Conselho Administrativo de Defesa Econômica
Departamento de Estudos Econômicos

Documento de Trabalho

Nº 003/2022

Aprendizado de máquina e antitruste

Tatiana de Macedo Nogueira Lima
(Coordenadora DEE/Cade)

Brasília, julho de 2022

CADE
Conselho Administrativo de Defesa Econômica

Ministério da Justiça e Segurança Pública

Conselho Administrativo de Defesa Econômica

Aprendizado de máquina e antitruste

Departamento de Estudos Econômicos – DEE

SEPN 515 Conjunto D, Lote 4, Ed. Carlos Taurisano

Cep: 70770-504 – Brasília-DF

www.gov.br/cade

ISSN 2764-1031

Esse documento foi produzido pelo Departamento de Estudos Econômicos do Conselho Administrativo de Defesa Econômica.

Tatiana de Macedo Nogueira Lima

(Coordenadora DEE/Cade)

Sumário Executivo

Aprendizado de máquina é a busca da compreensão da estrutura subjacente dos dados e sua regularidade por meio do desenvolvimento de modelos que podem ser usados para previsões (THEODORIS, 2015). Um conjunto de fatores provocou a popularização dessa área entre o final do século passado e o início deste. De um lado, tornou-se muito menos custoso coletar e manter dados em decorrência da digitalização da economia e da vida em geral (THE ECONOMIST, 2010), ainda que haja registros bastante antigos de coletas sistematizadas de dados. Paralelamente, o desenvolvimento computacional facilitou o emprego de modelos e técnicas para análise desses dados. Do outro lado, o interesse por análises baseadas em evidências aumentou (MCELHERAN; BRYNJOLFSSON, 2016).

No antitruste, autoridades têm criado unidades para a análise de dados ou diversificado seus departamentos econômicos, contratando engenheiros da computação, estatísticos e cientistas de dados (SCHREPEL; GROZA, 2022). No Conselho Administrativo de Defesa Econômica (Cade), o projeto Cérebro, desde 2013, usa ferramentas de análise de dados e de aprendizado de máquina para monitorar mercados, detectar cartéis e outras condutas anticompetitivas (GROSSMAN, 2018; PIMENTA, 2019).

Muitos outros procedimentos para análise de condutas e avaliação de fusões e aquisições podem beneficiar-se do emprego dessas ferramentas. No Departamento de Estudos Econômicos (DEE) do Cade, tem-se estudado o emprego de técnicas diversas na estimação de demanda, definição de mercado relevante e outros procedimentos típicos da análise antitruste. Para isso, o departamento diversificou seus quadros, dos quais fazem parte, atualmente, diferentes profissionais com experiência em ciência dos dados.

Tendo isso em vista, o objetivo deste documento é apresentar modelos e procedimentos de aprendizado de máquina que podem ser utilizados em diferentes etapas da análise antitruste. Em alguns casos, como se verá, esses modelos são o ponto de partida para a construção de modelos econométricos ou um meio para seu aprimoramento. Em outros, podem ser empregados para a realização de previsões, classificações e outras tarefas, sendo, por vezes, mais adequados que os modelos econométricos.

Com a apresentação desses modelos, pretende-se iniciar um diálogo com a comunidade antitruste sobre o uso dessas ferramentas para o aprimoramento das análises antitruste. Diversas discussões internas têm sido feitas. Acredita-se que, para além do que o Cade já tem feito, há muito que pode ainda ser desenvolvido.

Sumário

1. INTRODUÇÃO

Técnicas de aprendizado de máquina difundiram-se consideravelmente nos últimos anos e têm sido cada vez mais usadas por economistas. Segundo Athey (2017), em breve, não haverá artigo aceito nas principais revistas acadêmicas da área que não tenham incorporado algum modelo de aprendizado de máquina. Na literatura de organização industrial, estudos empíricos também têm utilizado modelos e técnicas baseados nessa literatura (por exemplo, BAJARI et al., 2015).

