente, além de ter capacidade de explicação de um único ponto ou de toda a amostra. A desvantagem, como é de se imaginar, é o alto custo computacional para calcular seu valor, que na maioria massante das vezes nos reduz apenas a estimá-lo.

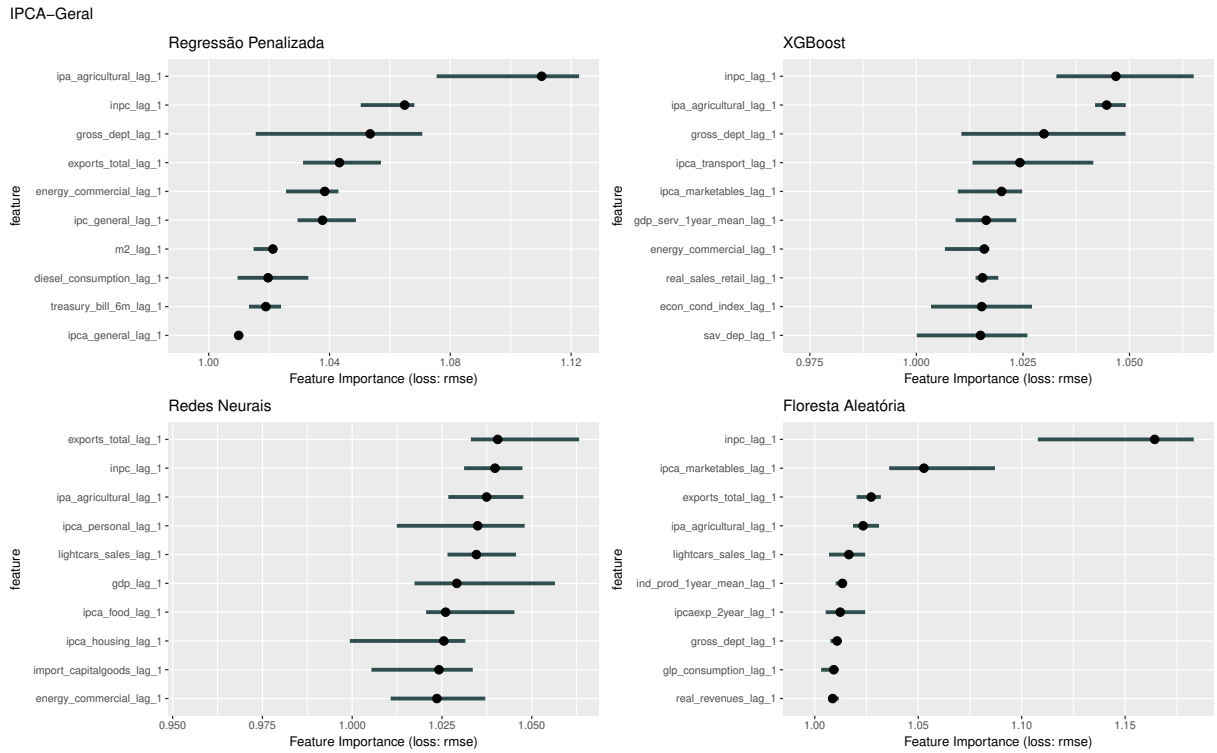
5.3 APLICAÇÃO DE MÉTODOS AGNÓSTICOS DE INTERPRETAÇÃO

No Capítulo 4 apresentamos os resultados de previsão para os modelos de aprendizagem de máquina comparativamente aos métodos tradicionais de séries temporais, em que os primeiros mostraram resultados mais satisfatórios. Entretanto, não

temos informações suficientes para entender o papel das covariáveis utilizadas na previsão da variável alvo. Nesse sentido, utilizaremos os métodos apresentados neste capítulo para auxiliar nossa compreensão destes modelos, discriminando cada uma das séries utilizadas, utilizando o pacote `iml` disponível na linguagem `R`.

5.3.1 Inflação (IPCA - Geral)

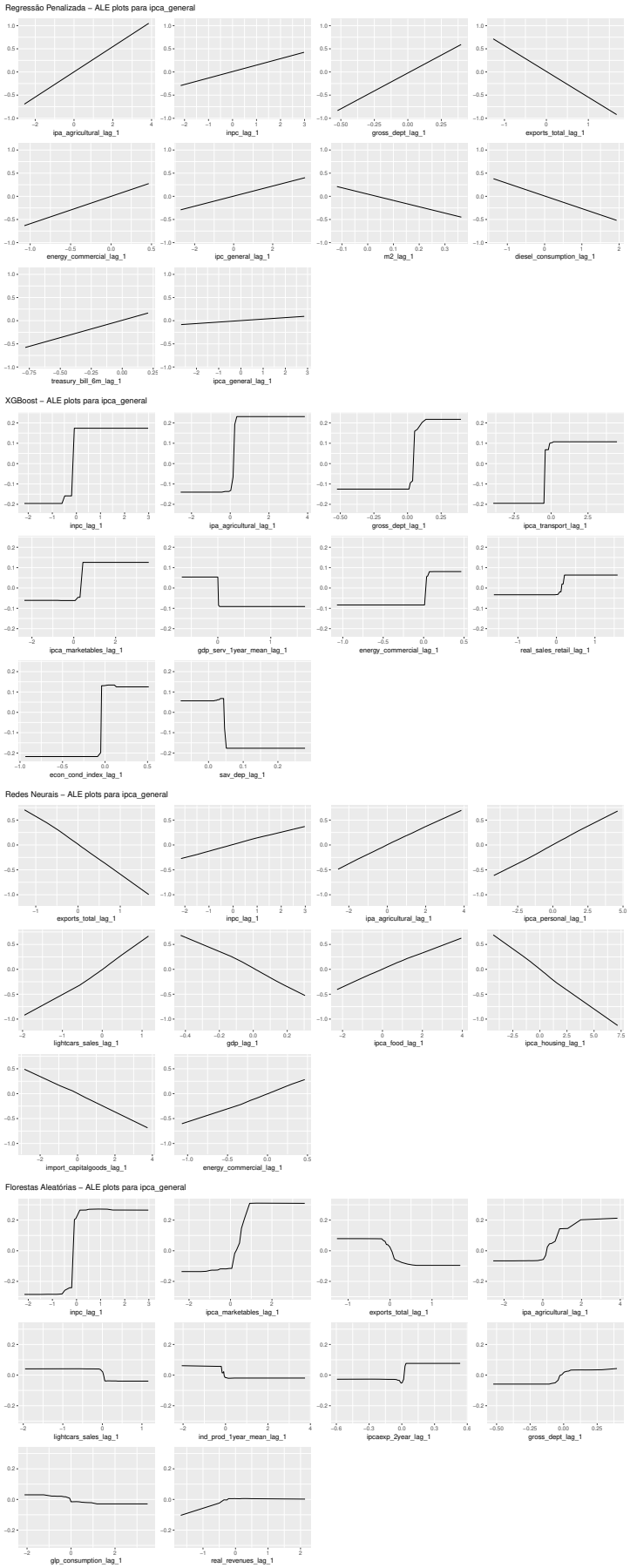
Figura 16 – Importância de Preditores para Série IPCA - Geral



Fonte: Produzido pelo autor.

Nossa primeira aplicação para interpretabilidade dos modelos de aprendizagem de máquina é a importância de variáveis via permutação, que nos mostra quais variáveis foram relevantes para a previsão das séries. Na Figura 16 temos as dez principais variáveis para a previsão da série, em cada um dos modelos de aprendizagem de máquina. Em geral, as variáveis que são referentes aos índices de preços são as mais presentes em todos os modelos, o que faz sentido porque todas estas variáveis de certa forma compõem o índice de preços mais geral, que é o caso do IPCA que prevemos. Variáveis relativas ao consumo, como compra de veículos leves e de combustíveis, também estão são relevantes para a previsão destes modelos. O caso do modelo de florestas aleatórias, modelo que apresentou menor RMSE comparativo, a série do INPC é a mais importante para a previsão do IPCA. Variáveis que apresentam `_Xyear_` são variáveis expectacionais, com expectativas formadas para `X` anos à frente.

Figura 17 – Efeitos Locais Acumulados para IPCA-Geral

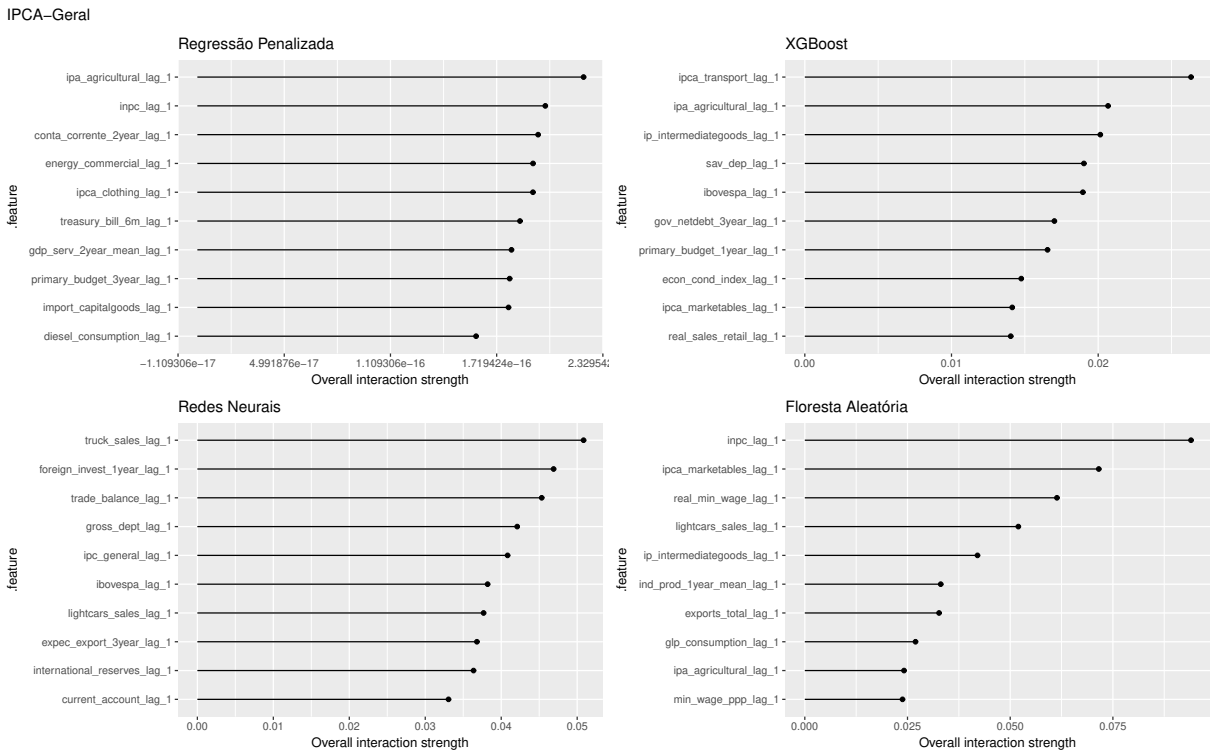


Fonte: Produzido pelo autor.

