Universidade Federal do Rio Grande do Sul
Instituto de Matemática e Estatística
Departamento de Estatística
Programa de Pós-Graduação em Estatística

Renato Pedroso Lauris

Estimação do efeito de tratamento heterogêneo em maiores dimensões utilizando métodos de aprendizado de máquina: um estudo comparativo

Porto Alegre
2023

Renato Pedroso Lauris

Estimação do efeito de tratamento heterogêneo em maiores dimensões utilizando métodos de aprendizado de máquina: um estudo comparativo

Dissertação apresentada ao Instituto de Matemática e Estatística da Universidade Federal do Rio Grande do Sul para obtenção do título de Mestre em Estatística pelo Programa de Pós-Graduação em Estatística.

Orientador: Prof. Dr. Eduardo de Oliveira Horta

Coorientador: Prof. Dr. Rodrigo Citton Padilha dos Reis

Porto Alegre

2023

CIP - Catalogação na Publicação

Lauris, Renato Pedroso
Estimação do efeito de tratamento heterogêneo em maiores dimensões utilizando métodos de aprendizado de máquina: um estudo comparativo / Renato Pedroso Lauris. -- 2023.
144 f.
Orientador: Eduardo de Oliveira Horta.

Coorientador: Rodrigo Citton Padilha dos Reis.

Dissertação (Mestrado) -- Universidade Federal do Rio Grande do Sul, Instituto de Matemática e Estatística, Programa de Pós-Graduação em Estatística, Porto Alegre, BR-RS, 2023.

1. maiores dimensionalidades. 2. aprendizado de máquina. 3. efeito médio de tratamento condicional. 4. floresta causal. 5. debiased machine learning (DML). I. Horta, Eduardo de Oliveira, orient. II. dos Reis, Rodrigo Citton Padilha, coorient. III. Título.

Elaborada pelo Sistema de Geração Automática de Ficha Catalográfica da UFRGS com os dados fornecidos pelo(a) autor(a).

Dissertação de autoria de Renato Pedroso Lauris, sob o título **"Estimação do efeito de tratamento heterogêneo em maiores dimensões utilizando métodos de aprendizado de máquina: um estudo comparativo"**, apresentada ao Instituto de Matemática e Estatística, da Universidade Federal do Rio Grande do Sul, para obtenção do título de Mestre em Estatística pelo Programa de Pós-graduação em Estatística, aprovada em 09 de agosto de 2023 pela comissão julgadora constituída pelos doutores:

Prof. Dr. José Luiz Padilha da Silva
Departamento de Estatística - UFPR

Prof. Dr. Luís Gustavo Silva e Silva
DATALAB/Food and Agriculture Organization of the United Nations (FAO)

Profa. Dra. Márcia Helena Barbian
Departamento de Estatística - UFRGS

Agradecimentos

Fazer o mestrado em Estatística e realizar a dissertação em inferência causal significou “desbravar mares nunca antes navegados” por mim. E para encarar esse novo caminho é preciso ter pessoas que dêem as coordenadas na medida certa para não ficar sem norte, mas também que permita aprender a trilhar o seu próprio caminho em futuros trabalhos acadêmicos. Nesse sentido, professores Doutores Eduardo e Rodrigo, obrigado pelos conselhos, correções e direcionamentos, vocês foram Orientadores completos e foi um privilégio tê-los presente ao longo desse trabalho.

Além disso, esse mestrado em Estatística representa o meu desejo de retomar a sede de aprender que a academia nos traz. Foi gratificante demais estudar quais as ferramentas são mais adequadas para transformar os mais diversos dados espalhados em entendimento dos fenômenos que acontecem no mundo. Essa teimosia de querer aprender mais não veio do nada, precisou ter um ambiente em casa que estimulasse isso. Por isso, mãe Mari Vani e pai Roberto, obrigado por me proporcionar esse ambiente ao longo da vida e assim deixar a chama de aprender sempre acesa.

Por último, essa jornada acadêmica é feita de altos e baixos, conquistas e expectativas, reveses e aflições. Estar cercado de apoio é essencial. Obrigado a minha esposa Nani e aos meus dois gatos, Lótus e Woodstock, por estarem ao meu lado com seus amores incondicionais, renovando as energias e as certezas de que tudo daria certo no dia a dia, vocês foram fundamentais. Aos meus familiares, especialmente mãe, tia Clô e prima Bibiana, vocês têm um lugar especial sempre e nesse processo vocês só seguiram sendo compreensivas e atenciosas, minha gratidão por estarem comigo hoje e sempre. Quero também dedicar esse trabalho aos meus amigos de longa data Érico, Minossi, Thomas, Tiago e Zé e suas respectivas parceiras que tem tido paciência e me acolheram em momentos de descontração necessários nessa fase intensa, vocês valem ouro. E no lugar onde trabalho, Tribunal de Contas, tem pessoas que preciso mencionar pois só me apoiaram e me encorajaram a encarar esse desafio nesses dois anos, Gustavo, Alfredo, Sandro, Henrique e Bruno, e particularmente as pessoas do meu setor, Centro de Orientação e Fiscalização de Políticas Públicas (CPP), César, Giuliani, Guilherme, Isana, Leonardo, Marcos, Vanessa e Vinícius, que possuem esse espírito de aprender e senso de propósito que me fazia relembrar o que representava fazer essa dissertação.

Resumo

A profusão de dados com maiores dimensões e o crescente interesse em inferir causalidade têm permitido avançar a pesquisa de métodos que buscam estimar, para além do efeito médio de tratamento, o efeito médio de tratamento condicional (CATE). Nessa direção, alguns métodos de aprendizado de máquina têm sido propostos para estimar o CATE e identificar efeitos heterogêneos baseado nos próprios dados, de forma a reduzir a possibilidade de escolha arbitrária de covariáveis (*p-hacking*). Dois métodos têm se apresentado como alternativas robustas a esse propósito: Floresta Causal (Causal Forest, CF, Wager and Athey (2018)) e Double Machine Learning (DML, Chernozhukov et al. (2022)). Tendo em vista a concorrência entre estas abordagens e a ausência de estudos comparativos, a presente dissertação têm como objetivo principal apresentar esses métodos e avaliar, por um estudo de simulação, qual deles melhor lida com dimensões com formas funcionais lineares e não-lineares, cenários com picos e vales e descontinuidade. Uma simulação de Monte Carlo baseada em casos que ilustrem os desafios de estimação e de inferência para cada um dos métodos foi implementada. Utilizando indicadores de desempenho dos estimadores quanto à acurácia e ao viés da estimação (Erro quadrático médio –MSE e Viés Absoluto) e à adequação do intervalo de confiança (Taxa Cobertura), foram encontrados alguns resultados dignos de nota. As estimativas por DML tiveram níveis de acurácia e viés próximos ao CF medidos pelo MSE e o Viés Absoluto somente para os cenários linear e não-linear. Ambos os métodos CF e DML, nos cenários propostos, apresentaram inadequadas taxas de cobertura, indicando a necessidade de se avançar na proposição de procedimentos para construção de intervalos de confiança (ICs) e na construção de estimadores para a variância do CATE. Em geral, o DML não apresenta propriedades melhores para superar os desafios de estimação em cenários funcionais do CATE com picos e vales ou com descontinuidades. Por outro lado, se constatou que o método alternativo ao Floresta Causal apresenta menor sensibilidade do desempenho da estimação em dimensionalidades maiores, especialmente para tamanhos de amostra superiores a $n = 2000$, o que abre a possibilidade de pesquisas futuras avançarem em modelos mais flexíveis usando DML que possam apresentar melhorias no ajuste da estimação nos referidos cenários. Este trabalho avança na proposição de cenários de simulação e comparação entre os métodos CF e DML que não haviam sido comparados em trabalhos anteriores. Além disso, trouxe uma implementação alternativa à estimação do CATE para o método DML em R, usando a interface R-Python a partir dos pacotes *DoubleML* (Bach et al., 2021) e *Reticulate* (Ushey et al., 2023).

Palavras-chaves: maiores dimensionalidades; aprendizado de máquina; efeito médio de tratamento condicional; floresta causal; debiased machine learning (DML); simulação de Monte Carlo; comparação de desempenho dos estimadores.

Abstract

The proliferation of data with higher dimensions and the growing interest in inferring causality have allowed for advancements in research methods that aim to estimate, beyond the average treatment effect, the conditional average treatment effect (CATE). In this direction, some machine learning methods have been proposed to estimate the CATE and identify heterogeneous effects based on the data itself, thus reducing the possibility of arbitrary covariate selection (p-hacking). Two methods have emerged as robust alternatives for this purpose: Causal Forest (CF, Wager and Athey (2018)) and Double Machine Learning (DML, Chernozhukov et al. (2022)). Considering the competition between these approaches and the lack of comparative studies, the main objective of this dissertation is to present these methods and evaluate, through a simulation study, which one better handles the estimation of heterogeneous treatment effects with linear and nonlinear functional forms, scenarios with peaks and valleys, and discontinuities. A Monte Carlo simulation based on cases that illustrate the challenges of estimation and inference for each method was implemented. Performance indicators such as Mean Squared Error (MSE) and Absolute Bias for estimation accuracy, as well as Coverage Rate for the adequacy of the confidence interval, were used to assess the results. The simulation results revealed some noteworthy findings. The DML estimates had accuracy levels and bias close to CF as measured by MSE and Absolute Bias, but only for linear and nonlinear scenarios. Both CF and DML methods exhibited inadequate coverage rates in the proposed scenarios, indicating the need for further advancement in proposing procedures for constructing confidence intervals (CIs) and developing estimators for the variance of the CATE. Overall, DML does not demonstrate better properties for overcoming estimation challenges in functional scenarios of the CATE with peaks and valleys or discontinuities. On the other hand, it was found that the alternative method to Causal Forest had lower sensitivity in estimation performance in higher dimensions settings, especially for sample sizes larger than $n = 2000$. This opens up the possibility for future research to advance in more flexible models using DML that may improve estimation fitting in the aforementioned scenarios. This work contributes to the proposition of simulation scenarios and the comparison between CF and DML methods that had not been compared in previous studies. Additionally, it provides an alternative implementation for estimating the CATE using the DML method in R, using the R-Python interface through the packages *DoubleML* (Bach et al., 2021) e *Reticulate* (Ushey et al., 2023).

Keywords: higher dimensionalities; machine learning; conditional average treatment effect; causal forest; debiased machine learning (DML); Monte Carlo simulation; comparison of estimators performance.

Lista de figuras

Lista de tabelas

Lista de abreviaturas e siglas

AIPW	*Augmented Inverse Probability Weighting*, ou ponderação aumentada pelo inverso da probabilidade
ATE	*average treatment effect*, ou efeito médio de tratamento
BART	*Bayesian Additive Regression Trees*, ou método de árvores de regressão aditivas bayesianas
CATE	*conditional average treatment effect*, ou efeito médio de tratamento condicional
CF	*causal forest*, ou florestas causais
CFit	*cross-fitting*
CT	*Causal Trees*, ou árvores causais
DML	*debiased machine learning*, ou método de aprendizado de máquina imparcial
GRF	*generalized random forests*, ou florestas aleatórias generalizadas
IPW	*Inverse probability weighting*, ou ponderação pelo inverso da probabilidade
KNN	*k-nearest neighbor*, ou método do vizinho mais próximo
Lasso	*least absolute shrinkage and selection operator*, ou operador mínimo absoluto de encolhimento e seleção de variáveis
ML	*machine learning*, ou aprendizado de máquina
MSE	*mean square error*, ou erro quadrático médio
RCT	*randomized control trials*, ou desenho de estudo experimental
SUTVA	*stable unit treatment value assumption*

Sumário

1 Introdução

Atribuir causalidade de determinadas ações sobre desfechos tem se apresentado como interesse comum para um grande número de pesquisas desenvolvidas em diferentes áreas do conhecimento e/ou do mercado, tais como epidemiologia (Hernán and Robins, 2006), medicina de precisão (Liang, 2018), ciências sociais aplicadas (Wager and Athey, 2018; Athey and Wager, 2019) ou empresas de tecnologia (Syrgkanis et al., 2021).

Compreender se a realização de uma certa cirurgia produz uma maior taxa de sobrevida entre pacientes com doença cardíaca na artéria coronária (Blackstone, 2019), ou avaliar se a implementação de ações de saneamento básico e medidas de higiene reduzem os casos de diarreia na infância (Wolf et al., 2018), podem ser vistos como um exercício de inferência de relações de causa (tratamento) e efeito (desfechos). O conjunto de métodos para inferir a presença e a magnitude das relações de causa e efeito vem sendo denotado por "inferência causal" (Aalen et al., 2012) e estas relações são referidas como "efeitos causais". Os efeitos causais devido a uma ação (causa ou tratamento) são denotados, de maneira genérica, por "efeitos de tratamento".

A aplicação destes métodos pode ser vista em diferentes áreas da ciência. Podemos citar, nos Estados Unidos, evidências que demonstram que o aumento de policiais do sexo feminino mediante política afirmativa promoveu um aumento da notificação de casos de violência doméstica e consequente redução de homicídio de parceiros e abusos domésticos não fatais (Miller and Segal, 2019) e que estudantes negros que frequentam salas de aula com professores da mesma cor têm maior desempenho educacional, sendo que o mesmo não ocorre com alunos latinos (Redding, 2019). Facebook, Google e Microsoft, empresas conhecidas do campo de tecnologia, obtém dados dos usuários e regularmente realizam experimentos para testar a efetividade dos seus sistemas de propaganda online, segundo Athey and Luca (2019).

Devido à profusão de conjuntos de dados suficientemente extensos em termos de tamanho amostral mas especialmente em termos de variáveis, a estimação de efeitos de tratamentos heterogêneos vem ganhando atenção nos últimos anos (Abrevaya et al., 2015; Athey and Imbens, 2016; Athey et al., 2019; Chernozhukov et al., 2018, 2022; Fan et al., 2022; Imai and Ratkovic, 2013; Jacob, 2021; Knaus et al., 2020; Künzel et al., 2019; Semenova and Chernozhukov, 2021; Semenova et al., 2022; Zhou and Zhu, 2021). A ideia

geral é que os efeitos do tratamento não são constantes para subgrupos das unidades. Estes subgrupos são formados por características dos indivíduos (também chamadas de covariáveis, variáveis explicativas ou preditoras).

Esse tipo de análise em um contexto de maior dimensionalidade do conjunto de covariáveis pode ser desafiador. A ausência de um critério claro pré-estabelecido de quais covariáveis podem apresentar uma heterogeneidade do efeito de tratamento pode comprometer a validade de tal achado. Neste sentido, o analista de dados corre o risco de buscar de forma indiscriminada por covariáveis que apresentem uma heterogeneidade no efeito de tratamento estatisticamente significativa, fenômeno conhecido como *p-hacking*. Este termo é descrito na literatura como a prática que pesquisadores usam, consciente ou inconscientemente, para encontrar p-valores artificialmente "satisfatórios" (em geral, $p < 0,05$), devido a dificuldade de conseguir publicar artigos cujos resultados sejam estatisticamente não significativos (veja, por exemplo, Brodeur et al., 2020).

Uma das alternativas para evitar o *p-hacking* é a elaboração de um plano de pré-análise ou um protocolo para ensaios clínicos a ser publicado antes do desenvolvimento do estudo. O objetivo desse plano ou protocolo é especificar um conjunto de covariáveis cujas hipóteses de efeito de tratamento heterogêneo pretendem ser testadas no estudo à luz da teoria e estudos anteriores.

Outra forma de lidar com o contexto de maior dimensionalidade e evitar a escolha arbitrária de covariáveis ou *p-hacking* é utilizar métodos para estimar os efeitos de tratamentos heterogêneos com base em subgrupos determinados pelo próprio conjunto de dados, também denominados de métodos *data-driven*. Neste sentido, diversas abordagens têm sido propostas na literatura recente para estimar heterogeneidade em efeitos de tratamento, dentre os quais destacam-se: o operador mínimo absoluto de encolhimento e seleção de variáveis (*least absolute shrinkage and selection operator*, Lasso) proposto por Tian et al. (2014); o método de dois estágios via ponderação pelo inverso da probabilidade (*Inverse probability weighting*, IPW) proposto por Abrevaya et al. (2015); o método floresta causal (*causal forest*) proposto por Wager and Athey (2018); o método aprendizado de máquina imparcial (*debiased machine learning*, DML) proposto por Chernozhukov et al. (2022); e o método de árvores de regressão aditivas bayesianas (*Bayesian Additive Regression Trees*, BART) proposto por Hahn et al. (2020).

1.1 Questões de interesse na pesquisa de efeitos de tratamento heterogêneos

Embora se verifique os recentes avanços na proposição de métodos de estimação do efeito de tratamento heterogêneo no contexto de maior dimensionalidade, a literatura relata ainda uma série de desafios à estimação e inferência. Um destes desafios é conhecido como a "maldição da dimensionalidade", em que quando o número de covariáveis na composição dos subgrupos aumenta se constata a diminuição da acurácia dos estimadores, em geral, medida pelo erro quadrático médio (Abrevaya et al., 2015; Athey et al., 2019). Segundo **?**, o termo foi introduzido por **?** e indica que o número de amostras necessárias para estimar uma função qualquer com um dado nível de acurácia cresce exponencialmente com o número de variáveis de entrada (dimensionalidade) da função.

Athey and Imbens (2016) mencionam que certos métodos de aprendizado de máquina não podem ser usados para inferência visto a dificuldade de estimar intervalos de confiança centrados e com distribuição assintótica normal devido a eventual viés ocorrido na amostra de treino não ser facilmente corrigido numa amostra teste, mesmo em tamanhos de amostras grandes.

Os autores em Wager and Athey (2018) e em Athey et al. (2019) salientam a baixa acurácia e alta variância na estimação em pontos extremos da região de suporte das covariáveis, denominados efeitos de borda.

Outro ponto mencionado na literatura é que abordagens que adotam em certa medida o método do vizinho mais próximo para estimar o efeito de tratamento heterogêneo, tal como o método floresta causal, tendem a suavizar pontos de vales e de cumes de uma função do efeito causal, especialmente nos pontos extremos (Wager and Athey, 2018; Athey et al., 2019).

E ainda, o uso de aprendizado de máquina como método de regularização, mecanismo de seleção das covariáveis, redução de dimensionalidade do modelo, e estimação direta do efeito de tratamento heterogêneo implica viés e sobreajuste do estimando de interesse (Chernozhukov et al., 2018). Diante desse problema, os autores propõe um método robusto da estimação chamado ortogonalização de Neyman e uma regra de partição da amostra denominado de *cross-fitting*.

1.2 Justificativa

Na literatura, os métodos CF e DML vêm se destacando entre os demais métodos citados para a estimação do efeito de tratamento heterogêneo (Athey and Imbens, 2016; Athey et al., 2019; Chernozhukov et al., 2018, 2022; Jacob, 2021; Wager and Athey, 2018; Semenova and Chernozhukov, 2021; Semenova et al., 2022) cujas formas funcionais apresentam formatos senoidais ou parábolas que representam picos e vales e regiões com presença de descontinuidades. Tendo em vista a grande atenção que os métodos citados vêm recebendo, a concorrência entre estas abordagens, e a ausência de estudos comparativos, se percebe a necessidade de aprofundar o estudo destas técnicas. Esta dissertação foca na avaliação do desempenho destes métodos com respeito à estimação.

1.3 Objetivos

Assim, o presente trabalho tem como objetivo principal verificar qual desses métodos selecionados melhor lida com estimação do efeito de tratamento heterogêneo sob a presença de formas funcionais com picos e vales bem como descontinuidades em um contexto de maior dimensionalidade.

E por meio de simulação de Monte Carlo baseado em casos que ilustrem os desafios de estimação e de inferência para cada um dos métodos, se pretende identificar qual deles apresenta as estimações mais adequadas para cada cenário, explicitando os motivos que levaram ao melhor desempenho.

1.4 Organização da dissertação

O restante da dissertação está organizada da seguinte forma. O Capítulo 2 apresenta uma breve revisão sobre o conceito de causalidade e inferência causal. Também é apresentada a estrutura do modelo causal de Rubin que define efeitos causais por meio de desfechos potenciais. O Capítulo 3 apresenta a definição do efeito médio de tratamento e do efeito médio de tratamento condicional, bem como os principais resultados de identificação destes efeitos. No Capítulo 4, é apresentada uma descrição dos métodos de estimação de efeitos heterogêneos. São abordados os métodos de dois estágios via pondera-

ção pelo inverso da probabilidade (IPW em dois estágios), floresta causal e o aprendizado de máquina imparcial (DML). No Capítulo 5, é apresentado o estudo de simulação para comparação dos métodos acima citados. Diferentes cenários, incluindo aqueles com maior dimensionalidade no conjunto de covariáveis e efeitos heterogêneos com formas funcionais alternativas, foram gerados para avaliação do desempenho dos métodos quanto à acurácia, viés, consistência e adequação dos intervalos de confiança (ICs). Por fim, no Capítulo 6, são apresentadas as conclusões deste trabalho, bem como as limitações e sugestões de temas de interesse de pesquisa futuros.

2 Causalidade e a Abordagem de Desfechos Potenciais

A compreensão da noção de causalidade é importante para fundamentar como as abordagens estatísticas, tais como Desfechos Potenciais ou Modelo Causal de Rubin, (Rubin, 1974), Grafos Acíclicos Direcionados (Pearl, 1995) e Modelos Causais Estruturais (Pearl, 2012), buscam realizar inferência causal com base na definição precisa dos efeitos causais.

A causalidade foi objeto de estudo de inúmeros filósofos e cientistas ao longo do tempo, tendo cada área delimitado sua noção de causa a partir das suas perspectivas particulares. Antes de apresentar o histórico de avanço da noção de causalidade, vale pontuarmos a noção de associação e sua relação com causalidade.

A noção de associação, em termos estatísticos, representa uma métrica de uma relação em que o conhecimento da ocorrência de uma variável aleatória X traz informação sobre a distribuição de outra variável Y e vice-versa. Trata-se de uma relação de dependência entre duas variáveis. Em notação de probabilidade, teremos $\Pr(Y|X) \neq \Pr(Y)$ ou $\Pr(X|Y) \neq \Pr(X)$.

Já a causalidade, onde X causa Y por exemplo, efetivamente implica existir associação, mas deve vir acompanhado com o atendimento de outras premissas: ordem temporal (X acontece antes de Y) e relação não espúria. Em uma relação espúria, encontramos numericamente uma associação com base na amostra disponível, mas que não tem sustentação em premissas e no conhecimento da área (Zhang and VanDyke, 2022). Por outro lado, também existem associações entre objetos que não representam relações causais, tais como associações provenientes de causas comuns, onde uma variável Z causa X e Y e assim estabelece uma relação de dependência não existente *a priori* (Altman and Krzywinski, 2015). Em um exemplo hipotético mencionado também em Altman and Krzywinski (2015), podemos observar que pessoas que tomam muito café por dia apresentam menor incidência de câncer de pele. Isso poderia levar a conclusão de que tomar café está associado a uma maior proteção contra o câncer de pele. Entretanto, um possível confundidor dessa relação é que pessoas que tomam bastante café podem ser aquelas que trabalham em ambientes fechados com menor exposição ao sol. Dessa forma, seria o tempo diário em ambientes fechados o causador de uma menor exposição ao sol e a

consequente redução de risco de câncer de pele e não o fato de o indivíduo tomar muito café diariamente.

Feito esse registro sobre a diferença entre associação e causalidade e retomando a perspectiva histórica da noção de causalidade, temos que Aristóteles, em suas obras Física e Metafísica, com o objetivo de conferir explicação para existência de quaisquer coisas (objetos e fenômenos) definiu quatro tipos de causas (Falcon, 2022):

- causa material: do que é feito? Exemplo: animais vertebrados são compostos de ossos e sangue.
- causa formal: o que dá forma? Exemplo: formato de uma estátua.
- causa eficiente: qual a fonte primária da mudança ou manutenção? Exemplo: para desenvolver a educação formal de adolescentes é preciso escolas com infraestrutura, professores, participação dos pais como causas eficientes.
- causa final: o fim para o qual uma coisa é feita. Exemplo: Para a causa final da saúde, as pessoas realizam exercícios físicos e tomam remédios.

De acordo com Holland (1986), o conceito de causa eficiente de Aristóteles é o que mais se aproxima dos efeitos das causas. Os efeitos das causas partem de uma causa específica e tenta identificar e quantificar seus efeitos tomando como referência uma causa alternativa. Por outro lado, as causas dos efeitos buscariam identificar e mensurar as diversas causas de um determinado efeito. Holland (1986) argumenta que as causas dos efeitos não seria a perspectiva adequada para análise de causalidade. Dessa forma, os efeitos das causas seria a noção de causalidade foco das abordagens estatísticas de inferência causal.

No século XVIII, David Hume em "Tratado da natureza humana" (1740), editado por Norton and Norton (2011), e em "Investigação sobre o Entendimento Humano" (1748), Beauchamp (1999), estabeleceu três critérios para conferir casualidade a determinado evento. Então, supondo o exemplo em que "ocorrência do evento A causa a ocorrência do evento B" ilustramos os conceitos propostos por Hume conforme segue:

- contiguidade espaço-temporal. Nesse caso, os eventos A e B são contíguos no tempo e no espaço;
- sucessão temporal. Isso denota que o evento A ocorre antes de B; e
- conjunção conjunta. Os dois eventos, A e B, sempre ocorrem ou não ocorrem conjuntamente.

Segundo Holland (1986), os dois primeiros critérios podem ser acomodados nas abordagens modernas de causalidade, porém o terceiro pode não valer na presença de erro de medida, o que poderia invalidar uma relação de causa e efeito efetivamente existente. Além disso, Hume não estabelece como condição necessária para aferir o efeito de uma causa a comparação com outra causa alternativa, nem contempla a ideia de experimento, pontos importantes da noção atual de casualidade. Vale ressalvar que os experimentos só vieram a se tornar um método científico propício a analisar causalidade no século XIX com os trabalhos de Lavoisier e outros, em momento posterior aos escritos de Hume (Morabia, 1991).

Já no século XIX, John Stuart Mill na sua obra "Sistema de lógica dedutiva e indutiva", de 1843 (Mill, 2011), propõe métodos para aferir causalidade em experimentos:

- método da variação concomitante: a variação conjunta entre dois eventos poderia indicar causalidade entre eles, sendo um deles causa ou efeito do outro. De acordo com Holland (1986), esse método somente vale para casos onde a associação entre os eventos acontece.
- método da diferença: a ocorrência de um fenômeno em circunstâncias comuns, exceto uma, a circunstância diferente será o efeito, a causa ou parte da causa do fenômeno. O método não leva em consideração o fato que não necessariamente uma causa produz efeitos.
- método dos resíduos: basicamente busca extrair o efeito residual de causas antecedentes considerando uma causa cujo os efeitos já são conhecidos e considerando os efeitos totais do fenômeno. Assim a diferença entre efeitos totais e efeitos conhecidos poderia identificar o efeito da causa de interesse.
- método da concordância: no caso de existência de somente uma circunstância em comum para várias instâncias de um fenômeno, essa circunstância é considerada a causa ou o efeito do referido fenômeno. Embora o método aparentemente busque identificar uma causa, na prática, ele contribui para testar a invalidade de possíveis causas de um fenômeno (efeito). Afinal, todo efeito tem uma causa, mas não necessariamente toda causa tem um efeito (efeito nulo ou ainda experimentos podem falhar).

Finalizando essa perspectiva histórica da noção de causalidade, já no século XX, os filósofos Hans Reichenbach e Patrick Suppes tentaram definir causalidade a partir da

noção de "aumento de probabilidade". Ou seja, A causa B se A aumenta a probabilidade de B (Pearl and Mackenzie, 2018, Capítulo 1). Entretanto, essa definição falha ao não considerar a possibilidade de causas comuns que levam a ocorrência de A e também ao aumento da probabilidade de B, o que não significa necessariamente que A causa B tal como esperado pela definição proposta. Por exemplo, é razoável pensarmos em um cenário em que pacientes com piora de seus quadros clínicos sejam submetidos a tratamentos mais arriscados (evento A) e efetivamente verificarmos maior probabilidade de óbito – $\Pr(B|A) > \Pr(B)$ – em um universo de pacientes de um hospital. No entanto, isso não significa que os tratamentos mais arriscados causam óbitos.

Além da evolução do conceito na história da filosofia, a noção de causa apresenta perspectivas particulares na medicina, na economia e em ciências sociais, conforme mencionado por Holland (1986). Na medicina, especificamente no campo da bacteriologia, os postulados de Koch-Henle buscaram estabelecer critérios para definir se um micro-organismo é causador de determinada doença da seguinte forma: (i) o organismo deve ser encontrado em todos os casos de indivíduos com a doença; (ii) o organismo precisa ser isolado do paciente e se reproduzir em uma cultura; e (iii) quando o organismo mantido nessa cultura é inoculado em um ser vivo, a doença é reproduzida (Holland, 1986).

Para outros campos da saúde, de acordo com Holland (1986), Hill (1965) sistematizou nove critérios que ajudam a identificar a associação como causalidade em observações não experimentais como é o caso em epidemiologia (relação entre ambiente e doenças): temporalidade (causa deve ser anterior ao efeito), experimentação (permitir a realização de experimentos), gradiente biológico (maior exposição ao agente, maior presença da doença causada), plausibilidade, coerência e analogia (dado pelo conhecimento prévio já adquirido para certas relações de causa efeito, podemos encontrar paralelos com outras relações), força associativa, consistência (generalização da associação através da população) e especificidade (efeitos específicos dado causas especificas). Morabia (1991) encontra similaridade dos critérios de Hill com as definições apresentadas por Hume, com exceção da plausibilidade e experimentação por razões históricas, visto que Hume era um pensador pré-experimentalista.

Já na área da economia se cunhou uma noção particular de causalidade para dados estruturados em séries temporais chamado *causalidade de Granger* (Granger, 1969). Uma variável causa outra se aquela for capaz de aumentar a capacidade de predição de valores futuros desta variável. Em termos probabilísticos, dado X e Z no tempo r, anterior a

ocorrência de Y no tempo s posterior ($r < s$), variáveis definidas na população, é dito que X não é *Granger causa* de Y, dada a informação de Z, se X e Y são condicionalmente independentes dado Z, $\Pr(Y|X,Z) = \Pr(Y|Z)$. A variável X será *Granger causa* de Y, dado Z, se X ajuda a predizer Y, levando em consideração Z. De acordo com Holland (1986), a *causalidade de Granger* é falha para definir causalidade no mesmo ponto que a definição de Patrick Suppes acima mencionada, visto que se Z for uma causa comum a X e a Y, onde ambos variam e aumenta o poder preditivo da variável Y, a *causalidade de Granger* de X sobre Y poderá ser uma causalidade espúria.

Por fim, as Ciências Sociais adotaram modelos ou diagramas de caminho (*path models, path diagrams*), uma representação gráfica de um sistema de equações lineares, para representar relações causais. Para fins ilustrativos, se X causa Y então temos o seguinte diagrama:

$$X \rightarrow Y$$

Embora essa metodologia permita representar uma noção de causalidade entre variáveis de interesse na área de Ciências Sociais, o uso do sistema de equações lineares tem um caráter associativo e não necessariamente a causalidade que estamos interessados em mensurar.

Por meio da revisão acima em relação aos esforços de filósofos e cientistas de diversas áreas para elaborar uma definição de causalidade, foi possível observar que todas as propostas citadas falham em alguma situação, incorrendo em casos de caracterização de causalidade espúria. Essa limitação só veio a ser superada por abordagens como Desfechos Potenciais, Grafos Acíclicos Direcionados e Modelos Causais Estruturais. Nesse trabalho nos restringiremos a aprofundar o arcabouço de Desfechos Potenciais na subseção a seguir.

2.1 O arcabouço de Desfechos Potenciais

Holland (1986) argumenta que a análise adequada da causalidade de um fenômeno tem como foco os efeitos das causas – para o autor, *causa* e *tratamento* são sinônimos. Nesse sentido, para codificar e estabelecer uma medida para causalidade, o arcabouço de Desfechos Potenciais (Rubin, 1974) introduz o conceito de *desfecho potencial*. O desfecho potencial é o resultado potencialmente encontrado fruto de um tratamento W incidente sobre uma unidade de análise i em um determinado tempo, representado por $Y_i^{(W)}$. O

tratamento é entendido como uma ação, manipulação, exposição ou intervenção aplicada a uma unidade (Imbens and Rubin, 2015). A unidade de análise a ser considerada pode ser um objeto individual (por exemplo, uma empresa, uma pessoa, um hospital) ou um objeto coletivo (uma sala de aula, um grupo de animais) em um instante de tempo. Um mesmo objeto físico em tempos distintos deve ser considerado como unidades distintas. Por exemplo, um indivíduo A toma um remédio pela manhã e à noite para curar sua dor de dente. Para fins de análise causal, esses eventos serão considerados como unidades diferentes visto que as reações ao tratamento podem ser distintas conforme o período do dia.

O conceito do efeito de um tratamento W só tem sentido substantivo a partir da comparação com outro tratamento de referência. Em um contexto dicotômico, estamos interessados no efeito de um tratamento, $w_i = 1$, em relação a um tratamento base, denominado *controle* ($w_i = 0$). A variável de tratamento W_i é a representação de uma variável (ou ainda um vetor) aleatório referente à designação ao tratamento, podendo assumir valores categóricos, contínuos ou multivariados. Especificamente no caso binário, assumirá os seguintes valores:

$$W_i = \begin{cases} 1, & \text{se } i \text{ foi tratado.} \\ 0, & \text{se } i \text{ não foi tratado.} \end{cases} \tag{2.1}$$

O efeito causal individual τ_i é uma quantidade determinada em função de dois *desfechos potenciais*, $Y_i^{(1)}$ e $Y_i^{(0)}$, novamente considerando o caso binário para fins de simplificação. O primeiro deles, $Y_i^{(1)}$, corresponde ao desfecho caso o indivíduo i seja alocado ao grupo tratamento. O segundo, $Y_i^{(0)}$, se refere à situação caso o *mesmo* indivíduo fosse alocado ao grupo controle. Formalmente, o efeito causal individual do tratamento W_i sobre a observação i no arcabouço de desfechos potenciais é definido da seguinte maneira para o contexto dicotômico.

Definição 2.1.1 (Efeito causal individual)**.** O **efeito causal individual** de uma variável de tratamento dicotômica W_i sobre uma unidade i é o parâmetro $\tau_i \in \mathbb{R}$ dado por

$$\tau_i = Y_i^{(1)} - Y_i^{(0)}. \tag{2.2}$$

Convém ressaltar que, alternativamente à forma aditiva (2.2), podemos representar o efeito causal em uma escala de log-diferença (e.g., $\log Y_i^{(1)} - \log Y_i^{(0)}$), por uma razão

em relação ao tratamento base (e.g., $\frac{Y_i^{(1)}}{Y_i^{(0)}}$), uma diferença relativa de tratamento (e.g., $\frac{Y_i^{(1)}-Y_i^{(0)}}{Y_i^{(0)}} \times 100$), entre outras formas.

Em um desenho de pesquisa que vá além do cenário dicotômico tratamento versus controle, os desfechos potenciais exprimem-se conforme os casos abaixo:

- categórico: $Y_i^{(w)}$, para $w = 0, 1, \ldots, K-1, K$, sendo $K \in \mathbb{N}$.
- contínuo: $Y_i^{(w)}$, para qualquer $w \in \mathbb{R}$.
- multivariado: $Y_i^{(w_1,\ldots,w_K)}$, para $w_K \in \tau_K$ e $K \in \mathbb{N}$.

Na realidade, somente um dentre os desfechos potenciais é observável. Por exemplo, no contexto dicotômico o cientista observa ou o valor $Y_i^{(1)}$ (caso a unidade i seja tratada), ou o valor $Y_i^{(0)}$ (caso contrário). Os demais cenários (i.e., os valores não observáveis) são denominados *contrafactuais*, representando resultados potencialmente verificáveis caso a situação de tratamento fosse outra. Essa restrição de observação é conhecido como o *problema fundamental da inferência causal* (Holland, 1986). Para dar um exemplo, pensemos em uma estudante de baixa renda, i, que em um dado ano terá perspectivas de notas distintas na prova: $Y_i^{(1)}$ caso seja contemplada com uma bolsa para um curso preparatório, ou $Y_i^{(0)}$ caso não seja contemplada (suponha, por simplicidade, que o recebimento ou não da bolsa baseia-se em um sorteio dentre estudantes que atendam a certo critério de elegibilidade). Nesse caso, ao final do concurso vestibular observaremos apenas uma nota, $Y_i^{(w)}$, em que $w_i = 1$ (resp., $w_i = 0$) representa o cenário em que a estudante foi contemplada (resp., não foi contemplada) com a bolsa. Assim o valor não observado representa a nota contrafactual da estudante, aquela que seria observada caso o resultado da seleção para a bolsa fosse distinto.