No Conselho Administrativo de Defesa Econômica (Cade), o Projeto Cérebro conjuga técnicas estatísticas e de mineração de dados a fim de detectar indícios de práticas anticompetitivas. Investigações antitruste têm sido iniciadas com base nos trabalhos desse projeto (PIMENTA, 2019). As ferramentas desenvolvidas em seu âmbito também são usadas na investigação de condutas já detectadas. Para além do trabalho do Projeto Cérebro, o Departamento Econômico do Cade (DEE) utilizou ferramental econométrico em conjunto com o de aprendizado de máquina em alguns estudos[1].

Na Europa, a Rede Europeia de Concorrência[2] tem discutido métodos de investigação por meios digitais (AUTORITÉ DE LA CONCURRENCE, 2020). A autoridade francesa de concorrência criou uma unidade de economia digital, que tem como um de seus objetivos desenvolver novas ferramentas digitais de investigação, baseadas em algoritmos, dados e inteligência artificial. No Reino Unido, a Unidade de dados, tecnologia e análise (DaTA) da Competition and Markets Authority (CMA) existe desde 2018 com o objetivo de empregar o que há de mais recente em técnicas de engenharia de dados, aprendizado de máquina e inteligência artificial para aumentar a eficiência do órgão (HUNT, 2018). Outros países europeus, como Espanha e Grécia, também criaram unidades voltadas ao desenvolvimento de técnicas de investigação e monitoramento de mercados com o uso da tecnologia ou contrataram cientistas da computação, engenheiros e cientistas de dados, em geral, para trabalharem em seus departamentos econômicos. Já nos Estados Unidos, o Departamento de Justiça norte-americano está treinando sua força de trabalho

[1] No documento "Técnicas de estimação de demanda usando aprendizagem de máquinas", do consultor Daniel Oliveira Cajueiro, é desenvolvida uma estratégia para estimação da demanda tendo como base sistemas de equações simultâneas e o lasso (*least absolute shrinkage and selection operator*). Nota técnica no 29/2018/DEE/CADE, em que se analisa o emprego de filtros para identificação de práticas colusivas de postos comubstíveis e se propõe uma modificação do filtro que vinha sendo empregado pelo Cade, também foram usados métodos de regressões regularizadas.

[2] A Rede europeia de Competição é um fórum no qual cooperam as autoridades nacionais de competição dos países europeus a fim de garantir a efetividade e consistência na aplicação das regras antitruste europeias (https://ec.europa.eu/competition-policy/european-competition-network_en).

para que seja capaz de lidar com as questões concorrenciais decorrentes do crescimento da economia digital (DELRAHIM, 2020).

Considerando-se o cenário de difusão de métodos computacionais no antitruste, este estudo visa, primeiramente, apresentar modelos e procedimentos de aprendizagem de máquina que podem ser utilizados na análise antitruste.

O termo aprendizado de máquina foi cunhado por Arthur Samuel, em 1959[3], e popularizou-se recentemente, com o aumento da capacidade de processamento dos computadores e dos dados disponíveis. Pode-se defini-lo como o estudo e desenvolvimento de algoritmos que se aprimoram automaticamente. O objetivo principal das técnicas e métodos de aprendizado de máquina é a previsão. Outros objetivos, contudo, devem ser citados, como a classificação e o agrupamento.

Os modelos que se pretende apresentar neste documento baseiam-se em modelos estatísticos de regressão e classificação e se aproximam do ferramental clássico da econometria. Há, ainda, modelos de agrupamento, bastante utilizados na estatística e facilmente compreendidos por economistas. Esses modelos passaram a ser ainda mais usados recentemente, com a popularização do aprendizado de máquina. Como se verá adiante, a principal diferença do aprendizado de máquina para a econometria são os objetivos. Enquanto a primeira tem por fim a previsão, classificação, agrupamento ou otimização, o principal objetivo da segunda é a inferência de causalidade. Há, assim, duas questões principais quando se discutem as relações entre ambas. A primeira é a possibilidade de se inferir causalidade a partir de modelos diversos. Analisar essa questão exige discutir tanto procedimentos usuais na econometria quanto entender os limites dos modelos de aprendizado de máquina.