Nesse sentido, nota-se a relevância da expectativa sobre o próprio IPCA 2 anos à frente para a previsão por meio de florestas aleatórias.

Na Figura 17, temos os gráficos para os efeitos locais acumulados, para as variáveis que mostraram maior importância. O modelo de regressão penalizada mostrou uma relação linear entre os preditores e a previsão da série do IPCA; enquanto redes neurais mostrou relações quase lineares entre as covariáveis e a previsão da variável alvo. No modelo de XGBoost, é possível observar que existem diversos segmentos horizontais, o que significa que, para esses valores dos preditores, há pouco efeito na previsão da variável de interesse. Estes segmentos horizontais também são muito presentes para o modelo de florestas aleatória. Conforme a variável tem menos importância, mais horizontal esta curva fica. Em geral, observam-se curvas com inclinação positiva – o que indica que há um aumento do efeito na previsão quando essas variáveis aumentam, mas que encontra algum platô (segmentos horizontais) para alguns modelos. Tanto regressão penalizada quanto floresta aleatória indicaram uma relação negativa entre exportações e a previsão da série do IPCA, indicando que valores menores para exportações indicam uma previsão de um valor maior para IPCA para estes modelos.

Figura 18 – Interação entre os preditores para a Série do IPCA-Geral

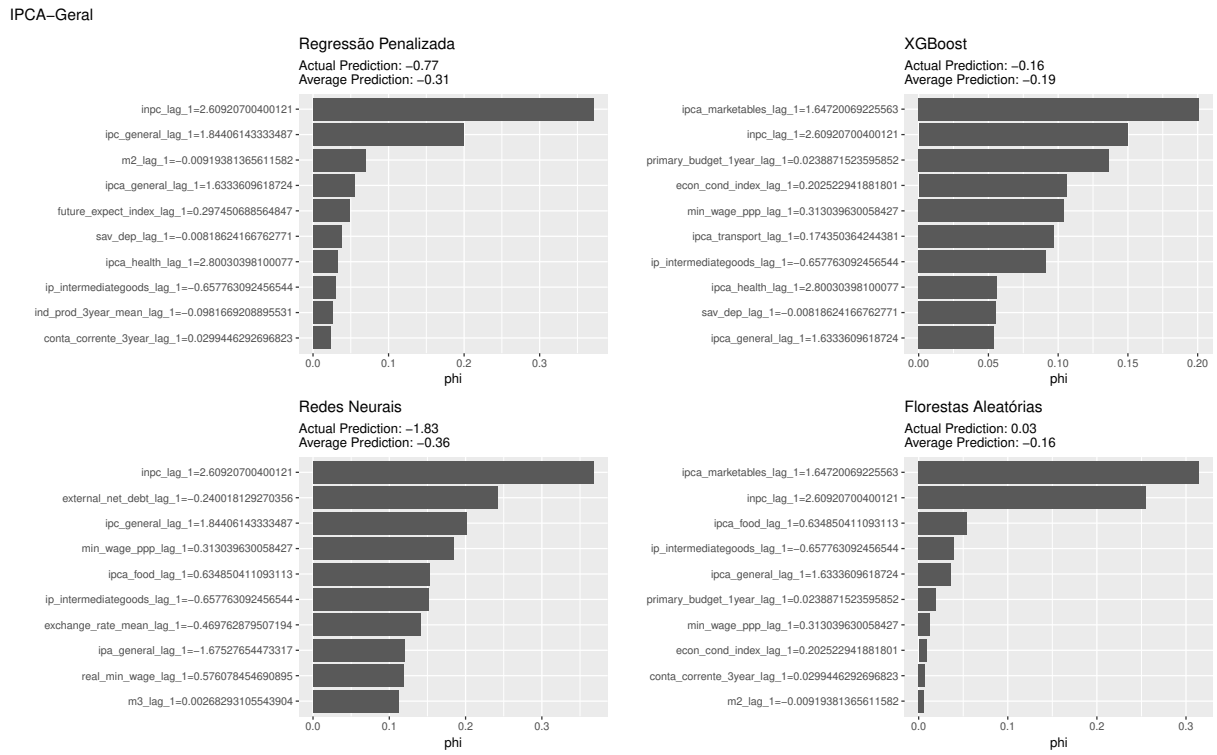


Fonte: Produzido pelo autor.

Já na Figura 18, temos o rank das 10 covariáveis que mais interagem com outras covariáveis. Em todos os modelos, os efeitos de interação das variáveis, representado

pela estatística-H, são muito fracos, sendo menor que 10% a variância explicada por cada preditor para previsão da série. No caso do modelo de regressão penalizada, o efeito da interação é estatisticamente desprezível, indicando que a interação entre as variáveis não é relevante para o modelo na previsão dos modelos.

Figura 19 – Valores de Shapley para a Série do IPCA

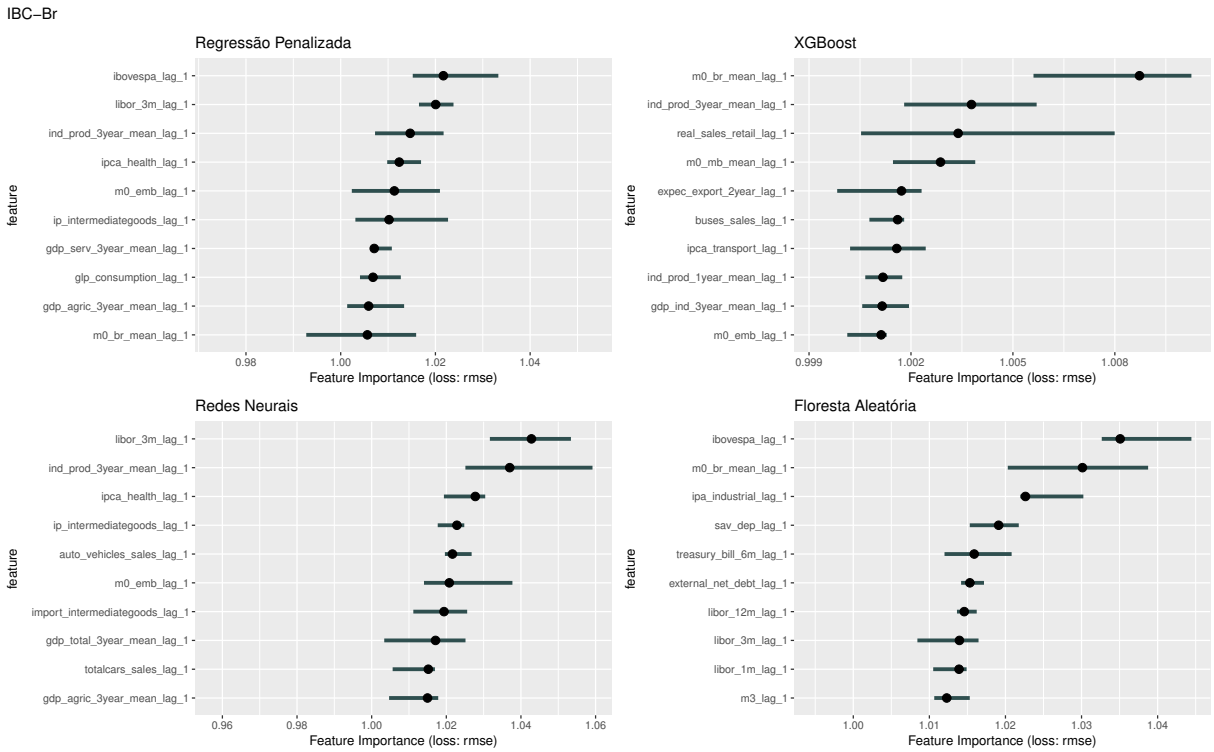


Fonte: Produzido pelo autor.

Nossa última ferramenta para interpretação dos resultados de previsão dos modelos de aprendizagem de máquina é o valor de Shapley, apresentado na Figura 19. O ponto específico para análise é o da primeira previsão, que utilizamos para todas as séries. Dada a diferença entre o valor atual de previsão (*actual prediction*) calculado pelo modelo e a previsão média (*average prediction*), o gráfico nos mostra a contribuição de cada uma das variáveis para esta diferença, mostrando as 10 primeiras variáveis deste rank. A série do INPC apareceu no topo como maior contribuidora para a diferença na previsão da série IPCA para todos os modelos, sendo a maior contribuidora para previsão dos modelos de regressão penalizada e redes neurais. Em geral observam-se a forte presença das variáveis relativas aos índices de preços. No caso de floresta aleatória, que apresentou o melhor desempenho comparativo, as série do IPCA - preços de mercado - negociáveis e INPC - geral foram as maiores contribuidoras, muito destoantes das demais.