No caso dicotômico, conforme as equações (2.1) e (2.2), vê-se que o valor observado da variável de interesse (por exemplo, a nota da estudante) pode ser expressa na forma

$$Y_i = W_i Y_i^{(1)} + (1 - W_i) Y_i^{(0)}. \tag{2.3}$$

O arcabouço de Desfechos Potenciais estabelece uma série de premissas que determinam claramente o objeto tratamento (variável W_i) e a sua relação com o resultado (Y_i): primeiramente, supõe-se que a *ordem causal* da relação entre o tratamento W_i e o resultado Y_i supõe que o tratamento causa o resultado ($W_i \rightarrow Y_i$), que inexiste causalidade reversa ($Y_i \nleftarrow W_i$) e tampouco simultaneidade ($W_i \nleftrightarrow Y_i$). Evidentemente, não devemos descartar a possibilidade de existir causalidade reversa ou simultaneidade. E

ajustar para existência de tais situações é objeto de estudo na literatura de inferência causal (**??**). No entanto, tal discussão foge do escopo desse trabalho e somente temos que fundamentar a validade da suposição *ordem causal* da relação entre o tratamento W_i e o resultado Y_i nos exercícios empíricos. Além disso, como forma de conferir certas condições de regularidade ao efeito causal do tratamento em uma observação, são impostas algumas suposições quanto às características do tratamento, conhecidas pela sigla SUTVA (*stable unit treatment value assumption*), conforme Rubin (1980).

Suposição 1 (SUTVA)**.** *A suposição SUTVA se desdobra nos seguintes componentes:*

1. ***Consistência****: indica a inexistência de versões ocultas do tratamento ou formas distintas de administração. Não importa como se recebe o tratamento, o resultado potencial será o mesmo que o efetivamente observado. Isso implica que se o indivíduo i é designado ao grupo de tratamento ($W_i = 1$), então o desfecho observado é igual ao desfecho potencial caso o indivíduo fosse alocado ao grupo tratamento. Por exemplo, o resultado de uma política de vacinação independe dos profissionais de saúde que a aplicam. Ou em outro caso se assume que as aspirinas tomadas possuem eficácia semelhante. Embora essa premissa nem sempre seja satisfeita em um caso concreto, é possível redefinir o tratamento de forma a satisfazer a suposição. Essa suposição não requer que todas as formas de cada nível de tratamento sejam idênticas através de todas as unidades, somente que a unidade i exposta ao nível de tratamento W especifica um desfecho potencial bem-definido.*

$$Y_i = Y_i^{(w)} \ , \forall i \ , \ W_i = w.$$

2. ***Não-interferência entre unidades****: significa que o resultado de tratamento em uma unidade não é afetado pelo tratamento sobre outras unidades. Um exemplo que viola essa premissa é o efeito da compra de um computador no aprendizado de um aluno, visto que existem grandes chances desse aluno compartilhar o uso com colegas próximos que não receberam esse equipamento. Uma alternativa para restabelecer a atendimento da suposição, é a redefinição do efeito tratamento para o nível de escola ao invés de aluno, separando o grupo de escolas tratadas e não-tratadas. Ainda assim, há situações onde a solução seria mais complexa tal como o efeito de equilíbrio geral de um programa de treinamento para o trabalho. Comum entre estudos econômicos, o efeito de um conjunto de indivíduos treinados pelo programa*

poderá afetar o mercado de trabalho como um todo através do aumento da oferta de trabalhadores qualificados e redução de salários, contrariando o efeito esperado do tratamento (Imbens and Rubin, 2015). Para um conjunto de n indivíduos, então, temos que o desfecho potencial do indivíduo i independe do tratamento sobre os demais indivíduos da população, conforme segue:

$$Y_i^{(W_1,W_2,\ldots,W_n)} = Y_i^{(W_i)}.$$

Em contextos de dados observacionais, em contraposição a estudos experimentais, encontramos que características anteriores à intervenção (vetor de covariáveis pré-tratamento X_i) podem vir a determinar a probabilidade de tratamento do indivíduo i, sendo modelado como $\Pr(W_i = 1|X_i) = f(X_i)$.

Por fim, um ponto controverso na caracterização de causalidade por meio de desfechos potenciais é o entendimento consagrado por Holland (1986) de que "não há causalidade sem manipulação" (*"No causation without manipulation"*, em inglês). Basicamente, o autor argumenta que não é possível definir efeitos causais decorrentes de atributos imutáveis de uma unidade de uma população, tais como gênero, raça, idade, entre outros. O interesse no efeito de atributos é muito comum em estudos econômicos sobre o mercado de trabalho em que se busca mensurar o discriminação de renda ou de empregabilidade de acordo com gênero, raça ou idade, por exemplo. O argumento de Holland (1986) pela impossibilidade de se estabelecer causalidade nesses casos se dá em virtude de que o atributo de um indivíduo somente assume uma das alternativas. A título ilustrativo, um indivíduo branco não pode assumir o atributo raça negra e vice-versa. Assim raça não pode ser designada aleatoriamente. Por outro lado, Heckman (2022) critica essa limitação proposta e ressalta as vantagens de modelos teóricos abstratos e de experimentos hipotéticos (*Thought Experiment*, em inglês.) como meio para realizar análises causais. A escolha desses modelos e das definições de parâmetros causais de interesse, já conhecidos na literatura de economia desde Haavelmo (1943) e Haavelmo (1944), são determinados pelos problemas científicos que se quer investigar. Nesse exemplo (do efeito causal da raça sobre renda) o uso de um modelo causal hipotético seria uma alternativa viável para buscar mensurar o impacto hipotético de raça em relação aos rendimentos, caracterizado pelo problema da discriminação racial. Pode-se relacionar essa diferenciação entre atributos vis-à-vis tratamentos com a distinção entre estudos observacionais vis-à-vis estudos randomizados.

Por fim, como se pôde apresentar neste capítulo, a noção de causalidade é objeto de reflexão há bastante tempo, evoluindo para definições e metodologia que superam limitações de definições anteriores que levavam a identificação espúria de efeito causal. O que se verifica é que o arcabouço de Desfechos Potenciais permite avançar no objetivo de inferir causalidade com seu conjunto de pressupostos e condições de regularidade. O capítulo que segue tratará da definição do efeito médio de tratamento e do efeito tratamento condicional, sendo este último o estimando de interesse principal para inferir efeitos causais heterogêneos.

3 Efeitos médio de tratamento e de tratamento condicional

Considerando o arcabouço de Desfechos Potenciais apresentado na Seção 2.1, nesse capítulo estamos interessados em definir o efeito causal médio do tratamento W em termos populacionais, denominado efeito médio de tratamento (*average treatment effect* – ATE, em inglês), indo além do nível individual definido pela equação (2.2). Ademais, estamos interessados em conceituar o efeito causal médio condicionado a um conjunto (vetor) de covariáveis X, denominado efeito tratamento médio condicional (*conditional average treatment effect* – CATE, em inglês). O CATE nos permitirá avançar no objetivo central da dissertação que é avaliar métodos para estimar e inferir a respeito de efeitos causais heterogêneos.

De acordo com Imbens and Rubin (2015), podemos entender *população* como um conjunto *finito* de unidades das quais observamos o valor respectivo de covariáveis (ou variáveis pré-tratamento, X_i), tratamentos (W_i) e desfechos realizados (Y_i, conforme equação 2.3), em que o mecanismo de alocação de tratamento é a única fonte de aleatoriedade. Essa abordagem pode ser aninhada em um cenário onde há uma *superpopulação*, onde o conjunto de unidades que observamos é aleatoriamente obtido de uma população infinita – no sentido de que é descrita por um modelo teórico (ver Ding et al., 2017; Hartley and Sielken Jr, 1975). Uma das diferenças entres as abordagens, é a estimativa da variância ser mais conservadora (maior) sob a perspectiva de uma superpopulação em relação a uma população finita (Ding et al., 2017). Vale ressaltar que as covariáveis, nesse contexto, não são afetados pelo tratamento.

3.1 O efeito médio de tratamento

Em muitas pesquisas científicas, o efeito médio de tratamento (*average treatment effects*, ATE) é o estimando de maior relevância, pois nos permite inferir, de maneira agregada, o efeito causal de determinada política pública, de um tratamento médico ou do impacto de determinados experimentos laboratoriais. Formalmente, o ATE é definido da seguinte forma.

Definição 3.1.1 (Efeito Médio de Tratamento – ATE)**.** Dada uma população de interesse $N = \{1, \ldots, n\}$, o **efeito médio de tratamento** (*average treatment effect* – ATE) é dado por:

$$\tau_{\text{ATE}} = \mathbb{E}\left[Y_i^{(1)} - Y_i^{(0)}\right]. \tag{3.1}$$

É importante perceber que está implícito na notação que o ATE não depende de i (isto ocorre, por exemplo, se as unidades são identicamente distribuídas). Além disso, nota-se que τ_{ATE} não é um parâmetro identificável, em virtude do problema fundamental da inferência causal (Holland, 1986): para cada elemento i, ou observamos $Y_i^{(1)}$ ou observamos $Y_i^{(0)}$, onde o desfecho observável Y_i é representado pela equação (2.3). De forma a garantir a identificabilidade do estimando ATE, estabelecemos algumas suposições adicionais a seguir:

Suposição 2 (Ignorabilidade)**.** *A suposição de ignorabilidade procura estabelecer que a* designação *de tratamento, dado um vetor de covariáveis, é independente dos desfechos potenciais:*

$$(Y_i^{(1)}, Y_i^{(0)}) \perp W_i \mid X_i = x \ , \ \forall x \textit{ no suporte de } X_i.$$

Suposição 3 (Suporte Comum)**.** *A premissa de suporte comum busca assegurar que toda a população tenha possibilidade de ter indivíduos tanto no grupo de controle como no de tratamento, dado um conjunto de características observadas pelas covariáveis X_i. Assim, garante-se a comparabilidade entre ambos os grupos em uma subpopulação e, consequentemente, a existência do efeito médio de tratamento nessa subpopulação.*

$$0 < \Pr(W_i = 1 \mid X_i = x) < 1 \ , \ \forall x \textit{ no suporte de } X_i.$$

Além das suposições acima elencadas, a consistência (suposição SUTVA 1) assegura que, se um indivíduo i é submetido a um tratamento $W_i = 1$, necessariamente observaremos $Y_i^{(1)}$; reciprocamente, se submetido a $W_i = 0$, observaremos $Y_i^{(0)}$. Dessa forma, o desfecho observado, representado pela equação (2.3), revela o valor do desfecho potencial correspondente ao tratamento efetivamente aplicado. Diante do conjunto de suposições

acima, é possível então encontrar o seguinte resultado de identificação para o efeito médio de tratamento, superando o problema fundamental da inferência causal:

$$\begin{aligned}
\tau_{\text{ATE}} &= \mathbb{E}\left[Y_i^{(1)} - Y_i^{(0)}\right] \\
&= \mathbb{E}\left[Y_i^{(1)} \;\middle|\; W_i = 1\right] - \mathbb{E}\left[Y_i^{(0)} \;\middle|\; W_i = 0\right] && \text{(Suposição 2 – Ignorabilidade)} \\
&= \mathbb{E}\left[Y_i \mid W_i = 1\right] - \mathbb{E}\left[Y_i \mid W_i = 0\right] && \text{(Suposição 1 SUTVA – Consistência)}
\end{aligned}$$

No caso particular de um desenho de estudo experimental (*randomized control trials* - RCT, em inglês), a suposição de ignorabilidade (Suposição 2) é plausível visto que a designação ao tratamento é aleatória por desenho. Além disso, são observáveis e estimáveis as quantidades $\mathbb{E}[Y_i \mid W_i = 1]$ e $\mathbb{E}[Y_i \mid W_i = 0]$, o que nos permite estimar o efeito médio de tratamento.

No tópico a seguir, iremos definir o estimando que nos permitirá estimar os efeitos causais heterogêneos: efeito tratamento condicional.

3.2 O efeito médio de tratamento condicional

Além do ATE, muitas vezes se está também interessado em eventuais *efeitos heterogêneos* ao longo de suporte de determinada covariável, em particular efeitos distintos condicionais a subgrupos estabelecidos pela combinação de covariáveis. Nesse sentido, introduz-se o efeito médio de tratamento condicional (*conditional average treatment effect* – CATE), estimando definido como segue.

Definição 3.2.1 (Efeito Médio de Tratamento Condicional – CATE)**.** Dada uma população de interesse $N = \{1, \ldots, n\}$, o **efeito médio de tratamento condicional** (*conditional average treatment effect* – CATE) é dado por:

$$\tau_{\text{CATE}}(x) = \mathbb{E}\left[Y_i^{(1)} - Y_i^{(0)} \middle| \; X_i = x\right]. \tag{3.2}$$

As suposições já estabelecidas para identificabilidade e estimação do ATE também devem ser consideradas para o CATE, agora em uma versão condicional ao vetor de cova-

riáveis: ignorabilidade (Suposição 2), suporte comum (Suposição 3) e SUTVA consistência (Suposição 1).

$$\begin{aligned}
\tau_{\text{CATE}}(x) &= \mathbb{E}\left[Y_i^{(1)} - Y_i^{(0)} \;\middle|\; X_i = x\right] \\
&= \mathbb{E}\left[Y_i^{(1)} \;\middle|\; W_i = 1, X_i = x\right] \\
&\quad - \mathbb{E}\left[Y_i^{(0)} \;\middle|\; W_i = 0, X_i = x\right] && \text{(Suposição 2 – Ignorabilidade)} \\
&= \mathbb{E}\left[Y \mid W_i = 1, X_i = x\right] \\
&\quad - \mathbb{E}\left[Y \mid W_i = 0, X_i = x\right] && \text{(Suposição 1 SUTVA – Consistência)} \\
&= \mu^{(1)}(x) - \mu^{(0)}(x),
\end{aligned}$$

onde

$$\mu^{(w)}(x) = \mathbb{E}[Y_i \mid W_i = w, X_i = x], \quad w \in \{0, 1\},\ x \in \text{suporte}(X). \tag{3.3}$$

Pode-se identificar o CATE, equação (3.2), assumindo ignorabilidade (Suposição 2), a partir da probabilidade inversa ponderada (*Inverse probability weighting* – IPW). Primeiramente, definimos a probabilidade de uma unidade i ser designada ao tratamento W via

$$e(x) = \Pr(W_i = 1 | X_i = x) = \mathbb{E}(W_i | X_i = x), \tag{3.4}$$

dita o *escore de propensão*. Assim, dado o escore de propensão e aplicando a propriedade das expectativas iteradas, o CATE pode ser reescrito da seguinte forma:

$$\tau_{\text{CATE}}(x) = \mathbb{E}\left[\frac{W_i Y_i}{e(x)} - \frac{(1 - W_i) Y_i}{1 - e(x)} \;\middle|\; X_i = x\right]. \tag{3.5}$$

A demonstração detalhada da identidade acima encontra-se no Apêndice A, seguindo Imbens and Wooldridge (2009).

No capítulo a seguir, apresentamos alguns métodos de aprendizado de máquina que buscam estimar o CATE e servirão de referência para comparação realizada no exercício de simulação.

4 Métodos de Estimação de Efeitos Tratamento Heterogêneos

Quando estamos interessados em estimar a forma funcional de uma função de regressão $\mu(x) = \mathbb{E}[Y \mid X = x]$, tanto os métodos de regressão tradicionais (mínimos quadrados ordinários, máxima verossimilhança, entre outros) quanto os métodos de *machine learning* (árvores de decisão, *bagging*, *support vector regression*, entre outros) estabelecem maneiras de se otimizar uma função perda do ajuste de uma amostra aleatória $\iota = (Y, X)$.

O estimando $\tau_{\text{CATE}}(\cdot)$ definido em (3.2) também representa uma forma funcional. Todavia, visto que no contexto de inferência causal somente mensuramos *um* dentre os possíveis desfechos potencias (para cada unidade), a estimação do CATE exige que sejam estabelecidas estratégias para identificação do contrafactual, seja através da imposição de suposições, seja via estimação local adaptativa de contrafactuais. Considerando-se que o escopo desse estudo se restringe à aplicação de técnicas de aprendizado de máquina para estimação dos efeitos causais heterogêneos em um contexto de maior dimensionalidade e de modelos com estrutura funcional mais flexíveis (não linearidades, termos de interação, descontinuidades, etc.) foram escolhidos os seguintes métodos de estimação que serão descritos no presente Capítulo 4: IPW em dois estágios (Abrevaya et al., 2015; Zhou and Zhu, 2021) e sua extensão duplamente robusta AIPW (*Augmented Inverse Probability Weighting*) (Glynn and Quinn, 2010); Florestas Causais (Wager and Athey, 2018; Athey et al., 2019); e *Debiased Machine Learning* (Chernozhukov et al., 2022).

4.1 IPW em dois estágios

No final do capítulo anterior vimos que o IPW, sob as suposições de ignorabilidade (Suposição 2), suporte comum (Suposição 3) e consistência (Suposição 1), permite identificar o estimando CATE, equação (3.2). Nesse sentido, a versão amostral para estimar o CATE pelo método IPW é dada pelo estimador em dois estágios:

$$\hat{\tau}_{\text{IPW}}(x) = \widehat{\mathbb{E}}\left[\frac{WY}{\hat{e}(x)} - \frac{(1-W)Y}{1-\hat{e}(x)} \;\middle|\; X = x\right]. \tag{4.1}$$

No primeiro estágio, se estima o escore de propensão, $e(x) = \Pr(W = 1 \mid X_i = x) = \mathbb{E}(W \mid X_i = x)$. Abrevaya et al. (2015) e Zhou and Zhu (2021) propõem que nesse estágio seja adotado o método não-paramétrico de regressão Kernel com certas condições de regularidade, ou algum modelo paramétrico como regressão logito ou probito,

por exemplo. Denotamos o estimador obtido por $\hat{e}(\cdot)$. No segundo estágio, estima-se um modelo de regressão utilizando como resposta as variáveis transformadas

$$\frac{W_i Y_i}{\hat{e}(X_i)} - \frac{(1 - W_i)Y_i}{1 - \hat{e}(X_i)}, \quad i \in N,$$

com vetor de covariáveis X_i, $i \in N$. Nesse particular, Abrevaya et al. (2015) e Zhou and Zhu (2021) propõem o método não-paramétrico de regressão Kernel. A ideia básica da transformação da variável desfecho é reponderar as observações do grupo de controle e de tratamento de acordo com a probabilidade de designação de tratamento, obtendo assim certa similaridade de características entre os grupos (medido pelas covariáveis) emulando um cenário de aleatoriedade.

Como exemplo, digamos que estamos interessados em medir o efeito médio condicional de tratamento de uma política de orientação nutricional para indivíduos mais velhos ($x_{\text{idade}} > 60$). Dentro dessa faixa populacional amostrada, verifica-se que pessoas com maior nível educacional medido em anos de estudo ($\mathbf{x} = [x_{\text{idade}} > 60; x_{\text{educ}} > 8]$) tem maior propensão ao tratamento. Ou seja, encontramos escores de propensão ao tratamento nutricional maiores em função da variável nível educacional. Isso implica uma situação de sobrerrepresentação de idosos com maior nível educacional na amostra do grupo de tratamento e superrepresentação de idosos com menor nível educacional no grupo de controle. Dessa forma, a transformação proposta pelo IPW para balancear essa sobrerrepresentação dará maior peso aos indivíduos menos propensos ao tratamento, pessoas idosas com menor nível educacional, e que foram tratados no cálculo do desfecho médio do grupo de tratamento – $W_i Y_i / \hat{e}(X_i)$, quase não afetando o peso dos indivíduos com alta propensão, idosos com maior nível educacional, e que foram tratados. Ao mesmo tempo, para o cálculo do desfecho médio do grupo de controle, $(1 - W_i)Y_i / \{1 - \hat{e}(X_i)\}$, se busca ponderar mais indivíduos com alta propensão que estarão subrepresentado no grupo de controle (não tratados). Assim, o IPW se propõe a melhor balancear as amostras dos grupos tratamento e controle, levando em consideração as propensões ao tratamento para o estimar o CATE. A Figura 1 ilustra essa transformação. No gráfico, temos no eixo x o escore de propensão e no eixo y a quantidade de indivíduos no grupo de tratamento, em *verde escuro*, e no grupo de controle, em *azul escuro*. A reponderação proporcionada pelo IPW é representada em *verde claro* para o grupo de tratamento, onde se dá maior peso para os indivíduos com menor propensão ao tratamento e que foram tratados (no exemplo, idosos com menor nível educacional tratados) e em *azul claro* para o grupo de

controle, onde se dá maior peso para os indivíduos com maior propensão ao tratamento e que não foram tratados (no exemplo, idosos com maior nível educacional não tratados). Dessa forma, temos um melhor balanceamento da amostra dos grupos de tratamento e controle ao longo dos níveis de escore de propensão.

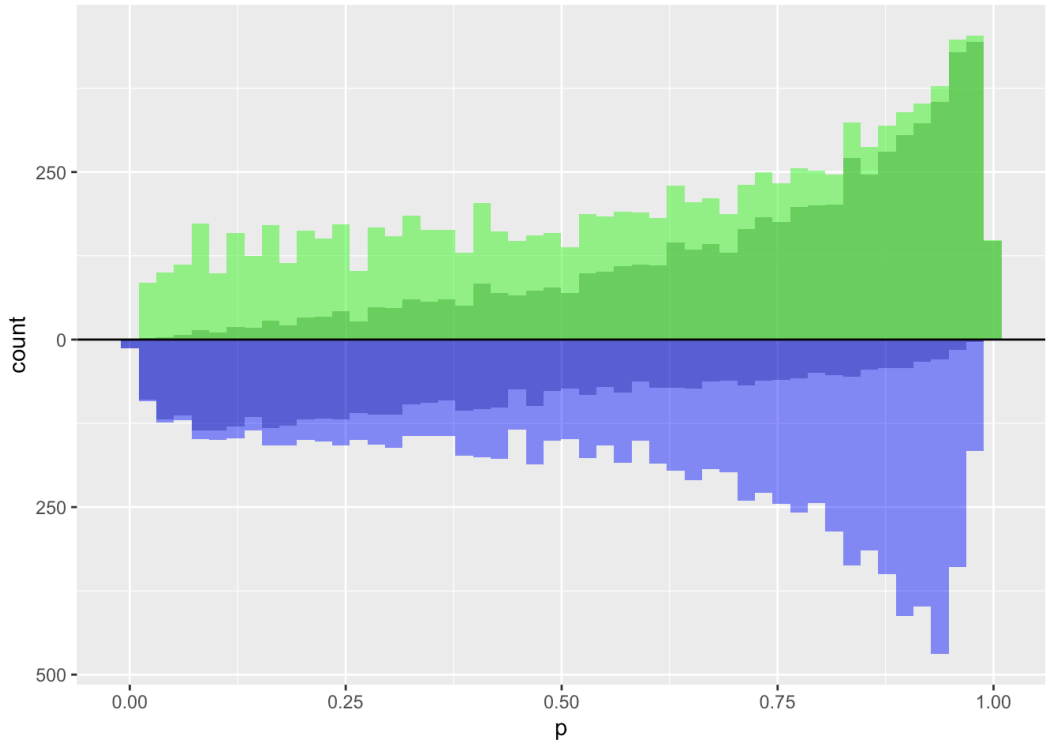

Figura 1 – Intuição do IPW.

Fonte: Lucy D'Agostino McGowan blog.

No processo de balanceamento acima ilustrado, um ponto que merece atenção é o fato de que valores de escore de propensão próximos a 0 ou 1 amplificam os pesos de forma extrema, comprometendo a precisão da estimação nessa faixa e de certo modo inviabilizando, na prática, a suposição de suporte comum (suposição 3). Crump et al. (2009) propuseram uma regra de ponto de corte simétrico de observações fora do intervalo de escore de propensão $[\eta, 1-\eta]$, chegando a uma regra de bolso ótima para $\eta = 0.1$. Stürmer et al. (2010), por sua vez, propuseram uma regra assimétrica de corte de acordo com um percentil inferior do escore de propensão no grupo dos tratados e um percentil superior no grupo de controle, como forma de reduzir o viés do estimador. Adicionalmente, Hirano et al. (2003) sugerem a normalização dos pesos, enquanto Liao and Rohde (2022) se inspiram em uma transformação de Rao-Blackwell Casella and Robert (1996) com o objetivo de reduzir a variância desse estimador.

Apesar das limitações mencionadas, por se tratar de um método clássico e bastante intuitivo, vamos adotá-lo no capítulo de simulação, estimando os dois estágios por meio de aprendizado de máquina (Floresta Aleatória).

Uma extensão do método, conhecido por AIPW (*Augmented Inverse Probability Weighting*), confere a propriedade de dupla robustez ao estimador e será apresentado a seguir.

4.1.1 IPW Aumentado – AIPW

Como comentado acima, o IPW é sensível à especificação do escore de propensão ($e(x) = \mathbb{E}\left[W_i = 1 \mid X_i = x\right]$). Nesse sentido, o *augmented IPW* – AIPW (Robins et al., 1994; Robins, 2000; Scharfstein et al., 1999) é visto como uma alternativa para superar essa limitação. Esse estimador é *duplamente* robusto, ou seja, permanece consistente mesmo em caso de má especificação do modelo de escore de propensão ou do modelo de desfechos do grupo dos tratados e dos não tratados, $\mu^{(w)}(x) = \mathbb{E}\left[Y_i \;\middle|\; W_i = w, X_i = x\right]$.

O AIPW é uma mescla do estimador IPW visto acima com o *meta-learner* S-learner, $\hat{\tau}_S(X) = \mathbb{E}\left[\hat{\mu}^{(1)}(x) - \hat{\mu}^{(0)}(x) \;\middle|\; X = x\right]$. O S-learner basicamente é obtido a partir da regressão $Y_i = \mu(X_i, W_i) + \epsilon_i$ e posterior estimação dos termos $Y_i^1 = \hat{\mu}(X_i, W_i = 1) = \hat{\mu}^{(1)}(x)$ e $Y_i^0 = \hat{\mu}(X_i, W_i = 0) = \hat{\mu}^{(0)}(x)$, resultando em $\hat{\tau}_S(X) = \hat{\mu}^{(1)}(x) - \hat{\mu}^{(0)}(x)$ (Jacob, 2021). Somente foi necessário definir o S-learner para descrever o AIPW abaixo especificado.

$$\hat{\tau}_{\text{AIPW}}(x) = \widehat{\mathbb{E}}\left[\hat{\mu}^{(1)}(x) - \hat{\mu}^{(0)}(x) + \frac{W\left(Y - \hat{\mu}^{(1)}(x)\right)}{\hat{e}(x)} - \frac{(1-W)\left(Y - \hat{\mu}^{(0)}(x)\right)}{1-\hat{e}(x)} \;\middle|\; X = x\right]. \tag{4.2}$$

Assim como no IPW, no primeiro estágio, precisamos estimar o modelo de escore de propensão, $\hat{e}(x)$, e o modelo de desfechos do grupo dos tratados e dos não tratados, $\hat{\mu}^{(w)}$. Enquanto no segundo estágio, se adota a versão transformada do desfecho plugando os resultados estimados no primeiro estágio para estimar um modelo de regressão condicionado ao vetor de covariáveis $\mathbf{X}$. Segundo Glynn and Quinn (2010), esse estimador exibe uma distribuição assintoticamente normal, e erros padrão válidos em amostras grandes. Por outro lado, o AIPW pode ter limitações em amostras pequenas. Se o escore de propensão estimado for altamente variável, o próprio estimador será variável também nesse cenário (Kang and Schafer, 2007). Por fim, utilizando simulação de Monte Carlo em vários cenários (grau de confundimento baixo, moderado e alto do modelo de designação de

tratamento, relação linear e não linear entre desfecho e covariáveis) e comparando com os estimadores IPW, *Matching* e Regressão, Glynn and Quinn (2010) demonstra que o AIPW teve resultados dramaticamente melhores em casos de má especificação, corroborando as vantagens da propriedade de dupla robustez.

No apêndice B demonstramos a propriedade de dupla robustez do estimador AIPW.

4.2 Árvores de decisão, Árvores causais e Florestas causais

Nessa seção descreveremos o método de estimação do CATE via Florestas causais, que é visto pela literatura como um dos métodos mais flexíveis e que será objeto de comparação com outros métodos igualmente flexíveis para estimar efeitos causais heterogêneos sob o contexto de maior dimensionalidade proposta nesta dissertação.

Antes de adentrar no método, é importante apresentar uma revisão de Árvores de decisão, de Florestas aleatórias e de Árvores causais para um melhor entendimento das Florestas causais.

4.2.1 Árvores de decisão e Florestas aleatórias

Compreender árvores de decisão (Breiman et al., 1984) é o primeiro passo para entendermos as adaptações realizadas nas árvores causais e nas florestas causais, métodos estes focados em estimar os parâmetros de efeitos causais em contraposição ao primeiro, centrado na predição de modo flexível da média condicional da variável desfecho Y – $\mu(X) = \mathbb{E}(Y|X)$.

Árvore de decisão é um método algorítmico de estimação em que se promove a partição recursiva do espaço de covariáveis ($\mathcal{X}$) baseado na otimização de uma função de perda. A partir de uma árvore inicial Π_0 contemplando todo o espaço de covariáveis de uma amostra $\mathcal{S}$, se particiona a árvore em duas folhas ou subárvores sob um critério de minimização do erro quadrático médio (MSE) dos valores contidos nas subárvores para maximizar a acurácia geral da estimação, e assim sucessivamente até atingir um critério de parada,usualmente o tamanho da folha ou ainda um patamar de minimização do erro.

Num exemplo didático extraído de Hastie et al. (2009), em uma amostra com um espaço de covariáveis bidimensional, X_1 e X_2, de acordo com o algoritmo acima descrito, obtemos as regiões das partições e árvore finais (Figuras 2 e 3 abaixo, respectivamente).

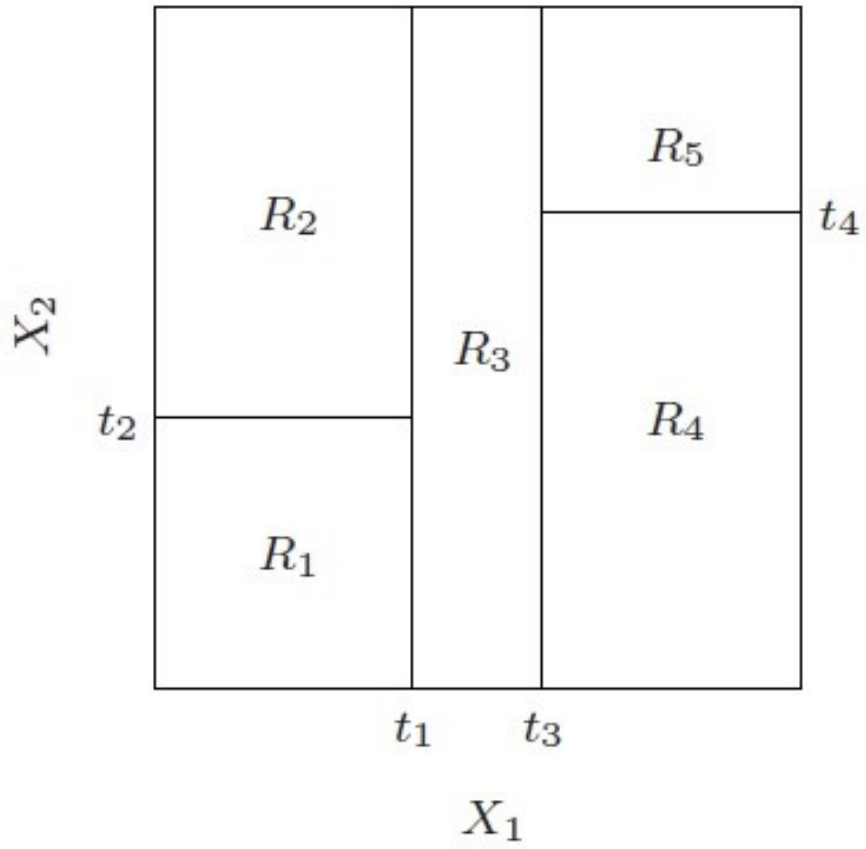

Figura 2 – Regiões de partição de um espaço de covariáveis (X_1,X_2).

Fonte: Adaptado de Hastie et al. (2009).

Em termos formais, dado uma árvore Π, definimos $\ell(x;\Pi)$ a folha $\ell \in \Pi$ tal que $x \in \ell$, ou seja, todas as observações cujos valores de x pertencem a partição estabelecida pela folha. Na população, a função da média condicional $\mu(x;\Pi)$ a uma folha pode ser expressa como:

$$\mu(x;\Pi) \equiv \mathbb{E}[Y_i|X_i \in \ell(x;\Pi)] = \mathbb{E}[\mu(X_i)|X_i \in \ell(x;\Pi)]. \tag{4.3}$$

Em termos amostrais, dada uma amostra $\mathcal{S}$ proveniente da árvore Π e sendo $\#(i \in \mathcal{S} : X_i \in \ell(x;\Pi))$ o número de observações dentro da folha $\ell(x;\Pi)$ que contém um x de interesse, um estimador da media condicional será:

$$\hat{\mu}(x;\mathcal{S},\Pi) = \frac{1}{\#(i \in \mathcal{S} : X_i \in \ell(x;\Pi))} \sum_{i \in \mathcal{S}: X_i \in \ell(x;\Pi)} Y_i. \tag{4.4}$$

No exemplo da Figura 2, a folha resultante da partição na região R_1 conterá todas as observações em que $X_1 \leqslant t_1$ e $X_2 \leqslant t_2$. A estimativa da média condicional dos valores nessa região será a média amostral (equação 4.4) dos desfechos Y das observações que satisfazem $X_1 \leqslant t_1$ e $X_2 \leqslant t_2$.

A cada rodada do processo recursivo, a escolha da partição do espaço de covariáveis se dará (i) pela escolha *aleatória* de uma covariável e (ii) a escolha do ponto de partição dessa covariável que minimiza uma função perda. A função perda pode ser simplesmente o MSE ou ainda uma composição entre o MSE e um termo de penalização do tamanho da árvore (número de partições indicando profundidade e resultando em determinado tamanho das folhas finais).

O MSE da estimativa da média condicional da amostra $\mathcal{S}$ é definida conforme segue:

$$\text{MSE}_\mu(S, \Pi) \equiv \frac{1}{\#(S)} \sum_{i \in S} (Y_i - \hat{\mu}(X_i; S, \Pi))^2. \tag{4.5}$$

Retomando o exemplo da Figura 3, vale ressaltar que o MSE geral é um somatório dos erros quadrados verificados em cada uma das folhas finais $(R_1, R_2, R_3, R_4, R_5)$ que

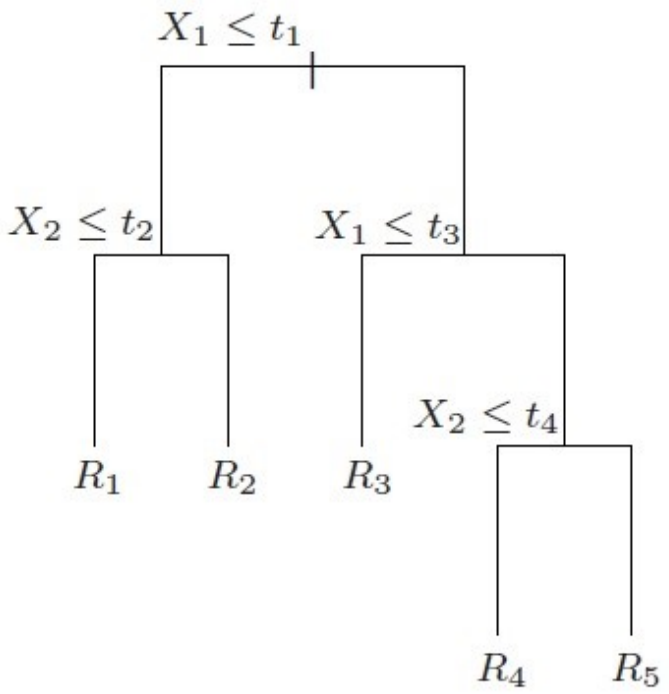

Figura 3 – Árvore de decisão particionada no espaço de covariáveis (X_1,X_2).

Fonte: Adaptado de Hastie et al. (2009).

nesse caso foram obtidos por não terem sido encontradas mais partições que satisfazem os critérios: redução do MSE ou outra função perda, e atingimento de critério de parada.

Já uma função perda feita por uma composição, proposta por Hastie et al. (2009), leva em conta a complexidade, sendo $|\Pi|$ o número de folhas finais e α um parâmetro de ajuste que balanceia o tamanho da árvore e a bondade do ajuste:

$$C_\alpha(\Pi) = \text{MSE}_\mu(S, \Pi) + \alpha|\Pi|. \tag{4.6}$$

A estimação do parâmetro α pode ser realizada por validação cruzada. Valores grandes do parâmetro α, significa que estamos impondo um custo maior de aumentar o tamanho da árvore em cada recursão para decidir pelo particionamento. Isso resultará em árvores menores e com maior tamanho das suas respectivas folhas. O inverso acontece com α menores.