A segunda questão trata da ampliação dos problemas analisados empiricamente por economistas ou da incorporação de novos métodos de análise. A econometria, na maior parte das vezes, baseia-se na utilização de modelos de regressão com vistas a estabelecer relações consistentes entre variáveis diversas. No aprendizado de máquina, os modelos classificados como de aprendizagem não supervisionada, que visam tão somente reconhecer padrões que permitam classificar um conjunto de observações, podem ter aplicações interessantes no antitruste. A definição de mercado relevante, por exemplo, pode ser compreendida como um problema de agrupamento: quais produtos podem ser agrupados como concorrentes; quais áreas fazem parte

[3] SAMUEL, ARTHUR. Some Studies in Machine Learning Using the Game of Checkers. IBM Journal, 1959.

do mesmo conjunto. Verificar se uma entrada é tempestiva, provável e suficiente é um problema de previsão, que, na maior parte dos casos, tem sido tratado apenas qualitativamente.

No que tange à apresentação de modelos, o objetivo não é descrevê-los detalhadamente, pois há extenso material sobre cada um disponível tanto na internet quanto em livros e artigos científicos, mas mostrar suas principais aplicações, características, como se relacionam com os métodos econométricos e sua aplicabilidade na defesa da concorrência. Sempre que possível, exemplos simples de cada método ilustrarão as descrições.

O trabalho divide-se em 6 seções, além da introdução. Na primeira, são discutidos conceitos que permitem entender as similaridades e diferenças entre os objetivos e técnicas econométricas e de aprendizado de máquinas. Na segunda, discute-se a aplicação de modelos de aprendizagem de máquina à defesa da concorrência. A partir da terceira seção, são apresentados procedimentos, discussões e modelos estudados na aprendizagem de máquina. Inicialmente, é apresentado o procedimento de validação cruzada, usado para a divisão de observações em conjuntos de treinamento, por meio dos quais os modelos são estimados, e conjuntos de teste, nos quais os modelos são testados. Na seção seguinte, o *tradeoff* entre viés e variância é discutido. Na quinta seção, são apresentadas as regressões regularizadas. Posteriormente, métodos de árvore de decisão, classificação e de agrupamento (*cluster*) são revisados.

2. ALGUMAS DEFINIÇÕES IMPORTANTES

Discussões sobre a aplicabilidade de modelos de aprendizado de máquina por economistas costumam versar sobre a possibilidade de se usar esses modelos para inferir causalidade (ATHEY; IMBENS, 2018; VARIAN, 2014). Isso porque, na maior parte das vezes, economistas não estão interessados apenas em verificar se uma variável se correlaciona com a outra ou se a partir de uma pode-se prever o comportamento da outra. O principal interesse nas aplicações empíricas econômicas é discernir os efeitos de uma ação em uma ou mais variáveis.

Uma pergunta econômica clássica, por exemplo, é como a demanda de certo produto é alterada se os preços aumentam. A resposta que parece óbvia a qualquer estudante de graduação é de que a demanda provavelmente irá diminuir. No entanto, um analista que apenas observe o comportamento de preços e quantidades vendidas pode chegar a conclusão diferente. Em mercados ligados ao turismo, a quantidade comercializada de produtos aumenta nos períodos de férias escolares, quando os preços também costumam aumentar, o que poderia levar a conclusão de que, maiores os preços, maior a demanda.

Um economista que analise esses mercados, no entanto, conhece os modelos teóricos de demanda e oferta e usa esse conhecimento na estimação que pretende fazer. Assim, a partir de premissas teóricas e de técnicas que permitem estimar os efeitos dos preços na demanda, mas que somente são empregadas porque existe o conhecimento teórico prévio, estima-se consistentemente[4] esses efeitos. A conclusão, ao final, é de que a demanda varia negativamente com o aumento dos preços.