5.3.2 Taxa de Crescimento do Produto (IBC-Br)

Figura 20 – Importância de Preditores para Série IBC-Br

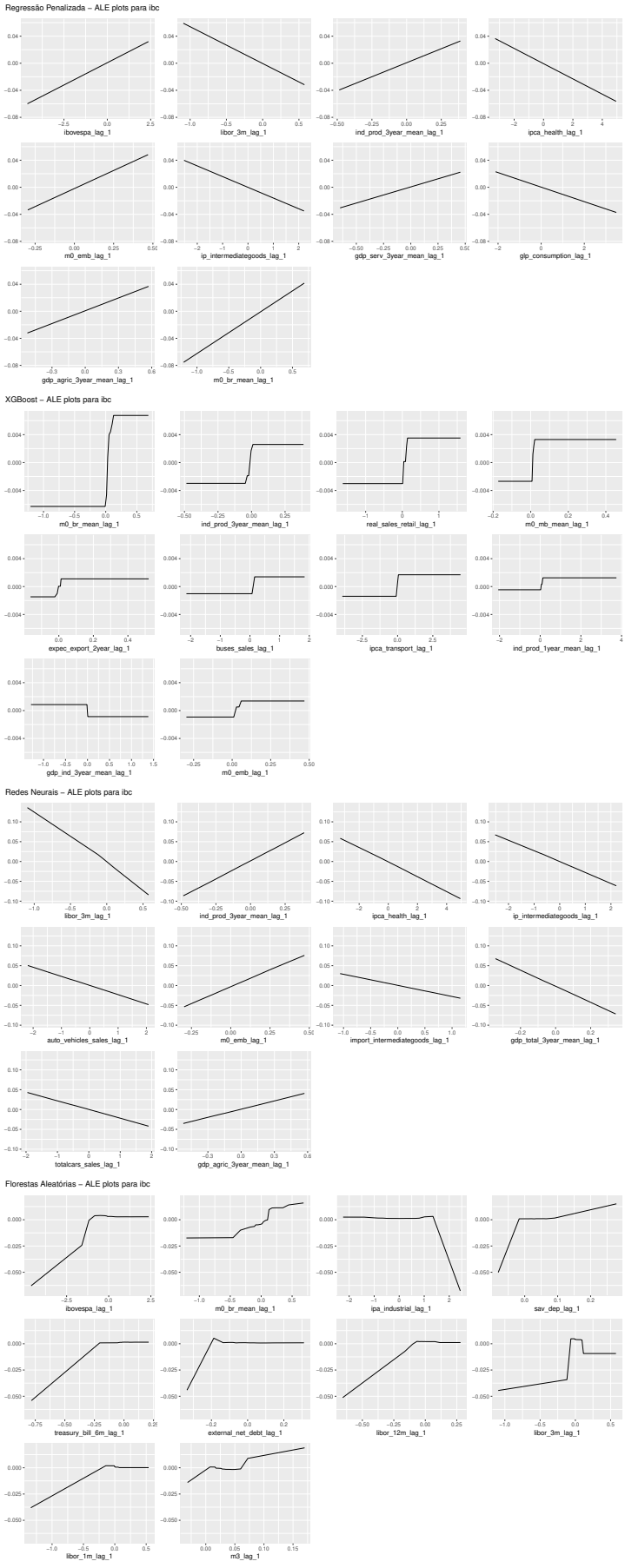


Fonte: Produzido pelo autor.

Na Figura 20 temos os gráficos de importância dos preditores para os modelos de aprendizagem de máquina. Em geral os preditores com maior importância são aqueles associados aos setores financeiros e monetário, além da presença de índices de preços. O modelo de redes neurais, que obteve o melhor desempenho comparativo com a estatística RMSE, surpreende ao mostrar a série da taxa de juros londrina LIBOR - 3 meses como maior contribuidora na previsão do IBC-Br. Ela é seguida pela expectativa de produção industrial para três anos, que teve importância pouco menor que a LIBOR, que também aparece como importante nos modelos de regressão penalizada e floresta aleatória. Regressão penalizada e florestas aleatórias indicam uma maior importância do índice do Bovespa.

Nos efeitos locais acumulados presentes na Figura 21, percebemos que o modelo de regressão penalizada continua diagnosticando relações lineares entre os preditores e as previsões da série, bem como o modelo de redes neurais. A série da taxa de juros LIBOR mostrou uma efeito negativo para previsão do IBC-Br nos modelos de regressão penalizada e de redes neurais. O contrário ocorre no modelo de florestas aleatórias, que apontou um efeito positivo. Florestas aleatórias também mostra um efeito positivo na previsão com o índice Bovespa, que faz certo sentido porque ao passo que este índice fecha em alta, esperamos períodos de maior prosperidade

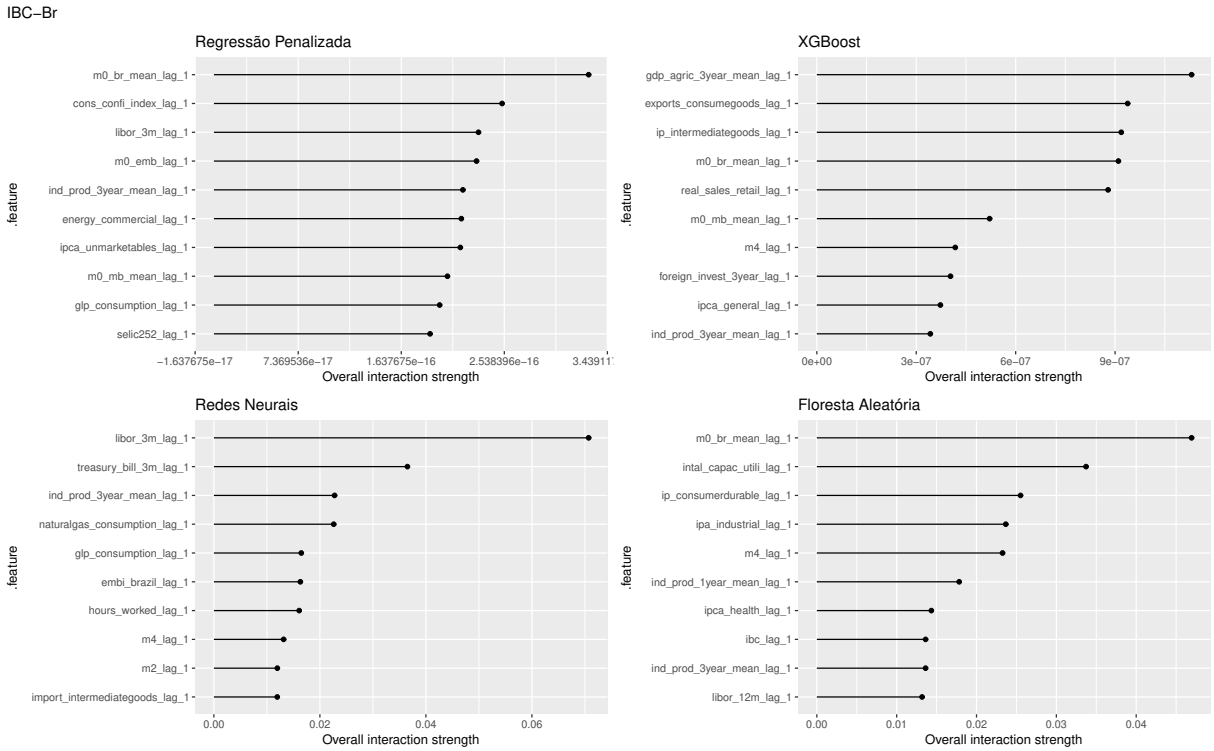
Figura 21 – Efeitos Locais Acumulados para IBC-Br



Fonte: Produzido pelo autor.

econômica. Ainda que nossa interpretação da relação negativa ou positiva possa se relacionar com a intuição econômica por trás desses resultados, alguns destes efeitos chamam atenção por fugir da lógica convencional do que se esperaria, como é o caso do efeito negativo das variáveis de vendas de veículos na previsão desta série que é proxy da taxa de crescimento do produto, no caso do modelo de redes neurais. Em geral esperamos que valores maiores para vendas de carros implicariam em maiores valores para o IBC-Br, mas o modelo revela um efeito negativo.

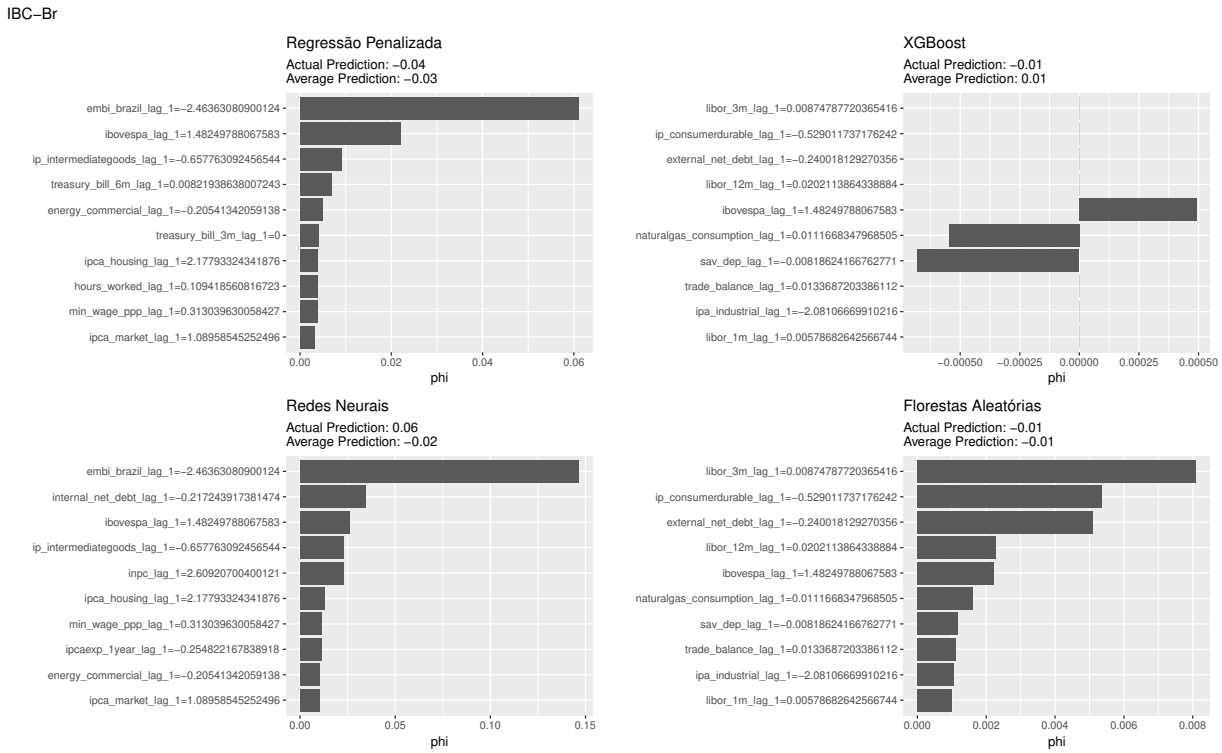
Figura 22 – Interação entre os preditores para a Série do IBC-Br



Fonte: Produzido pelo autor.

Na Figura 22, notamos que a interação entre as variáveis é muito fraca (ou desprezível no caso da regressão penalizada e XGBoost), expressas pela estatística-H sempre abaixo de 10%. Ainda assim, o modelo de redes neurais tem a taxa LIBOR como variável que mais interage dentro da estrutura do modelo para previsão da série.