De acordo com Hastie et al. (2009), o tamanho ótimo da árvore pode ser determinado adaptativamente de acordo com os dados (particionamento conforme atingimento de determinado patamar de redução do MSE), mas a melhor estratégia é estabelecer um critério de parada conforme um tamanho mínimo da folha.

Uma conclusão em relação ao tamanho de uma árvore e o *tradeoff* viés-variância que está no cerne de qualquer método de estimação, é que árvores de decisão muito grandes podem significar uma situação de sobreajuste (baixo viés e alta variância), enquanto árvores pequenas podem subajustar a estrutura dos dados que buscamos modelar.

Por fim, vale mencionar duas limitações do método de árvores de decisão. Primeiro, instabilidade, visto que pequenas diferenças nos dados podem representar configurações muito diferentes de árvores, com sequência de partições distintas. Em segundo lugar, a falta de suavização da função estimada da média condicional, $\hat{\mu}(x; \Pi)$. Basta pensarmos no exemplo das Figuras 2 e 3, onde o espaço das regiões R_1 $(X_1 \leqslant t_1; X_2 \leqslant t_2)$ e R_2 $(X_1 \leqslant t_1; X_2 > t_2)$ podem ter estimativas muito distintas, por exemplo, $\hat{\mu}(x \in R_1) = 10$ e $\hat{\mu}(x \in R_2) = 20$ mesmo nos pontos próximos entre eles, o que ilustra quebras de continuidade da forma funcional que desejamos estimar.

Métodos como *bagging* (Breiman, 1996) e florestas aleatórias ou *random forest* (Breiman, 2001) buscam reduzir a instabilidade e a falta de suavização características das árvores de decisão.

O *bagging* trata da média de um número de estimativas de árvores de decisão B via amostras *bootstrap*. O conjunto dessas amostras de árvores será *i.d.*, identicamente

distribuídas e *não* necessariamente independente, o que resulta em uma estimativa com esperança da média de B árvores semelhante a estimativa de cada árvore, o que não afeta o viés mas contribui para a redução da variância. De fato, assumindo que cada árvore tem variância σ^2 e uma correlação uma a uma de ρ, a variância da média de B árvores será (Hastie et al., 2009):

$$\rho\sigma^2 + \frac{1-\rho}{B}\sigma^2. \tag{4.7}$$

Denota-se que para $B \to \infty$, o segundo termo tende a zero, mantendo-se o primeiro, resultando em uma variância menor que o uso da estimação de uma árvore somente.

Já a floresta aleatória, em comparação ao *bagging*, incorpora a aleatoriedade na escolha da covariável que será particionada (escolha aleatória entre m variáveis, parâmetro de ajuste, de um total de p variáveis na amostra). Esse mecanismo torna as B árvores mais próximos de amostras *i.i.d.*, independente e identicamente distribuídas, o que permite a redução da correlação do primeiro termo da equação 4.7, além da redução do segundo termo para valores maiores de B. Para m menores aumenta a chance de independência e correlação menores. Segundo Hastie et al. (2009), quando a parcela de variáveis importantes para o melhor ajuste forem baixas, um m pequeno resultará em um ajuste ruim do modelo.

A estimação da média condicional por meio da floresta aleatória, seguindo a nomenclatura acima para árvores de decisão (equação 4.4) e indexando o conjunto de árvores Π_b em $b = 1, \ldots, B$ poderá ser expressa da seguinte forma:

$$\hat{\mu}^B(x) = \frac{1}{B}\sum_{b=1}^{B} \hat{\mu}(x; \Pi_b). \tag{4.8}$$

E para melhor sintetizar o método de floresta aleatória, a tabela a seguir descreve os passos do algoritmo, Tabela Algoritmo 1.

Essa subseção procurou ilustrar a dinâmica de estimação do método de árvores de decisão, pontuando suas limitações e vantagens, e especialmente o método de florestas aleatórias, que mantém a flexibilidade das árvores de decisão com ganhos quanto à acurácia e variância do estimador em contexto de predição.

No entanto, tais métodos foram pensados para predição e não para estimação imediata nos contextos de inferência causal onde os estimandos de interesse, seja ATE (3.1) ou CATE (3.2), exigem estratégias de identificação dos desfechos potenciais.

Via florestas aleatórias, podemos estimar tanto o escore de propensão para obter o IPW, equação (3.5), quanto a média condicional dos desfechos do grupo de tratados e

Algoritmo 1 Floresta Aleatória

Seja uma amostra $\mathcal{S}$

1. **procedimento** RANDOMFOREST(amostra $\mathcal{S}$, ponto de teste x)
2. **para** b = 1 até o número total de árvores B **faça**
3. Sortear uma amostra *bootstrap* de tamanho N de $\mathcal{S}$.
4. cresça a árvore P_b particionando as covariáveis conforme o critério de minimização da função perda MSE e até atingir o critério de parada do tamanho da folha (n_{min}) com a seguinte recursão:
5. selecione aleatoriamente m variáveis do total de p variáveis
6. escolha a melhor variável/ponto de partição entre os m, conforme o critério de minimização do MSE
7. particione a árvore em duas subárvores
8. **output** será $\hat{\mu}(x; \Pi_b)$ ▷ ver (4.4)
9. **output** após **loop** será $\hat{\mu}^B(x) = \frac{1}{B}\sum_{b=1}^{B} \hat{\mu}(x; \Pi_b)$

Fonte: Adaptado de Hastie et al. (2009).

não tratados para obter o AIPW, equação (4.2). Todavia, é notório que a função perda (MSE) a ser otimizada nos métodos de árvores de decisão e de florestas aleatórias não é identificável, visto que não encontramos $\tau_i = Y_i^{(1)} - Y_i^{(0)}$. Em particular, vemos que, em uma adaptação de (4.5), o MSE seria $\text{MSE}_\tau(S, \Pi) = \frac{1}{\#(S)}\sum_{i \in S}(\tau_i - \hat{\tau}(X_i; S, \Pi))^2$.

Por isso, nas próximas subseções apresentaremos as árvores causais e florestas causais que buscam estratégias para identificar o MSE e otimizar a heterogeneidade da função a ser estimada.

4.2.2 Árvores Causais

Como exposto acima, os métodos de estimação via árvores ou florestas aleatórias são bastante flexíveis e eficientes no contexto de predição. Entretanto, como mencionado no final da última seção, no contexto de estimação de efeitos causais (ATE ou CATE), onde o problema fundamental da inferência causal está presente, o critério de partição MSE se torna **não** identificável.

Diante dessa questão, Athey and Imbens (2016) e Wager and Athey (2018) propuseram o método de árvores causais, incorporando uma versão ajustada do MSE que permita identificação e privilegia a otimização da heterogeneidade entre as partições e consequentemente a busca de efeitos heterogêneos. A proposta também trouxe uma regra

de divisão da amostra para permitir resultados não-enviesados e estabeleceu condições para estimações com distribuição assintótica normal para fins de inferência.

O ajuste do critério de partição MSE é feito a partir da adição do termo $\mathbb{E}[Y_i^2]$, que não afeta o processo de otimização (minimização) da função perda para obtermos a estimativa. Assim, dada uma árvore Π, uma amostra para estimação da média condicional S^{est} e uma amostra de teste S^{te} para obter a estimativa dessa média, amostras independentes, sendo $\#(S)$ o tamanho da amostra, o MSE adaptado para média condicional do desfecho (MSE_μ) fica definido como:

$$\text{MSE}_\mu(S^{te}, S^{est}, \Pi) \equiv \frac{1}{\#(S^{te})} \sum_{i \in S^{te}} \left\{ (Y_i - \hat{\mu}(X_i; S^{est}, \Pi))^2 - Y_i^2 \right\}. \tag{4.9}$$

E, a partir da esperança do MSE_μ sobre diversas amostras de teste e de estimação independentes, obtemos o MSE ajustado esperado (EMSE) definido da seguinte forma:

$$\begin{aligned} \text{EMSE}(\Pi) \equiv & \mathbb{E}_{S^{te}, S^{est}}[\text{MSE}_\mu(S^{te}, S^{est}, \Pi)] = \\ & \mathbb{E}_{S^{te}, S^{est}, S^{tr}}[\text{MSE}_\mu(S^{te}, S^{est}, \pi(S^{tr})]. \end{aligned} \tag{4.10}$$

O uso de uma amostra para estimar a média condicional $\hat{\mu}$ e outra distinta (independente) para testar na função estimada é chamado por Athey and Imbens (2016) de critério *honesto*, enquanto o uso da mesma amostra para ambas finalidades é chamado de critério *adaptativo* pelos autores. O objetivo da estimação, nesse caso, é otimizar a função em (4.10) baseado em uma amostra de treino S^{tr} e conhecendo o tamanho da amostra de teste N^{te}, sendo que podemos expandir um termo para ilustrar a composição do estimador como segue:

$$\begin{aligned} \text{EMSE}_\mu(\Pi) = & \mathbb{E}\{\mathbb{E}_{(Y_i, X_i), S^{est}}[(Y_i - \mu(X_i; \Pi))^2 - Y_i^2] \\ & + \mathbb{E}_{X_i, S^{est}}[(\hat{\mu}(X_i; S^{est}, \Pi) - \mu(X_i; \Pi))^2]\} \\ = & - \mathbb{E}_{X_i}[\mu^2(X_i; \Pi)] + \mathbb{E}_{X_i, S_{est}}[\mathbb{V}(\hat{\mu}(X_i; S^{est}, \Pi))]. \end{aligned} \tag{4.11}$$

Veja que usamos a suposição que temos um estimador não enviesado onde:

$$\mathbb{E}_S[\hat{\mu}(x; \mathcal{S}, \Pi)] = \mu(x; \Pi). \tag{4.12}$$

A estimativa da variância da média condicional acima é determinada pela variância dentro de cada folha, $S^2_{\mathcal{S}^{tr}}(\ell)$, tal que:

$$\hat{\mathbb{V}}(\hat{\mu}(X_i; S^{est}, \Pi)) \equiv \frac{S^2_{\mathcal{S}^{tr}}(\ell(x; \Pi))}{N^{est}(\ell(x; \Pi))}.$$

Assumindo que a participação de cada folha ℓ, p_ℓ, seja aproximadamente iguais nas amostras de treino (S^{tr}) e de estimação (S^{est}), o estimador da variância pode ser aproximado por:

$$\mathbb{E}[\hat{\mathbb{V}}(\hat{\mu}(X_i; S^{est}, \Pi)|i \in \mathcal{S}^{te})] = \frac{1}{N^{est}} \sum_{\ell \in \Pi} S^2_{\mathcal{S}^{tr}}(\ell). \tag{4.13}$$

Já para estimar o primeiro termo de (4.11), $\mathbb{E}(\mu^2)$, podemos utilizar a fórmula da variância ($\mathbb{V}(X) = \mathbb{E}(X^2) - [\mathbb{E}(X)]^2$), a estimativa da variância de μ em (4.13) e da média de μ em (4.12), resultando em:

$$\mathbb{E}[\mu^2(x; \Pi)] = \hat{\mu}^2(x; S^{tr}, \Pi) - \frac{S^2_{\mathcal{S}^{tr}}(\ell)}{N^{tr}(\ell)}. \tag{4.14}$$

Assim com base nos valores estimados em (4.14) e em (4.13) aplicados em (4.11), obtemos o estimador não enviesado para $\text{EMSE}_\mu(\Pi)$ pelo critério de amostra honesto:

$$\begin{aligned} -\widehat{\text{EMSE}}_\mu(S^{tr}, N^{est}, \Pi) \equiv & \frac{1}{N^{tr}} \sum_{i \in \mathcal{S}^{tr}} \hat{\mu}^2(X_i; \mathcal{S}^{tr}, \Pi) \\ & - \left(\frac{1}{N^{tr}} + \frac{1}{N^{est}} \right) \sum_{\ell \in \Pi} S^2_{\mathcal{S}^{tr}}(\ell(x; \Pi)) \end{aligned} \tag{4.15}$$

Já o estimador do MSE pelo critério adaptativo tradicional para o método de árvores de decisão resulta na seguinte equação:

$$-\widehat{\text{MSE}}_\mu(S^{tr}, S^{tr}, \Pi) = \frac{1}{N^{tr}} \sum_{i \in \mathcal{S}^{tr}} \hat{\mu}^2(X_i; \mathcal{S}^{tr}, \Pi). \tag{4.16}$$

Nota-se que o que diferencia o estimador do MSE pelo critério honesto é o fato de penalizar a variância intra-folha.

Para aplicar os resultados acima no contexto de inferência causal precisamos de definições adicionais, visto que além do desfecho e covariáveis incluímos uma variável de tratamento (Y_i^{obs}, X_i, W_i) para obter o estimando do efeito tratamento (definições apresentadas nas seções 3.1 e 3.2). Dada $\mathcal{S}_{trat}$ uma amostra de observações tratadas de tamanho N_{trat} e $\mathcal{S}_{ctrl}$ uma amostra do grupo controle de tamanho N_{ctrl}, o percentual de observações tratadas $p = {}^{N_{trat}}/_{N_{ctrl}}$ e uma árvore Π e a folha $\ell(x; \Pi)$[1], então, *especificamente para o caso de árvores e suas folhas*, definimos o desfecho esperado, $\mu(w, x, \Pi)$, o efeito causal, $\tau(x; \Pi)$, e o MSE (tal como a equação 4.9) da seguinte forma:

1. Desfecho Esperado:

$$\mu(w, x, \Pi) \equiv \mathbb{E}[Y_i^{(w)}|X_i \in \ell(x; \Pi)]. \tag{4.17}$$

[1] Lembrando que a folha $\ell \in \Pi$ tal que $x \in \ell$.

2. Efeito Causal Médio:

$$\tau(x, \Pi) \equiv \mathbb{E}[Y_i^{(1)} - Y_i^{(0)} | X_i \in \ell(x; \Pi)] = \mu(1, x, \Pi) - \mu(0, x, \Pi). \tag{4.18}$$

3. MSE efeito causal:

$$\text{MSE}_\tau(S^{te}, S^{est}, \Pi) \equiv \frac{1}{\#(S^{te})} \sum_{i \in S^{te}} \left\{ (\tau_i - \hat{\tau}(X_i; S^{est}, \Pi))^2 - \tau_i^2 \right\}. \tag{4.19}$$

Para superar a questão intrínseca da inferência causal de identificação utilizamos o fato a seguir. Ou seja, o efeito causal de uma observação i pertencente a uma determinada folha será estimado pela estimativa do efeito causal dentro de uma folha:

$$\mathbb{E}_S^{te}[\tau_i | i \in S^{te} : i \in \ell(x, \Pi)] = \mathbb{E}_S^{te}[\hat{\tau}(x; S^{te}, \Pi)]. \tag{4.20}$$

Vale ressaltar que a estimativa do efeito causal dentro de uma folha pode ser dada por:

$$\hat{\tau}(x) = \frac{\sum_{\{i: W_i = 1, X_i \in \ell(x, \Pi)\}} Y_i}{|\{i : W_i = 1, X_i \in \ell(x, \Pi)\}|} - \frac{\sum_{\{i: W_i = 0, X_i \in \ell(x, \Pi)\}} Y_i}{|\{i : W_i = 0, X_i \in \ell(x, \Pi)\}|} \tag{4.21}$$

Assim, expandindo MSE_τ não identificável (4.19) como fizemos em (4.11) para função perda para estimação de μ, chegamos a:

$$-\text{EMSE}_\tau(\Pi) = \mathbb{E}_{X_i}[\tau^2(X_i; \Pi)] - \mathbb{E}_{X_i, S_{est}}[\mathbb{V}(\hat{\tau}^2(X_i; S^{est}, \Pi))]. \tag{4.22}$$

cuja versão estimada, sob a abordagem de separação honesta da amostra, será dependente somente de S^{tr} e N^{est} tal como representado abaixo:

$$\begin{aligned} -\widehat{\text{EMSE}}_\tau(S^{tr}, N^{est}, \Pi) \equiv & \frac{1}{N^{tr}} \sum_{i \in \mathcal{S}^{tr}} \hat{\tau}^2(X_i; \mathcal{S}^{tr}, \Pi) \\ & - \left(\frac{1}{N^{tr}} + \frac{1}{N^{est}} \right) \sum_{\ell \in \Pi} \left(\frac{S^2_{\mathcal{S}^{tr}_{trat}}(\ell)}{p} + \frac{S^2_{\mathcal{S}^{tr}_{crtl}}(\ell)}{1-p} \right) \end{aligned} \tag{4.23}$$

Para uso da validação cruzada, somente alteramos a amostra aplicada no critério acima, ao invés da amostra de treino completa aplicamos amostra de treino conforme o número de *folds*, $\widehat{\text{EMSE}}_\tau(S^{tr,cv}, N^{est}, \Pi)$.

Podemos depreender que o uso da função perda estimada acima para partição de uma árvore buscará privilegiar maior heterogeneidade entre as folhas filho ou subárvores resultantes, dado pelo primeiro termo. Um exemplo somente ilustrativo, se em uma folha de tamanho 100 estimamos um efeito causal $\hat{\tau} = 10$, se encontrarmos uma partição no

espaço de covariáveis $\mathcal{X}$ de forma a subdividir a árvore em duas folhas de igual tamanho ($N_E = N_D = 50$) de forma que a folha à esquerda ℓ_E tenha um efeito causal estimado $\hat{\tau}_E = 11$ e à direita $\hat{\tau}_D = 9$, teremos um valor maior no primeiro termo de (4.23) e assim o algoritmo irá buscar a partição que maximiza esse critério MSE.

Por outro lado, o segundo termo penaliza partições que aumentam a variância intra-folha. Seguindo o exemplo, mesmo se a partição aumentar a heterogeneidade entre as folhas, o aumento da variância intra-folha resultante pode acarretar na não escolha dessa opção de particionamento da árvore.

O método honesto, com o estabelecimento de amostras independentes, elimina o viés causado no adaptativo em caso de existência de valores extremos espúrios que serão levados em conta para determinação da partição. Como esses valores permanecem também na estimação do efeito causal, o uso do método adaptativo implicaria em médias amostrais do efeito também extremos e menor taxa de cobertura dos intervalos de confiança.

Assim, conforme exposto acima, Athey and Imbens (2016) e Wager and Athey (2018) propuseram as árvores causais (*Causal Trees*, **CT**) a partir da adaptação do critério MSE utilizado para partição de árvores no contexto de predição para permitir uma partição de árvores que permita otimizar a diferença de efeitos causais entre folhas, tendo o cuidado de não ampliar a variância intra-folha de forma a preservar a precisão dos efeitos causais heterogêneos estimados. Ainda, para evitar o sobreajuste de se utilizar a mesma amostra para particionar a árvore e para estimar o efeito causal, propôs-se o método de separação em amostras independentes, denominado método honesto.

4.2.3 Florestas Causais

Assim como o método de florestas aleatórias é uma alternativa para árvores de decisão para reduzir a variância e suavizar a função a ser estimada, o método de florestas causais proposto por Wager and Athey (2018) é uma média de árvores causais (Athey and Imbens, 2016).

Para demonstrar consistência da floresta causal para o verdadeiro valor de $\tau(x)$, os autores assumem certas condições. As funções de média condicional $\mu^{(1)} = \mathbb{E}[Y^{(1)}|X = x]$ e $\mu^{(0)} = \mathbb{E}[Y^{(0)}|X = x]$ devem ser Lipschitz contínuas. Também se assume condição de suporte comum (Suposição 3), o que garante a existência de observações do grupo controle

Algoritmo 2 Floresta Causal

1. **procedimento** DOUBLE-SAMPLE TREES(amostra teste $\mathcal{S}^{te} = \mathcal{I}$, amostra estimação $\mathcal{S}^{est} = \mathcal{J}$ da tríade (Y_i, X_i, W_i), ponto de teste x)
2. **para** b = 1 até o número total de árvores B **faça**
3. Sortear uma amostra de tamanho s sem reposição de uma amostra de treino de tamanho n de $\mathcal{S}$, dividindo entre duas subamostras de tamanho $|\mathcal{I}| = \lfloor s/2 \rfloor$ e $|\mathcal{J}| = \lceil s/2 \rceil$.
4. cresça a árvore Π particionando as covariáveis conforme o critério de minimização da função perda MSE ajustado e até atingir o critério de parada do tamanho mínimo da folha (k).
   As partições podem ser escolhidas usando a amostra $\mathcal{J}$ e as observações X e W da amostra $\mathcal{I}$, sem usar Y dessa amostra.
   Cada folha da árvore deve conter k ou mais observações da amostra $\mathcal{I}$ de cada tratamento.
   ▷ ver árvore causal na Seção 4.2.2
5. estimar o efeito causal de tratamento dentro das folhas, usando a amostra $\mathcal{I}$.
   ▷ ver equação (4.21)
6. **output** será $\hat{\tau}(x; \mathcal{S}^{est} = \mathcal{J}, \Pi_b)$ ▷ ver (4.4)
7. Após **loop**, o **output** será $\hat{\tau}^B(x) = \frac{1}{B} \sum_{b=1}^{B} \hat{\tau}(x; \mathcal{S}^{est} = \mathcal{J}, \Pi_b)$

Fonte: Adaptado de Wager and Athey (2018).

e tratamento próxima ao ponto de teste x. Essa condição é necessária para métodos de estimação local. Sabemos que floresta causal pode ser visto como um método ao estilo do vizinho mais próximo (*k-nearest neighbor, KNN*) onde ao invés de definir k vizinhos, se define os vizinhos, observações dentro de uma folha, por critérios adaptativos (MSE ou MSE adaptativo, por exemplo). Ademais, se busca observações mais próximas possíveis dentro da folha de forma a emular uma situação semelhante a um experimento aleatório. Por fim, para fins de consistência se exige que as árvores de decisão satisfaçam a condição de amostra honesta.

A tabela a seguir descreve os passos do algoritmo do método de estimação via floresta causal, algoritmo 2.

O uso de florestas causais conforme proposto em Wager and Athey (2018) permite estimarmos o nosso estimando de interesse, CATE (3.2), mas a metodologia de partição das B árvores causais se torna computacionalmente intensiva. Nesse sentido, Athey et al. (2019) propuseram um algoritmo que denominaram florestas aleatórias generalizadas (*generalized random forests - GRF*), uma extensão da floresta aleatória proposta por Breiman (2001), para estimar de maneira eficiente e flexivel qualquer estimando $\theta(x)$

mediante o estabelecimento de condição de momento local e a proposição de uma função peso adaptativa inspirada na função kernel.

Basicamente, o método objetiva estimar de forma local $\theta(x)$ (que em nosso caso seria $\tau(x)$), ou seja, no entorno de um valor x de interesse, de acordo com a seguinte condição de momento, dado $(X_i, O_i) \in \mathcal{X} \times \mathcal{O}$:

$$\mathbb{E}\left[\psi_{\theta(x),\nu(x)}(O_i)|X_i = x\right] = 0 \qquad \forall x \in X \tag{4.24}$$

onde $\psi(x)$ é uma função *score* e $\nu(x)$ é um parâmetro auxiliar opcional.

Por exemplo, se estamos interessados em modelar a distribuição de O_i condicionado em X_i, $f_{\theta(x),\nu(x)}(\cdot)$, a função *score* a seguir $\psi_{\theta(x),\nu(x)}(O) = \nabla \log(f_{\theta(x),\nu(x)}(O_i))$ aplicada na condição de momento (4.24) permite identificar os parâmetros de interesse $(\theta(x), \nu(x))$ por estimação via máxima verossimilhança local (Athey et al., 2019).

Considerando que o método busca aplicar a condição de momento para fins de eficiência e ao mesmo tempo utiliza árvores para obter flexibilidade, e sabendo que árvores têm por característica terem baixo viés mas alta variância o que implica em viés na condição de momento mesmo em uma média de árvores (florestas aleatórias), os autores propuseram a inclusão de uma função de ponderação adaptativa da vizinhança $\alpha(x)$ em cada ponto de teste x para estimar a função $\theta(x)$ de forma a capturar toda a sua heterogeneidade.

O algoritmo pode ser sumarizado em duas etapas, (i) geração de *pseudo-desfechos* a partir da uma aproximação linear baseada em gradiente de uma função não-linear cujos parâmetros foram estimados na folha pai, inspirado no *gradient boosting* (Friedman, 2001), para determinar o ponto de partição das covariáveis e (ii) regressão desses *pseudo-desfechos* sobre as covariáveis de acordo com rotina padrão em árvores.

E como estamos interessado em uma regressão para estimação local, o GRF incorpora essa ponderação α como uma espécie de estimador de vizinho mais próximo adaptativo no contexto da floresta aleatória. O objetivo é semelhante ao da abordagem utilizando ponderação kernel que dá maior importância para as observações mais próximas do ponto de teste x mas que apresenta limitações de performance com mais dimensões do espaço de covariáveis $\mathcal{X}$ – "maldição da dimensionalidade", Robins and Ritov (1997). Assim o peso de cada observação i, $\alpha_i(x)$, é dado pela proporção de árvores em que i encontra-se na mesma folha que o ponto de teste x. Ou seja, dado uma floresta com número de B árvores Π_b $(b = 1, \ldots, B)$ e sendo $\ell_b(x)$ o conjunto de observações contidas na mesma

folha que se encontra x, o peso $\alpha_i(x)$, uma medida de vizinhança adaptativa de x baseado na floresta, pode ser expressa como:

$$\alpha_{bi}(x) = \frac{\mathbf{1}(\{X_i \in \ell_b(x)\})}{\#\ell_b(x)}\ , \qquad \alpha_i(x) = \frac{1}{B}\sum_{b=1}^{B}\alpha_{bi}(x), \tag{4.25}$$

onde $\mathbf{1}(E)$ denota a função indicadora do evento E.

Então o componente de ponderação (4.25) é levado em consideração para estimar as funções de interesse $\hat{\theta}(x)$ e $\hat{\nu}(x)$ através da solução de minimização da equação representada abaixo:

$$\left(\hat{\theta}(x), \hat{\nu}(x)\right) \in \arg\min_{\theta,\nu}\left\{\left\|\sum_{i=1}^{n}\alpha_i(x)\psi_{\theta,\nu}(O_i)\right\|_2\right\}. \tag{4.26}$$

Como o GRF é uma generalização do método de floresta aleatória, por exemplo, para estimar a média condicional de Y sobre X, $\mu(x) = \mathbb{E}[Y_i|X_i = x]$, utilizamos a função escore $\psi_{\mu(x)}(Y_i) = Y_i - \mu(x)$ que satisfaz a condição de momento (4.24), e aplicando em (4.26), temos que a solução de minimização de $\sum_{i=1}^{n}\frac{1}{B}\sum_{b=1}^{B}\alpha_{bi}(x)(Y_i - \hat{\mu}(x))$ será $\hat{\mu}(x) = \frac{1}{B}\sum_{b=1}^{B}\hat{\mu}_b(x)$, onde $\hat{\mu}_b(x) = \sum_{\{i:X_i\in\ell_b(x)\}}{}^{Y_i}/_{|\ell_b(x)|}$, que é o resultado da regressão padrão por árvores. Todavia, o GRF se aplica à estimação de outras funções de interesse, especialmente aquela função de interesse desse trabalho, o $\tau(x)$.

Feita a descrição geral do método de floresta aleatória generalizadas (GRF), proposto por Athey et al. (2019), é importante apresentar o passo a passo algoritmico do método.

Para a primeira etapa de geração de *pseudo-desfechos*, iniciamos subdividindo a amostra inicial $\mathcal{S}$ em uma amostra $\mathcal{J}$, de forma que buscamos $(\hat{\theta}_P, \hat{\nu}_P)$ a solução ótima da função escore sobre o nó inicial, nó pai $P \subseteq X$, representado abaixo:

$$\left(\hat{\theta}_P, \hat{\nu}_P\right)(\mathcal{J}) \in \arg\min_{\theta,\nu}\left\{\left\|\sum_{\{i\in\mathcal{J}:X_i\in P\}}\psi_{\theta,\nu}(O_i)\right\|_2\right\}. \tag{4.27}$$

E assim como o método de árvore padrão, definimos uma função perda, $err(C_1, C_2)$, onde a minimização determinará o ponto de partição do nó pai P em dois nós filhos (C_1, C_2), buscando recursivamente uma maior acurácia da estimação de $\theta(x)$. O $\theta(x)$ deve satisfazer a condição de momento (4.24) e tal situação em geral implica estimativas enviesadas da função perda tomando como referência os estimadores até então propostos como exemplo de Athey and Imbens (2016) explicado na Seção 4.2.2.

Diante dessa limitação de estimação da função perda, e buscando otimizar o processo de estimação e tornar computacionalmente mais eficiente a definição do ponto de partição, os autores propuseram a solução de $err(C_1, C_2)$ por meio de uma aproximação de $\tilde{\theta}_{C_1}$ e $\tilde{\theta}_{C_2}$ baseado em gradiente. Dessa forma, inicialmente, se propõe computar A_p como uma estimação consistente da esperança de função escore, $\nabla\,\mathbb{E}[\psi_{\hat{\theta}_P,\hat{\nu}_P}(O_i)|X_i \in P]$, da seguinte forma, quando a função escore é continuamente diferenciável:

$$A_P = \frac{1}{|\{i : X_i \in P\}|} \sum_{\{i:X_i \in P\}} \nabla\psi_{\hat{\theta}_P,\hat{\nu}_P}(O_i). \tag{4.28}$$

E em seguida a própria aproximação linear de $\tilde{\theta}_{C_1}$ e $\tilde{\theta}_{C_2}$:

$$\tilde{\theta}_C = \hat{\theta}_C - \frac{1}{|\{i : X_i \in C\}|} \sum_{\{i:X_i \in C\}} \xi^\top A_P^{-1}\psi_{\hat{\theta}_P,\hat{\nu}_P}(O_i). \tag{4.29}$$

onde $(\hat{\theta}_P, \hat{\nu}_P)$ são obtidos por (4.27) e ξ é o vetor para extrair as coordenadas de θ do vetor (θ_P, ν_P). Vale notar que $\xi^\top A_P^{-1}\psi_{\hat{\theta}_P,\hat{\nu}_P}(O_i)$ representa a função de influência da observação i na estimação $\hat{\theta}_P$ no nó pai.

A partir dos valores calculados em (4.28) e em (4.29), podemos obter o *pseudo-desfecho* ρ_i:

$$\rho_i = -\xi^\top A_P^{-1}\psi_{\hat{\theta}_P,\hat{\nu}_P}(O_i) \in \mathbb{R}. \tag{4.30}$$

E por meio dessa versão modificada do desfecho efetuamos a regressão padrão de árvore para determinar o ponto de partição do nó pai P em (C_1, C_2) mediante a maximização do função abaixo:

$$\tilde{\Delta}(C_1, C_2) = \sum_{j=1}^{2} \frac{1}{|\{i : X_i \in C_j\}|} \left(\sum_{\{i:X_i \in C_j\}} \rho_i \right)^2. \tag{4.31}$$

Assim, em cada nó filho, C_1 e C_2, se promove os mesmos cálculos de forma a obter os respectivos *pseudo-desfechos* (4.30) e subsequente regressão e assim sucessivamente de forma recursiva.

Como exemplo, para o caso simples de regressão por mínimos quadrados ordinários, temos $\psi_{\theta(x)}(Y) = Y - \theta(x)$, sendo que (4.30) não é modificado e igual a $\rho_i = Y_i - \overline{Y}_p$ ($\overline{Y}_p$ média de Y no nó pai), resultando em processo de otimização e partição de covariáveis semelhante ao de regressão de árvores em Breiman (2001).

Superada a descrição do método de partição de cada árvore individualmente, e considerando que as árvores tendem a fornecer estimativas com alta variância de resultados, precisamos estabelecer a agregação de uma floresta de árvores para uma estimação

Algoritmo 3 Generalized random forest (GRF) com honestidade e regra de subamostragem

Todos os parâmetros de ajuste são pré-especificados, incluindo o número de árvores B e a taxa de subamostragem s usado em SUBSAMPLE.

procedure GENERALIZEDRANDOMFOREST(conjunto de observações da amostra $\mathcal{S}$, ponto de teste x)
  vetor peso $\alpha \leftarrow$ ZEROS($|\mathcal{S}|$)
  for $b = 1$ até o número total de árvores B **do**
    conjunto de observações da amostra $\mathcal{I} \leftarrow$ SUBSAMPLE($\mathcal{S}$, s)
    conjunto de observações da amostra $\mathcal{J}_1, \mathcal{J}_2 \leftarrow$ SPLITSAMPLE($\mathcal{S}$)
    tree $\tau \leftarrow$ GRADIENTTREE($\mathcal{J}_1, \mathcal{X}$) ▷ Olhar Algoritmo 4.
    $\mathcal{N} \leftarrow$ NEIGHBORS($x, \diamond, \mathcal{J}_2$) ▷ Retorna os elementos de $\mathcal{J}_2$ que permanecem na mesma folha de x na árvore Π.
    for toda observação $e \in \mathcal{N}$ **do**
      $\alpha[e] += {}^{1}/_{|\mathcal{N}|}$
    end for
  end for
  Output: $\hat{\theta}(x)$, a solução de (4.26) com o peso ${}^{\alpha}/_{B}$
end procedure

Fonte: Adaptado de Athey et al. (2019).

consistente de θ, assumindo certas condições de regularidade e utilizando o problema de otimização (4.26) juntamente com a função peso (4.25). O algoritmo completo baseado em floresta está descrito nas tabelas de algoritmo 3 e 4. No artigo, os autores demonstram que, sob certas taxas de subamostragem sob método honesto (amostras independentes para partição e regressão da árvore), obtemos uma distribuição assintótica normal da estimativa $\hat{\theta}$.

Para finalizar essa seção, feito resumo geral do passo no método de floresta aleatória generalizado (Athey et al., 2019) na tabelas do algoritmo em 3 e 4, vamos exemplificar a estimação específica do CATE por meio desse método, acrescentando um mecanismo de centralização local das variáveis proposto pelos autores.

Considerando uma amostra $(X_i, Y_i, W_i) \in \mathcal{X} \times \mathbb{R} \times \mathbb{R}$, onde $W_i \in 0, 1$ é a variável binária de designação de tratamento, estamos interessados em estimar o modelo $Y_i = W_i \cdot b_i + \epsilon_i$, especificamente o parâmetro $\beta(x) = \mathbb{E}[b_i | X_i = x]$. Então no contexto do GRF, o objetivo é estimar $\theta(x) = \xi \cdot \beta(x)$, sendo $\xi \in \mathbb{R}^p$, um vetor com os confundidores.

Algoritmo 4 Árvore baseado em gradiente

Árvores baseadas em gradiente crescem como uma subrotina da floresta aleatória generalizada (GRF).

procedure GRADIENTTREE(conjunto de observações da amostra $\mathcal{J}$, suporte $\mathcal{X}$)
    nó $P_0 \leftarrow$ CREATENODE($\mathcal{J}$, $\mathcal{X}$)
    fila $\mathcal{Q} \leftarrow$ INITIALIZEQUEUE(P_0)
    while NOTNULL(nó $P \leftarrow$ POP($\mathcal{Q}$)) **do**
        $(\hat{\theta}_P, \hat{\nu}_P, A_p) \leftarrow$ SOLVEESTIMATINGEQUATION(P) ▷ Computa (4.27) e (4.28).
        vetor $R_P \leftarrow$ GETPSEUDOOUTCOMES($\hat{\theta}_P, \hat{\nu}_P, A_p$) ▷ Aplica (4.30) sobre o nó pai P.
        split $\Sigma \leftarrow$ MAKECARTSPLIT(P, R_P) ▷ Otimiza (4.31).
        if SPLITSUCCEEDED(Σ) **then**
            SETCHILDREN(P,GETLEFTCHILD(Σ), GETRIGHTCHILD(Σ))
            ADDTOQUEUE($\mathcal{Q}$,GETLEFTCHILD(Σ
            ADDTOQUEUE($\mathcal{Q}$,GETRIGHTCHILD(Σ))
        end if
        Output: árvore com nó raiz P_0 crescida baseada em gradiente.
    end while
end procedure

A função chamada INITIALIZEQUEUE inicializa uma fila com um elemento único; POP retorna e remove o elemento mais antigo da fila $\mathcal{Q}$, exceto se $\mathcal{Q}$ estiver vazio em que retornará nulo (NULL). MAKECARTSPLIT processa uma partição de regressão de árvore padrão sobre os pseudo-desfechos, e retorna dois nós filhos (C_1, C_2) ou uma mensagem de falha caso nenhuma partição seja possível.

Fonte: Adaptado de Athey et al. (2019).

Sob a suposição de ignorabilidade (2), podemos identificar $\beta(x)$. Já $\theta(x) = \xi \cdot \beta(x)$ é identificado ao satisfazer a condição de momento (4.24), sendo $\psi_{\beta(x),c(x)}(Y_i, W_i) = (Y_i - \beta(x) \cdot W_i - c(x))(\,1\; W_i^\top\,)^\top$. Dada a função de ponderação $\alpha_i(x)$ (4.25), o estimador $\hat{\theta}$ será:

$$\hat{\theta}(x) = \xi^\top \left(\sum_{i=1}^{n} \alpha_i(x)(W_i - \overline{W}_\alpha)^{\otimes 2} \right)^{-1} \sum_{i=1}^{n} \alpha_i(x)(W_i - \overline{W}_\alpha)(Y_i - \overline{Y}_\alpha). \tag{4.32}$$

onde $\overline{W}_\alpha = \sum \alpha_i(x) W_i$ e $\overline{Y}_\alpha = \sum \alpha_i(x) Y_i$, definindo $\nu^{\otimes 2} = \nu\nu^\top$.