O exemplo acima traz à luz o primeiro ponto a ser observado: a inferência de causalidade somente é possível se o analista fizer hipóteses que vão além daquelas requeridas para fazer predições (ATHEY, 2017). Essas hipóteses, muitas vezes, não são testáveis e se baseiam em conhecimentos teóricos que vão além do ferramental econométrico. No caso da estimação da demanda, por exemplo, há um modelo teórico que fundamenta a hipótese de que um aumento nos preços diminui a demanda por um produto. Técnicas foram desenvolvidas para que se possa estimar os efeitos desse aumento consistentemente, como o uso de variáveis instrumentais. Essas técnicas, por vezes, diminuem o poder explicativo do modelo econométrico, mas, para o economista, um estimador sem viés ou consistente é, em geral, mais importante do que o poder explicativo de um modelo.

Sendo o núcleo da econometria a inferência causal, cabe defini-la apropriadamente. Imbens e Rubin (2015) explicam que causalidade está ligada a uma ação (manipulação, tratamento ou intervenção) aplicada a uma unidade. No exemplo anteriormente citado, a unidade seria o produto cuja demanda se pretende estimar, e a ação seria a modificação de seu preço. O efeito causal de uma ação em relação a outra é verificado pela comparação dos resultados potenciais. O efeito na demanda de um aumento nos preços, por exemplo, é verificado considerando-se qual teria sido a demanda se os preços não tivessem sido modificados e comparando-se esse potencial resultado ao obtido com o aumento dos preços. Há assim, três elementos essenciais para a inferência causal: a unidade que sofre a ação, que pode ser um produto, pessoa, grupo, enfim, qualquer elemento; uma ação; os resultados potenciais.

Enquanto tanto a unidade quanto a ação são observados, nem todos os resultados potenciais o são. O resultado que ocorreu é observado (por exemplo, a demanda após o aumento do preço), porém, todos os demais resultados potenciais não o são. Assim, Imbens e Rubin (2015) afirmam que o problema da inferência causal é, de certo modo, um problema de variável omitida. Os contrafactuais, ou seja, aquilo que teria acontecido se a ação não tivesse sido aplicada naquela

[4] Um estimador é consistente quando as estimativas geradas, à medida que o número de observações cresce, convergem em probabilidade para o valor verdadeiro do parâmetro que se pretende estimar.

unidade, não são observados. No caso da demanda por hotéis, por exemplo, não se observa o que teria acontecido se os preços não tivessem sido alterados. Por isso, apreender os efeitos causais usualmente requer a observação de mais de uma unidade. Pode-se observar unidades diferentes no mesmo período ou observar uma em diferentes momentos. Neste caso, trata-se a unidade como se fosse uma diferente a cada momento.

Há, ainda, um quarto elemento a ser considerado quando se trata de inferência causal, qual seja, um modelo teórico, anteriormente referido. Se houver um modelo que descreve a ação e seus possíveis efeitos, estabelecendo os mecanismos que fazem a primeira engendrar os últimos, pode-se traçar a relação causal ao se observar estimativas coerentes com as previsões teóricas. Pode-se, ainda, determinar quais modelos teóricos são consistentes com estimativas resultantes de procedimentos estatísticos. No exemplo da demanda por hotéis, há um modelo teórico de oferta e de demanda que, de um lado, leva os econometristas a adotarem certos procedimentos para estimação; do outro, permite que os resultados sejam comparados com a teoria de modo a se verificar se as conclusões desta são observadas empiricamente.