Figura 23 – Valores de Shapley para a Série do IBC-Br

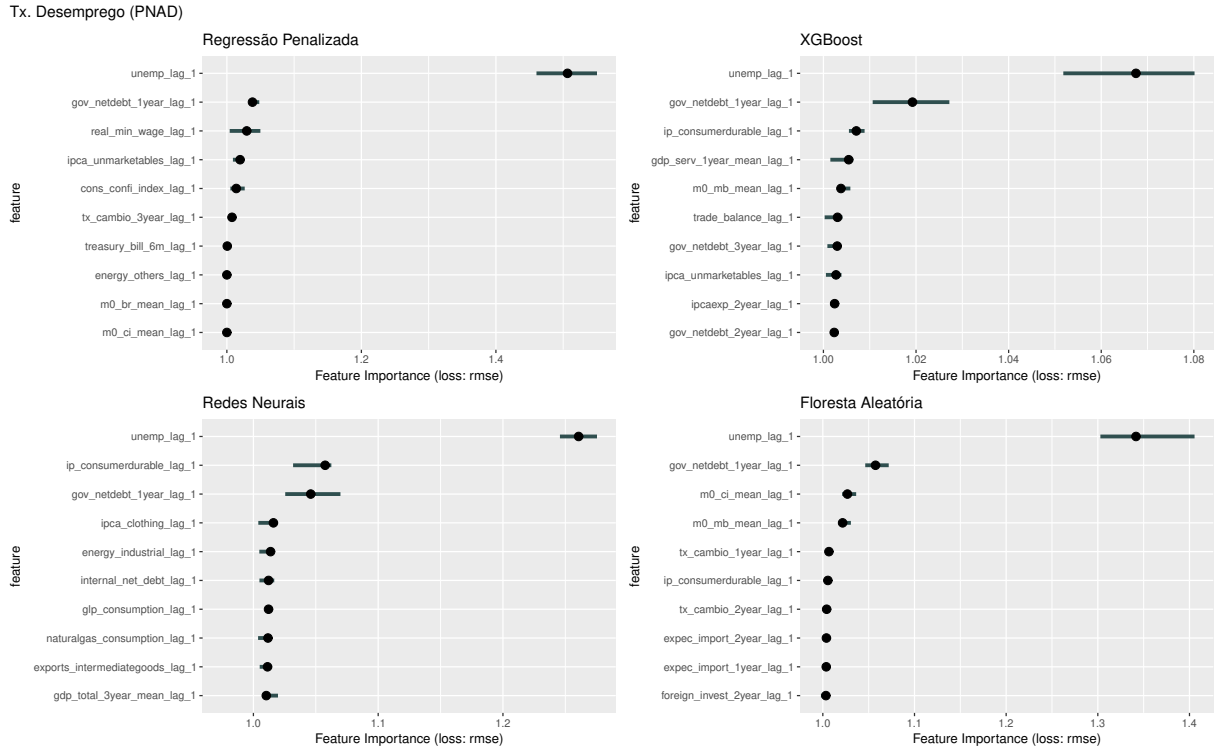


Fonte: Produzido pelo autor.

Por último, para o valor de Shapley, temos um contribuição bastante significativa do índice Embi-Brasil nos modelos de regressão penalizada e redes neurais. A taxa de juros LIBOR não aparece na análise destes dois modelos, mas surge para o modelo de floresta aleatórias como maior contribuinte. Os resultados do modelo XGBoost chamam atenção porque, para o ponto em questão, somente a série do IBOVESPA apresentou uma contribuição positiva na previsão do IBC-Br, ao passo que a série de consumo de gás natural e depósitos de poupança tiveram uma contribuição negativa para a previsão do IBC-Br.

5.3.3 Taxa de Desemprego (PNAD)

Figura 24 – Importância de Preditores para Taxa de Desemprego (PNAD)

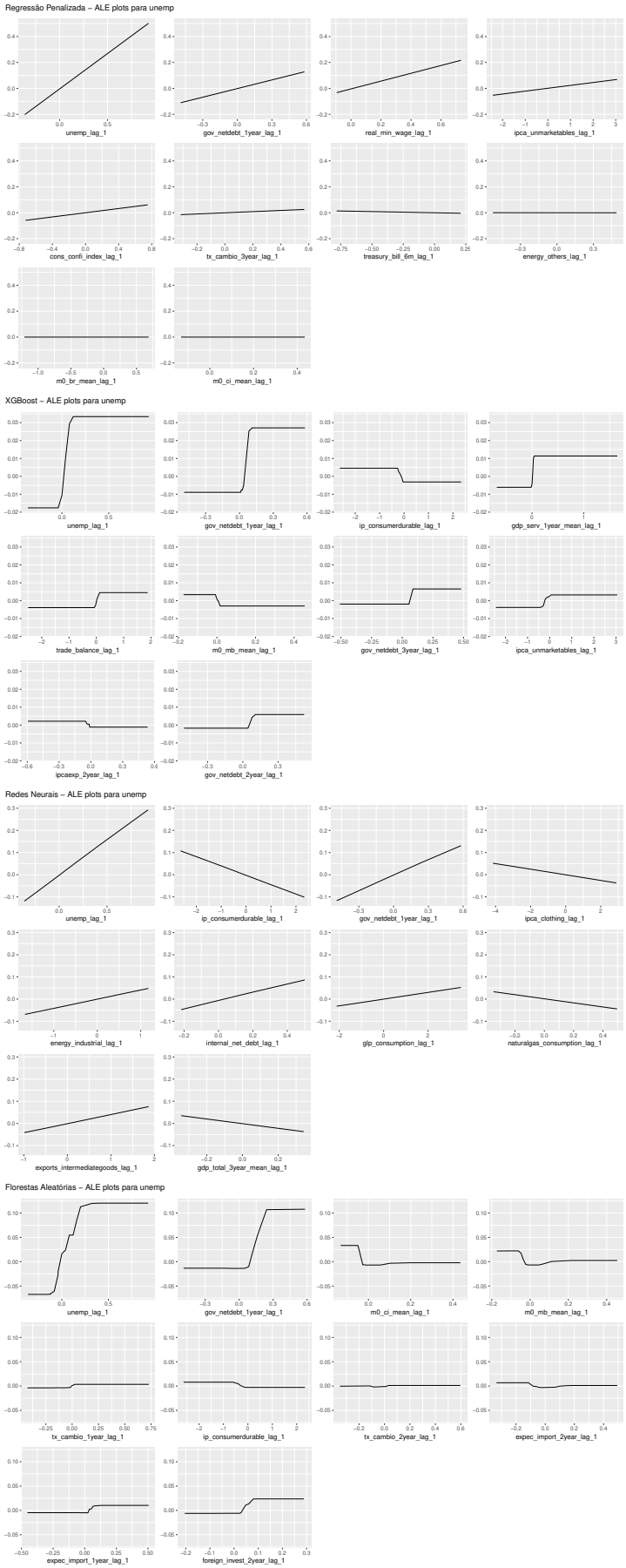


Fonte: Produzido pelo autor.

Para a taxa de desemprego, a importância dos preditores na Figura 24, nos mostra um papel bastante importante para o lag da própria série, em todos os modelos analisados. Este resultado é interessante porque na análise comparativa com os métodos tradicionais, o modelo ARMA apresentou os menores valores para RMSE e MAE, o que evidencia essa relação temporal também nos modelos de aprendizagem de máquina. Observa-se a presença da variável da expectativa um ano à frente da dívida líquida do governo como importante na previsão da série do desemprego.

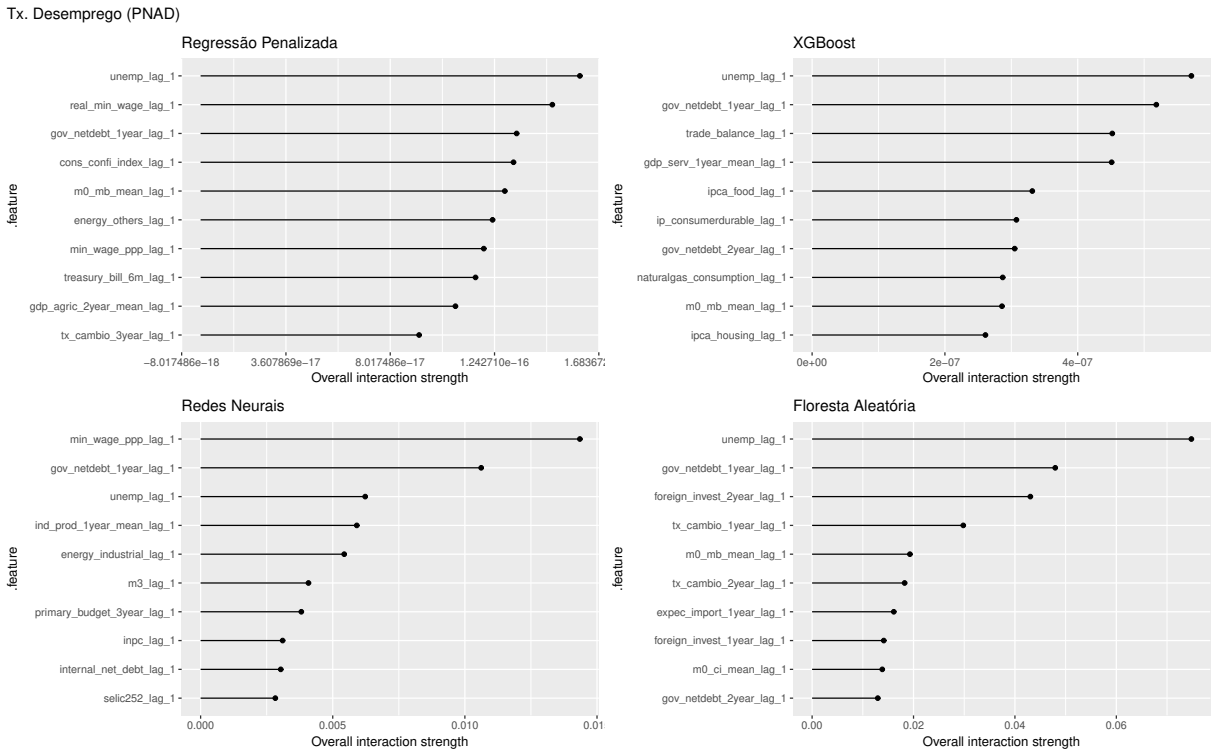
Na análise dos efeitos locais acumulados da Figura 26 para as variáveis mais importantes de cada modelo, é possível observar que de fato a taxa de desemprego defasada tem maior efeito para a previsão da série, para todos os modelos analisados. A variável de expectativa da dívida líquida do governo para um ano também possui um efeito significativo para previsão da série de desemprego, tendo um efeito positivo na previsão em todos os modelos analisados. Observam-se relações positivas entre os efeitos das variáveis e a previsão da série, com algumas exceções, como é o caso do modelo de redes neurais com as séries de venda total de carros e da produção industrial de bens de consumo duráveis.