Estimado $\hat{\theta}(x)$, a etapa seguinte do GRF propõe o cálculo do *pseudo-desfecho* (4.30) e de uma estimativa consistente do gradiente da função escore (4.28) abaixo, utilizados para crescer a árvore e definir o ponto de partição.

$$\begin{aligned} \rho_i &= -\xi^\top A_P^{-1}(W_i - \overline{W}_P)(Y_i - \overline{Y}_\alpha - (W_i - \overline{W}_P)\hat{\beta}_P) \\ A_P &= \frac{1}{|\{i : X_i \in P\}|} \sum_{\{i : X_i \in P\}} (W_i - \overline{W}_P)^{\otimes 2} \end{aligned} \tag{4.33}$$

onde $\overline{W}_P$ e $\overline{Y}_P$ são as médias obtidas no nó pai P e $\hat{\beta}_P$ é a estimativa da regressão de mínimo quadrados de coeficiente de Y_i sobre W_i.

Citando propostas similares em outros autores (Chernozhukov et al., 2018; Newey, 1994; Neyman, 1979; Robinson, 1988), Athey et al. (2019) propõem um processo de centralização local ou ainda uma ortogonalização das variáveis frente às covariáveis X. Define-se como desfechos centralizados $\tilde{Y}_i = Y_i - \hat{y}^{(-i)}(X_i)$ e $\tilde{W}_i = W_i - \hat{w}^{(-i)}(X_i)$, como os valores residualizados da estimação *leave-one-out* das respectivas esperanças condicionais com a retirada da própria observação i da regressão. Essa estimação pode ser realizada por *k-fold cross-fitting*, proposto por Chernozhukov et al. (2018).

De acordo com Athey et al. (2019), a centralização local de Y_i e de W_i antes de calcular a recursão da floresta aleatória pode tornar o estimador (4.32) robusto ao efeito de confundidores mesmo quando a função de ponderação $\alpha_i(x)$ não seja adequadamente concentrada ao redor de x.

Por fim, depreende-se que o método GRF proposto em Athey et al. (2019) avança em relação às florestas causais propostas em Wager and Athey (2018) ao incorporar a centralização local descrita acima, bem como inova ao desenvolver uma função perda baseada em gradiente (4.29), o que permite uma aproximação de função de perda não-linear e uma otimização na identificação do ponto ótimo. Ademais, esse método também traz uma proposta de ponderação da vizinhança de x de acordo com o conjunto de árvores, ao invés de uma média dos resultados estimados em cada árvore individualmente.

A próxima seção será destinada a mostrar o método de estimação dos efeitos heterogêneos conhecido por *Double/Debiased Machine Learning*, demonstrando suas vantagens e limitações, visto que o método será testado em comparação com o método de florestas causais em exercício de simulação.

4.3 Double/Debiased Machine Learning

O método *Double/Debiased Machine Learning* (ou somente DML), proposto inicialmente por Chernozhukov et al. (2018, 2022), estabelece um estimador do CATE a partir de um modelo estrutural composto de duas equações, cujos parâmetros auxiliares são estimados por métodos não paramétricos de aprendizado de maquina. O estimador do CATE, baseando-se nos estimadores auxiliares, deve satisfazer as condições de mo-

mento da função escore e as chamadas condições de ortogonalidade de Neyman (Neyman, 1959, 1979). Para situações de maior dimensionalidade, vários métodos de aprendizado de máquina podem ser utilizados para estimar os parâmetros auxiliares. Aliado ao processo de ortogonalização das equações auxiliares, o DML propõe o método de separação de amostras denominado *cross-fitting*, que objetiva corrigir o sobreajuste característico de um método de aprendizado de máquina. O *cross-fitting* foi brevemente descrito na seção anterior, mas será aprofundado na atual seção.

Inicialmente, considerando a variável de tratamento binária $W \in \{0, 1\}$, a função verdadeira do escore de propensão $e(x) = \mathbb{E}[W_i|X_i = x]$ (equação 3.4), a função verdadeira da esperança condicional do desfecho quando não tratado $\mu^{(0)}(X) = \mathbb{E}[Y_i|W_i = 0, X_i = x]$, a função efeito médio do tratamento condicional (CATE) $\tau(X)$, equação (3.2), e o vetor aleatório formado por $(Y, W, \mathbf{X})$, temos o conjunto de equações estruturais:

$$Y = \mu^{(0)}(X) + \tau(X)'W + \zeta, \qquad \mathbb{E}[\zeta|X, W] = 0, \tag{4.34}$$

$$W = e(X) + \nu, \qquad \mathbb{E}[\nu|X] = 0. \tag{4.35}$$

Como normalmente não conhecemos a forma das funções auxiliares $\mu^{(0)}(X)$ e $e(X)$, Chernozhukov et al. (2018) sugerem a utilização de aprendizado de máquina para obter estimativas dessas funções baseadas em modelos mais flexíveis e com grande poder preditivo. A estimativa dessas funções auxiliares serve de suporte para podermos estimar o CATE, nosso estimando de interesse, sob a suposição de ignorabilidade ou não-confundimento (equação 2) e suporte comum (equação 3).

Para fins de identificar o parâmetro de interesse $\tau(x)$, o DML propõe uma função escore $\psi(W; \tau, \eta)$ que satisfaça a condição de momento, $\mathbb{E}[\psi(W; \tau, \eta_0)] = 0$, e a condição de ortogonalidade, $\partial_\eta \mathbb{E}[\psi(W; \tau, \eta_0)][\eta - \eta_0] = 0$, sendo $\eta_0 = (\mu^{(0)}, e)$ os parâmetros auxiliares verdadeiros. A função de $Y(\eta)$ que satisfaz essas condições de ortogonalidade, de acordo com Chernozhukov et al. (2018) e Semenova and Chernozhukov (2021), pode ser representada da seguinte forma:

$$Y(\eta) := \mu^{(1)}(X) - \mu^{(0)}(X) + \frac{W(Y - \mu^{(1)}(X))}{e(X)} - \frac{(1 - W)(Y - \mu^{(0)}(X))}{1 - e(X)}, \tag{4.36}$$

onde $\mu^{(w)}(X) = \mathbb{E}[Y|W = w, X = x]$. A condição de momento guarda semelhança com o proposto no método de floresta aleatória generalizada (Athey et al., 2019) descrito na seção anterior. Já a condição de momento ortogonal se baseia em Robins and Rotnitzky (1995). Como a ortogonalização, conhecida como *Ortogonalidade de Neyman* (Neyman,

1959, 1979), envolve uma variação instantânea (derivada) dos parâmetros auxiliares, é possível depreender que esse fato torna o estimador do parâmetro τ robusto a variações (vieses) nas estimativas dos parâmetros auxiliares.

Uma outra condição do método de DML para fins de garantir a distribuição assintótica normal do estimador é que a taxa de convergência da estimação dos parâmetros auxiliares multiplicados seja inferior a $\sqrt{n}$, o que normalmente é obtidos pelos métodos de aprendizado de máquina.

Semenova et al. (2022) propuseram estabelecer uma aproximação da forma funcional do CATE – $\tau(x)$ por meio de uma combinação linear de um conjunto de transformações de X, $\kappa = f(X)$ (possíveis termos lineares, quadráticos, de ordem n, interações entre covariáveis, etc.), dado um vetor paramétrico β:

$$\tau(X) \approx \kappa'\beta. \tag{4.37}$$

Fundamentado no teorema de Frisch-Waugh-Lovell (Frisch and Waugh, 1933; Lovell, 1963) e seguindo a proposta original de Robinson (1988), os autores propuseram o DML a partir da obtenção do resíduo da regressão de Y sobre $\ell(X) = \mu^{(0)}(X) + K(X)'\beta \cdot e(X)$, definido por $\tilde{Y} = Y - \mathbb{E}[Y|X]$, ou seja a variação de Y não explicada pelo termo $\ell(X)$ equivalente aos termos da direita da equação (4.34) considerando a aproximação em (4.37). Utilizando também o resíduo da equação (4.35), $\tilde{W} = W - \mathbb{E}[W|X]$, ou seja a variação de W não explicada pelo mesmo conjunto de covariáveis no termo em $e(X)$, obtém-se a seguinte equação de regressão bivariada que permite estimar exclusivamente a forma funcional do CATE:

$$\tilde{Y} = \tilde{W}'\beta + \epsilon. \tag{4.38}$$

Antes de estimar a equação (4.38), para obtenção das variáveis ortogonalizadas $\tilde{Y}$ e $\tilde{W}$, o método DML propõe primeiro a estimação das funções auxiliares $\mathbb{E}[Y|X]$ e $\mathbb{E}[W|X]$ por métodos não-paramétricos de aprendizado de máquina (ML). Nesse contexto, para remover a tendência de viés de sobreajuste dos métodos de ML, Chernozhukov et al. (2018) propõe o uso da regra de partição da amostra chamado *cross-fitting* (CFit). O *cross-fitting* particiona uma amostra aleatória independente e identicamente distribuída (i.i.d.) de tamanho N em K partes $(I_k)_{k=1}^K$ de tamanho $n = \frac{N}{K}$. A amostra complementar a I_k é definida como $I_k^c := 1, \ldots, N \setminus I_k$. Assim, para cada $k \in 1, \ldots, K$, utilizamos a amostra complementar I_k^c para estimar as funções auxiliares acima mencionadas e a partição I_k para obtenção dos resíduos (ou os escores que satisfazem as condições de

momento) e estimar os parâmetros do CATE $\hat{\tau}_k(X)$ na equação (4.38) como a média $\hat{\tau}(X) = \frac{1}{k}\sum_{k=1}^{K}\hat{\tau}_k(X)$.

Segundo Chernozhukov et al. (2018), a escolha do valor de k pode ser limitada em amostras pequenas e valores maiores de k implicam em tamanho de amostra maiores em I_k^c, o que pode ser benéfico se supormos alta dimensionalidade nas funções auxiliares. Valores de k entre 4 e 5 funcionam melhor que $k = 2$ em diversos exemplos empíricos e simulações (Chernozhukov et al., 2018). Nie and Wager (2021) menciona que tipicamente se fixa $k = 5$ ou $k = 10$ para o processo de *cross-fitting*.

O valor do k não tem efeito sobre a distribuição assintótica da estimativa do CATE $\hat{\tau}(X)$, não obstante a taxa de convergência depende somente da complexidade de $\tau(X)$ e não das funções auxiliares do DML (Nie and Wager, 2021).

A depender da suposição de dimensionalidade estabelecida, podemos adotar estimadores distintos para estimar a função do CATE $\hat{\tau}(X)$ por meio da estimativa do vetor de coeficientes β na equação (4.38).

No caso de baixa dimensionalidade $d = dim(\beta) << N$ (vetor de β relativamente pequeno em relação ao tamanho da amostra N), podemos utilizar o tradicional mínimos quadrados ordinários (*ordinary least squares* - OLS), chamado de mínimos quadrados ortogonal (*orthogonal least squares* - OLS) por Semenova et al. (2022):

$$\hat{\beta}_{\text{OLS}} = \arg\min_{\beta\in\mathbb{R}^d}\frac{1}{N}\sum_{i=1}^{N}(\tilde{Y} - \tilde{W}'\beta)^2. \tag{4.39}$$

Já no contexto de maior dimensionalidade do CATE, que é o principal foco desse trabalho, onde $d = dim(\beta) >> N$ (vetor de β relativamente grande em relação ao tamanho da amostra N) mas assumindo a suposição de esparsidade, em que somente s coeficientes tem efeito diferente de zero $\|\beta\|_0 = s$, Semenova et al. (2022) sugere a utilização do chamado lasso ortogonal (*orthogonal lasso* - L). Dado $\lambda_\beta = C_\beta\sqrt{\log d/N}$ e C_β o parâmetro de penalização, chegaremos a estimativa a partir da otimização da função perda abaixo:

$$\hat{\beta}_{\text{L}} = \arg\min_{\beta\in\mathbb{R}^d}\frac{1}{N}\sum_{i=1}^{N}(\tilde{Y} - \tilde{W}'\beta)^2 + \lambda_\beta\sum_{j=1}^{d}|\beta_j|. \tag{4.40}$$

A escolha do parâmetro de penalização de uma forma adaptativa ao conjunto de dados da amostra são discutidos em Belloni et al. (2012, 2017) e Chernozhukov et al. (2016).

Embora essa proposta de estimação do CATE através do método DML e do lasso ortogonal seja uma alternativa interessante de aproximação do estimando em um contexto

de maior dimensão, estimadores com penalização tem como característica apresentarem estimativas enviesadas. Para fins de inferência, esse viés exige uma adaptação do estimador da variância para alcançarmos uma distribuição normal assintótica do estimador do CATE.

A proposta de adaptação da estimativa da variância feita por Semenova et al. (2022) se baseia em Van de Geer et al. (2014) e Zhang and Zhang (2014) da seguinte forma. Dada a matriz de covariância dos resíduos Q, com inversa Q^{-1}:

$$Q = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}\,\tilde{W}\tilde{W}'.$$

Com isso, temos a matriz de covariância amostral dos resíduos $\hat{Q}$ da forma:

$$\hat{Q} := \frac{1}{N}\sum_{i=1}^{N} \hat{\tilde{W}}\hat{\tilde{W}}'.$$

Propõe-se uma estimativa aproximada da inversa de $\hat{Q}$ a partir o problema de otimização abaixo:

$$\hat{\Omega} = \underset{\Omega\in\mathbb{R}^{d\times d}}{\arg\min}\|\Omega\|_1 : \|\hat{Q}\Omega - I_d\|_\infty \leqslant \lambda_Q.$$

em que

$$\lambda_Q := C_Q; \qquad \kappa_N := \sqrt{\log^3(d^2\log(N))\frac{\log N}{N}} \tag{4.41}$$

onde C_Q é uma constante do parâmetro de ajuste.

E uma forma de tornar uma matriz simétrica à aproximação da inversa $\hat{\Omega}$ (4.41) conforme Cai et al. (2011):

$$\hat{\Omega}^{CLIME} = (\hat{\omega}_{ij}^{CLIME}) \ , \qquad \hat{\omega}_{ij}^{CLIME} = \hat{\omega}_{ij}1_{|\hat{\omega}_{ij}|<|\hat{\omega}_{ji}|} + \hat{\omega}_{ji}1_{|\hat{\omega}_{ij}|>|\hat{\omega}_{ji}|}$$

Dessa forma, chegamos a adaptação necessária para obtermos o denominado lasso ortogonal "desviesado" (*debiased orthogonal lasso* - DL), como proposto por Semenova et al. (2022):

$$\hat{\beta}_{DL} := \hat{\beta}_L + \hat{\Omega}^{CLIME}\frac{1}{N}\sum_{i=1}^{N}\hat{\tilde{W}}(\hat{\tilde{Y}} - \hat{\tilde{W}}'\hat{\beta}_L). \tag{4.42}$$

E sob certas condições de regularidade, $\sqrt{N}(\hat{\beta}_{DL} - \beta)$ tem distribuição assintótica normal $N(0, \Sigma)$, sendo a matriz de covariância Σ definida por:

$$\Sigma := Q^{-1}\Gamma Q^{-1} = Q^{-1}\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}\,\hat{\tilde{W}}\hat{\tilde{W}}'\epsilon^2 Q^{-1}.$$

Podemos resumir o passo a passo do método DML de estimação do CATE, $\hat{\tau}(X)$, da seguinte forma:

1. estimar os resíduos $\tilde{Y}$ e $\tilde{W}$ por meio de qualquer método de aprendizado de máquina com regra de partição *cross-fitting* com k partes.
2. estimar a função CATE (4.37) na equação (4.38), via lasso ortogonal (4.40), para o caso de *maior dimensionalidade*, ou via mínimos quadrados ortogonal (4.39), para menores dimensões de $d = dim(\beta)$.
3. a inferência dos parâmetros da função CATE a partir de estimador com distribuição assintoticamente normal será dada pelo estimador *Debiased Orthogonal Lasso* (4.42) no contexto de maior dimensionalidade de $d = dim(\beta)$, enquanto para valores baixos de d poderá ser usado OLS.

Em terminando a descrição do método DML, proposto por Semenova et al. (2022), no próximo capítulo procederemos a comparação dos métodos de estimação dos efeitos heterogêneos IPW, IPW Aumentado (AIPW), Florestas Causais (CF) e *Double/Debiased Machine Learning* (DML) a partir de exercícios de simulação que permitam identificar as vantagens e as limitações dos métodos especialmente em um contexto de maior dimensionalidade.

5 Estudo de simulação

No presente capítulo realizamos um estudo de simulação para avaliar os métodos de estimação e inferência de efeitos causais heterogêneos, o CATE, conforme definido na Equação (3.2). O objetivo é comparar especialmente a qualidade de estimação dos métodos Floresta Causal (CF, Seção 4.2.3) e Double/Debiased Machine Learning (DML, Seção 4.3), aplicando florestas aleatórias para os dois estágios, frente a cenários de maior dimensionalidade e sob diferentes formas funcionais do CATE. Também foram incluídos os resultados para os métodos IPW (Seção 4.1) e AIPW (Seção 4.1.1), aplicando florestas aleatórias para os dois estágios, para servirem como *benchmarks*[1], dado que são métodos tradicionalmente consolidados na literatura de inferência causal (Kurz, 2022).

5.1 Configuração dos cenários

Foram realizadas simulações de Monte Carlo com $r = 500$ replicações e tamanho de amostra $n \in \{500, 2000, 5000\}$, de forma a verificar o comportamento dos estimadores em um contexto de amostras pequenas e avaliar sua performance a partir do aumento do tamanho das amostras. No apêndice D são apresentadas tabelas adicionais que complementam as análises deste capítulo.

No caso específico do método de floresta causal, foram utilizados como parâmetros de *tuning* um número de árvores $B = 400$, um tamanho mínimo do nó da árvore como uma proporção de 0.5% do tamanho da amostra ($\text{min\_node} = 0.005 \times n$), e um tamanho da amostra para partição honesta equivalente à metade do tamanho amostra $s = n/2$.

O processo gerador de dados leva em consideração as seguintes especificações de distribuição: para cada unidade amostral $i \in \{1, \ldots, n\}$, a equação estrutural é

$$Y_i = m(X_i) + (W_i - 0.5) \times \tau(X_i) + \sigma\epsilon_i,$$

onde,

$$X_i \sim U_d[0,1], \qquad W_i|X_i \sim \text{Bernoulli}\{e(X_i)\}, \qquad \epsilon_i \sim N(0,1).$$

em que $U_d[0,1]$ é a distribuição uniforme em $[0,1]^d$, $\tau(X)$ representa o CATE, $e(X)$ o escore de propensão, ϵ o termo de erro e $m(X)$ o efeito principal. Em todos os cenários, os vetores

[1] Métodos que servem de referência na simulação.

$(X_i, Y_i, W_i, \epsilon_i)_{i=1}^n$ são independentes e identicamente distribuídos, com ϵ_i independente de (X_i, W_i).

Quanto às formas funcionais, foram definidos as seguintes especificações, baseando-se em Wager and Athey (2018); Athey et al. (2019); Jacob (2021); Knaus et al. (2020); Nie and Wager (2021):

A. linear:

$$\begin{aligned} m(X) &= X_1 + 0.5X_2 + 0.4X_3, \\ e(X) &= \text{trim}_{0.1}\{\text{sen}(\pi X_1 X_2)\}, \\ \tau(X) &= 0.3X_1 + 0.6X_2 + 0.6X_3 + 2.2X_4, \end{aligned}$$

onde a função $\text{trim}_\eta(x) = \max\{\eta, \min(x, 1-\eta)\}$ estabelece pontos de corte mínimos e máximos em função de η para melhor assegurar a satisfação da suposição de suporte comum (suposição 3).

B. não-linear:

$$\begin{aligned} m(X) &= 0.2X_1 + X_1^2 + 0.5X_2X_3 + 0.4X_3 + 0.8X_3^2, \\ e(X) &= \text{trim}_{0.1}\{X_1 - 0.2X_2^2\}, \\ \tau(X) &= 0.3X_1 + 0.6X_2 + 0.6X_3 + 2.2X_4. \end{aligned}$$

C. picos e vales:

$$\begin{aligned} m(X) &= 0, \\ e(X) &= 0.5X_1 + 0.5X_2, \\ \tau(X) &= \zeta(X_1)\zeta(X_2)\zeta(X_3), \\ \zeta(X) &= 1 + \frac{1}{1 + e^{-20(x-\frac{1}{3})}}. \end{aligned}$$

D. descontinuidade:

$$\begin{aligned} m(X) &= 2 \times \mathbf{1}\{X_1 > 0.4\} + 0.3X_2, \\ e(X) &= \text{trim}_{0.1}\{0.7X_1 - 0.7X_2 + 0.7X_3\}, \\ \tau(X) &= 2 \times \mathbf{1}\{X_1 > 0.6\} + 1.5 \times \mathbf{1}\{X_2 > 0.6\} + 0.3X_3 - 0.7X_4. \end{aligned}$$

Para cada uma das formas funcionais, além de diferentes tamanhos de amostra, avaliamos os cenários conforme o número de covariáveis no processo gerador de dados

$d \in \{4, 10, 20\}$ com o objetivo de verificar as implicações de cenários com maior dimensionalidade sobre a estimação e inferência dos efeitos heterogêneos para cada um dos métodos utilizados nesse exercício de simulação. Além disso, em todos os cenários estudados o desvio-padrão do termo de erro, σ_ϵ, é igual a 3.

As Figuras 4 e 5 a seguir apresentam o comportamento da função CATE, $\tau(X)$, nas respectivas formas funcionais dos processos geradores linear (A), não-linear (B), picos e vales (C) e descontinuidade (D). Como o propósito é ilustrar os cenários, foram calculados os valores da função CATE ao longo de uma grade de $X_1 = [-1, 1]$, com $X_2 = X_3 = X_4 = 0.5$ constantes, e ao longo de uma grade de $X_2 = [-1, 1]$, com $X_1 = X_3 = X_4 = 0.5$ constantes, sob as especificações $n = 500, d = 4, \sigma_\epsilon = 3$. Vale notar que os Cenários A e B embora possuam função $\tau(X)$ lineares nas variáveis X_1 e X_2, se diferenciam quanto ao efeito principal $m(X)$, o que implica em desafios para estimação.

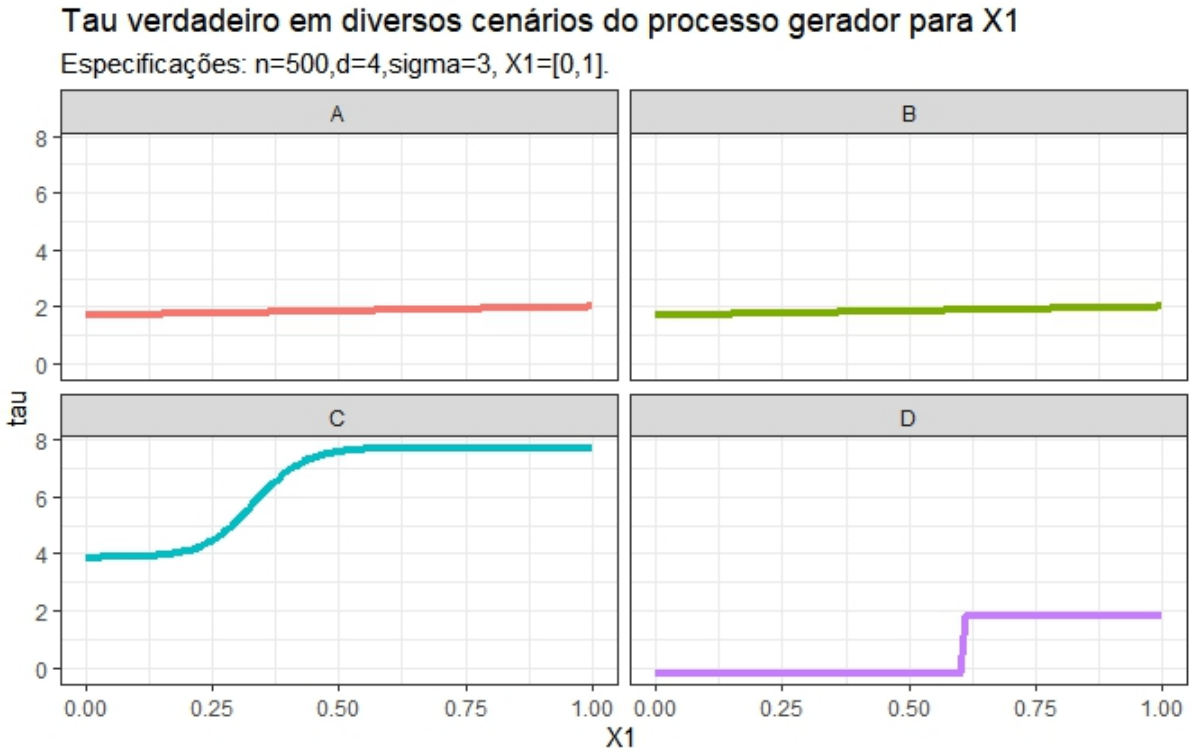

Figura 4 – Comportamento dos processos geradores na variável X_1.

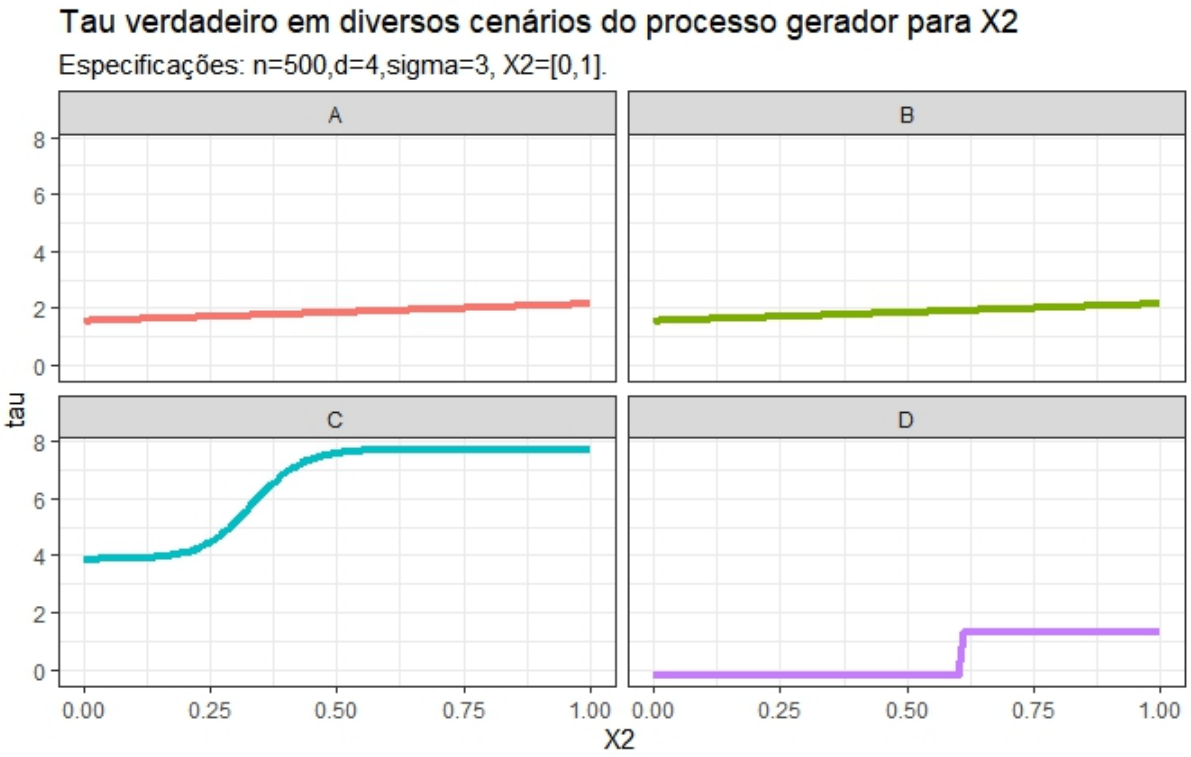

Figura 5 – Comportamento dos processos geradores na variável X_2.

5.2 Medidas de avaliação

A fim de analisar a qualidade dos estimadores quanto à acurácia e o viés da estimação e ainda para verificar a adequação dos intervalos de confiança produzidos, quando o método assim permitir, foram calculadas, respectivamente, as medidas de performance de estimadores (Knaus et al., 2020) erro quadrático médio, viés absoluto médio e taxa de cobertura (Walther and Moore, 2005). Dado uma amostra de tamanho n, sendo cada unidade amostral $i \in \{1, \ldots, n\}$, e dado o número de replicações *i.i.d.*, $r \in \{1, \ldots, R\}$, de uma simulação de Monte Carlo, em que $\mathbb{E}_{\text{Monte Carlo}}$ é a média com respeito ao número de replicações de Monte Carlo, temos:

- Erro quadrático médio (*mean square error* - MSE)

$$\text{MSE} = \mathbb{E}_{\text{Monte Carlo}} \left\{ \frac{1}{n} \sum_{i=1}^{n} (\tau(x_i) - \hat{\tau}(x_i))^2 \right\}. \tag{5.1}$$

- Viés absoluto médio ($|Bias|$)

$$|\text{Bias}| = \mathbb{E}_{\text{Monte Carlo}} \left\{ \frac{1}{n} \sum_{i=1}^{n} |\tau(x_i) - \hat{\tau}(x_i)| \right\}. \tag{5.2}$$

- Taxa de cobertura (*Covered*), onde s é o erro-padrão:

$$\text{Covered} = \mathbb{E}_{\text{Monte Carlo}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}\{|\tau(x_i) - \hat{\tau}(x_i)| \leqslant 1.96 \times s\} \right\}. \tag{5.3}$$

5.3 *Implementação computacional*

Foi utilizado o pacote *grf* (Tibshirani et al., 2022) para operacionalização dos cálculos do CF no *software* R (R Core Team, 2022), versão 4.2.2 e RStudio (Posit team, 2023), versão 2023.6.0.421.

Para a estimação do CATE através do método DML, foi necessário adaptar uma versão em Python do pacote *DoubleML (Bach et al., 2021, 2022)*[2] juntamente com o pacote *reticulate* (Ushey et al., 2023), o qual permite uma integração entre R e Python. Quanto às configurações do pacote para implementação do método DML, temos

1. o uso de florestas aleatórias para a estimação das funções auxiliares;
2. o ponto de corte do escore de propensão $\eta = 0.01$;
3. o parâmetro do número de replicações *bootstrap* $nboot = 1000$ para cálculo da variância
4. o número de *folds* para *cross-fitting* $nfolds = 5$;
5. o uso de uma base canônica de termos lineares como conjunto de transformações de X, $K = K(X)$, de acordo com Semenova et al. (2022) e as equações 4.37 e 4.38.

A íntegra do código em R para implementação da simulação de Monte Carlo está disponível no endereço <https://github.com/renatolauris/disserta_sim> e descrito no apêndice C.

5.4 *Resultados e discussão*

Primeiramente, temos os resultados quanto à acurácia e ao viés da estimação do CATE, $\tau(X)$, para $n = 2000$, $d = 4$, segundo cada um dos métodos apresentados no Capítulo 4, medidos pelo MSE (5.1) e Viés Absoluto Médio (5.2). Ao avaliar a relação viés-variância dos métodos para estimação do CATE, nota-se que as estimativas produzidas pelo método de florestas causais (CF amostra honesta, CF, e amostra adaptativa, CF_{adapt})

[2] Exemplo em Python disponivel aqui. Acesso em 26/06/2023.

apresentam maior acurácia que aquelas produzidas pelo método DML nos cenários B, C e D (Tabela 1), sendo $43,3\%$, $44,8\%$ e $56,6\%$ menores, respectivamente. Já para o cenário A, o resultado inverte, com o DML apresentando melhor desempenho (valor de MSE $53,8\%$ menor). Os métodos CF e DML tiveram ambos menor viés que os métodos IPW (IPW_{rf}) e AIPW (AIPW_{rf}). Em relação ao viés dos métodos, os resultados da Tabela 2 apresentam conclusões semelhantes àquelas com respeito à relação viés-variância. Na comparação entre o CF e o DML, o último se demonstrou preferível no cenário linear A, tomando por base as métricas de viés e de RMSE. Por outro lado, adicionando não-linearidades mais complexas (cenários B, C e D), verifica-se que o CF sistematicamente torna-se o método mais adequado.

Tabela 1 – MSE médio de 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) com número de covariáveis $d = 4$, tamanho de amostra $n = 2000$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	2000	**0.240**	0.590	3.296	3.054	**0.111**
B		**0.115**	0.690	3.010	3.070	**0.203**
C		**0.712**	0.777	5.870	10.202	**1.291**
D		**0.676**	0.662	6.505	7.578	**1.558**

Tabela 2 – Viés absoluto médio para 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) com número de covariáveis $d = 4$, tamanho de amostra $n = 2000$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	2000	**0.408**	0.621	1.234	1.102	**0.271**
B		**0.269**	0.655	1.101	1.223	**0.372**
C		**0.693**	0.714	1.961	2.541	**0.906**
D		**0.678**	0.633	1.601	1.793	**1.040**

Diante de especificações pré-estabelecidas de tamanho de amostra, de número de covariáveis, nota-se que a acurácia e o viés das estimações, medidas pelo MSE (Tabela 1, 10 e 11 do Apêndice D) e pelo Viés absoluto médio (Tabela 2, 12 e 13 do Apêndice D), respectivamente, são claramente melhores para estimações através do método de florestas causais (amostra honesta, CF, ou amostra adaptativa, CF_{adapt}) que pelo método Double/-Debiased Machine Learning (DML) nos cenários picos e vales e descontinuidade (Cenários C e D). Nos cenários linear e não-linear (Cenários A e B) o DML mantém um desempenho adequado, especialmente no cenário linear em virtude da estrutura de modelo utilizada: somente a combinação de termos lineares de X. Além disso, o método DML apresenta

estimativas com maior acurácia e menor viés em relação aos métodos tradicionais IPW e AIPW em todos os cenários. Tal fato nos permite afirmar que o DML é um método cujas estimações têm desempenhos que o torna um método com potencial de concorrer ao CF.

Em uma avaliação da consistência dos estimadores, ou seja, melhoria da acurácia do viés com o aumento do tamanho amostral, podemos ver que todos os métodos tendem a valores menores de MSE e de viés nos vários cenários, conforme as Tabelas 3 e 4 a seguir, exceto para o DML para os Cenários A e D na comparação entre as amostras de tamanho $n = 2000$ e $n = 5000$.

Tabela 3 – MSE médio de 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) por tamanho de amostra com número de covariáveis $d = 4$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	500	**0.501**	1.988	16.540	5.440	**0.919**
	2000	**0.240**	0.590	3.296	3.054	**0.111**
	5000	**0.080**	0.347	1.251	1.535	**0.157**
B	500	**0.229**	2.212	14.888	6.032	**0.399**
	2000	**0.115**	0.690	3.010	3.070	**0.203**
	5000	**0.112**	0.282	1.316	1.749	**0.113**
C	500	**1.207**	2.191	10.911	13.218	**2.249**
	2000	**0.712**	0.777	5.870	10.202	**1.291**
	5000	**0.364**	0.227	5.029	8.868	**1.110**
D	500	**1.197**	2.391	22.517	19.863	**3.088**
	2000	**0.676**	0.662	6.505	7.578	**1.558**
	5000	**0.541**	0.446	3.015	4.291	**1.831**

Tabela 4 – Viés absoluto médio de 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) por tamanho de amostra com número de covariáveis $d = 4$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	500	**0.588**	1.152	2.300	1.758	**0.787**
	2000	**0.408**	0.621	1.234	1.102	**0.271**
	5000	**0.230**	0.468	0.780	0.916	**0.327**
B	500	**0.381**	1.180	2.053	1.716	**0.505**
	2000	**0.269**	0.655	1.101	1.223	**0.372**
	5000	**0.276**	0.428	0.849	0.995	**0.275**
C	500	**0.941**	1.220	2.673	2.870	**1.192**
	2000	**0.693**	0.714	1.961	2.541	**0.906**
	5000	**0.495**	0.388	1.833	2.394	**0.850**
D	500	**0.839**	1.227	2.991	2.652	**1.433**
	2000	**0.678**	0.633	1.601	1.793	**1.040**
	5000	**0.589**	0.516	1.324	1.568	**1.081**

Especificamente na comparação entre Floresta Causal e DML, o MSE médio do DML para os cenários linear e não-linear (cenários A e B), ao elevarmos o tamanho amostral de $n = 500$ para $n = 5000$, tem reduções mais próximas da redução encontrada no CF ($82,9\%$ e $82,5\%$, respectivamente, no cenário A, e $71,7\%$ e $87,3\%$ no Cenário B). Essa maior acurácia do DML para esses cenários indica que o método consegue obter estimativas relativamente competitivas em amostras grandes e formas funcionais menos complexas. Esse comportamento também é encontrado nas especificações com número de covariáveis $d = 10$ e $d = 20$, conforme consta no Apêndice D na Seção Consistência Estimação.