A partir desses elementos, inferência causal pode ser definida como a dedução dos efeitos de uma ação em uma unidade. Essa dedução parte de premissas que, como afirmado anteriormente, nem sempre podem ser verificadas ou testadas, e dos resultados observados da ação sobre a unidade. Esse é o núcleo da maior parte dos problemas econométricos. Quando um econometrista se preocupa com a presença de viés em uma estimativa ou com a consistência de um estimador, quer garantir que aquilo que está sendo medido são (i) os efeitos estimados da ação investigada e que (ii) as estimativas do efeito convergem em probabilidade para o próprio efeito (consistência do estimador) ou que o valor esperado do estimador do efeito é igual ao próprio efeito (ausência de viés do estimador).

Enquanto o núcleo da econometria é a inferência causal, o do aprendizado de máquina é a busca da compreensão da estrutura subjacente dos dados e sua regularidade por meio do desenvolvimento de modelos que podem ser usados para previsões, classificação ou outros objetivos (THEODORIS, 2015). Esses modelos são automatizados por meio de algoritmos que, ao serem aplicados aos dados, apreendem padrões que os permitem fazer previsões, classificações e diversas outras tarefas. O procedimento usual no aprendizado de máquina consiste em quatro etapas:

1. Definição da estrutura do problema (previsão, classificação, agrupamento, por exemplo);

2. Definição do modelo que se acredita adequado para o problema ou o conjunto de dados (por exemplo, uma regressão linear);

3. Implementação desse modelo por meio de um algoritmo;

4. Avaliação do modelo com base nos resultados do passo anterior.

Na prática, esse procedimento requer também a coleta e limpeza de dados, bem como análise prévia de suas características (análise descritiva). As etapas concernentes à coleta e à limpeza dos dados são comuns à econometria e ao aprendizado de máquina.

Para a realização do procedimento acima, os dados disponíveis costumam ser divididos em subconjuntos (treinamento, validação e teste). O conjunto de treinamento é usado para estimar o modelo, o de validação para escolhê-lo e o de teste para avaliar seu desempenho. Muitas vezes, os conjuntos de teste e de validação são combinados (VARIAN, 2014).

A utilização de diferentes conjuntos é feita para que se possa analisar os resultados do modelo quando aplicado a observações fora da amostra, minimizando-se, assim, a chance de que aquele não seja generalizável. Se o modelo tiver o desempenho esperado fora do conjunto de treinamento, pode ser usado para predição, assumindo-se que o conjunto de dados atuais (treinamento, validação e teste) têm a mesma estrutura do conjunto de dados a serem usados nas previsões.

Para encontrar o modelo ótimo, é preciso definir um critério, que se traduz matematicamente por meio de uma função que quantifica o erro entre os valores preditos e os observados. Ao minimizá-la, obtém-se os parâmetros estimados mais adequados segundo o critério adotado. Essa função é denominada de função custo. Para cada método ou problema, diferentes funções custo podem ser usadas.

Na estimação de uma regressão linear, por exemplo, os parâmetros são encontrados por meio da minimização da norma euclidiana $((\boldsymbol{y} - \boldsymbol{X\beta})'(\boldsymbol{y} - \boldsymbol{X\beta}))$[5]. Na linguagem de aprendizagem de máquinas essa norma seria a função custo, e a melhor aproximação seria o estimador de mínimos quadrados $((\boldsymbol{X'X})^{-1}\boldsymbol{Xy})$. Nada impediria, contudo, que se definisse a melhor aproximação como os valores estimados a partir da otimização de $|\boldsymbol{y} - \boldsymbol{X\beta}|$, por exemplo. Nesse caso, $|\boldsymbol{y} - \boldsymbol{X\beta}|$ seria a função custo.

[5] No decorrer do texto, termos em negrito são vetores ou matrizes e termos não negritados são escalares.