Figura 26 – Efeitos Locais Acumulados para Taxa de Desemprego (PNAD)



Fonte: Produzido pelo autor.

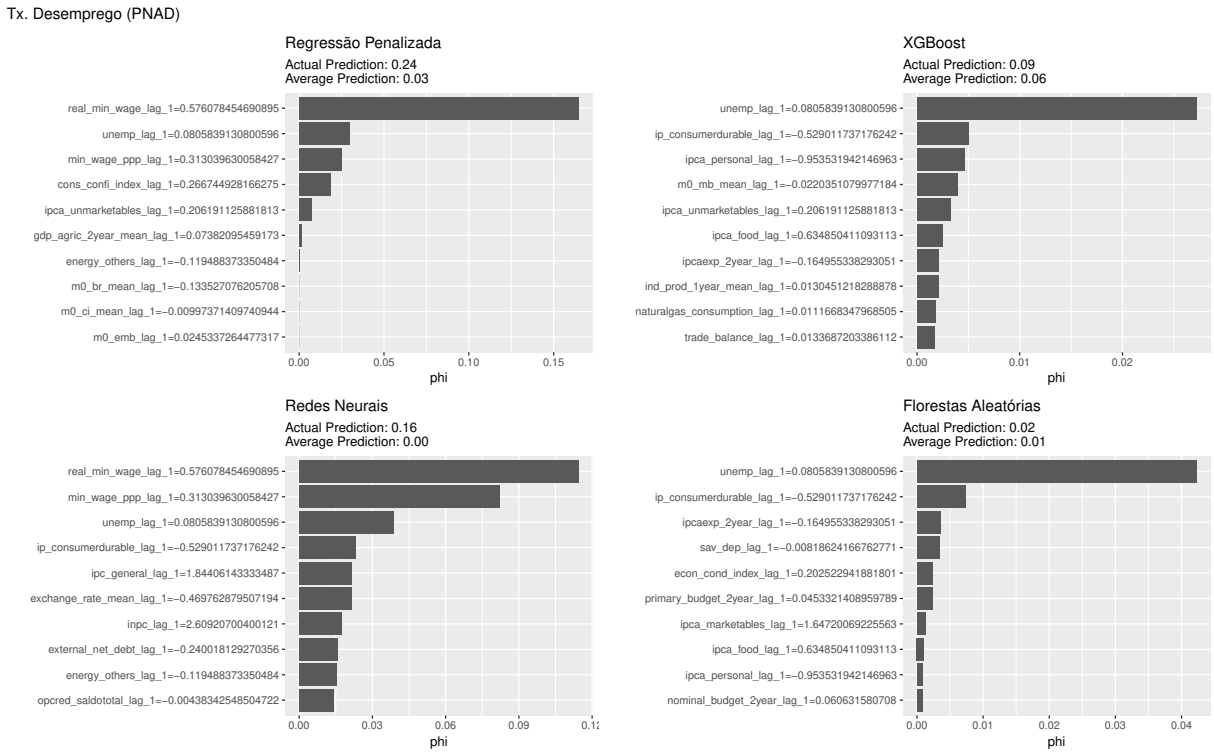
Figura 25 – Interação entre os preditores para a Taxa de Desemprego (PNAD)



Fonte: Produzido pelo autor.

A interação entre preditores na Figura 25, não temos uma uma estatística-H que mostra uma relevância de pelo menos 10% de suas variâncias explicadas pela interação com outros preditores, para nenhum dos modelos.

Figura 27 – Valores de Shapley para a Taxa de Desemprego (PNAD)

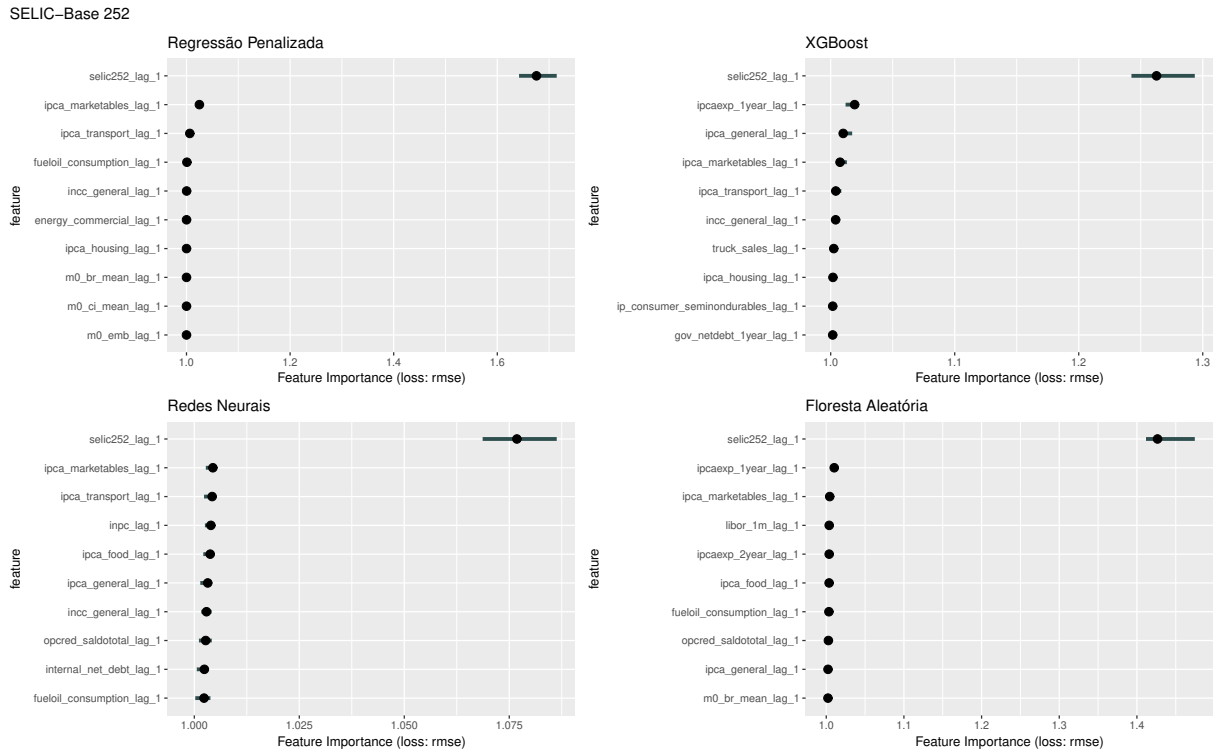


Fonte: Produzido pelo autor.

Nos valores de Shapley, a Figura 27 nos mostra que o modelo XGBoost e Florestas Aleatórias teve resultados parecidos ao atribuir maior contribuição da taxa de desemprego defasada. Esta série aparece também em regressão penalizada e em redes neurais, mas estes modelos foi identificado maior contribuição do contribuição do salário mínimo real, além do salário mínimo pela paridade do poder de compra.

5.3.4 Taxa de Juros Básica (SELIC-Base 252)

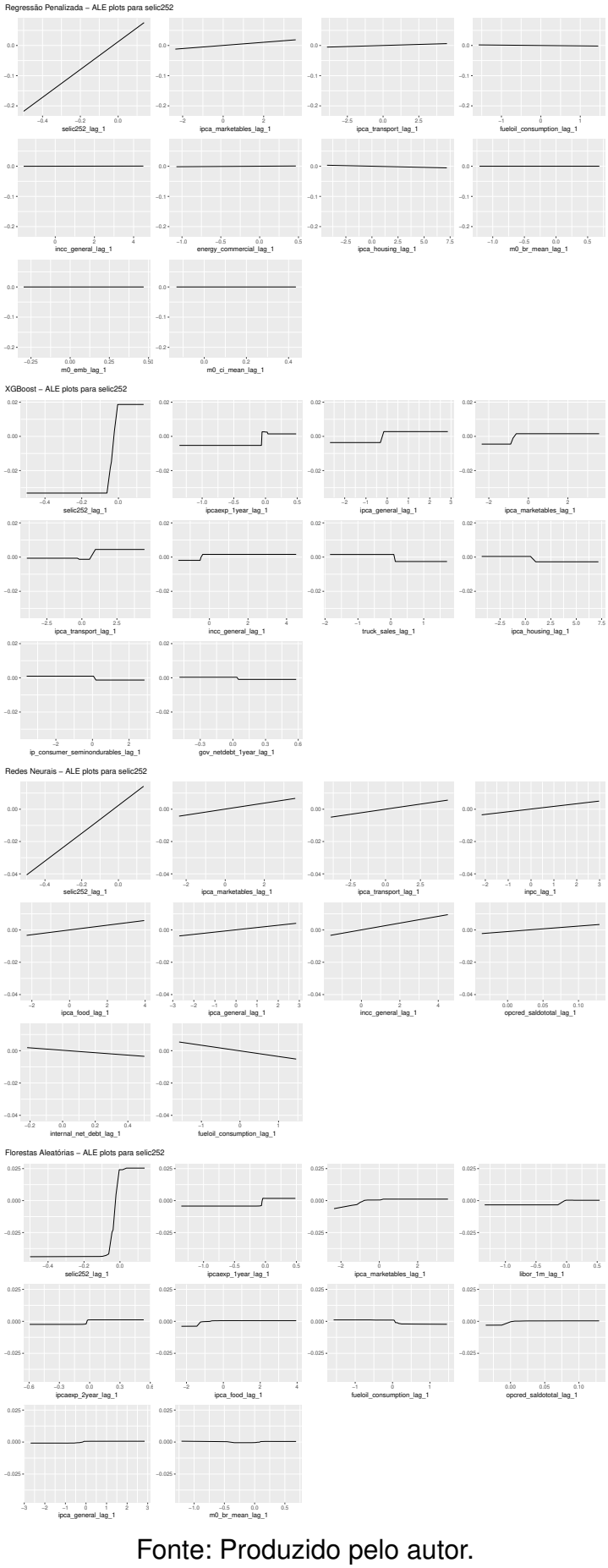
Figura 28 – Importância de Preditores para SELIC-Base 252



Fonte: Produzido pelo autor.