Adicionalmente, analisamos se os métodos CF e DML alcançam uma taxa de cobertura compatível com o esperado em uma distribuição normal com intervalo de confiança ao nível de 95%. Tomamos como referência a taxa de cobertura do método do k vizinho mais próximo KNN (*k-nearest neighbor*, KNN-10 – KNN_{10} e KNN-100 – KNN_{100}) para comparação realizada entre os métodos CF e DML, tal como realizado em Wager and Athey (2018). Segundo Wager and Athey (2018), "como florestas causais são uma espécie de método de vizinho mais próximo adaptativo, é natural o KNN, vizinho mais próximo não adaptativo como base de referência". Assim o intervalo de confiança do KNN segue o proposto por Wager and Athey (2018), enquanto do CF se baseia em Athey et al. (2019) e o DML utiliza o método *bootstrap* como especificado na Seção 5.3.

Tabela 5 – Taxa de cobertura para 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, KNN-10, KNN-100 e DML) com número de covariáveis $d = 4$, tamanho de amostra $n = 2000$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	KNN_{10}	KNN_{100}	**DML**
A	2000	**0.82**	1.00	0.95	0.91	**1.00**
B		**0.94**	1.00	0.93	0.90	**1.00**
C		**0.66**	0.99	0.93	0.53	**1.00**
D		**0.70**	1.00	0.91	0.71	**1.00**

Tabela 6 – Taxa de cobertura de 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, KNN-10, KNN-100 e DML) por tamanho de amostra com número de covariáveis $d = 4$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	KNN_{10}	KNN_{100}	**DML**
A	500	**0.96**	1.00	0.93	0.83	**1.00**
	2000	**0.82**	1.00	0.95	0.91	**1.00**
	5000	**0.89**	1.00	0.94	0.94	**1.00**
B	500	**0.98**	1.00	0.94	0.97	**1.00**
	2000	**0.94**	1.00	0.93	0.90	**1.00**
	5000	**0.83**	1.00	0.94	0.93	**1.00**
C	500	**0.88**	1.00	0.89	0.29	**1.00**
	2000	**0.66**	0.99	0.93	0.53	**1.00**
	5000	**0.59**	1.00	0.94	0.77	**1.00**
D	500	**0.80**	1.00	0.92	0.48	**1.00**
	2000	**0.70**	1.00	0.91	0.71	**1.00**
	5000	**0.61**	0.98	0.91	0.71	**1.00**

As Tabelas 5 e 6, para $d = 4, n = 2000, \sigma_\epsilon = 3$ e por tamanho de amostra $n = 500, 2000, 5000$, indicam que tanto o Floresta Causal (CF) como o DML, em geral, não apresentam boas propriedades para fins de inferência. O primeiro alcançou taxas de cobertura relativamente próximas da taxa nominal de 95% para os Cenários A e B, mas inadequadas para os Cenários C e D. As taxas de cobertura do segundo método indicam a hipótese de estimativa do desvio-padrão enviesada para cima, pois em todos os cenários o DML apresentou taxa de cobertura de 100%, o indica a necessidade de se avançar no estudo de estimadores para a variância do CATE para esse método.

Por fim, a análise nas Tabelas 7, 8 e 9 das métricas de acurácia e de viés para fins de estimação, MSE e Viés Absoluto, e de adequação dos intervalos de confiança para fins de inferência, Taxa de Cobertura, para os métodos CF e DML sob os cenários avaliados responde ao objetivo principal dessa dissertação: (i) avaliar os métodos para estimação de efeitos heterogêneos em contexto de maior dimensionalidade e, (ii) na comparação entre os métodos Floresta Causal e DML verificar se o último se apresenta como alternativa frente aos desafios de estimação do primeiro em cenários funcionais do CATE com picos e vales (Cenário C) ou com descontinuidades (Cenário D) acrescidos de maior dimensionalidade.

Sob especificações de tamanho de amostra $n = 2000$ e $\sigma_\epsilon = 3$, à medida que ampliamos o número de covariáveis inseridas na amostra a ser treinada para estimar o formato verdadeiro do parâmetro de interesse $\tau(X)$, $d = 4, 10, 20$, é possível verificar uma perda de acurácia e um maior viés de $d = 10$ para $d = 20$ para as estimações realizadas com

Floresta Causal (Tabelas 7 e 8). O mesmo padrão de queda de desempenho é verificada em termos de taxa de cobertura de $d = 10$ para $d = 20$ pelo método CF, especialmente nos cenários linear e não-linear (Cenários A e B) que até então apresentava níveis de taxa satisfatórios conforme o intervalo de confiança nominal. As estimações por DML, embora tenha níveis de acurácia e de viés próximos ao CF medidos pelo MSE e pelo Viés Absoluto Médio para os cenários linear e não-linear (Cenários A e B), aparentam sofrer menor deterioração de acurácia e de viés nos cenários picos e vales e descontinuidade (Cenários C e D). Tal fato indica que a alternativa proposta ao Floresta Causal, DML, em geral não apresenta propriedades melhores para superar os desafios de estimação em cenários funcionais do CATE com picos e vales (Cenário C) ou com descontinuidades (Cenário D). Ao mesmo tempo, a menor sensibilidade do desempenho da estimação com o DML em maiores dimensionalidades abre a possibilidade de pesquisas futuras avançarem em modelos mais flexíveis usando DML possam apresentar melhorias no ajuste da estimação nos referidos cenários desafiantes. Entretanto, ainda persiste a falta de boas propriedades de adequação dos intervalos de confiança com o esperado nominalmente do DML ao longo de especificações com maior dimensionalidade (número de covariáveis), conforme podemos ver na Tabela 9.

Tabela 7 – MSE médio de 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) por número de covariáveis com tamanho de amostra $n = 2000$ e $\sigma_\epsilon = 3$.

Cenário	d	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	4	**0.240**	0.590	3.296	3.054	**0.111**
	10	**0.121**	0.487	1.716	1.252	**0.398**
	20	**0.306**	0.451	1.152	1.010	**0.301**
B	4	**0.115**	0.690	3.010	3.070	**0.203**
	10	**0.138**	0.653	2.500	1.273	**0.199**
	20	**0.306**	0.504	0.889	1.143	**0.337**
C	4	**0.712**	0.777	5.870	10.202	**1.291**
	10	**0.540**	0.636	5.345	7.438	**1.348**
	20	**1.390**	0.927	4.872	6.474	**1.526**
D	4	**0.676**	0.662	6.505	7.578	**1.558**
	10	**0.662**	1.391	3.145	4.120	**2.444**
	20	**1.095**	0.975	2.451	2.674	**2.174**

Tabela 8 – Viés absoluto médio para 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) por número de covariáveis com tamanho de amostra $n = 2000$ e $\sigma_\epsilon = 3$.

Cenário	d	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
	4	**0.408**	0.621	1.234	1.102	**0.271**
A	10	**0.278**	0.554	0.833	0.872	**0.505**
A	20	**0.450**	0.535	0.782	0.801	**0.440**
	4	**0.269**	0.655	1.101	1.223	**0.372**
B	10	**0.287**	0.653	0.882	0.900	**0.360**
	20	**0.456**	0.568	0.755	0.824	**0.467**
	4	**0.693**	0.714	1.961	2.541	**0.906**
C	10	**0.603**	0.637	1.901	2.214	**0.939**
	20	**0.953**	0.777	1.825	2.079	**0.986**
	4	**0.678**	0.633	1.601	1.793	**1.040**
D	10	**0.637**	0.941	1.368	1.573	**1.267**
	20	**0.828**	0.789	1.231	1.298	**1.191**

Tabela 9 – Taxa de cobertura para 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, KNN-10, KNN-100 e DML) por número de covariáveis com tamanho de amostra $n = 2000$ e $\sigma_\epsilon = 3$.

Cenário	d	**CF**	CF_{adapt}	KNN_{10}	KNN_{100}	**DML**
	4	**0.82**	1.00	0.95	0.91	**1.00**
A	10	**0.96**	1.00	0.94	0.90	**1.00**
	20	**0.74**	1.00	0.93	0.88	**1.00**
	4	**0.94**	1.00	0.93	0.90	**1.00**
B	10	**0.94**	1.00	0.94	0.92	**1.00**
	20	**0.74**	1.00	0.94	0.81	**1.00**
	4	**0.66**	0.99	0.93	0.53	**1.00**
C	10	**0.71**	1.00	0.90	0.37	**1.00**
	20	**0.51**	1.00	0.82	0.32	**1.00**
	4	**0.70**	1.00	0.91	0.71	**1.00**
D	10	**0.79**	1.00	0.92	0.64	**1.00**
	20	**0.56**	1.00	0.89	0.52	**1.00**

Já cotejando os resultados sob especificações de tamanho de amostra menores, $n = 500$, e maiores, $n = 5000$ que se encontram no Apêndice D (Seção Dimensionalidade), especialmente para os cenários mais desafiantes para estimação pelo Floresta Causal, Cenários C e D, verificamos que as estimativas são menos afetadas quanto à acurácia e viés à medida que se aumenta o número de covariáveis e de tamanho da amostra. As estimativas pelo método DML se mantêm competitivas frente ao Floresta Causal quando temos uma forma funcional linear e não-linear em um contexto de aumento das dimensões e de tamanho da amostra.

6 Considerações finais

A presente dissertação buscou abordar sobre métodos de aprendizado de máquina que propõem estimar efeitos causais heterogêneos e que especialmente sejam adequados em contextos de maior dimensionalidade.

Para isso foram revisados o arcabouço de Desfechos Potenciais (Rubin, 1974, 2005), as suposições necessárias para identificação dos estimandos de interesse para estimação dos efeitos causais, ATE e, particularmente, CATE para o caso de efeitos causais heterogêneos.

O objetivo principal dessa dissertação foi avaliar os métodos concorrentes, Floresta Causal e DML, para estimação de efeitos heterogêneos em contexto de maior dimensionalidade. Através de um estudo de simulação, os métodos foram comparados a partir de medidas de desempenho das estimativas (MSE, Viés e Taxa de Cobertura) afim de verificar se o último se apresenta como alternativa frente aos desafios de estimação do primeiro em cenários funcionais do CATE com picos e vales (Cenário C) ou com descontinuidades (Cenário D) acrescidos de maior dimensionalidade.

Para atingir esses objetivos foram descritos como os métodos de aprendizado de máquina Floresta Causal e DML estimam o CATE, o estimando que permite identificar efeitos causais heterogêneos.

Este trabalho propôs cenários de simulação e comparação entre os métodos CF e DML que não haviam sido comparados em trabalhos anteriores. Além disso, trouxe uma implementação alternativa à estimação do CATE para o método DML em R, usando a interface R-Python a partir dos pacotes *DoubleML* (Bach et al., 2021) e *Reticulate* (Ushey et al., 2023).

Dentre as limitações do trabalho, podemos citar a ausência de um exemplo com dados reais de análise utilizando os métodos estudados. Também se utilizou somente uma proposta de aproximação linear do modelo para estimar o CATE via DML. Basicamente se usou uma base canônica de termos lineares como conjunto de transformações de X, $K = K(X)$, de acordo com Semenova et al. (2022) e as equações (4.37) e (4.38). Espera-se que outras proposições possam conferir maior flexibilidade e melhor desempenho do DML, especialmente nos cenários mais desafiadores em maior dimensionalidade, picos e vales ou com descontinuidades.

Dentre os resultados que merecem destaque, temos que inadequadas taxas de cobertura encontradas no exercício de simulação indicam a necessidade de se avançar na proposição de procedimentos para construção de ICs e na construção de estimadores para a variância do CATE para ambos métodos CF e DML que possam superar esses resultados não satisfatórios.

Particularmente, as estimativas por DML tiveram níveis de acurácia próximos ao CF medidos pelo MSE e o Viés somente para os cenários linear e não-linear (Cenários A e B). Em geral, o DML não apresenta propriedades melhores para superar os desafios de estimação em cenários funcionais do CATE com picos e vales (Cenário C) ou com descontinuidades (Cenário D). Não obstante, se constatou que o método alternativo ao Floresta Causal apresenta menor sensibilidade do desempenho da estimação em maiores dimensionalidades, especialmente para tamanho de amostra grandes tal como $n = 5000$ estudado. Isso abre a possibilidade de pesquisas futuras avançarem em modelos mais flexíveis usando DML que possam apresentar melhorias no ajuste da estimação nos referidos cenários desafiadores.

Referências

Aalen, O.O., Røysland, K., Gran, J.M., Ledergerber, B., 2012. Causality, Mediation and Time: A Dynamic Viewpoint. Journal of the Royal Statistical Society Series A: Statistics in Society 175, 831–861. URL: <https://doi.org/10.1111/j.1467-985X.2011.01030.x>, doi:<10.1111/j.1467-985X.2011.01030.x>.

Abrevaya, J., Hsu, Y.C., Lieli, R.P., 2015. Estimating Conditional Average Treatment Effects. Journal of Business & Economic Statistics 33, 485–505. URL: <https://doi.org/10.1080/07350015.2014.975555>, doi:<10.1080/07350015.2014.975555>. publisher: Taylor & Francis _eprint: https://doi.org/10.1080/07350015.2014.975555.

Altman, N., Krzywinski, M., 2015. Points of significance: Association, correlation and causation. Nature methods 12.

Athey, S., Imbens, G., 2016. Recursive partitioning for heterogeneous causal effects. Proceedings of the National Academy of Sciences 113, 7353–7360. URL: <https://www.pnas.org/doi/full/10.1073/pnas.1510489113>, doi:<10.1073/pnas.1510489113>. publisher: Proceedings of the National Academy of Sciences.

Athey, S., Luca, M., 2019. Economists (and economics) in tech companies. Journal of Economic Perspectives 33, 209–30.

Athey, S., Tibshirani, J., Wager, S., 2019. Generalized random forests. The Annals of Statistics 47, 1148–1178.

Athey, S., Wager, S., 2019. Estimating treatment effects with causal forests: An application. Observational Studies 5, 37–51. Publisher: University of Pennsylvania Press.

Bach, P., Chernozhukov, V., Kurz, M.S., Spindler, M., 2021. DoubleML – An object-oriented implementation of double machine learning in R. URL: <https://arxiv.org/abs/2103.09603>.

Bach, P., Chernozhukov, V., Kurz, M.S., Spindler, M., 2022. DoubleML – An object-oriented implementation of double machine learning in Python. Journal of Machine Learning Research 23, 1–6. URL: <http://jmlr.org/papers/v23/21-0862.html>.

Beauchamp, D.H.E.b.T.L. (Ed.), 1999. An Enquiry concerning Human Understanding. Oxford Philosophical Texts, Oxford University Press, Oxford, New York.

Belloni, A., Chen, D., Chernozhukov, V., Hansen, C., 2012. Sparse models and methods for optimal instruments with an application to eminent domain. Econometrica 80, 2369–2429.

Belloni, A., Chernozhukov, V., Fernández-Val, I., Hansen, C., 2017. Program evaluation and causal inference with high-dimensional data. Econometrica 85, 233–298.

Blackstone, E.H., 2019. Precision medicine versus evidence-based medicine: individual treatment effect versus average treatment effect. Circulation 140, 1236–1238.

Breiman, L., 1996. Bagging predictors. Machine learning 24, 123–140.

Breiman, L., 2001. Random forests. Machine learning 45, 5–32.

Breiman, L., Friedman, J., Olshen, R., Stone, C., 1984. Cart. Classification and regression trees .

Brodeur, A., Cook, N., Heyes, A., 2020. Methods matter: P-hacking and publication bias in causal analysis in economics. American Economic Review 110, 3634–60.

Cai, T., Liu, W., Luo, X., 2011. A constrained ℓ 1 minimization approach to sparse precision matrix estimation. Journal of the American Statistical Association 106, 594–607.

Casella, G., Robert, C.P., 1996. Rao-blackwellisation of sampling schemes. Biometrika 83, 81–94.

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., Robins, J., 2018. Double/debiased machine learning for treatment and structural parameters.

Chernozhukov, V., Hansen, C., Spindler, M., 2016. High-dimensional metrics in r. arXiv preprint arXiv:1603.01700 .

Chernozhukov, V., Newey, W.K., Singh, R., 2022. Automatic debiased machine learning of causal and structural effects. Econometrica 90, 967–1027.

Crump, R.K., Hotz, V.J., Imbens, G.W., Mitnik, O.A., 2009. Dealing with limited overlap in estimation of average treatment effects. Biometrika 96, 187–199.

Ding, P., Li, X., Miratrix, L.W., 2017. Bridging finite and super population causal inference. Journal of Causal Inference 5.

Falcon, A., 2022. Aristotle on Causality, in: Zalta, E.N. (Ed.), The Stanford Encyclopedia of Philosophy. Spring 2022 ed.. Metaphysics Research Lab, Stanford University.

Fan, Q., Hsu, Y.C., Lieli, R.P., Zhang, Y., 2022. Estimation of Conditional Average Treatment Effects With High-Dimensional Data. Journal of Business & Economic Statistics 40, 313–327. URL: <https://doi.org/10.1080/07350015.2020.1811102>, doi:<10.1080/07350015.2020.1811102>. publisher: Taylor & Francis _eprint: https://doi.org/10.1080/07350015.2020.1811102.

Friedman, J.H., 2001. Greedy function approximation: a gradient boosting machine. Annals of statistics , 1189–1232.

Frisch, R., Waugh, F.V., 1933. Partial time regressions as compared with individual trends. Econometrica: Journal of the Econometric Society , 387–401.

Van de Geer, S., Bühlmann, P., Ritov, Y., Dezeure, R., 2014. On asymptotically optimal confidence regions and tests for high-dimensional models. Annals of Statistics , 1166–202.

Glynn, A.N., Quinn, K.M., 2010. An introduction to the augmented inverse propensity weighted estimator. Political analysis 18, 36–56.

Granger, C.W., 1969. Investigating causal relations by econometric models and cross-spectral methods. Econometrica: journal of the Econometric Society , 424–438.

Haavelmo, T., 1943. The statistical implications of a system of simultaneous equations. Econometrica, Journal of the Econometric Society , 1–12.

Haavelmo, T., 1944. The probability approach in econometrics. Econometrica: Journal of the Econometric Society , iii–115.

Hahn, P.R., Murray, J.S., Carvalho, C.M., 2020. Bayesian regression tree models for causal inference: Regularization, confounding, and heterogeneous effects (with discussion). Bayesian Analysis 15, 965–1056.

Hartley, H., Sielken Jr, R.L., 1975. A"super-population viewpoint"for finite population sampling. Biometrics , 411–422.

Hastie, T., Tibshirani, R., Friedman, J.H., Friedman, J.H., 2009. The elements of statistical learning: data mining, inference, and prediction. volume 2. Springer.

Heckman, J.J., 2022. Interview with James Heckman. Observational Studies 8, 7–22. URL: <https://muse.jhu.edu/article/867086>, doi:<10.1353/obs.2022.0006>. publisher: University of Pennsylvania Press.

Hernán, M.A., Robins, J.M., 2006. Estimating causal effects from epidemiological data. Journal of Epidemiology & Community Health 60, 578–586.

Hill, A.B., 1965. The environment and disease: association or causation?

Hirano, K., Imbens, G.W., Ridder, G., 2003. Efficient estimation of average treatment effects using the estimated propensity score. Econometrica 71, 1161–1189.

Holland, P.W., 1986. Statistics and Causal Inference. Journal of the American Statistical Association 81, 945–960. URL: <https://www.jstor.org/stable/2289064>, doi:<10.2307/2289064>. publisher: [American Statistical Association, Taylor & Francis, Ltd.].

Imai, K., Ratkovic, M., 2013. Estimating treatment effect heterogeneity in randomized program evaluation. The Annals of Applied Statistics 7, 443–470.

Imbens, G.W., Rubin, D.B., 2015. Causal inference in statistics, social, and biomedical sciences. Cambridge University Press.

Imbens, G.W., Wooldridge, J.M., 2009. Recent Developments in the Econometrics of Program Evaluation. Journal of Economic Literature 47, 5–86. URL: <https://www.aeaweb.org/articles?id=10.1257/jel.47.1.5>, doi:<10.1257/jel.47.1.5>.

Jacob, D., 2021. Cate meets ml–the conditional average treatment effect and machine learning. arXiv preprint arXiv:2104.09935 .

Kang, J.D., Schafer, J.L., 2007. Demystifying double robustness: A comparison of alternative strategies for estimating a population mean from incomplete data. Statistical science 22, 523–539.

Knaus, M.C., Lechner, M., Strittmatter, A., 2020. Machine learning estimation of heterogeneous causal effects: Empirical Monte Carlo evidence. The Econometrics Journal 24, 134–161. URL: <https://doi.org/10.1093/ectj/utaa014>, doi:<10.1093/ectj/utaa014>.

Künzel, S.R., Sekhon, J.S., Bickel, P.J., Yu, B., 2019. Metalearners for estimating heterogeneous treatment effects using machine learning. Proceedings of the national academy of sciences 116, 4156–4165.

Kurz, C.F., 2022. Augmented inverse probability weighting and the double robustness property. Medical Decision Making 42, 156–167.

Liang, M., 2018. Subgroup identification and estimation of individualized causal effects in precision medicine. The University of Wisconsin-Madison.

Liao, J., Rohde, C., 2022. Variance reduction in the inverse probability weighted estimators for the average treatment effect using the propensity score. Biometrics 78, 660–667.

Lovell, M.C., 1963. Seasonal adjustment of economic time series and multiple regression analysis. Journal of the American Statistical Association 58, 993–1010.

Mill, J.S., 2011. A System of Logic, Ratiocinative and Inductive: Being a Connected View of the Principles of Evidence, and the Methods of Scientific Investigation. volume 1 of *Cambridge Library Collection - Philosophy*. Cambridge University Press, Cambridge. URL: <https://www.cambridge.org/core/books/system-of-logic-ratiocinative-and-inductive/290C43FBA4DC7022540D58E7EC49B1C2>, doi:<10.1017/CBO9781139149839>.

Miller, A.R., Segal, C., 2019. Do female officers improve law enforcement quality? effects on crime reporting and domestic violence. The review of economic studies 86, 2220–2247.

Morabia, A., 1991. On the origin of hill's causal criteria. Epidemiology , 367–369.

Newey, W.K., 1994. The asymptotic variance of semiparametric estimators. Econometrica: Journal of the Econometric Society , 1349–1382.

Neyman, J., 1959. Optimal asymptotic tests of composite hypotheses. Probability and statsitics , 213–234.

Neyman, J., 1979. C (α) tests and their use. Sankhyā: The Indian Journal of Statistics, Series A , 1–21.

Nie, X., Wager, S., 2021. Quasi-oracle estimation of heterogeneous treatment effects. Biometrika 108, 299–319.

Norton, E.b.D.F., Norton, M.J. (Eds.), 2011. David Hume: A Treatise of Human Nature: Two-volume set. Clarendon Hume Edition Series, Oxford University Press, Oxford, New York.

Pearl, J., 1995. Causal Diagrams for Empirical Research. Biometrika 82, 669–688. URL: <https://www.jstor.org/stable/2337329>, doi:<10.2307/2337329>. publisher: [Oxford University Press, Biometrika Trust].

Pearl, J., 2012. The causal foundations of structural equation modeling. Technical Report. California Univ Los Angeles Dept of Computer Science.

Pearl, J., Mackenzie, D., 2018. The book of why: the new science of cause and effect. Basic books.

Posit team, 2023. RStudio: Integrated Development Environment for R. Posit Software, PBC. Boston, MA. URL: <http://www.posit.co/>.

R Core Team, 2022. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing. Vienna, Austria. URL: <https://www.R-project.org/>.

Redding, C., 2019. A teacher like me: A review of the effect of student–teacher racial/ethnic matching on teacher perceptions of students and student academic and behavioral outcomes. Review of educational research 89, 499–535.

Robins, J.M., 2000. Robust estimation in sequentially ignorable missing data and causal inference models, in: Proceedings of the American Statistical Association, Indianapolis, IN. pp. 6–10.

Robins, J.M., Ritov, Y., 1997. Toward a curse of dimensionality appropriate (coda) asymptotic theory for semi-parametric models. Statistics in medicine 16, 285–319.

Robins, J.M., Rotnitzky, A., 1995. Semiparametric efficiency in multivariate regression models with missing data. Journal of the American Statistical Association 90, 122–129.

Robins, J.M., Rotnitzky, A., Zhao, L.P., 1994. Estimation of regression coefficients when some regressors are not always observed. Journal of the American statistical Association 89, 846–866.

Robinson, P.M., 1988. Root-n-consistent semiparametric regression. Econometrica: Journal of the Econometric Society , 931–954.

Rubin, D., 1980. Discussion of"randomization analysis of experimental data in the fisher randomization test"by d. basu. Journal of the American statistical association 75, 591–593.

Rubin, D.B., 1974. Estimating causal effects of treatments in randomized and nonrandomized studies. Journal of educational Psychology 66, 688.

Rubin, D.B., 2005. Causal Inference Using Potential Outcomes. Journal of the American Statistical Association 100, 322–331. URL: <https://doi.org/10.1198/016214504000001880>, doi:<10.1198/016214504000001880>. publisher: Taylor & Francis _eprint: https://doi.org/10.1198/016214504000001880.

Scharfstein, D.O., Rotnitzky, A., Robins, J.M., 1999. Adjusting for nonignorable drop-out using semiparametric nonresponse models. Journal of the American Statistical Association 94, 1096–1120.

Semenova, V., Chernozhukov, V., 2021. Debiased machine learning of conditional average treatment effects and other causal functions. The Econometrics Journal 24, 264–289.

Semenova, V., Goldman, M., Chernozhukov, V., Taddy, M., 2022. Estimation and inference on heterogeneous treatment effects in high-dimensional dynamic panels under weak dependence.

Stürmer, T., Rothman, K.J., Avorn, J., Glynn, R.J., 2010. Treatment effects in the presence of unmeasured confounding: dealing with observations in the tails of the propensity score distribution—a simulation study. American journal of epidemiology 172, 843–854.

Syrgkanis, V., Lewis, G., Oprescu, M., Hei, M., Battocchi, K., Dillon, E., Pan, J., Wu, Y., Lo, P., Chen, H., et al., 2021. Causal inference and machine learning in practice with econml and causalml: Industrial use cases at microsoft, tripadvisor, uber, in: Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining, pp. 4072–4073.

Tian, L., Alizadeh, A.A., Gentles, A.J., Tibshirani, R., 2014. A simple method for estimating interactions between a treatment and a large number of covariates. Journal of the American Statistical Association 109, 1517–1532.

Tibshirani, J., Athey, S., Sverdrup, E., Wager, S., 2022. grf: Generalized Random Forests. URL: <https://CRAN.R-project.org/package=grf>. r package version 2.2.1.

Ushey, K., Allaire, J., Tang, Y., 2023. reticulate: Interface to 'Python'. URL: <https://CRAN.R-project.org/package=reticulate>. r package version 1.30.

Wager, S., Athey, S., 2018. Estimation and Inference of Heterogeneous Treatment Effects using Random Forests. Journal of the American Statistical Association 113, 1228–1242. URL: <https://doi.org/10.1080/01621459.2017.1319839>, doi:<10.1080/01621459.2017.1319839>. publisher: Taylor & Francis _eprint: https://doi.org/10.1080/01621459.2017.1319839.

Walther, B.A., Moore, J.L., 2005. The concepts of bias, precision and accuracy, and their use in testing the performance of species richness estimators, with a literature review of estimator performance. Ecography 28, 815–829.

Wolf, J., Hunter, P.R., Freeman, M.C., Cumming, O., Clasen, T., Bartram, J., Higgins, J.P., Johnston, R., Medlicott, K., Boisson, S., et al., 2018. Impact of drinking water, sanitation and handwashing with soap on childhood diarrhoeal disease: updated meta-analysis and meta-regression. Tropical medicine & international health 23, 508–525.

Zhang, C.H., Zhang, S.S., 2014. Confidence intervals for low dimensional parameters in high dimensional linear models. Journal of the Royal Statistical Society: Series B: Statistical Methodology , 217–242.

Zhang, W., VanDyke, M.S., 2022. Association Versus Causation. Springer International Publishing, Cham. pp. 53–55. URL: <https://doi.org/10.1007/978-3-319-32010-6_15>, doi:<10.1007/978-3-319-32010-6_15>.

Zhou, N., Zhu, L., 2021. On IPW-based estimation of conditional average treatment effects. Journal of Statistical Planning and Inference 215, 1–22. URL: <https://www.sciencedirect.com/science/article/pii/S0378375821000197>, doi:<10.1016/j.jspi.2021.02.003>.

Apêndices

Apêndice A – Identificação do CATE via IPW

Conforme demonstrado em Imbens and Wooldridge (2009), sendo $e(x) = \Pr(W = 1 \mid X_i = x) = \mathbb{E}(W \mid X_i = x)$ o escore de propensão e aplicando a propriedade das expectativas iteradas, o IPW identifica o CATE da seguinte forma:

$$
\begin{aligned}
\tau_{\text{CATE}}(x) &= \mathbb{E}\left[\frac{WY}{e(x)} - \frac{(1-W)Y}{1-e(x)} \;\middle|\; X_i = x\right] \\
&= \mathbb{E}\left[\frac{WY}{e(x)} \;\middle|\; X_i = x\right] - \mathbb{E}\left[\frac{(1-W)Y}{1-e(x)} \;\middle|\; X_i = x\right] \\
&= \mathbb{E}\left[\frac{WY^{(1)}}{e(x)} \;\middle|\; X_i = x\right] - \mathbb{E}\left[\frac{(1-W)Y^{(0)}}{1-e(x)} \;\middle|\; X_i = x\right] \\
&= \mathbb{E}\left[\mathbb{E}\left[\frac{WY^{(1)}}{e(x)} \;\middle|\; X\right] \;\middle|\; X_i = x\right] - \mathbb{E}\left[\mathbb{E}\left[\frac{(1-W)Y^{(0)}}{1-e(x)} \;\middle|\; X\right] \;\middle|\; X_i = x\right] \\
&= \mathbb{E}\left[\frac{\mathbb{E}(W \mid X)\,\mathbb{E}(Y^{(1)} \mid X)}{e(x)} \;\middle|\; X_i = x\right] \\
&\qquad - \mathbb{E}\left[\frac{(1-\mathbb{E}(W \mid X))\,\mathbb{E}(Y^{(0)} \mid X)}{1-e(x)} \;\middle|\; X_i = x\right] \\
&= \mathbb{E}\left[\frac{e(x)\,\mathbb{E}(Y^{(1)} \mid X)}{e(x)} \;\middle|\; X_i = x\right] \\
&\qquad - \mathbb{E}\left[\frac{(1-e(x))\,\mathbb{E}(Y^{(0)} \mid X)}{1-e(x)} \;\middle|\; X_i = x\right] \\
&= \mathbb{E}\left[\mathbb{E}(Y^{(1)} \mid X) \mid X_i = x\right] - \mathbb{E}\left[\mathbb{E}(Y^{(0)} \mid X) \mid X_i = x\right] \\
&= \mathbb{E}\left[Y^{(1)} \mid X_i = x\right] - \mathbb{E}\left[Y^{(0)} \mid X_i = x\right] \\
&= \mathbb{E}\left[Y^{(1)} - Y^{(0)} | X_i = x\right].
\end{aligned}
$$

Apêndice B – Dupla robustez do estimador AIPW

Nesse apêndice iremos demonstrar a consistência do estimador AIPW nos casos onde (i) o modelo de escore de propensão é conhecido e assim corretamente especificado, mesmo que o modelo de desfechos seja mal especificado; e (ii) a situação inversa, onde o modelo de desfechos está bem especificada e o escore de propensão não.

Primeiro, devemos assumir SUTVA e ignorabilidade (suposição 2), e reapresentamos o estimador (4.2):

$$\hat{\tau}_{\text{AIPW}}(x) = \left[\hat{\mu}_1(x) - \hat{\mu}_0(x) + \frac{W}{\hat{e}(x)}\left(Y - \hat{\mu}_1(x)\right) - \frac{(1-W)}{1-\hat{e}(x)}\left(Y - \hat{\mu}_0(x)\right) \;\middle|\; X = x\right].$$

No caso (i), $\hat{e}(x)$ é corretamente especificado, logo $\mathbb{E}\left[\hat{e}(x)\right] = \mathbb{E}\left[W = 1 \mid X = x\right]$ e $(W - \hat{e}(x))$ converge para zero quando o tamanho da amostra tende ao infinito, $[W - \hat{e}(x)] \xrightarrow{p} 0$. Assim rearranjando a equação do estimador AIPW (4.2), os dois últimos termos tendem a zero e chegamos a equação do IPW (3.5), que já demonstramos que identifica o CATE, conforme demonstrado a seguir:

$$\begin{aligned}
\hat{\tau} &\xrightarrow{p} \mathbb{E}\left[\frac{WY}{\hat{e}(x)} - \frac{(1-W)Y}{1-\hat{e}(x)} + \frac{\hat{e}(x) - W}{\hat{e}(x)}\hat{\mu}^{(1)}(x) - \frac{(1-\hat{e}(x)) - (1-W)}{1-\hat{e}(x)}\hat{\mu}^{(0)}(x) \;\middle|\; X = x\right] \\
&= \mathbb{E}\left[\frac{WY}{\hat{e}(x)} - \frac{(1-W)Y}{1-\hat{e}(x)} \;\middle|\; X = x\right] \\
&= \mathbb{E}\left[Y^{(1)} - Y^{(0)} \mid X = x\right] \\
&= \tau.
\end{aligned}$$

Já no caso (ii), consideramos $\hat{\mu}$ corretamente especificado, ou seja, $\mathbb{E}\left[\hat{\mu}^{(w)}(x)\right] = \mathbb{E}\left[Y \mid X = x, W = w\right]$. Para $w = 1$ e assumindo ignorabilidade e SUTVA, $\mathbb{E}\left[\hat{\mu}^{(1)}(x)\right] = \mathbb{E}\left[Y \mid X = x, W = 1\right] = \mathbb{E}\left[Y \mid X = x\right]$. Para $w = 0$ temos um resultado semelhante sob as suposições mencionadas: $\mathbb{E}\left[\hat{\mu}^{(0)}(x)\right] = \mathbb{E}\left[Y \mid X = x, W = 0\right] = \mathbb{E}\left[Y \mid X = x\right]$.

Então a partir da equação do estimador AIPW (4.2), os dois últimos termos tendem a zero e chegamos a equação do S-leaner e como assumimos correta especificação de $\hat{\mu}$, demonstra-se abaixo que o estimador é consistente também no caso (ii):

$$\begin{aligned}
\hat{\tau} &\xrightarrow{p} \mathbb{E}\left[\hat{\mu}^{(1)}(x) - \hat{\mu}^{(0)}(X) + \frac{W\left(Y - \hat{\mu}^{(1)}(x)\right)}{\hat{e}(x)} - \frac{(1-W)\left(Y - \hat{\mu}^{(0)}(x)\right)}{1-\hat{e}(x)} \;\middle|\; X = x\right] \\
&= \mathbb{E}\left[\hat{\mu}^{(1)}(x) - \hat{\mu}^{(0)}(x) \mid X = x\right] \\
&= \mathbb{E}\left[Y^{(1)} - Y^{(0)} \mid X = x\right] \\
&= \tau.
\end{aligned}$$

Apêndice C – Código em R Simulação de Monte Carlo

Além da versão disponível em <https://github.com/renatolauris/disserta_sim>, a íntegra do código com a implementação da simulação de acordo com os métodos e com os processos geradores de dados segue abaixo:

```
rm(list = ls())

# loading libraries ----------------------------------------------------

library(grf)
# library(mgcv)
# library(randomForestCI) #computing causalForest variance
library(FNN)
library(Hmisc)
library(xtable)
library(pacman)
p_load(tidyverse)
p_load(np)
p_load(writexl)
p_load(ranger)
library(hdm)
library(glmnet)
library(purrr)
require(splines)
library(readxl)

library(DoubleML)
library(mlr3)
library(mlr3learners)
library(data.table)
library(ggplot2)
library(reticulate)
# citation("FNN")
# citation("grf")
# citation("np")

# 0. loading working directory and data
   ------------------------------------------------------------
```

```
33
34 getwd()
35 end.temp <- "C:/Users/renat/Dropbox/Dissertacao-Renato/2.simulacoes de
       causalidade"
36 setwd(end.temp)
37
38 # 2a. KNN estimator ------------------------------------
39
40 # knn causal function
41 causal.kn <- function(kn,X,W,X.test,Y){
42   # function inputs
43   # kn: K-nearest neighbour
44   # X: vector of covariates, train data
45   # W: treatment variable, train data
46   # Y: outcome variable, train data
47   # X.test: vector of covariates, test data
48
49   knn.0.mu = knn.reg(X[W==0,], X.test, Y[W==0], k = kn)$pred
50   knn.1.mu = knn.reg(X[W==1,], X.test, Y[W==1], k = kn)$pred
51
52   knn.0.mu2 = knn.reg(X[W==0,], X.test, Y[W==0]^2, k = kn)$pred
53   knn.1.mu2 = knn.reg(X[W==1,], X.test, Y[W==1]^2, k = kn)$pred
54
55   knn.0.var = (knn.0.mu2 - knn.0.mu^2) / (kn - 1)
56   knn.1.var = (knn.1.mu2 - knn.1.mu^2) / (kn - 1)
57
58   knn.tau = knn.1.mu - knn.0.mu
59   knn.se = sqrt(knn.0.var + knn.1.var)
60   cbind(knn.tau,knn.se) %>% as.data.frame()
61   # function outputs:
62   # knn.tau: K-nearest neighbour tau estimate
63   # knn.se: K-nearest neighbour standard errors of tau estimate
64 }
65
66 # 2b. Inverse propensity-weighted estimator ------------------------------
67
68 # adjust X vector to model matrix
69 make_matrix = function(x) stats::model.matrix(~.-1, x)
70 # transform X vector to splines for lasso regression
```

```r
make_matrix_splines = function(x) {
  col=1
  X_ns = do.call(cbind, lapply(1:dim(x)[2], function(col){matrix(splines::
    ns(x[,col],df=7), nrow(x), 7)}))
  dim_ns = dim(X_ns)[2]
  X_ns = stats::model.matrix(~.*.-1, data.frame(X_ns)) # pairwise
    interaction (not including squared term for each column)
  X_ns_sq = do.call(cbind, lapply(1:dim_ns, function(col){matrix(X_ns[,col
    ]^2)})) # squared term for each column
  X_ns = cbind(X_ns, X_ns_sq)
  make_matrix(data.frame(X_ns))
}

# function to trim the propensity score function probability space
# to hold common support hypotesis
trimmed_ps <- function(x,p=0.01){
  x = ifelse(x<p, p, ifelse(x>1-p,1-p, x))
}

# ipw estimator by random forest regression with predetermined ps threshold
ipw.rf.estimator <- function(X,Y,W,n_fold=5,p_threshold=0.01){
  # function inputs
  # X: vector of covariates, train data
  # W: treatment variable, train data
  # Y: outcome variable, train data
  # n_fold: n fold for cross-fit first stage estimations
  # p_threshold: threshold for trimmed propensity score

  # first stage
  # A list of vectors indicating the left-out subset
  n <- nrow(X)
  n.folds <- n_fold
  # indices <- split(seq(n), sort(seq(n) %% n.folds))
  foldid <- rep.int(1:n.folds,times = ceiling(n/n.folds))[sample.int(n)] #
    define folds indices
  indices <- split(1:n, foldid)  #split observation indices into folds
  ipw.scores <- lapply(indices, function(idx) {

    # Fitting the propensity score model
```

```
    # Comment / uncomment the lines below as appropriate.
    # OBSERVATIONAL SETTING (with unconfoundedness+overlap):
    model.e <- ranger(W~., max.depth=8, data=data.frame(cbind(W=W[-idx],X[-idx,])))
    e.hat <- predict(model.e, X[idx,], type="response")$predictions
    e.hat <- trimmed_ps(e.hat,p=p_threshold)
    # RANDOMIZED SETTING
    # e.hat <- rep(0.5, length(idx))


    # Compute IPW scores
    ipw.scores <- Y[idx] * (W[idx]/e.hat - (1-W[idx])/(1-e.hat))
    ipw.scores
  })
  ipw.scores<- unname(do.call(c, ipw.scores))

  # second stage
  tau.scores <- lapply(indices, function(idx) {
    # Fitting the outcome model
    model.e <- ranger(W[-idx]~., max.depth=8, data=data.frame(cbind(W[-idx],X[-idx,])))

    outcome.model <- ranger(Y~., max.depth=8, data=data.frame(cbind(Y=ipw.scores[-idx],X[-idx,])))
    tau.hat <- predict(outcome.model, X[idx,], type="response")$predictions
    tau.hat
  })
  tau.scores<- unname(do.call(c, tau.scores))
  return(tau.scores)
  # function outputs:
  # tau.scores: IPW with randomforest tau estimate
}

# ipw estimator by lasso regression with predetermined ps threshold
ipw.lasso.estimator <- function(X,Y,W,n_fold=5,p_threshold=0.01){
  # function inputs
  # X: vector of covariates, train data
  # W: treatment variable, train data
  # Y: outcome variable, train data
```

```
# n_fold: n fold for cross-fit first stage estimations
# p_threshold: threshold for trimmed propensity score

# first stage
XX <- make_matrix_splines(X)
# A list of vectors indicating the left-out subset
n <- nrow(XX)
n.folds <- n_fold
# indices <- split(seq(n), sort(seq(n) %% n.folds))
foldid <- rep.int(1:n.folds,times = ceiling(n/n.folds))[sample.int(n)] #
 define folds indices
indices <- split(1:n, foldid)  #split observation indices into folds
ipw.scores <- lapply(indices, function(idx) {

  # Fitting the propensity score model
  # Comment / uncomment the lines below as appropriate.
  # OBSERVATIONAL SETTING (with unconfoundedness+overlap):
  model.e <- cv.glmnet(XX[-idx,], W[-idx], family="binomial")
  e.hat <- predict(model.e, XX[idx,], s="lambda.min", type="response")
  e.hat <- trimmed_ps(e.hat,p=p_threshold)
  # RANDOMIZED SETTING
  # e.hat <- rep(0.5, length(idx))

  # Compute IPW scores
  ipw.scores <- Y[idx] * (W[idx]/e.hat - (1-W[idx])/(1-e.hat))

  ipw.scores
})
ipw.scores<- unname(do.call(c, ipw.scores))

# second stage
 tau.scores <- lapply(indices, function(idx) {
  # Fitting the outcome model
  outcome.model <- cv.glmnet(x=XX[-idx,], y=ipw.scores[-idx], family="
 gaussian")
  tau.hat <- predict(outcome.model, XX[idx,], s = "lambda.min", type="
 response")
  tau.hat
})
```

```
178   tau.scores<- unname(do.call(c, tau.scores))
179   return(tau.scores)
180   # function outputs:
181   # tau.scores: IPW with lasso regression tau estimate
182
183 }
184
185 # 2c. Augmented inverse propensity-weighted (AIPW) estimator -------------
186
187 # aipw estimator by random forest regression with predetermined ps
      threshold
188 aipw.rf.estimator <- function(X,Y,W,n_fold=5,p_threshold=0.01){
189   # function inputs
190   # X: vector of covariates, train data
191   # W: treatment variable, train data
192   # Y: outcome variable, train data
193   # n_fold: n fold for cross-fit first stage estimations
194   # p_threshold: threshold for trimmed propensity score
195
196   # first stage
197
198   # A list of vectors indicating the left-out subset
199   n <- nrow(X)
200   n.folds <- n_fold
201   # indices <- split(seq(n), sort(seq(n) %% n.folds))
202   foldid <- rep.int(1:n.folds,times = ceiling(n/n.folds))[sample.int(n)] #
      define folds indices
203   indices <- split(1:n, foldid)  #split observation indices into folds
204
205   # Preparing data
206   Y <- Y
207   W <- W
208   X <- X
209   data <- data.frame(Y,W,X)
210   covariates <- paste0("X",1:ncol(X))
211   treatment <- "W"
212
213   # # Matrix of (transformed) covariates used to estimate E[Y|X,W]
```

```
214  # fmla.xw <- formula(paste("~ 0 +", paste0("bs(", covariates, ", df=3)",
       "*", treatment, collapse=" + ")))
215  # XW <- model.matrix(fmla.xw, data)
216  XW <- data.frame(cbind(X,W))
217  # Matrix of (transformed) covariates used to predict E[Y|X,W=w] for each
       w in {0, 1}
218  data.1 <- XW
219  data.1[,treatment] <- 1
220  XW1 <- data.1
221  # XW1 <- model.matrix(fmla.xw, data.1)  # setting W=1
222  data.0 <- XW
223  data.0[,treatment] <- 0
224  XW0 <- data.0
225  # XW0 <- model.matrix(fmla.xw, data.0)  # setting W=0
226
227  # # Matrix of (transformed) covariates used to estimate and predict e(X)
       = P[W=1|X]
228  # fmla.x <- formula(paste(" ~ 0 + ", paste0("bs(", covariates, ", df=3)",
         collapse=" + ")))
229  # XX <- model.matrix(fmla.x, data)
230  #
231  # Cross-fitted estimates of E[Y|X,W=1], E[Y|X,W=0] and e(X) = P[W=1|X]
232  mu.hat.1 <- rep(NA, n)
233  mu.hat.0 <- rep(NA, n)
234  e.hat <- rep(NA, n)
235
236  for (idx in indices) {
237    # Estimate outcome model and propensity models
238    # Note how cross-validation is done (via cv.glmnet) within cross-
       fitting!
239    outcome.model <- ranger(Y~., max.depth=8, data=data.frame(cbind(Y=Y[-
       idx],XW[-idx,])))
240    propensity.model <- ranger(W~., max.depth=8, data=data.frame(cbind(W=W
       [-idx],X[-idx,])))
241
242    # Predict with cross-fitting
243    mu.hat.1[idx] <- predict(outcome.model, XW1[idx,], type="response")$
       predictions
```

```
    mu.hat.0[idx] <- predict(outcome.model, XW0[idx,], type="response")$
    predictions
    e.hat[idx] <- predict(propensity.model, X[idx,], type="response")$
    predictions
  }

  # Commpute the summand in AIPW estimator
  aipw.scores <- (mu.hat.1 - mu.hat.0
                  + W / e.hat * (Y -  mu.hat.1)
                  - (1 - W) / (1 - e.hat) * (Y -  mu.hat.0))

  # second stage
  tau.scores <- lapply(indices, function(idx) {
    # Fitting the outcome model
    outcome.model <- ranger(Y~., max.depth=8, data=data.frame(cbind(Y=aipw.
    scores[-idx],XW[-idx,])))
    tau.hat <- predict(outcome.model, XW[idx,],type="response")$predictions
    tau.hat
  })
  tau.scores<- unname(do.call(c, tau.scores))
  return(tau.scores)
  # function outputs:
  # tau.scores: IPW with lasso regression tau estimate
}

# aipw estimator by lasso regression with predetermined ps threshold
aipw.lasso.estimator <- function(X,Y,W,n_fold=5,p_threshold=0.01){
  # function inputs
  # X: vector of covariates, train data
  # W: treatment variable, train data
  # Y: outcome variable, train data
  # n_fold: n fold for cross-fit first stage estimations
  # p_threshold: threshold for trimmed propensity score

  # first stage

  XX <- make_matrix_splines(X)
  # A list of vectors indicating the left-out subset
  n <- nrow(XX)
```

```
n.folds <- n_fold
# indices <- split(seq(n), sort(seq(n) %% n.folds))
foldid <- rep.int(1:n.folds,times = ceiling(n/n.folds))[sample.int(n)] # define folds indices
indices <- split(1:n, foldid)  #split observation indices into folds

# Preparing data
Y <- Y
W <- W
X <- X
data <- data.frame(Y,W,X)
covariates <- paste0("X",1:ncol(X))
treatment <- "W"

# Matrix of (transformed) covariates used to estimate E[Y|X,W]
fmla.xw <- formula(paste("~ 0 +", paste0("bs(", covariates, ", df=3)", "*", treatment, collapse=" + ")))
XW <- model.matrix(fmla.xw, data)
# Matrix of (transformed) covariates used to predict E[Y|X,W=w] for each w in {0, 1}
data.1 <- data
data.1[,treatment] <- 1
XW1 <- model.matrix(fmla.xw, data.1)  # setting W=1
data.0 <- data
data.0[,treatment] <- 0
XW0 <- model.matrix(fmla.xw, data.0)  # setting W=0

# Matrix of (transformed) covariates used to estimate and predict e(X) = P[W=1|X]
fmla.x <- formula(paste(" ~ 0 + ", paste0("bs(", covariates, ", df=3)", collapse=" + ")))
XX <- model.matrix(fmla.x, data)

# (Optional) Not penalizing the main effect (the coefficient on W)
penalty.factor <- rep(1, ncol(XW))
penalty.factor[colnames(XW) == treatment] <- 0

# Cross-fitted estimates of E[Y|X,W=1], E[Y|X,W=0] and e(X) = P[W=1|X]
mu.hat.1 <- rep(NA, n)
```

```
  mu.hat.0 <- rep(NA, n)
  e.hat <- rep(NA, n)
  for (idx in indices) {
    # Estimate outcome model and propensity models
    # Note how cross-validation is done (via cv.glmnet) within cross-fitting!
    outcome.model <- cv.glmnet(x=XW[-idx,], y=Y[-idx], family="gaussian", penalty.factor=penalty.factor)
    propensity.model <- cv.glmnet(x=XX[-idx,], y=W[-idx], family="binomial")

    # Predict with cross-fitting
    mu.hat.1[idx] <- predict(outcome.model, newx=XW1[idx,], type="response")
    mu.hat.0[idx] <- predict(outcome.model, newx=XW0[idx,], type="response")
    e.hat[idx] <- predict(propensity.model, newx=XX[idx,], type="response")
  }

  # Commpute the summand in AIPW estimator
  aipw.scores <- (mu.hat.1 - mu.hat.0
                  + W / e.hat * (Y -  mu.hat.1)
                  - (1 - W) / (1 - e.hat) * (Y -  mu.hat.0))

  # second stage
  tau.scores <- lapply(indices, function(idx) {
    # Fitting the outcome model
    outcome.model <- cv.glmnet(x=XX[-idx,], y=aipw.scores[-idx], family="gaussian")
    tau.hat <- predict(outcome.model, XX[idx,], s = "lambda.min", type="response")
    tau.hat
  })
  tau.scores<- unname(do.call(c, tau.scores))
  return(tau.scores)
  # function outputs:
  # tau.scores: IPW with lasso regression tau estimate
}

```

```
346 # 2d. DML estimator -----------------------------------------------------
347
348 # estimate W residual
349 wreg <- function(X_train,W,X_test){
350   wfit=ranger(W~., max.depth=8, data=data.frame(cbind(W,X_train)))
351   predict(wfit, data=X_test)$predictions
352 } #ML method=Forest
353
354 # estimate Y residual
355 yreg <- function(X_train,Y,X_test){
356   yfit=ranger(Y~., max.depth=8, data=data.frame(cbind(Y,X_train)))
357   predict(yfit, data=X_test)$predictions
358 } #ML method=Forest
359
360 # cross-fitting residuals
361 DML.LearnResiduals <- function(X, W, Y, nfold=5) {
362   nobs <- nrow(X) #number of observations
363   foldid <- rep.int(1:nfold,times = ceiling(nobs/nfold))[sample.int(nobs)]
      #define folds indices
364   Id <- split(1:nobs, foldid)  #split observation indices into folds
365   ytil <- wtil <- rep(NA, nobs)
366   # cat("fold: ")
367   for(b in 1:length(Id)){
368     what <- wreg(X_train=X[-Id[[b]],], W=W[-Id[[b]]], X_test = X[Id[[b]],]
      ) #take a fold out
369     yhat <- yreg(X_train=X[-Id[[b]],], Y=Y[-Id[[b]]], X_test = X[Id[[b]],]
      ) # take a fold out
370     wtil[Id[[b]]] <- (W[Id[[b]]] - what) #record residual for the left-out
      fold
371     ytil[Id[[b]]] <- (Y[Id[[b]]] - yhat) #record residual for the left-out
      fold
372     # cat(b," ")
373   }
374   #
375   # cat(sprintf("Controls explain %g per cent of variance of Outcome",
      round( max(1-var(ytil)/var(y),0)*100, digits=3)) )
376   # cat(sprintf("\n Controls explain %g per cent of variance of Treatment",
        round( max(1-var(wtil)/var(W),0)*100, digits=3)) )
377   return( list(wtil=wtil, ytil=ytil) ) #save output and residuals
```

```
}

# double lasso regression ytil on wtil
# with b term as covariates as main outcome and multiplied on wtil
DML.CATE.DLasso<- function(wtil, ytil, b, name="wtil"){
  # ytil is y resid; #wtil is W resid
  # name is name of variable whose coefficient we want to infer
  # name could be a list of names
  # name=grep("wtil", colnames(X.lasso))[1:3] will give the first three
    coefficiens
  ytil = ytil
  wtil = wtil
  b <- b[, which(apply(b, 2, var) != 0)] # exclude all constant variables
  demean<- function (x){ x- mean(x)}
  b<- apply(as.matrix(b), 2, FUN=demean)
  X.lasso= model.matrix( ~ wtil+ wtil:b + b)
  # index.treatment = name
  # if all interactons are of interest write:
  index.treatment = grep(name, colnames(X.lasso))
  effects.treatment <- rlassoEffects(x = X.lasso, y = ytil, index = index.
    treatment, post=FALSE)
  result=summary(effects.treatment)
  return(coef=result$coef)
}

# double/debiased machine learning estimator
dml.estimator <- function(Y,W,X){
  # function inputs
  # X: vector of covariates, test data
  # W: treatment variable, test data
  # Y: outcome variable, test data
    set.seed(1)
  res <- DML.LearnResiduals(X,W,Y)
  B=X
  result <- DML.CATE.DLasso(wtil=res$wtil,ytil=res$ytil, b=B,
                            name="wtil")
  B <- B[, which(apply(B, 2, var) != 0)] # exclude all constant variables
  data <- cbind(Y=Y,W=W,B) %>% as.data.frame()
  data2 <- stats::model.matrix(~1 + ., data.frame(data[,-c(1,2)]))
```

```
  tau_hat <- data2 %*% result[,1]%>% as.vector()
  se_hat <- data2 %*% result[,2]%>% as.vector()
  return( list(tau_hat=tau_hat, se_hat=se_hat) )
  # function outputs:
  # tau_hat: dml tau estimate
  # se_hat: dml standard errors of tau estimate
}

# import function created in python for estimate CATE by DoubleML package
# Python: Conditional Average Treatment Effects (CATEs) - DoubleML
    documentation
# https://docs.doubleml.org/stable/examples/py_double_ml_cate.html
# Sys.which("python")
# use_python("C:\\Users\\renat\\AppData\\Local\\Programs\\Python\\PYTHON~
    1\\python.exe")
source_python("dml.py")
# dml_py(data_train_full = dados_train_full,
#                 data_train = dados_train,
#                 outcome = "Y",treatment = "W",
#                 covariates = cova_train,data_test = dados_test)
# function inputs
# data_train_full: vector of outcome + treatment + covariates (this order),
     train data
# data_train: vector of covariates, X_train, train data
# outcome: outcome variable name
# treatment: treatment variable name
# covariates: vector of covariates name
# data_test: vector of covariates, X_test, test data

# function outputs:
# tau_hat: dml tau estimate
# se_hat: dml standard errors of tau estimate


# SIMULATION ------------------------------------------------------------

simu.fun = function(n,d,sigma,setup) {
  # function inputs
  # n: sample size
```

```
451  # d: number of covariates
452  # sigma: variance of error term
453  # setup: data generating process (DGP) setup choice
454
455  # function to choose the data generating process (DGP) parameters
456  # the output is a list with X=X, b=b, tau=tau, e=e parameters
457  if (setup == 'A') {#linear setup
458    get.params = function() {
459      X = matrix(runif(n * d, min=0, max=1), n, d) # covariates
460      b = X[,1] + 0.5*X[,2] + 0.4*X[,3] # baseline main effect
461      tau = 0.3*X[,1] + 0.6*X[,2] + 0.6*X[,3] + 2.2*X[,4] # effect = tau(
     CATE)
462      eta = 0.1
463      e = pmax(eta, pmin(sin(pi * X[,1] * X[,2]), 1-eta)) #propensity score
      trimmed eta 0.1
464      list(X=X, b=b, tau=tau, e=e)
465    }
466  } else if (setup == 'B') {# nonlinear setup
467    get.params = function() {
468      X = matrix(runif(n*d, min=0, max=1), n, d) # covariates
469      b = 0.2*X[,1] + X[,1]^2 + 0.5*X[,2]*X[,3] + 0.4*X[,3] + 0.8*X[,3]^2 #
      baseline main effect
470      eta = 0.1
471      e = pmax(eta, pmin(X[,1]- 0.2*X[,2]^2, 1-eta)) #propensity score
     trimmed eta 0.1
472      tau = 0.3*X[,1] + 0.6*X[,2] + 0.6*X[,3] + 2.2*X[,4]  # effect = tau(
     CATE)
473      list(X=X, b=b, tau=tau, e=e)
474    }
475  } else if (setup == 'C') {# peaks and valleys setup
476    get.params = function() {
477      X = matrix(runif(n * d, min=0, max=1), n, d) # covariates
478      b = 0 # baseline main effect
479      tau = (1 + 1 / (1 + exp(-20 * (X[,1] - 1/3)))) * (1 + 1 / (1 + exp
     (-20 * (X[,2] - 1/3))))* (1 + 1 / (1 + exp(-20 * (X[,3] - 1/3)))) #
     effect = tau(CATE)
480      e = 0.5*X[,1] + 0.5*X[,2] #propensity score
481      list(X=X, b=b, tau=tau, e=e)
482    }
```

```
} else if (setup == 'D') {# discontinuities setup
  get.params = function() {
    X = matrix(rnorm(n * d), n, d) # covariates
    b = 2 * (X[,1]>0.4) + 0.3*X[,2] # baseline main effect
    eta = 0.1
    e = pmax(eta, pmin(0.7*X[,1]- 0.7*X[,2]+ 0.7*X[,3], 1-eta)) #
  propensity score trimmed eta 0.1
    tau = 2 * (X[,1]>0.6) + 1.5 * (X[,2]>0.6) + 0.3*X[,3] - 0.7*X[,4]  #
  effect = tau(CATE)
    list(X=X, b=b, tau=tau, e=e)
  }
} else {

  stop("bad setup")
}
params_train = get.params()
W_train = rbinom(n, 1, params_train$e)
Y_train = params_train$b + (W_train - 0.5) * params_train$tau + sigma *
  rnorm(n)

params_test = get.params()
W_test = rbinom(n, 1, params_test$e)
Y_test = params_test$b + (W_test - 0.5) * params_test$tau + sigma * rnorm
  (n)

make_matrix = function(x) stats::model.matrix(~.-1, x)

X_train = make_matrix(data.frame(params_train$X))
X_test = make_matrix(data.frame(params_test$X))

# causal forest honest split
# estimate causal forest
cf = causal_forest(X_train, Y_train, W_train,num.trees = ntree,min.node.
  size = min_node)
# prediction causal forest
cf.pred <- predict(cf, X_test, estimate.variance = TRUE)

# se estimate causal forest
se.hat = sqrt(cf.pred$variance.estimates)
```

```
517 # cover rate
518 cf.cov = abs(cf.pred$predictions - params_test$tau) <= 1.96 * se.hat
519 cf.covered = mean(cf.cov)
520
521 # mse
522 cf.mse = mean((cf.pred$predictions - params_test$tau)^2)
523 # bias
524 cf.bias = mean(abs(cf.pred$predictions - params_test$tau))
525
526 # same for causal forest adaptative split
527 cf.adapt = causal_forest(X_train, Y_train, W_train,num.trees = ntree,min.
     node.size = min_node,honesty = F)
528 cf.pred.adapt <- predict(cf.adapt, X_test, estimate.variance = TRUE)
529
530 se.hat.adapt = sqrt(cf.pred.adapt$variance.estimates)
531 cf.cov.adapt = abs(cf.pred.adapt$predictions - params_test$tau) <= 1.96 *
       se.hat.adapt
532 cfadapt.covered = mean(cf.cov.adapt)
533
534 cfadapt.mse = mean((cf.pred.adapt$predictions - params_test$tau)^2)
535 cfadapt.bias = mean(abs(cf.pred.adapt$predictions - params_test$tau))
536
537 # same for 10-nearest neighbour
538 k.small = 10
539 knn.small=causal.kn(kn=k.small,X_train,W_train,X_test,Y_train)
540 knn.cov = abs(knn.small$knn.tau - params_test$tau) <= 1.96 * knn.small$
     knn.se
541 knn.covered = mean(knn.cov)
542
543 knn.mse = mean((knn.small$knn.tau - params_test$tau)^2)
544 knn.bias = mean(abs(knn.small$knn.tau - params_test$tau))
545
546 # same for 100-nearest neighbour
547 k.big = 100
548 knn.big=causal.kn(kn=k.big,X_train,W_train,X_test,Y_train)
549 knnbig.cov = abs(knn.big$knn.tau - params_test$tau) <= 1.96 * knn.big$knn
     .se
550 knnbig.covered = mean(knnbig.cov)
551
```

```
552 knnbig.mse = mean((knn.big$knn.tau - params_test$tau)^2)
553 knnbig.bias = mean(abs(knn.big$knn.tau - params_test$tau))
554
555 # # same for inverse probability weight estimator (IPW) with lasso
556 # ipwlasso.pred = ipw.lasso.estimator(X_test,Y_test,W_test)
557 # ipwlasso.mse = mean((ipw.pred - params_test$tau)^2)
558 # ipwlasso.bias = mean(abs(ipw.pred - params_test$tau))
559
560 # same for inverse probability weight estimator (IPW) with random forest
561 ipwrf.pred = ipw.rf.estimator(X_test,Y_test,W_test)
562 ipwrf.mse = mean((ipwrf.pred - params_test$tau)^2)
563 ipwrf.bias = mean(abs(ipwrf.pred - params_test$tau))
564
565 # # same for augmented inverse probability weight estimator (AIPW) with
     lasso
566 # aipwlasso.pred = aipw.lasso.estimator(X_test,Y_test,W_test)
567 # aipwlasso.mse = mean((aipw.pred - params_test$tau)^2)
568 # aipwlasso.bias = mean(abs(aipw.pred - params_test$tau))
569
570 # same for augmented inverse probability weight estimator (AIPW) with
     random forest
571 aipwrf.pred = aipw.rf.estimator(X_test,Y_test,W_test)
572 aipwrf.mse = mean((aipwrf.pred - params_test$tau)^2)
573 aipwrf.bias = mean(abs(aipwrf.pred - params_test$tau))
574
575 # same for double/debiased machine learning (DML)
576 dml.pred = dml.estimator(X=X_test,W=W_test,Y=Y_test)$tau_hat
577 dml.mse = mean((dml.pred - params_test$tau)^2)
578 dml.bias = mean(abs(dml.pred - params_test$tau))
579
580 # same for double/debiased machine learning (DML) by DoubleML python
     package
581 dados_train_full <- data.frame(Y=Y_train,W=W_train,X_train)
582 cova_train=colnames(X_train)
583 dados_train <- data.frame(X_train)
584 dados_test <- data.frame(X_test)
585 dmlpy_results <- dml_py(data_train_full = dados_train_full,
586                         data_train = dados_train,
587                         outcome = "Y",treatment = "W",
```

```
                  covariates = cova_train,data_test = dados_test,nboot =
  nbootdml)

# lista <- c(200,400,600,1000,2000) %>% as.integer()
# for(i in lista){
#   start_time <- Sys.time()
#   dmlpy_results <- dml_py(data_train_full = dados_train_full,
#                           data_train = dados_train,
#                           outcome = "Y",treatment = "W",
#                           covariates = cova_train,data_test = dados_
  test,nboot = i)
#
#   end_time <- Sys.time()
#   time_taken <- difftime(end_time, start_time, units='mins')
#   print(time_taken)
# }

# prediction dml
dmlpy.pred = dmlpy_results$effect

# cover rate
dmlpy.cov = dmlpy_results$effect >= dmlpy_results$`2.5 %` &
            dmlpy_results$effect <= dmlpy_results$`97.5 %`
dmlpy.covered = mean(dmlpy.cov)

# mse
dmlpy.mse = mean((dmlpy.pred - params_test$tau)^2)
# bias
dmlpy.bias = mean(abs(dmlpy.pred - params_test$tau))

c(
  cf_covered = cf.covered,
  cf_mse = cf.mse,
  cf_bias = cf.bias,
  cfadapt_covered = cfadapt.covered,
  cfadapt_mse = cfadapt.mse,
  cfadapt_bias = cfadapt.bias,
  knn_covered = knn.covered,
  knn_mse = knn.mse,
```

```
625     knn_bias = knn.bias,
626     knnbig_covered = knnbig.covered,
627     knnbig_mse = knnbig.mse,
628     knnbig_bias = knnbig.bias,
629     # ipwlasso_mse = ipwlasso.mse,
630     # ipwlasso_bias = ipwlasso.bias,
631     ipwrf_mse = ipwrf.mse,
632     ipwrf_bias = ipwrf.bias,
633     aipwrf_mse = aipwrf.mse,
634     aipwrf_bias = aipwrf.bias,
635     # aipwlasso_mse = aipwlasso.mse,
636     # aipwlasso_bias = aipwlasso.bias,
637     dml_mse = dml.mse,
638     dml_bias = dml.bias,
639     dmlpy_covered = dmlpy.covered,
640     dmlpy_mse = dmlpy.mse,
641     dmlpy_bias = dmlpy.bias
642     )
643   # function outputs
644   # cover rate (covered), mean squared error (mse) and bias (bias)
645   # for all estimation methods:
646   # causal forest, honest (cf) and adaptative split method (cfadapt)
647   # k-nearest neighbor with a small (knn) and a big k (knnbig)
648   # inverse probability weighting (ipw) and augmented (aipw)
649   # double machine learning by R (dml) e by python (dmlpy)
650 }
651
652
653 # A. Set parameters -------------------------------------------------------
654
655 sigma=3;
656 # n=500;
657 d=20
658 # s = n/2 # sample size to honest partition method in causal forest
659 ntree = 400 # number of trees(B)
660 # min_node =  round(n*0.005) # minimum size node of tree
661 nbootdml = as.integer(1000) # number of bootstrap repetition
662 simu.reps = 500  #simulation replication
663 # sigma=c(0.5,1,3)
```

```
nvals= c(500)
# nvals= c(500,2000,5000)
# dvals=c(4,10,20)
setupvals=LETTERS[1:4]

# loop for all DGP ----------------------------------------------------

for(i in nvals){
  n=i
  s = n/2 # sample size to honest partition method in causal forest
  min_node =  round(n*0.005) # minimum size node of tree
  # n=500;d=4
start_time <- Sys.time()
results.raw = lapply(setupvals, function(type) {
  print(paste("RUNNING FOR SETUP",type," WITH ",d, "COVARIATES, ",
              sigma, "SIGMA, ",
              n," SAMPLE SIZE."))
  res.d = sapply(1:simu.reps, function(iter) {
    print(paste0("starting iteration ", iter))
    simu.fun(n,d,sigma,setup = type)})
  # res.fixed = data.frame(t(res.d))
  # print(paste("RESULT AT", d, "IS", colMeans(res.fixed)))
  # res.fixed
  res.d
})
# fnm_rept = paste("results/saidarept", n, d, sigma,simu.reps,  "full.xlsx
    ", sep="-")
# write_xlsx(results.raw,path = fnm_rept)

# results.condensed = rowMeans(data.frame(results.raw)) %>% as.data.frame()
     %>%
#   rownames_to_column() %>% separate_wider_delim(rowname,
#                                                 delim = "_",
#                                                 names = c("method", "
    measure")) %>%
#   rename(value=".")

# results.condensed = lapply(results.raw, function(RR) {
#   RR.mu = colMeans(RR)
```

```
#    RR.var = sapply(RR, var) / (nrow(RR) - 1)
#    rbind("mu"=RR.mu, "se"=sqrt(RR.var))
#    })

results.condensed2 = lapply(results.raw, function(RR) {
  RR.mu = rowMeans(data.frame(RR)) %>% as.data.frame() %>%
    rownames_to_column() %>% separate_wider_delim(rowname,
                                                  delim = "_",
                                                  names = c("method", "measure")) %>%
    rename(value=".")

    })
results.condensed <- map2(results.condensed2, setupvals, ~cbind(.x, setup = .y)) %>%
  bind_rows()
end_time <- Sys.time()
time_taken <- difftime(end_time, start_time, units='mins')
results.condensed2 <- cbind(results.condensed,time_taken=as.numeric(time_taken),
      n, s, ntree, min_node, sigma,d,simu.reps)
fnm = paste("results/oficial", n, d, sigma,simu.reps,  "full.xlsx", sep="-")
write_xlsx(results.condensed2,path = fnm)
print(time_taken)
}

# B. construct tables
    ------------------------------------------------------------

# read all worksheets for selected folder
end.temp <- "C:/Users/renat/Dropbox/Dissertacao-Renato/2.simulacoes de causalidade/results2106"
setwd(end.temp)
file.list <- list.files(pattern='*.xlsx')
alldata <- file.list %>%
  map_dfr(~read_excel(.x))

# list of methods excluded in table
```

```
exclusion_list <- c("knn","knnbig","dml")

# B1. Estimation MSE for n = 2000, d = 4, sigma = 3, for all setups (A, B, C, D)
    ----------------------------------------------------------------------
# filtering table Comparing RF - Honesty Sample & Adaptive, IPW, AIPW, DML
alldata %>%
  filter(!method %in% exclusion_list,measure=="mse",
         n==2000,d==4,sigma==3) %>%
  select(method,value,setup,n) %>%
  mutate(value=round(value, 3),n=as.integer(n)) %>%
  pivot_wider(names_from = method ,values_from = value
  ) %>%
  arrange(setup,n) -> dataselected
xtab = xtable(dataselected,
              caption = "MSE médio de 500 replicações dos métodos estudados (Floresta Causal -- Amostra Honesta e Adaptativa, IPW, AIPW e DML)
        com número de covariáveis d=4, tamanho de amostra n=2000 e sigma=3."
    ,
              label = "tab:mse-cov4-n2000-sigma3",
              digits = 3,
              align = "rlr|rrrr|r")
print(xtab, include.rownames = FALSE)

# B1a1. Estimation MSE for n = 2000, d = 10, sigma = 3, for all setups (A, B, C, D)
    ----------------------------------------------------------------------
# filtering table Comparing RF - Honesty Sample & Adaptive, IPW, AIPW, DML
alldata %>%
  filter(!method %in% exclusion_list,measure=="mse",
         n==2000,d==10,sigma==3) %>%
  select(method,value,setup,n) %>%
  mutate(value=round(value, 3),n=as.integer(n)) %>%
  pivot_wider(names_from = method ,values_from = value
  ) %>%
  arrange(setup,n) -> dataselected
xtab = xtable(dataselected,
              caption = "MSE médio de 500 replicações dos métodos estudados (Floresta Causal -- Amostra Honesta e Adaptativa, IPW, AIPW e DML)
```

```
765         com número de covariáveis d=10, tamanho de amostra n=2000 e sigma=3.
    ",
766         label = "tab:mse-cov10-n2000-sigma3",
767         digits = 3,
768         align = "rlr|rrrr|r")
769 print(xtab, include.rownames = FALSE)
770
771 # B1a2. Estimation MSE for n = 2000, d = 20, sigma = 3, for all setups (A,
    B, C, D)
    ----------------------------------------------------------------------
772 # filtering table Comparing RF - Honesty Sample & Adaptive, IPW, AIPW, DML
773 alldata %>%
774   filter(!method %in% exclusion_list,measure=="mse",
775          n==2000,d==20,sigma==3) %>%
776   select(method,value,setup,n) %>%
777   mutate(value=round(value, 3),n=as.integer(n)) %>%
778   pivot_wider(names_from = method ,values_from = value
779   ) %>%
780   arrange(setup,n) -> dataselected
781 xtab = xtable(dataselected,
782               caption = "MSE médio de 500 replicações dos métodos estudados
     (Floresta Causal -- Amostra Honesta e Adaptativa, IPW, AIPW e DML)
783         com número de covariáveis d=20, tamanho de amostra n=2000 e sigma=3.
    ",
784         label = "tab:mse-cov20-n2000-sigma3",
785         digits = 3,
786         align = "rlr|rrrr|r")
787 print(xtab, include.rownames = FALSE)
788
789 # B1.2. Estimation Bias for n = 2000, d = 4, sigma = 3, for all setups (A,
    B, C, D)
    ----------------------------------------------------------------------
790 # filtering table Comparing RF - Honesty Sample & Adaptive, IPW, AIPW, DML
791 alldata %>%
792   filter(!method %in% exclusion_list,measure=="bias",
793          n==2000,d==4,sigma==3) %>%
794   select(method,value,setup,n) %>%
795   mutate(value=round(value, 3),n=as.integer(n)) %>%
796   pivot_wider(names_from = method ,values_from = value
```

```
797   ) %>%
798   arrange(setup,n) -> dataselected
799 xtab = xtable(dataselected,
800                caption = "Viés absoluto médio para 500 replicações dos
      métodos estudados (Floresta Causal -- Amostra Honesta e Adaptativa, IPW,
       AIPW e DML)
801          com número de covariáveis d=4, tamanho de amostra n=2000 e sigma=3."
      ,
802                label = "tab:bias-cov4-n2000-sigma3",
803                digits = 3,
804                align = "rlr|rrrr|r")
805 print(xtab, include.rownames = FALSE)
806
807 # B1.2a1. Estimation Bias for n = 2000, d = 10, sigma = 3, for all setups (
      A, B, C, D)
      ----------------------------------------------------------------------
808 # filtering table Comparing RF - Honesty Sample & Adaptive, IPW, AIPW, DML
809 alldata %>%
810   filter(!method %in% exclusion_list,measure=="bias",
811          n==2000,d==10,sigma==3) %>%
812   select(method,value,setup,n) %>%
813   mutate(value=round(value, 3),n=as.integer(n)) %>%
814   pivot_wider(names_from = method ,values_from = value
815   ) %>%
816   arrange(setup,n) -> dataselected
817 xtab = xtable(dataselected,
818                caption = "Viés absoluto médio para 500 replicações dos
      métodos estudados (Floresta Causal -- Amostra Honesta e Adaptativa, IPW,
       AIPW e DML)
819          com número de covariáveis d=10, tamanho de amostra n=2000 e sigma=3.
      ",
820                label = "tab:bias-cov10-n2000-sigma3",
821                digits = 3,
822                align = "rlr|rrrr|r")
823 print(xtab, include.rownames = FALSE)
824
825 # B1.2a2. Estimation Bias for n = 2000, d = 20, sigma = 3, for all setups (
      A, B, C, D)
      ----------------------------------------------------------------------
```

```
826 # filtering table Comparing RF - Honesty Sample & Adaptive, IPW, AIPW, DML
827 alldata %>%
828   filter(!method %in% exclusion_list,measure=="bias",
829          n==2000,d==20,sigma==3) %>%
830   select(method,value,setup,n) %>%
831   mutate(value=round(value, 3),n=as.integer(n)) %>%
832   pivot_wider(names_from = method ,values_from = value
833   ) %>%
834   arrange(setup,n) -> dataselected
835 xtab = xtable(dataselected,
836               caption = "Viés absoluto médio para 500 replicações dos
    métodos estudados (Floresta Causal -- Amostra Honesta e Adaptativa, IPW,
     AIPW e DML)
837        com número de covariáveis d=20, tamanho de amostra n=2000 e sigma=3.
    ",
838        label = "tab:bias-cov20-n2000-sigma3",
839        digits = 3,
840        align = "rlr|rrrr|r")
841 print(xtab, include.rownames = FALSE)
842
843 # B1.3. Covered Rate for n = 2000, d = 4, sigma = 3, for all setups (A, B,
    C, D)
    ------------------------------------------------------------------------
844
845 alldata %>%
846   filter(method %in% c("cf","cfadapt","knn","knnbig","dmlpy"),measure=="
    covered",
847          n==2000,d==4,sigma==3) %>%
848   select(method,value,setup,n) %>%
849   mutate(value=round(value, 2),n=as.integer(n)) %>%
850   pivot_wider(names_from = method ,values_from = value
851   ) %>%
852   arrange(setup,n) -> dataselected
853 xtab = xtable(dataselected,
854               caption = "Taxa de cobertura para 500 replicações dos métodos
     estudados (Floresta Causal -- Amostra Honesta e Adaptativa, KNN-10, KNN
    -100 e DML)
855        com número de covariáveis d=4, tamanho de amostra n=2000 e sigma=3."
    ,
```

```
856         label = "tab:covered-cov4-n2000-sigma3",
857         digits = 2,
858         align = "rlr|rrrr|r")
859 print(xtab, include.rownames = FALSE)
860
861 # B1.3a1. Covered Rate for n = 2000, d = 10, sigma = 3, for all setups (A,
      B, C, D)
      ----------------------------------------------------------------------
862
863 alldata %>%
864   filter(method %in% c("cf","cfadapt","knn","knnbig","dmlpy"),measure=="
      covered",
865          n==2000,d==10,sigma==3) %>%
866   select(method,value,setup,n) %>%
867   mutate(value=round(value, 2),n=as.integer(n)) %>%
868   pivot_wider(names_from = method ,values_from = value
869   ) %>%
870   arrange(setup,n) -> dataselected
871 xtab = xtable(dataselected,
872               caption = "Taxa de cobertura para 500 replicações dos métodos
       estudados (Floresta Causal -- Amostra Honesta e Adaptativa, KNN-10, KNN
      -100 e DML)
873         com número de covariáveis d=10, tamanho de amostra n=2000 e sigma=3.
      ",
874               label = "tab:covered-cov10-n2000-sigma3",
875               digits = 2,
876               align = "rlr|rrrr|r")
877 print(xtab, include.rownames = FALSE)
878
879 # B1.3a2. Covered Rate for n = 2000, d = 20, sigma = 3, for all setups (A,
      B, C, D)
      ----------------------------------------------------------------------
880
881 alldata %>%
882   filter(method %in% c("cf","cfadapt","knn","knnbig","dmlpy"),measure=="
      covered",
883          n==2000,d==20,sigma==3) %>%
884   select(method,value,setup,n) %>%
885   mutate(value=round(value, 2),n=as.integer(n)) %>%
```

```
886   pivot_wider(names_from = method ,values_from = value
887   ) %>%
888   arrange(setup,n) -> dataselected
889 xtab = xtable(dataselected,
890               caption = "Taxa de cobertura para 500 replicações dos métodos
      estudados (Floresta Causal -- Amostra Honesta e Adaptativa, KNN-10, KNN
      -100 e DML)
891         com número de covariáveis d=20, tamanho de amostra n=2000 e sigma=3.
      ",
892         label = "tab:covered-cov20-n2000-sigma3",
893         digits = 2,
894         align = "rlr|rrrr|r")
895 print(xtab, include.rownames = FALSE)
896
897 # B2. Consistency MSE for d = 4, sigma = 3, for all setups (A, B, C, D) and
       n=500,2000,5000
      ------------------------------------------------------------------------
898 alldata %>%
899   filter(!method %in% exclusion_list,measure=="mse",
900          d==4,sigma==3) %>%
901   select(method,value,setup,n) %>%
902   mutate(value=round(value, 3),n=as.integer(n)) %>%
903   pivot_wider(names_from = method ,values_from = value
904   ) %>%
905   arrange(setup,n) -> dataselected
906 xtab = xtable(dataselected,
907               caption = "MSE médio de 500 replicações dos métodos estudados
      (Floresta Causal -- Amostra Honesta e Adaptativa, IPW, AIPW e DML)
908         por tamanho de amostra com número de covariáveis d=4 e sigma=3.",
909               label = "tab:mse-por-amostra-cov4",
910               digits = 3,
911               align = "rlr|rrrr|r")
912 print(xtab, include.rownames = FALSE)
913
914 # B2a1. Consistency MSE for d = 10, sigma = 3, for all setups (A, B, C, D)
      and n=500,2000,5000
      ------------------------------------------------------------------------
915 alldata %>%
916   filter(!method %in% exclusion_list,measure=="mse",
```

```
917           d==10,sigma==3) %>%
918   select(method,value,setup,n) %>%
919   mutate(value=round(value, 3),n=as.integer(n)) %>%
920   pivot_wider(names_from = method ,values_from = value
921   ) %>%
922   arrange(setup,n) -> dataselected
923 xtab = xtable(dataselected,
924                 caption = "MSE médio de 500 replicações dos métodos estudados
      (Floresta Causal -- Amostra Honesta e Adaptativa, IPW, AIPW e DML)
925         por tamanho de amostra com número de covariáveis d=10 e sigma=3.",
926         label = "tab:mse-por-amostra-cov10",
927         digits = 3,
928         align = "rlr|rrrr|r")
929 print(xtab, include.rownames = FALSE)
930
931 # B2a2. Consistency MSE for d = 20, sigma = 3, for all setups (A, B, C, D)
      and n=500,2000,5000
      ------------------------------------------------------------------------
932 alldata %>%
933   filter(!method %in% exclusion_list,measure=="mse",
934           d==20,sigma==3) %>%
935   select(method,value,setup,n) %>%
936   mutate(value=round(value, 3),n=as.integer(n)) %>%
937   pivot_wider(names_from = method ,values_from = value
938   ) %>%
939   arrange(setup,n) -> dataselected
940 xtab = xtable(dataselected,
941                 caption = "MSE médio de 500 replicações dos métodos estudados
      (Floresta Causal -- Amostra Honesta e Adaptativa, IPW, AIPW e DML)
942         por tamanho de amostra com número de covariáveis d=20 e sigma=3.",
943         label = "tab:mse-por-amostra-cov20",
944         digits = 3,
945         align = "rlr|rrrr|r")
946 print(xtab, include.rownames = FALSE)
947
948 # B2.2. Consistency Bias for d = 4, sigma = 3, for all setups (A, B, C, D)
      and n=500,2000,5000
      ------------------------------------------------------------------------
949 alldata %>%
```

```
950   filter(!method %in% exclusion_list, measure=="bias",
951          d==4,sigma==3) %>%
952   select(method,value,setup,n) %>%
953   mutate(value=round(value, 3),n=as.integer(n)) %>%
954   pivot_wider(names_from = method ,values_from = value
955   ) %>%
956   arrange(setup,n) -> dataselected
957 xtab = xtable(dataselected,
958                caption = "Viés absoluto médio de 500 replicações dos métodos
      estudados (Floresta Causal -- Amostra Honesta e Adaptativa, IPW, AIPW e
      DML)
959        por tamanho de amostra com número de covariáveis d=4 e sigma=3.",
960                label = "tab:bias-por-amostra-cov4",
961                digits = 3,
962                align = "rlr|rrrr|r")
963 print(xtab, include.rownames = FALSE)
964
965 # B2.2a1. Consistency Bias for d = 10, sigma = 3, for all setups (A, B, C,
      D) and n=500,2000,5000
      ------------------------------------------------------------------------
966 alldata %>%
967   filter(!method %in% exclusion_list, measure=="bias",
968          d==10,sigma==3) %>%
969   select(method,value,setup,n) %>%
970   mutate(value=round(value, 3),n=as.integer(n)) %>%
971   pivot_wider(names_from = method ,values_from = value
972   ) %>%
973   arrange(setup,n) -> dataselected
974 xtab = xtable(dataselected,
975                caption = "Viés absoluto médio de 500 replicações dos métodos
      estudados (Floresta Causal -- Amostra Honesta e Adaptativa, IPW, AIPW e
      DML)
976        por tamanho de amostra com número de covariáveis d=10 e sigma=3.",
977        label = "tab:bias-por-amostra-cov10",
978        digits = 3,
979        align = "rlr|rrrr|r")
980 print(xtab, include.rownames = FALSE)
981
```

```
# B2.2a2. Consistency Bias for d = 20, sigma = 3, for all setups (A, B, C, D) and n=500,2000,5000
    ----------------------------------------------------------------------
alldata %>%
  filter(!method %in% exclusion_list, measure=="bias",
         d==20,sigma==3) %>%
  select(method,value,setup,n) %>%
  mutate(value=round(value, 3),n=as.integer(n)) %>%
  pivot_wider(names_from = method ,values_from = value
  ) %>%
  arrange(setup,n) -> dataselected
xtab = xtable(dataselected,
              caption = "Viés absoluto médio de 500 replicações dos métodos estudados (Floresta Causal -- Amostra Honesta e Adaptativa, IPW, AIPW e DML)
       por tamanho de amostra com número de covariáveis d=20 e sigma=3.",
       label = "tab:bias-por-amostra-cov20",
       digits = 3,
       align = "rlr|rrrr|r")
print(xtab, include.rownames = FALSE)

# B3. Covered Rate for d = 4, sigma = 3, for all setups (A, B, C, D) and n=500,2000,5000
    ----------------------------------------------------------------------
alldata %>%
  filter(method %in% c("cf","cfadapt","knn","knnbig","dmlpy"),measure=="covered",
         d==4,sigma==3) %>%
  select(method,value,setup,n) %>%
  mutate(value=round(value, 2),n=as.integer(n)) %>%
  pivot_wider(names_from = method ,values_from = value
  ) %>%
  arrange(setup,n) -> dataselected
xtab = xtable(dataselected,
              caption = "Taxa de cobertura de 500 replicações dos métodos estudados (Floresta Causal -- Amostra Honesta e Adaptativa, KNN-10, KNN-100 e DML)
       por tamanho de amostra com número de covariáveis d=4 e sigma=3.",
              label = "tab:covered-por-amostra-cov4",
```

```
1012              digits = 2,
1013              align = "rlr|rrrr|r")
1014 print(xtab, include.rownames = FALSE)
1015
1016 # B3a1. Covered Rate for d = 10, sigma = 3, for all setups (A, B, C, D) and
       n=500,2000,5000
     ------------------------------------------------------------------------
1017 alldata %>%
1018   filter(method %in% c("cf","cfadapt","knn","knnbig","dmlpy"),measure==""
       covered",
1019          d==10,sigma==3) %>%
1020   select(method,value,setup,n) %>%
1021   mutate(value=round(value, 2),n=as.integer(n)) %>%
1022   pivot_wider(names_from = method ,values_from = value
1023   ) %>%
1024   arrange(setup,n) -> dataselected
1025 xtab = xtable(dataselected,
1026              caption = "Taxa de cobertura de 500 replicações dos métodos
       estudados (Floresta Causal -- Amostra Honesta e Adaptativa, KNN-10, KNN
       -100 e DML)
1027        por tamanho de amostra com número de covariáveis d=10 e sigma=3.",
1028        label = "tab:covered-por-amostra-cov10",
1029        digits = 2,
1030        align = "rlr|rrrr|r")
1031 print(xtab, include.rownames = FALSE)
1032
1033 # B3a2. Covered Rate for d = 20, sigma = 3, for all setups (A, B, C, D) and
       n=500,2000,5000
     ------------------------------------------------------------------------
1034 alldata %>%
1035   filter(method %in% c("cf","cfadapt","knn","knnbig","dmlpy"),measure==""
       covered",
1036          d==20,sigma==3) %>%
1037   select(method,value,setup,n) %>%
1038   mutate(value=round(value, 2),n=as.integer(n)) %>%
1039   pivot_wider(names_from = method ,values_from = value
1040   ) %>%
1041   arrange(setup,n) -> dataselected
1042 xtab = xtable(dataselected,
```

```
1043              caption = "Taxa de cobertura de 500 replicações dos métodos
    estudados (Floresta Causal -- Amostra Honesta e Adaptativa, KNN-10, KNN
    -100 e DML)
1044        por tamanho de amostra com número de covariáveis d=20 e sigma=3.",
1045        label = "tab:covered-por-amostra-cov20",
1046        digits = 2,
1047        align = "rlr|rrrr|r")
1048 print(xtab, include.rownames = FALSE)
1049
1050 # B4. Dimensionality MSE for n = 2000, sigma = 3, for all setups (A, B, C,
    D) and d=4,10,20
    ------------------------------------------------------------------------
1051 alldata %>%
1052   filter(!method %in% exclusion_list,measure=="mse",
1053          n==2000,sigma==3) %>%
1054   select(method,value,setup,d) %>%
1055   mutate(value=round(value, 3),d=as.integer(d)) %>%
1056   pivot_wider(names_from = method ,values_from = value
1057   ) %>%
1058   arrange(setup,d) -> dataselected
1059 xtab = xtable(dataselected,
1060              caption = "MSE médio de 500 replicações dos métodos estudados
    (Floresta Causal -- Amostra Honesta e Adaptativa, IPW, AIPW e DML)
1061        por número de covariáveis com tamanho de amostra n=2000 e sigma=3.",
1062              label = "tab:mse-por-cov-n2000",
1063              digits = 3,
1064              align = "rlr|rrrr|r")
1065 print(xtab, include.rownames = FALSE)
1066
1067 # B4a1. Dimensionality MSE for n = 500, sigma = 3, for all setups (A, B, C,
    D) and d=4,10,20
    ------------------------------------------------------------------------
1068 alldata %>%
1069   filter(!method %in% exclusion_list,measure=="mse",
1070          n==500,sigma==3) %>%
1071   select(method,value,setup,d) %>%
1072   mutate(value=round(value, 3),d=as.integer(d)) %>%
1073   pivot_wider(names_from = method ,values_from = value
1074   ) %>%
```

```
1075   arrange(setup,d) -> dataselected
1076 xtab = xtable(dataselected,
1077                caption = "MSE médio de 500 replicações dos métodos estudados
       (Floresta Causal -- Amostra Honesta e Adaptativa, IPW, AIPW e DML)
1078          por número de covariáveis com tamanho de amostra n=500 e sigma=3.",
1079          label = "tab:mse-por-cov-n500",
1080          digits = 3,
1081          align = "rlr|rrrr|r")
1082 print(xtab, include.rownames = FALSE)
1083
1084 # B4a2. Dimensionality MSE for n = 5000, sigma = 3, for all setups (A, B, C
       , D) and d=4,10,20
       ----------------------------------------------------------------------
1085 alldata %>%
1086   filter(!method %in% exclusion_list,measure=="mse",
1087          n==5000,sigma==3) %>%
1088   select(method,value,setup,d) %>%
1089   mutate(value=round(value, 3),d=as.integer(d)) %>%
1090   pivot_wider(names_from = method ,values_from = value
1091   ) %>%
1092   arrange(setup,d) -> dataselected
1093 xtab = xtable(dataselected,
1094                caption = "MSE médio de 500 replicações dos métodos estudados
       (Floresta Causal -- Amostra Honesta e Adaptativa, IPW, AIPW e DML)
1095          por número de covariáveis com tamanho de amostra n=5000 e sigma=3.",
1096          label = "tab:mse-por-cov-n5000",
1097          digits = 3,
1098          align = "rlr|rrrr|r")
1099 print(xtab, include.rownames = FALSE)
1100
1101 # B4.2. Dimensionality Bias for n = 2000, sigma = 3, for all setups (A, B,
       C, D) and d=4,10,20
       ----------------------------------------------------------------------
1102 alldata %>%
1103   filter(!method %in% exclusion_list,measure=="bias",
1104          n==2000,sigma==3) %>%
1105   select(method,value,setup,d) %>%
1106   mutate(value=round(value, 3),d=as.integer(d)) %>%
1107   pivot_wider(names_from = method ,values_from = value
```

```
1108   ) %>%
1109   arrange(setup,d) -> dataselected
1110 xtab = xtable(dataselected,
1111                caption = "Viés absoluto médio para 500 replicações dos
       métodos estudados (Floresta Causal -- Amostra Honesta e Adaptativa, IPW,
        AIPW e DML)
1112        por número de covariáveis com tamanho de amostra n=2000 e sigma=3.",
1113                label = "tab:bias-por-cov-n2000",
1114                digits = 3,
1115                align = "rlr|rrrr|r")
1116 print(xtab, include.rownames = FALSE)
1117
1118 # B4.2a1. Dimensionality Bias for n = 500, sigma = 3, for all setups (A, B,
       C, D) and d=4,10,20
       ------------------------------------------------------------------------
1119 alldata %>%
1120   filter(!method %in% exclusion_list,measure=="bias",
1121          n==500,sigma==3) %>%
1122   select(method,value,setup,d) %>%
1123   mutate(value=round(value, 3),d=as.integer(d)) %>%
1124   pivot_wider(names_from = method ,values_from = value
1125   ) %>%
1126   arrange(setup,d) -> dataselected
1127 xtab = xtable(dataselected,
1128                caption = "Viés absoluto médio para 500 replicações dos
       métodos estudados (Floresta Causal -- Amostra Honesta e Adaptativa, IPW,
        AIPW e DML)
1129        por número de covariáveis com tamanho de amostra n=500 e sigma=3.",
1130        label = "tab:bias-por-cov-n500",
1131        digits = 3,
1132        align = "rlr|rrrr|r")
1133 print(xtab, include.rownames = FALSE)
1134
1135 # B4.2a2. Dimensionality Bias for n = 5000, sigma = 3, for all setups (A, B
       , C, D) and d=4,10,20
       ------------------------------------------------------------------------
1136 alldata %>%
1137   filter(!method %in% exclusion_list,measure=="bias",
1138          n==5000,sigma==3) %>%
```

```
  select(method,value,setup,d) %>%
  mutate(value=round(value, 3),d=as.integer(d)) %>%
  pivot_wider(names_from = method ,values_from = value
  ) %>%
  arrange(setup,d) -> dataselected
xtab = xtable(dataselected,
              caption = "Viés absoluto médio para 500 replicações dos
    métodos estudados (Floresta Causal -- Amostra Honesta e Adaptativa, IPW,
     AIPW e DML)
       por número de covariáveis com tamanho de amostra n=5000 e sigma=3.",
       label = "tab:bias-por-cov-n5000",
       digits = 3,
       align = "rlr|rrrr|r")
print(xtab, include.rownames = FALSE)

# B4.3. Dimensionality Covered for n = 2000, sigma = 3, for all setups (A,
    B, C, D) and d=4,10,20
    ----------------------------------------------------------------------
alldata %>%
  filter(method %in% c("cf","cfadapt","knn","knnbig","dmlpy"),measure=="
    covered",
         n==2000,sigma==3) %>%
  select(method,value,setup,d) %>%
  mutate(value=round(value, 2),d=as.integer(d)) %>%
  pivot_wider(names_from = method ,values_from = value
  ) %>%
  arrange(setup,d) -> dataselected
xtab = xtable(dataselected,
              caption = "Taxa de cobertura para 500 replicações dos métodos
     estudados (Floresta Causal -- Amostra Honesta e Adaptativa, KNN-10, KNN
    -100 e DML)
       por número de covariáveis com tamanho de amostra n=2000 e sigma=3.",
              label = "tab:covered-por-cov-n2000",
              digits = 2,
              align = "rlr|rrrr|r")
print(xtab, include.rownames = FALSE)