Nossa última série analisada por métodos agnósticos de interpretação de modelos de aprendizagem de máquina mostrou unanimidade na importância de preditores. Na Figura 28 é possível notarmos que a série SELIC defasada é a mais importante para a previsão da própria série. Em geral também observa-se séries relativas aos níveis de preços defasados (especialmente a expectativa do IPCA um ano a frente nos modelos de floresta aleatória e XGBoost), ainda que comparativamente a série SELIC defasadas, suas importâncias sejam quase insignificantes. Essa importância, mesmo que singela, das variáveis relativas a inflação nos chama atenção porque nos modelos teóricos de regra de política monetária em regimes de metas de inflação, temos esta relação para a fixação da taxa de juros básica da economia.

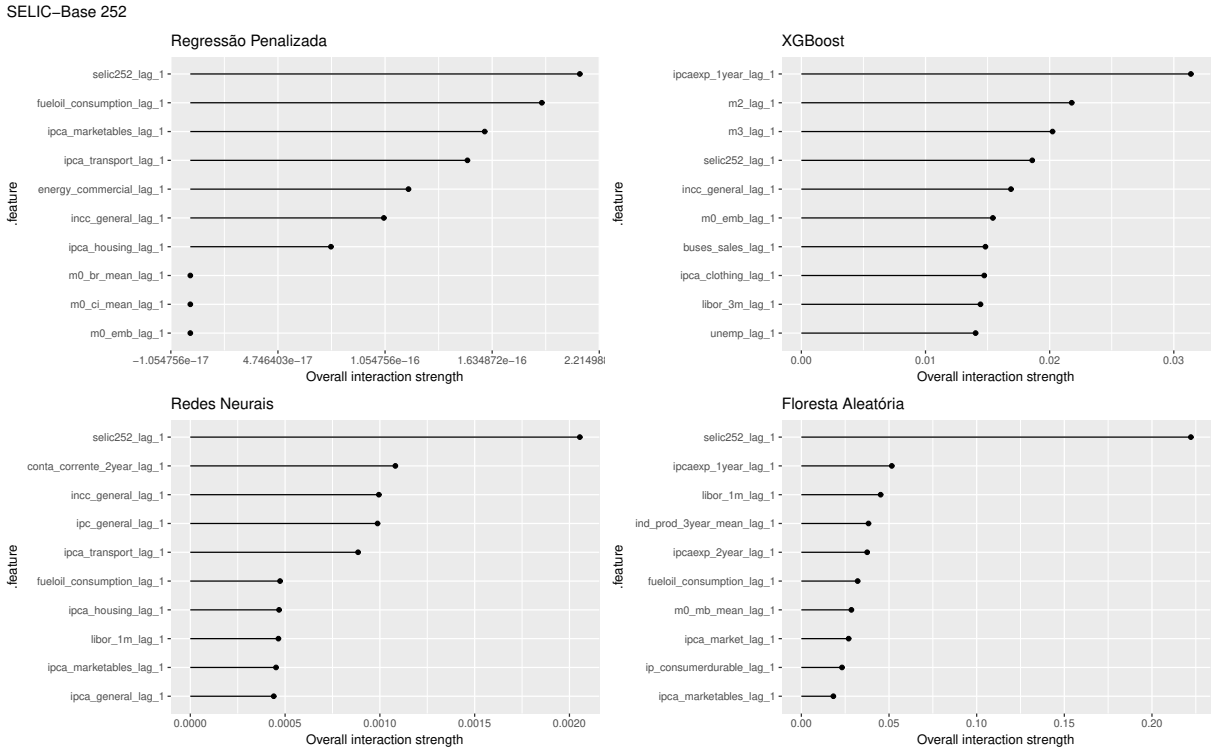
Figura 29 – Efeitos Locais Acumulados para SELIC-Base 252



Fonte: Produzido pelo autor.

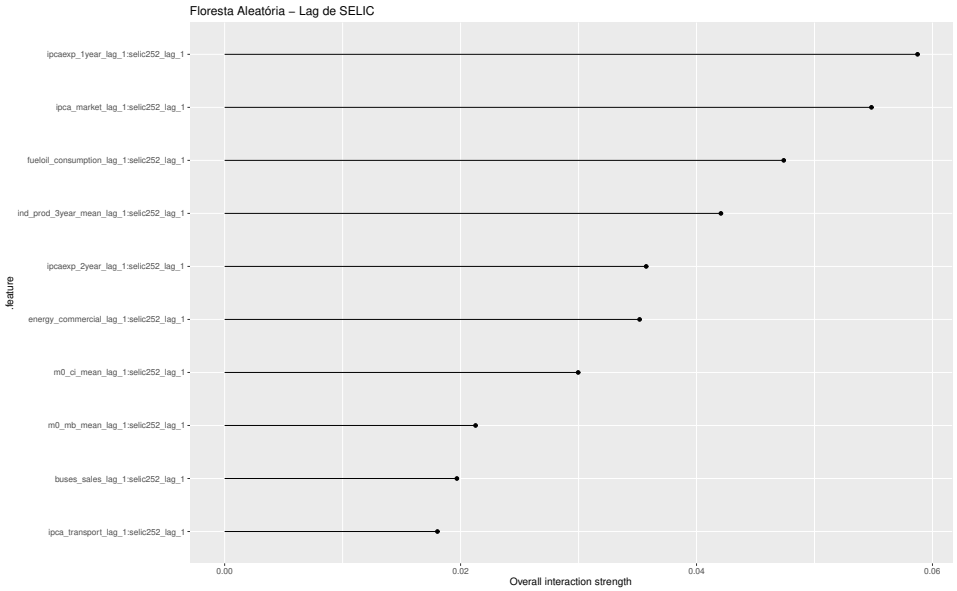
Na análise dos efeitos locais acumulados, o resultado é semelhante ao que observamos na importância das variáveis da figura anterior. A Figura 29 mostra que a série SELIC defasada é a que apresenta maior efeito na previsão da própria série, sendo os outros preditores praticamente irrelevantes para previsão da mesma, notadamente pelas curvas mais planas. Somente no caso de redes neurais notamos um efeito fraco das variáveis relativas aos índices de preços com uma relação linear positiva para a previsão da taxa de juros. Nota-se também uma efeito positivo no caso do modelo de florestas aleatórias, da expectativa do IPCA para previsão da SELIC, ainda que essa janela seja muito pequena.

Figura 30 – Interação entre os preditores para a SELIC-Base 252



Fonte: Produzido pelo autor.

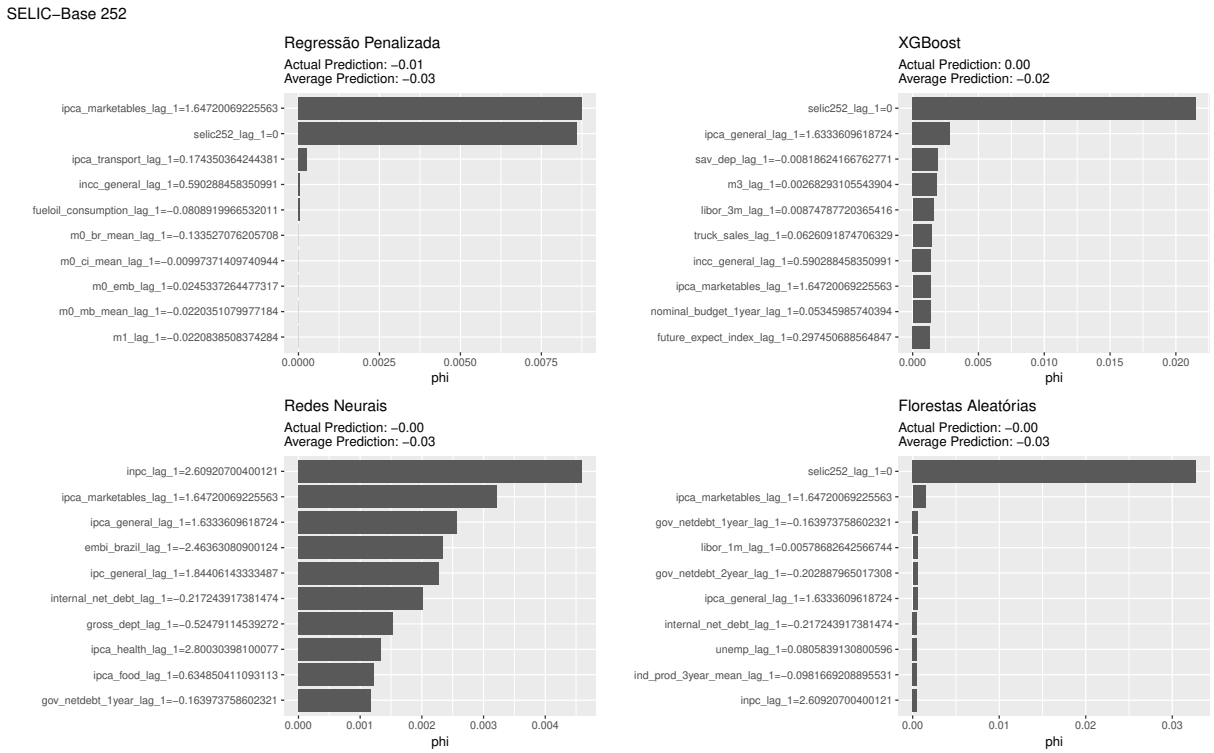
Figura 31 – Interação entre os preditores e o lag da SELIC-Base 252 no modelo de Floresta Aleatória



Fonte: Produzido pelo autor.

A interação entre os preditores na Figura 30 mais uma vez mostrou-se muito fraca para a maioria dos modelos. Somente para o modelo de floresta aleatória, temos a variável SELIC defasada tendo pouco mais de 20% de sua variância explicada pela interação com outros preditores. Esta variável aparece em todos os modelos, mais uma vez. Na Figura 31, é possível notar que a maior interação para o lag da SELIC é a expectativa do IPCA um ano à frente, seguido do IPCA - preços de mercado e consumo de óleos combustíveis.

Figura 32 – Valores de Shapley para SELIC-Base 252



Fonte: Produzido pelo autor.