```

```
1169 # B4.3a1. Dimensionality Covered for n = 500, sigma = 3, for all setups (A,
         B, C, D) and d=4,10,20
     ----------------------------------------------------------------------
1170 alldata %>%
1171   filter(method %in% c("cf","cfadapt","knn","knnbig","dmlpy"),measure=="
       covered",
1172          n==500,sigma==3) %>%
1173   select(method,value,setup,d) %>%
1174   mutate(value=round(value, 2),d=as.integer(d)) %>%
1175   pivot_wider(names_from = method ,values_from = value
1176   ) %>%
1177   arrange(setup,d) -> dataselected
1178 xtab = xtable(dataselected,
1179               caption = "Taxa de cobertura para 500 replicações dos métodos
       estudados (Floresta Causal -- Amostra Honesta e Adaptativa, KNN-10, KNN
      -100 e DML)
1180        por número de covariáveis com tamanho de amostra n=500 e sigma=3.",
1181        label = "tab:covered-por-cov-n500",
1182        digits = 2,
1183        align = "rlr|rrrr|r")
1184 print(xtab, include.rownames = FALSE)
1185
1186 # B4.3a2. Dimensionality Covered for n = 5000, sigma = 3, for all setups (A
      , B, C, D) and d=4,10,20
     ----------------------------------------------------------------------
1187 alldata %>%
1188   filter(method %in% c("cf","cfadapt","knn","knnbig","dmlpy"),measure=="
       covered",
1189          n==5000,sigma==3) %>%
1190   select(method,value,setup,d) %>%
1191   mutate(value=round(value, 2),d=as.integer(d)) %>%
1192   pivot_wider(names_from = method ,values_from = value
1193   ) %>%
1194   arrange(setup,d) -> dataselected
1195 xtab = xtable(dataselected,
1196               caption = "Taxa de cobertura para 500 replicações dos métodos
       estudados (Floresta Causal -- Amostra Honesta e Adaptativa, KNN-10, KNN
      -100 e DML)
1197        por número de covariáveis com tamanho de amostra n=5000 e sigma=3.",
```

```
1198        label = "tab:covered-por-cov-n5000",
1199        digits = 2,
1200        align = "rlr|rrrr|r")
1201 print(xtab, include.rownames = FALSE)
1202 
1203 # 4. Plots ------------------------------------------------------------
1204 
1205 setupvals=LETTERS[1:4]
1206 ntree = 400 # number of trees(B)
1207 nbootdml = as.integer(1000) # number of bootstrap repetition
1208 
1209 # function to create a X1 variable grid and other constants
1210 # also provide true tau (CATE) and estimated values for all methods
1211 # on X defined vector of variables.
1212 params.x1.setup = function(n=500,d=4,sigma=3,eta = 0.1,setup = "A") {
1213   s = n/2 # sample size to honest partition method in causal forest
1214   min_node =  round(n*0.005) # minimum size node of tree
1215   # function to choose the data generating process (DGP) parameters
1216   # the output is a list with X=X, b=b, tau=tau, e=e parameters
1217   if (setup == 'A') {#linear setup
1218     get.params = function() {
1219       X = matrix(runif(n * d, min=0, max=1), n, d) # covariates
1220       b = X[,1] + 0.5*X[,2] + 0.4*X[,3] # baseline main effect
1221       tau = 0.3*X[,1] + 0.6*X[,2] + 0.6*X[,3] + 2.2*X[,4] # effect = tau(
    CATE)
1222       eta = 0.1
1223       e = pmax(eta, pmin(sin(pi * X[,1] * X[,2]), 1-eta)) #propensity score
     trimmed eta 0.1
1224       list(X=X, b=b, tau=tau, e=e)
1225     }
1226   } else if (setup == 'B') {# nonlinear setup
1227     get.params = function() {
1228       X = matrix(runif(n*d, min=0, max=1), n, d) # covariates
1229       b = 0.2*X[,1] + X[,1]^2 + 0.5*X[,2]*X[,3] + 0.4*X[,3] + 0.8*X[,3]^2 #
     baseline main effect
1230       eta = 0.1
1231       e = pmax(eta, pmin(X[,1]- 0.2*X[,2]^2, 1-eta)) #propensity score
    trimmed eta 0.1
```

```
1232      tau = 0.3*X[,1] + 0.6*X[,2] + 0.6*X[,3] + 2.2*X[,4]  # effect = tau(
    CATE)
1233      list(X=X, b=b, tau=tau, e=e)
1234    }
1235  } else if (setup == 'C') {# peaks and valleys setup
1236    get.params = function() {
1237      X = matrix(runif(n * d, min=0, max=1), n, d) # covariates
1238      b = 0 # baseline main effect
1239      tau = (1 + 1 / (1 + exp(-20 * (X[,1] - 1/3)))) * (1 + 1 / (1 + exp
    (-20 * (X[,2] - 1/3))))* (1 + 1 / (1 + exp(-20 * (X[,3] - 1/3)))) #
    effect = tau(CATE)
1240      e = pmax(eta, pmin(0.5*X[,1] + 0.5*X[,2], 1-eta)) #propensity score
    trimmed eta 0.1
1241      list(X=X, b=b, tau=tau, e=e)
1242    }
1243  } else if (setup == 'D') {# discontinuities setup
1244    get.params = function() {
1245      X = matrix(rnorm(n * d), n, d) # covariates
1246      b = 2 * (X[,1]>0.4) + 0.3*X[,2] # baseline main effect
1247      eta = 0.1
1248      e = pmax(eta, pmin(0.7*X[,1]- 0.7*X[,2]+ 0.7*X[,3], 1-eta)) #
    propensity score trimmed eta 0.1
1249      tau = 2 * (X[,1]>0.6) + 1.5 * (X[,2]>0.6) + 0.3*X[,3] - 0.7*X[,4]  #
    effect = tau(CATE)
1250      list(X=X, b=b, tau=tau, e=e)
1251    }
1252  } else {
1253
1254    stop("bad setup")
1255  }
1256
1257  # function to choose the data generating process (DGP) parameters
1258  # the output is a list with X=X, b=b, tau=tau, e=e parameters
1259  if (setup == 'A') {#linear setup
1260    get.params.t = function() {
1261      X.grid=matrix(c(seq(-1, 1, by= 0.01),rep(0.5,times=201*(d-1))),length
    (seq(-1, 1, by= 0.01)),d)
1262      X = X.grid
1263      b = X[,1] + 0.5*X[,2] + 0.4*X[,3] # baseline main effect
```

```
1264      tau = 0.3*X[,1] + 0.6*X[,2] + 0.6*X[,3] + 2.2*X[,4] # effect = tau(
    CATE)
1265      eta = 0.1
1266      e = pmax(eta, pmin(sin(pi * X[,1] * X[,2]), 1-eta)) #propensity score
     trimmed eta 0.1
1267      list(X=X, b=b, tau=tau, e=e)
1268    }
1269  } else if (setup == 'B') {# nonlinear setup
1270    get.params.t = function() {
1271      X.grid=matrix(c(seq(-1, 1, by= 0.01),rep(0.5,times=201*(d-1))),length
    (seq(-1, 1, by= 0.01)),d)
1272      X = X.grid
1273      b = 0.2*X[,1] + X[,1]^2 + 0.5*X[,2]*X[,3] + 0.4*X[,3] + 0.8*X[,3]^2 #
     baseline main effect
1274      eta = 0.1
1275      e = pmax(eta, pmin(X[,1]- 0.2*X[,2]^2, 1-eta)) #propensity score
    trimmed eta 0.1
1276      tau = 0.3*X[,1] + 0.6*X[,2] + 0.6*X[,3] + 2.2*X[,4]  # effect = tau(
    CATE)
1277      list(X=X, b=b, tau=tau, e=e)
1278    }
1279  } else if (setup == 'C') {# peaks and valleys setup
1280    get.params.t = function() {
1281      X.grid=matrix(c(seq(-1, 1, by= 0.01),rep(0.5,times=201*(d-1))),length
    (seq(-1, 1, by= 0.01)),d)
1282      X = X.grid
1283      b = 0 # baseline main effect
1284      tau = (1 + 1 / (1 + exp(-20 * (X[,1] - 1/3)))) * (1 + 1 / (1 + exp
    (-20 * (X[,2] - 1/3))))* (1 + 1 / (1 + exp(-20 * (X[,3] - 1/3)))) #
    effect = tau(CATE)
1285      e = pmax(eta, pmin(0.5*X[,1] + 0.5*X[,2], 1-eta)) #propensity score
    trimmed eta 0.1
1286      list(X=X, b=b, tau=tau, e=e)
1287    }
1288  } else if (setup == 'D') {# discontinuities setup
1289    get.params.t = function() {
1290      X.grid=matrix(c(seq(-1, 1, by= 0.01),rep(0.5,times=201*(d-1))),length
    (seq(-1, 1, by= 0.01)),d)
1291      X = X.grid
```

```
1292      b = 2 * (X[,1]>0.4) + 0.3*X[,2] # baseline main effect
1293      eta = 0.1
1294      e = pmax(eta, pmin(0.7*X[,1]- 0.7*X[,2]+ 0.7*X[,3], 1-eta)) #
    propensity score trimmed eta 0.1
1295      tau = 2 * (X[,1]>0.6) + 1.5 * (X[,2]>0.6) + 0.3*X[,3] - 0.7*X[,4]  #
    effect = tau(CATE)
1296      list(X=X, b=b, tau=tau, e=e)
1297     }
1298   } else {
1299
1300     stop("bad setup")
1301   }
1302   params_train = get.params()
1303   W_train = rbinom(n, 1, params_train$e)
1304   Y_train = params_train$b + (W_train - 0.5) * params_train$tau + sigma *
    rnorm(n)
1305
1306   params_test = get.params.t()
1307   n2 <- length(params_test$e)
1308   W_test = rbinom(n2, 1, params_test$e)
1309   Y_test = params_test$b + (W_test - 0.5) * params_test$tau
1310
1311   make_matrix = function(x) stats::model.matrix(~.-1, x)
1312
1313   X_train = make_matrix(data.frame(params_train$X))
1314   X_test = make_matrix(data.frame(params_test$X))
1315
1316   # causal forest honest split
1317   # estimate causal forest
1318   cf = causal_forest(X_train, Y_train, W_train,num.trees = ntree,min.node.
    size = min_node)
1319   # prediction causal forest
1320   cf.pred <- predict(cf, X_test, estimate.variance = TRUE)
1321
1322   # se estimate causal forest
1323   se.hat = sqrt(cf.pred$variance.estimates)
1324
1325   # same for causal forest adaptative split
```

```
1326  cf.adapt = causal_forest(X_train, Y_train, W_train,num.trees = ntree,min.
       node.size = min_node,honesty = F)
1327  cf.pred.adapt <- predict(cf.adapt, X_test, estimate.variance = TRUE)
1328  se.hat.adapt = sqrt(cf.pred.adapt$variance.estimates)
1329
1330  # same for 10-nearest neighbour
1331  k.small = 10
1332  knn.small=causal.kn(kn=k.small,X_train,W_train,X_test,Y_train)
1333  se.hat.kkn.small= knn.small$knn.se
1334
1335  # same for 100-nearest neighbour
1336  k.big = 100
1337  knn.big=causal.kn(kn=k.big,X_train,W_train,X_test,Y_train)
1338  se.hat.kkn.small= knn.big$knn.se
1339
1340  # same for inverse probability weight estimator (IPW) with random forest
1341  ipwrf.pred = ipw.rf.estimator(X_test,Y_test,W_test)
1342
1343  # same for augmented inverse probability weight estimator (AIPW) with
       random forest
1344  aipwrf.pred = aipw.rf.estimator(X_test,Y_test,W_test)
1345
1346  # same for double/debiased machine learning (DML)
1347  dml.pred = dml.estimator(X=X_test,W=W_test,Y=Y_test)$tau_hat
1348
1349  # same for double/debiased machine learning (DML) by DoubleML python
       package
1350  dados_train_full <- data.frame(Y=Y_train,W=W_train,X_train)
1351  cova_train=colnames(X_train)
1352  dados_train <- data.frame(X_train)
1353  dados_test <- data.frame(X_test)
1354  dmlpy_results <- dml_py(data_train_full = dados_train_full,
1355                          data_train = dados_train,
1356                          outcome = "Y",treatment = "W",
1357                          covariates = cova_train,data_test = dados_test,
       nboot = nbootdml)
1358
1359  # prediction dml
1360  dmlpy.pred = dmlpy_results$effect
```

```
1361
1362   results <- list(
1363     params_test=params_test,
1364     predtruth = list(tau = params_test$tau,cf = cf.pred$predictions,
1365                      cfadpt=cf.pred.adapt$predictions,dml = dmlpy.pred,
1366                      ipw=ipwrf.pred,aipw=aipwrf.pred)
1367     # W_test=W_test,X_test=X_test,Y_test=Y_test,
1368     # cf.pred=cf.pred,se.hat=se.hat,
1369     # cf.pred.adapt=cf.pred.adapt,se.hat.adapt=se.hat.adapt,
1370     # knn.small=knn.small,se.hat.kkn.small=se.hat.kkn.small,
1371     # knn.big=knn.big,se.hat.kkn.small=se.hat.kkn.small,
1372     # ipwrf.pred=ipwrf.pred,aipwrf.pred=aipwrf.pred,
1373     # dml.pred=dml.pred
1374     # dmlpy.pred=dmlpy.pred
1375   )
1376   results
1377 }
1378
1379 # function to create a X2 variable grid and other constants
1380 # also provide true tau (CATE) and estimated values for all methods
1381 # on X defined vector of variables.
1382 params.x2.setup = function(n=500,d=4,sigma=3,eta = 0.1,setup = "A") {
1383   # function to choose the data generating process (DGP) parameters
1384   # the output is a list with X=X, b=b, tau=tau, e=e parameters
1385   s = n/2 # sample size to honest partition method in causal forest
1386   min_node =  round(n*0.005) # minimum size node of tree
1387   if (setup == 'A') {#linear setup
1388     get.params = function() {
1389       X = matrix(runif(n * d, min=0, max=1), n, d) # covariates
1390       b = X[,1] + 0.5*X[,2] + 0.4*X[,3] # baseline main effect
1391       tau = 0.3*X[,1] + 0.6*X[,2] + 0.6*X[,3] + 2.2*X[,4] # effect = tau(
     CATE)
1392       eta = 0.1
1393       e = pmax(eta, pmin(sin(pi * X[,1] * X[,2]), 1-eta)) #propensity score
      trimmed eta 0.1
1394       list(X=X, b=b, tau=tau, e=e)
1395     }
1396   } else if (setup == 'B') {# nonlinear setup
1397     get.params = function() {
```

```
1398      X = matrix(runif(n*d, min=0, max=1), n, d) # covariates
1399      b = 0.2*X[,1] + X[,1]^2 + 0.5*X[,2]*X[,3] + 0.4*X[,3] + 0.8*X[,3]^2 #
    baseline main effect
1400      eta = 0.1
1401      e = pmax(eta, pmin(X[,1]- 0.2*X[,2]^2, 1-eta)) #propensity score
    trimmed eta 0.1
1402      tau = 0.3*X[,1] + 0.6*X[,2] + 0.6*X[,3] + 2.2*X[,4]  # effect = tau(
    CATE)
1403      list(X=X, b=b, tau=tau, e=e)
1404    }
1405  } else if (setup == 'C') {# peaks and valleys setup
1406    get.params = function() {
1407      X = matrix(runif(n * d, min=0, max=1), n, d) # covariates
1408      b = 0 # baseline main effect
1409      tau = (1 + 1 / (1 + exp(-20 * (X[,1] - 1/3)))) * (1 + 1 / (1 + exp
    (-20 * (X[,2] - 1/3))))* (1 + 1 / (1 + exp(-20 * (X[,3] - 1/3)))) #
    effect = tau(CATE)
1410      e = pmax(eta, pmin(0.5*X[,1] + 0.5*X[,2], 1-eta)) #propensity score
    trimmed eta 0.1
1411      list(X=X, b=b, tau=tau, e=e)
1412    }
1413  } else if (setup == 'D') {# discontinuities setup
1414    get.params = function() {
1415      X = matrix(rnorm(n * d), n, d) # covariates
1416      b = 2 * (X[,1]>0.4) + 0.3*X[,2] # baseline main effect
1417      eta = 0.1
1418      e = pmax(eta, pmin(0.7*X[,1]- 0.7*X[,2]+ 0.7*X[,3], 1-eta)) #
    propensity score trimmed eta 0.1
1419      tau = 2 * (X[,1]>0.6) + 1.5 * (X[,2]>0.6) + 0.3*X[,3] - 0.7*X[,4]  #
    effect = tau(CATE)
1420      list(X=X, b=b, tau=tau, e=e)
1421    }
1422  } else {
1423
1424    stop("bad setup")
1425  }
1426
1427  # function to choose the data generating process (DGP) parameters
1428  # the output is a list with X=X, b=b, tau=tau, e=e parameters
```

```
1429  if (setup == 'A') {#linear setup
1430    get.params.t = function() {
1431      X.grid=matrix(c(seq(-1, 1, by= 0.01),rep(0.5,times=201*(d-1))),length
      (seq(-1, 1, by= 0.01)),d)
1432      X = X.grid
1433      b = X[,1] + 0.5*X[,2] + 0.4*X[,3] # baseline main effect
1434      tau = 0.3*X[,1] + 0.6*X[,2] + 0.6*X[,3] + 2.2*X[,4] # effect = tau(
      CATE)
1435      eta = 0.1
1436      e = pmax(eta, pmin(sin(pi * X[,1] * X[,2]), 1-eta)) #propensity score
       trimmed eta 0.1
1437      list(X=X, b=b, tau=tau, e=e)
1438    }
1439  } else if (setup == 'B') {# nonlinear setup
1440    get.params.t = function() {
1441      X.grid=matrix(c(seq(-1, 1, by= 0.01),rep(0.5,times=201*(d-1))),length
      (seq(-1, 1, by= 0.01)),d)
1442      X = X.grid
1443      b = 0.2*X[,1] + X[,1]^2 + 0.5*X[,2]*X[,3] + 0.4*X[,3] + 0.8*X[,3]^2 #
       baseline main effect
1444      eta = 0.1
1445      e = pmax(eta, pmin(X[,1]- 0.2*X[,2]^2, 1-eta)) #propensity score
      trimmed eta 0.1
1446      tau = 0.3*X[,1] + 0.6*X[,2] + 0.6*X[,3] + 2.2*X[,4]  # effect = tau(
      CATE)
1447      list(X=X, b=b, tau=tau, e=e)
1448    }
1449  } else if (setup == 'C') {# peaks and valleys setup
1450    get.params.t = function() {
1451      X.grid=matrix(c(seq(-1, 1, by= 0.01),rep(0.5,times=201*(d-1))),length
      (seq(-1, 1, by= 0.01)),d)
1452      X = X.grid
1453      b = 0 # baseline main effect
1454      tau = (1 + 1 / (1 + exp(-20 * (X[,1] - 1/3)))) * (1 + 1 / (1 + exp
      (-20 * (X[,2] - 1/3))))* (1 + 1 / (1 + exp(-20 * (X[,3] - 1/3)))) #
      effect = tau(CATE)
1455      e = pmax(eta, pmin(0.5*X[,1] + 0.5*X[,2], 1-eta)) #propensity score
      trimmed eta 0.1
1456      list(X=X, b=b, tau=tau, e=e)
```

```
1457    }
1458  } else if (setup == 'D') {# discontinuities setup
1459    get.params.t = function() {
1460      X.grid=matrix(c(seq(-1, 1, by= 0.01),rep(0.5,times=201*(d-1))),length
      (seq(-1, 1, by= 0.01)),d)
1461      X = X.grid
1462      b = 2 * (X[,1]>0.4) + 0.3*X[,2] # baseline main effect
1463      eta = 0.1
1464      e = pmax(eta, pmin(0.7*X[,1]- 0.7*X[,2]+ 0.7*X[,3], 1-eta)) #
      propensity score trimmed eta 0.1
1465      tau = 2 * (X[,1]>0.6) + 1.5 * (X[,2]>0.6) + 0.3*X[,3] - 0.7*X[,4]  #
      effect = tau(CATE)
1466      list(X=X, b=b, tau=tau, e=e)
1467    }
1468  } else {
1469
1470    stop("bad setup")
1471  }
1472
1473  # function to choose the data generating process (DGP) parameters
1474  # the output is a list with X=X, b=b, tau=tau, e=e parameters
1475  if (setup == 'A') {#linear setup
1476    get.params.t = function() {
1477      X.grid=matrix(c(rep(0.5,times=201),seq(-1, 1, by= 0.01),rep(0.5,times
      =201*(d-2))),length(seq(-1, 1, by= 0.01)),d)
1478      X = X.grid
1479      b = X[,1] + 0.5*X[,2] + 0.4*X[,3] # baseline main effect
1480      tau = 0.3*X[,1] + 0.6*X[,2] + 0.6*X[,3] + 2.2*X[,4] # effect = tau(
      CATE)
1481      eta = 0.1
1482      e = pmax(eta, pmin(sin(pi * X[,1] * X[,2]), 1-eta)) #propensity score
       trimmed eta 0.1
1483      list(X=X, b=b, tau=tau, e=e)
1484    }
1485  } else if (setup == 'B') {# nonlinear setup
1486    get.params.t = function() {
1487      X.grid=matrix(c(rep(0.5,times=201),seq(-1, 1, by= 0.01),rep(0.5,times
      =201*(d-2))),length(seq(-1, 1, by= 0.01)),d)
1488      X = X.grid
```

```
1489      b = 0.2*X[,1] + X[,1]^2 + 0.5*X[,2]*X[,3] + 0.4*X[,3] + 0.8*X[,3]^2 #
     baseline main effect
1490      eta = 0.1
1491      e = pmax(eta, pmin(X[,1]- 0.2*X[,2]^2, 1-eta)) #propensity score
    trimmed eta 0.1
1492      tau = 0.3*X[,1] + 0.6*X[,2] + 0.6*X[,3] + 2.2*X[,4]  # effect = tau(
    CATE)
1493      list(X=X, b=b, tau=tau, e=e)
1494    }
1495  } else if (setup == 'C') {# peaks and valleys setup
1496    get.params.t = function() {
1497      X.grid=matrix(c(rep(0.5,times=201),seq(-1, 1, by= 0.01),rep(0.5,times
    =201*(d-2))),length(seq(-1, 1, by= 0.01)),d)
1498      X = X.grid
1499      b = 0 # baseline main effect
1500      tau = (1 + 1 / (1 + exp(-20 * (X[,1] - 1/3)))) * (1 + 1 / (1 + exp
    (-20 * (X[,2] - 1/3))))* (1 + 1 / (1 + exp(-20 * (X[,3] - 1/3)))) #
    effect = tau(CATE)
1501      e = pmax(eta, pmin(0.5*X[,1] + 0.5*X[,2], 1-eta)) #propensity score
    trimmed eta 0.1
1502      list(X=X, b=b, tau=tau, e=e)
1503    }
1504  } else if (setup == 'D') {# discontinuities setup
1505    get.params.t = function() {
1506      X.grid=matrix(c(rep(0.5,times=201),seq(-1, 1, by= 0.01),rep(0.5,times
    =201*(d-2))),length(seq(-1, 1, by= 0.01)),d)
1507      X = X.grid
1508      b = 2 * (X[,1]>0.4) + 0.3*X[,2] # baseline main effect
1509      eta = 0.1
1510      e = pmax(eta, pmin(0.7*X[,1]- 0.7*X[,2]+ 0.7*X[,3], 1-eta)) #
    propensity score trimmed eta 0.1
1511      tau = 2 * (X[,1]>0.6) + 1.5 * (X[,2]>0.6) + 0.3*X[,3] - 0.7*X[,4]  #
    effect = tau(CATE)
1512      list(X=X, b=b, tau=tau, e=e)
1513    }
1514  } else {
1515
1516    stop("bad setup")
1517  }
```

```
1518  params_train = get.params()
1519  W_train = rbinom(n, 1, params_train$e)
1520  Y_train = params_train$b + (W_train - 0.5) * params_train$tau + sigma *
       rnorm(n)
1521
1522  params_test = get.params.t()
1523  n2 <- dim(params_test$X)[1]
1524  W_test = rbinom(n2, 1, params_test$e)
1525  Y_test = params_test$b + (W_test - 0.5) * params_test$tau
1526
1527  make_matrix = function(x) stats::model.matrix(~.-1, x)
1528
1529  X_train = make_matrix(data.frame(params_train$X))
1530  X_test = make_matrix(data.frame(params_test$X))
1531
1532  # causal forest honest split
1533  # estimate causal forest
1534  cf = causal_forest(X_train, Y_train, W_train,num.trees = ntree,min.node.
       size = min_node)
1535  # prediction causal forest
1536  cf.pred <- predict(cf, X_test, estimate.variance = TRUE)
1537
1538  # se estimate causal forest
1539  se.hat = sqrt(cf.pred$variance.estimates)
1540
1541  # same for causal forest adaptative split
1542  cf.adapt = causal_forest(X_train, Y_train, W_train,num.trees = ntree,min.
       node.size = min_node,honesty = F)
1543  cf.pred.adapt <- predict(cf.adapt, X_test, estimate.variance = TRUE)
1544  se.hat.adapt = sqrt(cf.pred.adapt$variance.estimates)
1545
1546  # same for 10-nearest neighbour
1547  k.small = 10
1548  knn.small=causal.kn(kn=k.small,X_train,W_train,X_test,Y_train)
1549  se.hat.kkn.small= knn.small$knn.se
1550
1551  # same for 100-nearest neighbour
1552  k.big = 100
1553  knn.big=causal.kn(kn=k.big,X_train,W_train,X_test,Y_train)
```

```
1554  se.hat.kkn.small= knn.big$knn.se
1555
1556  # same for inverse probability weight estimator (IPW) with random forest
1557  ipwrf.pred = ipw.rf.estimator(X_test,Y_test,W_test)
1558
1559  # same for augmented inverse probability weight estimator (AIPW) with
       random forest
1560  aipwrf.pred = aipw.rf.estimator(X_test,Y_test,W_test)
1561
1562  # same for double/debiased machine learning (DML)
1563  dml.pred = dml.estimator(X=X_test,W=W_test,Y=Y_test)$tau_hat
1564
1565  # same for double/debiased machine learning (DML) by DoubleML python
       package
1566  dados_train_full <- data.frame(Y=Y_train,W=W_train,X_train)
1567  cova_train=colnames(X_train)
1568  dados_train <- data.frame(X_train)
1569  dados_test <- data.frame(X_test)
1570  dmlpy_results <- dml_py(data_train_full = dados_train_full,
1571                          data_train = dados_train,
1572                          outcome = "Y",treatment = "W",
1573                          covariates = cova_train,data_test = dados_test,
       nboot = nbootdml)
1574
1575  # prediction dml
1576  dmlpy.pred = dmlpy_results$effect
1577
1578
1579  results <- list(
1580    params_test=params_test,
1581    predtruth = list(tau = params_test$tau,cf = cf.pred$predictions,
1582                     cfadpt=cf.pred.adapt$predictions,dml = dmlpy.pred,
1583                     ipw=ipwrf.pred,aipw=aipwrf.pred)
1584    # W_test=W_test,X_test=X_test,Y_test=Y_test,
1585    # cf.pred=cf.pred,se.hat=se.hat,
1586    # cf.pred.adapt=cf.pred.adapt,se.hat.adapt=se.hat.adapt,
1587    # knn.small=knn.small,se.hat.kkn.small=se.hat.kkn.small,
1588    # knn.big=knn.big,se.hat.kkn.small=se.hat.kkn.small,
1589    # ipwrf.pred=ipwrf.pred,aipwrf.pred=aipwrf.pred,
```

```
    # dml.pred=dml.pred
    # dmlpy.pred=dmlpy.pred
  )
  results
}

# tt <- params.x1.setup(n=500,d=4,sigma=3,setup = "D")
# data1 <- data.frame(tt$params_test$X,
#                     tau=tt$params_test$tau)
# ggplot(data1)+
#   geom_line(aes(x=X1, y=tau)) + scale_y_continuous(
#     limits =   c(-2,10),
#     expand = expansion(mult = c(0,0.05)))
#
# ss <- params.x2.setup(n=500,d=4,sigma=3,setup = "D")
# data2 <- data.frame(ss$params_test$X,
#                     tau=ss$params_test$tau)
# ggplot(data2) +
#   geom_line(aes(x=X2, y=tau))+ scale_y_continuous(
#     limits =   c(-2,10),
#     expand = expansion(mult = c(0,0.05)))


# 4A. true tau for X1 ------------------------------------------------------
tautruth <- lapply(setupvals, function(type) {
  tt <- params.x1.setup(n=500,d=4,sigma=3,setup = type)
  data1 <- data.frame(tt$params_test$X,
                      tau=tt$params_test$tau,setup = type)
  data1
  })
tautruth2 <- do.call(rbind.data.frame, tautruth)

ggplot(tautruth2) +
  geom_line(aes(x=X1, y=tau,color = setup),size=1.5,show.legend = FALSE)+
  facet_wrap(~setup)+
  labs(title = "Tau verdadeiro em diversos cenários do processo gerador
    para X1",
       subtitle = "Especificações: n=500,d=4,sigma=3, X1=[-1,1].") +
  theme_bw()
```

```
1628
1629 # 4B. true tau for X2 ----------------------------------------------------
1630 tautruth3 <- lapply(setupvals, function(type) {
1631   tt <- params.x2.setup(n=500,d=4,sigma=3,setup = type)
1632   data1 <- data.frame(tt$params_test$X,
1633                       tau=tt$params_test$tau,setup = type)
1634   data1
1635 })
1636 tautruth4 <- do.call(rbind.data.frame, tautruth3)
1637
1638 ggplot(tautruth4)+
1639   geom_line(aes(x=X2, y=tau,color = setup),size=1.5,show.legend = FALSE)+
1640   facet_wrap(~setup)+
1641   labs(title = "Tau verdadeiro em diversos cenários do processo gerador
       para X2",
1642        subtitle = "Especificações: n=500,d=4,sigma=3, X2=[-1,1].") +
1643   theme_bw()
1644
1645 # 4C. Comparing estimated tau CF and DML for X1
       ----------------------------------
1646 comparetauX1 <- lapply(setupvals, function(type) {
1647   tt <- params.x1.setup(n=500,d=4,sigma=3,setup = type)
1648   data1 <- data.frame(tt$predtruth %>% as.data.frame(),
1649                       X1=tt$params_test$X[,1],
1650                       setup = type)
1651   data1
1652 })
1653 comparetauX12 <- do.call(rbind.data.frame, comparetauX1)
1654 ggplot(comparetauX12, aes(x=X1))+
1655   geom_line(aes(y=tau,color="Tau"),size=1.5)+
1656   geom_line(aes(y=cf,color="CF"),size=1.5)+
1657   geom_line(aes(y=dml,color="DML"),size=1.5)+
1658   # geom_line(aes(y=ipw,color="IPW"),size=1.5)+
1659   # geom_line(aes(y=aipw,color="AIPW"),size=1.5)+
1660   facet_wrap(~setup,ncol = 1)+
1661   labs(color = "",y = "tau",
1662        title = "Comparação entre tau verdadeiro e valores estimados para X1
       ",
1663        subtitle = "Floresta Causal X DML, cenários selecionados.") +
```

```
  theme_bw()

# 4D. Comparing estimated tau CF and DML for X2
    ----------------------------------
comparetauX2 <- lapply(setupvals, function(type) {
  tt <- params.x2.setup(n=500,d=4,sigma=3,setup = type)
  data1 <- data.frame(tt$predtruth %>% as.data.frame(),
                      X2=tt$params_test$X[,2],
                      setup = type)
  data1
})
comparetauX22 <- do.call(rbind.data.frame, comparetauX2)
ggplot(comparetauX22, aes(x=X2))+
  geom_line(aes(y=tau,color="Tau"),size=1.5)+
  geom_line(aes(y=cf,color="CF"),size=1.5)+
  geom_line(aes(y=dml,color="DML"),size=1.5)+
  # geom_line(aes(y=ipw,color="IPW"),size=1.5)+
  # geom_line(aes(y=aipw,color="AIPW"),size=1.5)+
  facet_wrap(~setup)+
  labs(color = "",y = "tau",
       title = "Comparação entre tau verdadeiro e valores estimados para X2
    ",
       subtitle = "Floresta Causal X DML, cenários selecionados.") +
  theme_bw()

# X.grid=matrix(rep(seq(-1, 1, by= 0.01),times=d),length(seq(-1, 1, by=
    0.01)),d)
# te = apply(X.grid, 1, effect) # true effect for test sample
# cf.estCI <- predict(cf, X.grid, estimate.variance = TRUE)
# cf.adapt.estCI <- predict(cf.adapt, X.grid, estimate.variance = TRUE)
# knn.small.estCI=causal.kn(kn=k.small,X.grid,Y)
# knn.big.estCI=causal.kn(kn=k.big,X.grid,Y)
#
# data1 = cbind(X=X.grid[,1],truth=te,cf=cf.estCI$predictions,
#               cf.adapt=cf.adapt.estCI$predictions,
#               knn.small=knn.small.estCI$knn.tau,
#               knn.big=knn.big.estCI$knn.tau) %>% as.data.frame
#
# data1_long <- data1 %>%
```

```
#   pivot_longer(cols = truth:knn.big, names_to = "grp",
#                values_to = "mean")


# comparing methods
# (p1 <- ggplot(data1_long, aes(x=X, y=mean, color = grp))+
#     geom_line(aes(x=X, y=mean, color=grp)) +
#     # scale_color_viridis_d()+
#     labs(color = "",y = "tau") +
#     theme_bw()
# )
#
# data2 = cbind(X=X.grid[,1],truth=te,cf=cf.estCI$predictions,
#               cf.se=sqrt(cf.estCI$variance.estimates)) %>% as.data.frame
#
# head(data2)
# data2 <-  mutate(data2,
#                  upper.ci=cf+1.96*cf.se,
#                  lower.ci=cf-1.96*cf.se,
#                  cf.se=NULL)
#
# # confidence interval band
# (p2 <- ggplot(data2, aes(x=X))+
#     # geom_point()+
#     geom_line(aes(y=truth,color="truth"))+
#     geom_ribbon(aes(ymin=lower.ci,ymax=upper.ci),alpha=0.3)+
#     geom_line(aes(y=cf,color="cf"))+
#     labs(color = "",y = "tau") +
#     theme_bw()
# )
#
#
#
#
# # Concatenate the two results.
# res <- rbind(forest.ate, ols.ate)
#
# # Plotting the point estimate of average treatment effect
# # and 95% confidence intervals around it.
```

```
# ggplot(res) +
#   aes(x = ranking, y = estimate, group=method, color=method) +
#   geom_point(position=position_dodge(0.2)) +
#   geom_errorbar(aes(ymin=estimate-2*std.err, ymax=estimate+2*std.err),
    width=.2, position=position_dodge(0.2)) +
#   ylab("") + xlab("") +
#   ggtitle("Average CATE within each ranking (as defined by predicted CATE
    )") +
#   theme_minimal() +
#   theme(legend.position="bottom", legend.title = element_blank())
#
#
# # plot partial dependence
# selected.covariate <- "polviews"
# other.covariates <- covariates[which(covariates != selected.covariate)]
#
# # Fitting a forest
# # (commented for convenience; no need re-fit if already fitted above)
# fmla <- formula(paste0("~ 0 + ", paste0(covariates, collapse="+")))
# # Note: For smaller confidence intervals, set num.trees ~ sample size
# # X <- model.matrix(fmla, data)
# # W <- data[,treatment]
# # Y <- data[,outcome]
# # forest.tau <- causal_forest(X, Y, W, W.hat=.5)  # few trees for speed
    here
#
# # Compute a grid of values appropriate for the selected covariate
# grid.size <- 7
# covariate.grid <- seq(min(data[,selected.covariate]), max(data[,selected.
    covariate]), length.out=grid.size)
#
# # Other options for constructing a grid:
# # For a binary variable, simply use 0 and 1
# # grid.size <- 2
# # covariate.grid <- c(0, 1)
#
# # For a continuous variable, select appropriate percentiles
# # percentiles <- c(.1, .25, .5, .75, .9)
# # grid.size <- length(percentiles)
```

```
# # covariate.grid <- quantile(data[,selected.covariate], probs=percentiles)
#
# # Take median of other covariates
# medians <- apply(data[, other.covariates, F], 2, median)
#
# # Construct a dataset
# data.grid <- data.frame(sapply(medians, function(x) rep(x, grid.size)), covariate.grid)
# colnames(data.grid) <- c(other.covariates, selected.covariate)
#
# # Expand the data
# X.grid <- model.matrix(fmla, data.grid)
#
# # Point predictions of the CATE and standard errors
# forest.pred <- predict(forest.tau, newdata = X.grid, estimate.variance=TRUE)
# tau.hat <- forest.pred$predictions
# tau.hat.se <- sqrt(forest.pred$variance.estimates)
#
# # Plot predictions for each group and 95% confidence intervals around them.
# data.pred <- transform(data.grid, tau.hat=tau.hat, ci.low = tau.hat - 2*tau.hat.se, ci.high = tau.hat + 2*tau.hat.se)
# ggplot(data.pred) +
#   geom_line(aes_string(x=selected.covariate, y="tau.hat", group = 1), color="black") +
#   geom_errorbar(aes_string(x=selected.covariate, ymin="ci.low", ymax="ci.high", width=.2), color="blue") +
#   ylab("") +
#   ggtitle(paste0("Predicted treatment effect varying '", selected.covariate, "' (other variables fixed at median)")) +
#   scale_x_continuous("polviews", breaks=covariate.grid, labels=signif(covariate.grid, 2)) +
#   theme_minimal() +
#   theme(plot.title = element_text(size = 11, face = "bold"))
#
# # confidence interval band
# (p2 <- ggplot(mp, aes(wav, wow))+
```

```
#      geom_point()+
#      geom_line(data=predframe)+
#      geom_ribbon(data=predframe,aes(ymin=lwr,ymax=upr),alpha=0.3))
```

Apêndice D – Simulação - Estimação e Inferência - Tabelas Adicionais

D.1 Estimação - especificação selecionada

Tabela 10 – MSE médio de 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) com número de covariáveis $d = 10$, tamanho de amostra $n = 2000$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	2000	**0.121**	0.487	1.716	1.252	**0.398**
B		**0.138**	0.653	2.500	1.273	**0.199**
C		**0.540**	0.636	5.345	7.438	**1.348**
D		**0.662**	1.391	3.145	4.120	**2.444**

Tabela 11 – MSE médio de 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) com número de covariáveis $d = 20$, tamanho de amostra $n = 2000$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	2000	**0.306**	0.451	1.152	1.010	**0.301**
B		**0.306**	0.504	0.889	1.143	**0.337**
C		**1.390**	0.927	4.872	6.474	**1.526**
D		**1.095**	0.975	2.451	2.674	**2.174**

Tabela 12 – Viés absoluto médio para 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) com número de covariáveis $d = 10$, tamanho de amostra $n = 2000$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	2000	**0.278**	0.554	0.833	0.872	**0.505**
B		**0.287**	0.653	0.882	0.900	**0.360**
C		**0.603**	0.637	1.901	2.214	**0.939**
D		**0.637**	0.941	1.368	1.573	**1.267**

Tabela 13 – Viés absoluto médio para 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) com número de covariáveis $d = 20$, tamanho de amostra $n = 2000$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	2000	**0.450**	0.535	0.782	0.801	**0.440**
B		**0.456**	0.568	0.755	0.824	**0.467**
C		**0.953**	0.777	1.825	2.079	**0.986**
D		**0.828**	0.789	1.231	1.298	**1.191**

Tabela 14 – Taxa de cobertura para 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, KNN-10, KNN-100 e DML) com número de covariáveis $d = 10$, tamanho de amostra $n = 2000$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	KNN_{10}	KNN_{100}	**DML**
A	2000	**0.96**	1.00	0.94	0.90	**1.00**
B		**0.94**	1.00	0.94	0.92	**1.00**
C		**0.71**	1.00	0.90	0.37	**1.00**
D		**0.79**	1.00	0.92	0.64	**1.00**

Tabela 15 – Taxa de cobertura para 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, KNN-10, KNN-100 e DML) com número de covariáveis $d = 20$, tamanho de amostra $n = 2000$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	KNN_{10}	KNN_{100}	**DML**
A	2000	**0.74**	1.00	0.93	0.88	**1.00**
B		**0.74**	1.00	0.94	0.81	**1.00**
C		**0.51**	1.00	0.82	0.32	**1.00**
D		**0.56**	1.00	0.89	0.52	**1.00**

D.2 Consistência Estimação

Tabela 16 – MSE médio de 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) por tamanho de amostra com número de covariáveis $d = 10$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	500	**0.243**	1.425	2.429	2.995	**0.745**
	2000	**0.121**	0.487	1.716	1.252	**0.398**
	5000	**0.223**	0.382	0.736	1.209	**0.088**
B	500	**0.227**	1.180	3.288	2.065	**0.392**
	2000	**0.138**	0.653	2.500	1.273	**0.199**
	5000	**0.128**	0.343	0.935	1.067	**0.216**
C	500	**2.055**	1.419	7.564	9.376	**1.325**
	2000	**0.540**	0.636	5.345	7.438	**1.348**
	5000	**0.528**	0.343	4.571	7.407	**1.123**
D	500	**1.715**	2.828	4.305	4.984	**2.846**
	2000	**0.662**	1.391	3.145	4.120	**2.444**
	5000	**0.682**	0.537	2.304	3.271	**1.650**

Tabela 17 – MSE médio de 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) por tamanho de amostra com número de covariáveis $d = 20$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	500	**0.897**	2.120	2.314	1.686	**1.682**
	2000	**0.306**	0.451	1.152	1.010	**0.301**
	5000	**0.296**	0.269	0.704	0.924	**0.228**
B	500	**0.455**	1.870	1.951	1.629	**1.137**
	2000	**0.306**	0.504	0.889	1.143	**0.337**
	5000	**0.188**	0.415	0.712	0.804	**0.335**
C	500	**3.487**	3.055	5.915	6.649	**4.015**
	2000	**1.390**	0.927	4.872	6.474	**1.526**
	5000	**0.420**	0.392	4.416	6.573	**1.213**
D	500	**1.271**	2.952	2.930	3.182	**3.724**
	2000	**1.095**	0.975	2.451	2.674	**2.174**
	5000	**0.626**	0.713	1.999	2.553	**1.805**

Tabela 18 – Viés absoluto médio de 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) por tamanho de amostra com número de covariáveis $d = 10$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	500	**0.391**	0.970	1.165	1.263	**0.694**
	2000	**0.278**	0.554	0.833	0.872	**0.505**
	5000	**0.374**	0.492	0.676	0.885	**0.240**
B	500	**0.381**	0.841	1.278	1.103	**0.516**
	2000	**0.287**	0.653	0.882	0.900	**0.360**
	5000	**0.277**	0.469	0.753	0.826	**0.377**
C	500	**1.234**	0.935	2.206	2.451	**0.919**
	2000	**0.603**	0.637	1.901	2.214	**0.939**
	5000	**0.551**	0.472	1.747	2.204	**0.850**
D	500	**1.052**	1.309	1.624	1.738	**1.337**
	2000	**0.637**	0.941	1.368	1.573	**1.267**
	5000	**0.661**	0.588	1.174	1.426	**1.050**

Tabela 19 – Viés absoluto médio de 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) por tamanho de amostra com número de covariáveis $d = 20$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	500	**0.786**	1.223	1.148	1.039	**1.059**
	2000	**0.450**	0.535	0.782	0.801	**0.440**
	5000	**0.456**	0.415	0.660	0.767	**0.386**
B	500	**0.567**	1.120	1.104	0.993	**0.851**
	2000	**0.456**	0.568	0.755	0.824	**0.467**
	5000	**0.348**	0.513	0.669	0.715	**0.464**
C	500	**1.497**	1.417	1.939	2.080	**1.624**
	2000	**0.953**	0.777	1.825	2.079	**0.986**
	5000	**0.515**	0.503	1.727	2.086	**0.880**
D	500	**0.896**	1.402	1.323	1.414	**1.547**
	2000	**0.828**	0.789	1.231	1.298	**1.191**
	5000	**0.635**	0.616	1.149	1.275	**1.095**

D.3 Consistência do intervalo de confiância e Inferência

Tabela 20 – Taxa de cobertura de 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, KNN-10, KNN-100 e DML) por tamanho de amostra com número de covariáveis $d = 10$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	KNN_{10}	KNN_{100}	**DML**
A	500	**1.00**	1.00	0.89	0.78	**1.00**
	2000	**0.96**	1.00	0.94	0.90	**1.00**
	5000	**0.71**	1.00	0.93	0.83	**1.00**
B	500	**0.99**	1.00	0.92	0.60	**1.00**
	2000	**0.94**	1.00	0.94	0.92	**1.00**
	5000	**0.82**	1.00	0.92	0.82	**1.00**
C	500	**0.57**	1.00	0.83	0.25	**1.00**
	2000	**0.71**	1.00	0.90	0.37	**1.00**
	5000	**0.71**	1.00	0.89	0.39	**1.00**
D	500	**0.73**	1.00	0.83	0.41	**1.00**
	2000	**0.79**	1.00	0.92	0.64	**1.00**
	5000	**0.56**	1.00	0.90	0.62	**1.00**

Tabela 21 – Taxa de cobertura de 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, KNN-10, KNN-100 e DML) por tamanho de amostra com número de covariáveis $d = 20$ e $\sigma_\epsilon = 3$.

Cenário	n	**CF**	CF_{adapt}	KNN_{10}	KNN_{100}	**DML**
A	500	**0.84**	1.00	0.89	0.59	**1.00**
	2000	**0.74**	1.00	0.93	0.88	**1.00**
	5000	**0.52**	1.00	0.93	0.83	**1.00**
B	500	**0.96**	1.00	0.93	0.73	**1.00**
	2000	**0.74**	1.00	0.94	0.81	**1.00**
	5000	**0.71**	1.00	0.92	0.82	**1.00**
C	500	**0.55**	1.00	0.80	0.34	**1.00**
	2000	**0.51**	1.00	0.82	0.32	**1.00**
	5000	**0.74**	1.00	0.85	0.35	**1.00**
D	500	**0.82**	1.00	0.85	0.38	**1.00**
	2000	**0.56**	1.00	0.89	0.52	**1.00**
	5000	**0.64**	1.00	0.89	0.55	**1.00**

D.4 Dimensionalidade - Estimação

Tabela 22 – MSE médio de 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) por número de covariáveis com tamanho de amostra $n = 500$ e $\sigma_\epsilon = 3$.

Cenário	d	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	4	**0.501**	1.988	16.540	5.440	**0.919**
	10	**0.243**	1.425	2.429	2.995	**0.745**
	20	**0.897**	2.120	2.314	1.686	**1.682**
B	4	**0.229**	2.212	14.888	6.032	**0.399**
	10	**0.227**	1.180	3.288	2.065	**0.392**
	20	**0.455**	1.870	1.951	1.629	**1.137**
C	4	**1.207**	2.191	10.911	13.218	**2.249**
	10	**2.055**	1.419	7.564	9.376	**1.325**
	20	**3.487**	3.055	5.915	6.649	**4.015**
D	4	**1.197**	2.391	22.517	19.863	**3.088**
	10	**1.715**	2.828	4.305	4.984	**2.846**
	20	**1.271**	2.952	2.930	3.182	**3.724**

Tabela 23 – MSE médio de 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) por número de covariáveis com tamanho de amostra $n = 5000$ e $\sigma_\epsilon = 3$.

Cenário	d	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	4	**0.080**	0.347	1.251	1.535	**0.157**
	10	**0.223**	0.382	0.736	1.209	**0.088**
	20	**0.296**	0.269	0.704	0.924	**0.228**
B	4	**0.112**	0.282	1.316	1.749	**0.113**
	10	**0.128**	0.343	0.935	1.067	**0.216**
	20	**0.188**	0.415	0.712	0.804	**0.335**
C	4	**0.364**	0.227	5.029	8.868	**1.110**
	10	**0.528**	0.343	4.571	7.407	**1.123**
	20	**0.420**	0.392	4.416	6.573	**1.213**
D	4	**0.541**	0.446	3.015	4.291	**1.831**
	10	**0.682**	0.537	2.304	3.271	**1.650**
	20	**0.626**	0.713	1.999	2.553	**1.805**

Tabela 24 – Viés absoluto médio para 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) por número de covariáveis com tamanho de amostra $n = 500$ e $\sigma_\epsilon = 3$.

Cenário	d	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	4	**0.588**	1.152	2.300	1.758	**0.787**
	10	**0.391**	0.970	1.165	1.263	**0.694**
	20	**0.786**	1.223	1.148	1.039	**1.059**
B	4	**0.381**	1.180	2.053	1.716	**0.505**
	10	**0.381**	0.841	1.278	1.103	**0.516**
	20	**0.567**	1.120	1.104	0.993	**0.851**
C	4	**0.941**	1.220	2.673	2.870	**1.192**
	10	**1.234**	0.935	2.206	2.451	**0.919**
	20	**1.497**	1.417	1.939	2.080	**1.624**
D	4	**0.839**	1.227	2.991	2.652	**1.433**
	10	**1.052**	1.309	1.624	1.738	**1.337**
	20	**0.896**	1.402	1.323	1.414	**1.547**

Tabela 25 – Viés absoluto médio para 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, IPW, AIPW e DML) por número de covariáveis com tamanho de amostra $n = 5000$ e $\sigma_\epsilon = 3$.

Cenário	d	**CF**	CF_{adapt}	IPW_{rf}	AIPW_{rf}	**DML**
A	4	**0.230**	0.468	0.780	0.916	**0.327**
	10	**0.374**	0.492	0.676	0.885	**0.240**
	20	**0.456**	0.415	0.660	0.767	**0.386**
B	4	**0.276**	0.428	0.849	0.995	**0.275**
	10	**0.277**	0.469	0.753	0.826	**0.377**
	20	**0.348**	0.513	0.669	0.715	**0.464**
C	4	**0.495**	0.388	1.833	2.394	**0.850**
	10	**0.551**	0.472	1.747	2.204	**0.850**
	20	**0.515**	0.503	1.727	2.086	**0.880**
D	4	**0.589**	0.516	1.324	1.568	**1.081**
	10	**0.661**	0.588	1.174	1.426	**1.050**
	20	**0.635**	0.616	1.149	1.275	**1.095**

D.5 Dimensionalidade - Inferência

Tabela 26 – Taxa de cobertura para 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, KNN-10, KNN-100 e DML) por número de covariáveis com tamanho de amostra $n = 500$ e $\sigma_\epsilon = 3$.

Cenário	d	**CF**	CF_{adapt}	KNN_{10}	KNN_{100}	**DML**
A	4	**0.96**	1.00	0.93	0.83	**1.00**
	10	**1.00**	1.00	0.89	0.78	**1.00**
	20	**0.84**	1.00	0.89	0.59	**1.00**
B	4	**0.98**	1.00	0.94	0.97	**1.00**
	10	**0.99**	1.00	0.92	0.60	**1.00**
	20	**0.96**	1.00	0.93	0.73	**1.00**
C	4	**0.88**	1.00	0.89	0.29	**1.00**
	10	**0.57**	1.00	0.83	0.25	**1.00**
	20	**0.55**	1.00	0.80	0.34	**1.00**
D	4	**0.80**	1.00	0.92	0.48	**1.00**
	10	**0.73**	1.00	0.83	0.41	**1.00**
	20	**0.82**	1.00	0.85	0.38	**1.00**

Tabela 27 – Taxa de cobertura para 500 replicações dos métodos estudados (Floresta Causal – Amostra Honesta e Adaptativa, KNN-10, KNN-100 e DML) por número de covariáveis com tamanho de amostra $n = 5000$ e $\sigma_\epsilon = 3$.

Cenário	d	**CF**	CF_{adapt}	KNN_{10}	KNN_{100}	**DML**
A	4	**0.89**	1.00	0.94	0.94	**1.00**
	10	**0.71**	1.00	0.93	0.83	**1.00**
	20	**0.52**	1.00	0.93	0.83	**1.00**
B	4	**0.83**	1.00	0.94	0.93	**1.00**
	10	**0.82**	1.00	0.92	0.82	**1.00**
	20	**0.71**	1.00	0.92	0.82	**1.00**
C	4	**0.59**	1.00	0.94	0.77	**1.00**
	10	**0.71**	1.00	0.89	0.39	**1.00**
	20	**0.74**	1.00	0.85	0.35	**1.00**
D	4	**0.61**	0.98	0.91	0.71	**1.00**
	10	**0.56**	1.00	0.90	0.62	**1.00**
	20	**0.64**	1.00	0.89	0.55	**1.00**