Por último, o valor de Shapley na Figura 32, com exceção de redes neurais, teve a SELIC defasada como maior contribuidor para a previsão da série. No caso da regressão penalizada, a SELIC defasada só perde para a série do IPCA - preços de mercado - negociáveis. Em geral as maiores contribuidoras são variáveis relacionadas com índices de preços. No caso de XGBoost e florestas aleatórias, a contribuição do lag da SELIC é significadamente maior que outro, fazendo sentido com a importância desta variável nas técnicas anteriores e também chamando atenção pelo fato de que o modelo ARMA identificou um modelo AR(1) com coeficiente muito próximo de 1.

6 CONCLUSÕES

Esta dissertação buscou comparar o desempenho preditivo de modelos de aprendizagem de máquina – regressão penalizada, redes neurais, XGBoost e florestas aleatórias – com modelos clássicos de previsão de séries macroeconômicas – ARMA, VAR e FAVAR. Dada a crítica sobre a dificuldade de interpretar os resultados dos modelos de aprendizagem de máquina, buscamos métodos agnósticos de interpretação destes modelos – importância do recurso de permutação, efeitos locais acumulados, interação de preditores e valores de Shapley – permitindo uma maior compreensão dos resultados obtidos pelas máquinas.

Utilizando dados da economia brasileira no período de 02/2003 até 04/2021, com um conjunto de 126 variáveis, nos propomos a prever o período compreendido entre 11/2015 até 04/2021 para as séries do IPCA - geral, que é uma proxy da inflação brasileira; a série do IBC-Br, que é uma proxy para a taxa de crescimento do produto; a série da PNAD para a taxa de desemprego e a série SELIC acumulada base 252, taxa de juros básica da economia brasileira. Nossos resultados mostraram um desempenho superior para os modelos de aprendizagem de máquina, em três de quatro séries – somente para a série da taxa de desemprego o modelo ARMA teve um desempenho superior. Na média, para as quatro séries, florestas aleatórias teve os menores valores para RMSE, ao passo que regressão penalizada teve o menor valor para MAE. Estes resultados se alinham com aqueles encontrados na literatura, que mostram um resultado superior dos modelos de aprendizagem de máquina na previsão de séries temporais comparativamente aos métodos tradicionais de previsão. Na série do IPCA, o modelo de floresta aleatória foi o melhor modelo, apresentando os menores valores para a RMSE, ao passo que o MAE foi menor no caso de regressão penalizada. No caso da série do IBC, o modelo de redes neurais foi ligeiramente melhor do que os demais, ainda que o MAE tenha sido melhor para o FAVAR. A taxa de desemprego acabou mostrando um melhor desempenho do modelo ARMA, seguido pelo modelo FAVAR, que mostraram valores RMSE e MAE relativamente menores aos demais modelos. Quando analisamos apenas os modelos de aprendizagem de máquina, florestas aleatórias também teve um resultado melhor para esta série. Para a SELIC, apesar do melhor desempenho da regressão penalizada, todos os modelos, com exceção de VAR e redes neurais, tiveram um desempenho bastante semelhante em termos de RMSE e MAE. Os resultados da metodologia de Model Confidence Set também mostram os modelos de aprendizagem de máquina entre o conjunto de melhores para a previsão das séries – com exceção da taxa de desemprego. Florestas aleatórias e regressão penalizada aparecem sempre disputando o primeiro lugar, ainda que seus resultados sejam estatisticamente iguais quando comparamos os p-valores MCS dos modelos.

Os métodos agnósticos utilizados para interpretação dos resultados nos forne-

ceu importantes pistas para entendermos os resultados dos modelos. A análise da interação entre os preditores não mostrou significância na previsão de nenhuma das séries para nenhum dos modelos, com exceção do modelo de floresta aleatória para a SELIC, que indicou a interação do lag da própria série como relevante para a previsão deste modelo – mostrando que a interação deste variável com a expectativa do IPCA um ano à frente foi importante para a previsão da própria série por meio do modelo. No caso da série do IPCA, o fato de que incluímos séries que compunham o próprio IPCA - geral propiciaram uma previsão melhor para os modelos de aprendizagem de máquina, evidenciado pela análise da importância de preditores. Para florestas aleatórias, algoritmo que teve o menor RMSE, a série INPC e IPCA - preços de mercado - negociáveis, foram as mais relevantes para a previsão, seguido do total de exportação, IPA - agricultura e a venda de carros leves. A análise dos efeitos acumulados mostra em geral uma relação linear e positiva para o IPCA - geral e estas variáveis, exceto para exportação e venda de carros leves, de acordo com o modelo de florestas aleatórias. Na análise dos valores de Shapley, a contribuição maior também é destas variáveis, mas em outra ordem de importância inversa, tendo primeiro IPCA - preços de mercado - negociáveis.

No caso da série do IBC, redes neurais teve o melhor desempenho preditivo, mostraram as séries da taxa de juros londrina LIBOR (3 meses) e a expectativa de produção industrial para três anos, como mais relevantes para a previsão da taxa de crescimento do produto. Estas variáveis também aparecem como importantes para os outros modelos, especialmente a LIBOR. Redes neurais mostraram uma relação positiva e linear por meio de efeito para a expectativa de produção industrial, ao passo que para a LIBOR esta relação mostrou-se negativa para a previsão do IBC, denotadas pelo gráfico dos efeitos locais acumulados. Quando avaliamos os valores de Shapley, em ambos os modelos a variável EMBI-Brasil foi a que mais contribuiu para a previsão da série, seguida de variáveis financeiras e relativas a divida pública, financeiras e aos índices de preços da economia brasileira. A presença da série EMBI-Brasil como principal contribuinte na previsão desta série é interessante porque sugere a identificação um papel relevante sobre as expectativas dos agentes para a previsão da taxa de crescimento do produto no modelo de redes neurais.

Para a previsão da série da taxa de desemprego da PNAD, a análise de importância de preditores nos mostra que a variável mais importante para todos os modelos foi o lag dessa própria variável prevista. Esse resultado chama a atenção porque na análise dos valores de RMSE e MAE, o melhor desempenho foi dos modelos autorregressivos – ARMA e FAVAR – em que identificamos um modelo ARMA(1,2) para esta série. Na comparação entre os modelos de aprendizagem de máquina, florestas aleatórias mais uma vez leva a melhor, com menores RMSE e MAE. No caso dos valores de Shapley, esta variável só aparece como maior contribuinte no caso dos

modelos XGBoost e Florestas Aleatórias, seguida de variáveis de produção industrial de consumo semi e não duráveis e a expectativa do IPCA em 2 anos; em regressão penalizada e redes neurais identificamos o salário mínimo real e salário mínimo pela paridade do poder de compra como maiores contribuintes, o que faz sentido porque são variáveis que fazem parte do mercado de trabalho.

A série da SELIC também mostrou resultados interessantes. Para todos os modelos de aprendizagem de máquina, a importância da série defasada foi de longe a mais importante para o resultado de previsão dos modelos. Esse resultado chama a atenção porque para ARMA identificamos um processo autorregressivo de ordem um. Na análise dos efeitos acumulados, nota-se uma relação linear e positiva entre a previsão e o lag da variável, sendo em todos os modelos, outras variáveis praticamente insignificantes para previsão. Em valores de Shapley, somente em redes neurais e regressão penalizada não vemos uma maior contribuição para a previsão: no primeiro, variáveis relativas ao índice de preços são mais relevantes, ao passo que para o segundo, temos o IPCA - preços de mercado - negociáveis contribuindo um pouco melhor que o lag da SELIC. Em geral variáveis relativas aos preços são grandes contribuintes para a previsão, aparecendo em todos os modelos.

Sob a perspectiva de Blanchard (2018) ao classificar diferentes modelos para diferentes fins, algoritmos de aprendizagem são bons modelos para previsão: para o autor, referência em macroeconomia, um bom modelo de previsão é aquele que possui a previsão mais precisa, não necessariamente aquele que está mais de acordo com a teoria macroeconômica – na verdade, ele afirma que se a teoria não for relevante para a previsão, então ela deve ser descartada. Além das referências internacionais, nosso trabalho também mostrou um resultado bastante empolgante, mostrando um caminho promissor de aplicações de aprendizagem de máquina. As técnicas de interpretação também trouxeram informações relevantes para entendermos a importância do chamado *big data* para aplicações em macroeconomia – nossa base de dados, apesar de relativamente pequena em termos temporais, possui uma quantidade razoável de variáveis – permitindo alternativas interessantes para os economistas. Com os resultados preliminares desta pesquisa, não é possível enxergar esta classe de modelos como substitutos das técnicas e teorias convencionais, principalmente se tratando de políticas econômicas. A própria avaliação para a série do desemprego mostrou um resultado mais preciso do modelo ARMA, talvez o mais simples de todos os modelos apresentados nesta dissertação. Modelos de aprendizagem de máquina são, como vemos em Fernández-Villaverde (2021), ferramentas complementares para o arsenal do economista, permitindo que expandamos cada vez mais a fronteira do conhecimento em macroeconomia.

A revisão da literatura mostrou uma quantidade bastante considerável de algoritmos de aprendizagem de máquina que potencialmente podem ser aplicados em

macroeconomia, mas que devido o tempo curto de um mestrado, não foram possíveis de serem melhores explorados. Acreditamos que para trabalhos futuros, utilizando dados da economia brasileira, há espaço de exploração de outros modelos que eventualmente possam ter um desempenho melhor para a previsão das séries aqui estudadas. Modelos híbridos, como mostrados em Nosratabadi *et al.* (2020), também são opções viáveis para fins preditivos que poderiam ser aplicados em dados macroeconômicos. Outro caminho promissor seria aliar o poder de aprendizagem de máquinas com as ferramentas usuais dos macroeconomistas. Como afirma Fernández-Villaverde e Guerrón-Quintana (2021), o futuro de estimação dos modelos DSGE, que são modelos robustos e estruturais muito utilizados para análise macroeconômica, podem se beneficiar com a estimação ao utilizar técnicas de aprendizagem de máquina – o que poderia ser útil também se aplicarmos para os dados da economia brasileira – propiciando eventualmente resultados melhores para estes modelos. O uso de métodos agnósticos para interpretação também trazem importantes informações que nos permitem refletir sobre a teoria macroeconômica e como a aplicamos em metodologias de trabalhos empíricos para fins de previsão. Entender como as máquinas obtiveram determinadas previsões propiciou um melhor aproveitamento dos resultados obtidos nesta pesquisa e naturalmente é uma ferramenta viável para futuras aplicações na literatura macroeconométrica.

REFERÊNCIAS

ALMOSOVA, Anna. **Essays on monetary macroeconomics**. 2019. Tese (Doutorado) – Humboldt-Universität zu Berlin.

ATHEY, Susan; IMBENS, Guido W. Machine learning methods that economists should know about. **Annual Review of Economics**, Annual Reviews, v. 11, p. 685–725, 2019.

BABENKO, VITALINA; PANCHYSHYN, ANDRIY; ZOMCHAK, LARYSA; NEHREY, MARYNA; ARTYM-DROHOMYRETSKA, ZORIANA; LAHOTSKYI, TARAS. Classical machine learning methods in economics research: Macro and micro level example. **WSEAS Transactions on Business and Economics**, v. 18, p. 209–217, 2021.

BABII, Andrii; GHYSELS, Eric; STRIAUKAS, Jonas. Machine learning time series regressions with an application to nowcasting. **Journal of Business & Economic Statistics**, Taylor & Francis, p. 1–23, 2021.

BERGMEIR, Christoph; HYNDMAN, Rob J; KOO, Bonsoo. A note on the validity of cross-validation for evaluating autoregressive time series prediction. **Computational Statistics & Data Analysis**, Elsevier, v. 120, p. 70–83, 2018.

BERGSTRA, James; BENGIO, Yoshua. Random search for hyper-parameter optimization. **Journal of machine learning research**, v. 13, n. 2, 2012.

BERNANKE, Ben S; BOIVIN, Jean; ELIASZ, Piotr. Measuring the effects of monetary policy: a factor-augmented vector autoregressive (FAVAR) approach. **The Quarterly journal of economics**, MIT Press, v. 120, n. 1, p. 387–422, 2005.

BLANCHARD, Olivier. On the future of macroeconomic models. **Oxford Review of Economic Policy**, Oxford University Press UK, v. 34, n. 1-2, p. 43–54, 2018.

BLUWSTEIN, Kristina; BUCKMANN, Marcus; JOSEPH, Andreas; KAPADIA, Sujit; SIMSEK, Özgür. Credit growth, the yield curve and financial crisis prediction: evidence from a machine learning approach. ECB Working Paper, 2021.

BOX, George EP; JENKINS, Gwilym M; REINSEL, Gregory C. **Time series analysis, forecasting and control. Englewood Clifs**. 3. ed. Englewood Cliffs, NJ: Prentice Hall, 1994.

BREIMAN, Leo. Random forests. **Machine learning**, Springer, v. 45, n. 1, p. 5–32, 2001.

BUCKMANN, Marcus; JOSEPH, Andreas; ROBERTSON, Helena. Opening the Black Box: Machine Learning Interpretability and Inference Tools with an Application to

Economic Forecasting. *In*: DATA Science for Economics and Finance. [*S.l.*]: Springer, Cham, 2021. p. 43–63.

CHAIBOONSRI, Chukiat; WANNAPAN, Satawat. Big data and machine learning for economic cycle prediction: application of Thailand's economy. *In*: SPRINGER. INTERNATIONAL Symposium on Integrated Uncertainty in Knowledge Modelling and Decision Making. [*S.l.*: *s.n.*], 2019. p. 347–359.

CHAIBOONSRI, Chukiat; WANNAPAN, Satawat. Nowcasting and Forecasting for Thailand's Macroeconomic Cycles Using Machine Learning Algorithms. *In*: SPRINGER. INTERNATIONAL Symposium on Integrated Uncertainty in Knowledge Modelling and Decision Making. [*S.l.*: *s.n.*], 2020. p. 270–282.

CHEN, Tianqi; GUESTRIN, Carlos. Xgboost: A scalable tree boosting system. *In*: PROCEEDINGS of the 22nd acm sigkdd international conference on knowledge discovery and data mining. San Francisco, CA: [*s.n.*], 2016. p. 785–794.

CHLEBUS, Marcin; ŚWITAŁA, Maciej Stefan. **So close and so far. Finding similar tendencies in econometrics and machine learning papers. Topic models comparison**. [*S.l.*], 2020.

CICCERI, Giovanni; INSERRA, Giuseppe; LIMOSANI, Michele. A machine learning approach to forecast economic recessions—an italian case study. **Mathematics**, Multidisciplinary Digital Publishing Institute, v. 8, n. 2, p. 241, 2020.

COULOMBE, Philippe Goulet. The macroeconomy as a random forest. **Available at SSRN 3633110**, 2020.

COULOMBE, Philippe Goulet; LEROUX, Maxime; STEVANOVIC, Dalibor; SURPRENANT, Stéphane. How is machine learning useful for macroeconomic forecasting? **arXiv preprint arXiv:2008.12477**, 2020.

D'ORAZIO, Paola. Big data and complexity: Is macroeconomics heading toward a new paradigm? **Journal of Economic Methodology**, Taylor & Francis, v. 24, n. 4, p. 410–429, 2017.

DIMOSKI, Matej; PETTERSEN, Markus. **Predicting housing prices with machine learning: a macroeconomic analysis of the Norwegian housing market**. 2020. Diss. (Mestrado) – Norwegian School of Economics.

FEN, Cameron; UNDAVIA, Samir. Improving External Validity of Machine Learning, Reduced Form, and Structural Macroeconomic Models using Panel Data, 2021.

FERNÁNDEZ-VILLAVERDE, Jesús. Has machine learning rendered simple rules obsolete? **European Journal of Law and Economics**, Springer, p. 1–15, 2021.

FERNÁNDEZ-VILLAVERDE, Jesús; GUERRÓN-QUINTANA, Pablo A. Estimating DSGE models: Recent advances and future challenges. **Annual Review of Economics**, Annual Reviews, v. 13, 2021.

FRIEDMAN, Jerome; HASTIE, Trevor; TIBSHIRANI, Robert. **The elements of statistical learning**. New York, NY: Springer, 2001. v. 1.

GARCIA, Márcio GP; MEDEIROS, Marcelo C; VASCONCELOS, Gabriel FR. Real-time inflation forecasting with high-dimensional models: The case of Brazil. **International Journal of Forecasting**, Elsevier, v. 33, n. 3, p. 679–693, 2017.

GENBERG, Hans; KARAGEDIKLI, Özer. Machine Learning and Central Banks: Ready for Prime Time? South East Asian Central Banks (SEACEN) Research e Training Centre, 2021.

HALL, Aaron Smalter *et al.* Machine learning approaches to macroeconomic forecasting. **The Federal Reserve Bank of Kansas City Economic Review**, v. 103, n. 63, p. 2, 2018.

HANSEN, Peter R; LUNDE, Asger; NASON, James M. The model confidence set. **Econometrica**, Wiley Online Library, v. 79, n. 2, p. 453–497, 2011.

HIRSCHBÜHL, Dominik; ONORANTE, Luca; SAIZ, Lorena. Using machine learning and big data to analyse the business cycle. **Economic Bulletin Articles**, European Central Bank, v. 5, 2021.

ISKHAKOV, Fedor; RUST, John; SCHJERNING, Bertel. Machine learning and structural econometrics: contrasts and synergies. **The Econometrics Journal**, Oxford University Press, v. 23, n. 3, s81–s124, 2020.

LIASHCHYNSKYI, Petro; LIASHCHYNSKYI, Pavlo. Grid search, random search, genetic algorithm: a big comparison for NAS. **arXiv preprint arXiv:1912.06059**, 2019.

LIN, Chin-Shien; KHAN, Haider A; CHANG, Ruei-Yuan; WANG, Ying-Chieh. A new approach to modeling early warning systems for currency crises: Can a machine-learning fuzzy expert system predict the currency crises effectively? **Journal of International Money and Finance**, Elsevier, v. 27, n. 7, p. 1098–1121, 2008.

MAEHASHI, Kohei; SHINTANI, Mototsugu. Macroeconomic forecasting using factor models and machine learning: an application to Japan. **Journal of the Japanese and International Economies**, Elsevier, v. 58, p. 101104, 2020.

MARTIN, Lisa-Cheree. **Machine learning vs. traditional forecasting methods An application to South African GDP**. [*S.l.*: *s.n.*], 2019.

MASINI, Ricardo P; MEDEIROS, Marcelo C; MENDES, Eduardo F. Machine Learning Advances for Time Series Forecasting. **arXiv preprint arXiv:2012.12802**, 2020.

MEDEIROS, Marcelo C; VASCONCELOS, Gabriel FR; VEIGA, Álvaro; ZILBERMAN, Eduardo. Forecasting inflation in a data-rich environment: the benefits of machine learning methods. **Journal of Business & Economic Statistics**, Taylor & Francis, v. 39, n. 1, p. 98–119, 2019.

MILLER, Tim. Explanation in artificial intelligence: Insights from the social sciences. **Artificial intelligence**, Elsevier, v. 267, p. 1–38, 2019.

MOLNAR, Christoph. **Interpretable Machine Learning**: A Guide for Making Black Box Models Explainable. [*S.l.*: *s.n.*], 2019. `https://christophm.github.io/interpretable-ml-book/`.

NOSRATABADI, Saeed; MOSAVI, Amirhosein; DUAN, Puhong; GHAMISI, Pedram; FILIP, Ferdinand; BAND, Shahab S; REUTER, Uwe; GAMA, Joao; GANDOMI, Amir H. Data science in economics: comprehensive review of advanced machine learning and deep learning methods. **Mathematics**, Multidisciplinary Digital Publishing Institute, v. 8, n. 10, p. 1799, 2020.

PARUCHURI, Harish. Conceptualization of Machine Learning in Economic Forecasting. **Asian Business Review**, v. 11, n. 2, p. 51–58, 2021.

SCHNORRENBERGER, Richard. **Fixed-income portfolio optimization based on dynamic Nelson-Siegel models with macroeconomic factors for the Brazilian yield curve**. 2017. Diss. (Mestrado) – Universidade Federal de Santa Catarina.

SPERANZA, Talitha F.; TANSCHEIT, Ricardo; VELLASCO, Marley M. B. R. A Monetary Policy Strategy Based on Genetic Algorithms and Neural Networks. **International Conference on Engineering Applications of Neural Networks**, v. 1, p. 605–616, 2020.

TSAY, Ruey S. **Analysis of financial time series**. 3. ed. Chicago, IL: JOHN WILEY & SONS, 2010.

XU, Lei. Machine learning and causal analyses for modeling financial and economic data. *In*: SPRINGEROPEN, 1. APPLIED Informatics. [*S.l.*: *s.n.*], 2018. v. 5, p. 1–42.

YOON, Jaehyun. Forecasting of real GDP growth using machine learning models: Gradient boosting and random forest approach. **Computational Economics**, Springer US, p. 1–19, 2020.