A partir das descrições e conceitos apresentados, as diferenças entre os métodos econométricos e os de aprendizado de máquina ficam evidentes. Enquanto na primeira, busca-se inferir causalidade e, para isso, os estimadores devem ser não viesados, na segunda, o principal objetivo é prever, classificar ou agrupar as observações não utilizadas nas estimações. Como se verá adiante, estimadores viesados podem ser usados se houver ganho na diminuição da variância e, consequentemente, melhora nas previsões. Assim, no aprendizado de máquina, faz-se uma escolha consciente entre precisão e consistência (ou ausência de viés). A depender do objetivo, pode-se prescindir da segunda em favor da primeira. Na econometria, diferentemente, a escolha tende a ser sempre pela ausência de viés. Paralelamente, muitas vezes, assume-se que a escolha de uma amostra aleatória ou adequada à heterogeneidade dos efeitos é suficiente para garantir a generalização do modelo (MUELLER, 2015).

Ainda que os principais objetivos sejam diversos, a econometria e o aprendizado de máquina têm pontos em comum. Uma parcela importante dos métodos do último, chamado de aprendizado supervisionado, parte de modelos estatísticos também usados na econometria. Nesses métodos, usa-se um conjunto de variáveis para se prever um determinado resultado. Tanto as variáveis exógenas quanto as endógenas são observadas. Se a variável dependente for contínua, um dos principais métodos usados no aprendizado de máquina supervisionado são as regressões. Se a variável for categórica, são usados modelos de classificação, entre os quais os modelos probit e logit, bastante utilizados por econometristas.

Uma consequência direta do estudo do aprendizado de máquina por economistas é a ampliação do conjunto de métodos disponíveis para a solução de problemas. Conhecendo-se a especificidade de cada um desses modelos e suas limitações, pode-se usá-los para a resolução de problemas econômicos. Nesse sentido, Athey e Imbens (2018) ressaltam que muitos modelos econômicos podem ser decompostos em duas partes. A primeira teria o objetivo de inferir causalidade e, a segunda, o de prever.

A aplicação de métodos de aprendizado de máquina para a resolução de problemas do segundo tipo é direta. Para a inferência causal, Varian (2014) argumenta que, mesmo se um modelo preditivo não permitir inferir causalidade, pode ajudar a estimar o impacto de uma intervenção quando esta ocorrer. Para isso, pode-se estimar um modelo preditivo antes da intervenção, fazer as previsões e compará-las com os resultados pós-intervenção. As previsões seriam estimativas dos contrafactuais ou, nos termos de Imbens e Rubin (2015), um meio de lidar com o problema de variáveis omitidas (resultados não ocorridos).

Essa é uma forma de se inferir a causalidade por meio de modelos de aprendizado de máquina. Tomando-se os devidos cuidados tanto na seleção de variáveis quanto na caracterização da função objetivo (ou função custo) e, adotando-se premissas que vão além das matemáticas e, muitas vezes não podem ser testadas, é possível usá-los para inferência causal.

Outra possibilidade para a economia é usar as técnicas de aprendizado para entender os dados e analisar como as variáveis interagem entre si. O uso de algoritmos seria um meio de testar inúmeros modelos e escolher o mais adequado para um problema. Nesse sentido, embora os economistas argumentem serem seus modelos construídos a partir da teoria econômica, as variáveis a serem incluídas e a forma como interagem muitas vezes não são determinadas pela teoria (ATHEY; HORVITZ, 2018). Tanto os modelos quanto os procedimentos do aprendizado de máquina podem ser utilizados para aprimorar etapas do processo de estimação econométrica, como a seleção de variáveis, os critérios de escolha de um modelo, o processo de análise da robustez e sua validação (ATHEY; IMBENS, 2018).

Além do aprendizado supervisionado, cujos métodos se aproximam dos econométricos, existe um outro conjunto de modelos que visam reconhecer padrões. Esses métodos são classificados como não supervisionados. Os dados usados, diferentemente daqueles dos métodos supervisionados, não são rotulados, ou seja, não há informação anterior que permita definir o que seriam as variáveis de saída (ou variáveis endógenas, como nos modelos supervisionados) (IZENMAN, 2008). Os principais métodos desse grupo visam agrupar os dados (*clustering*) ou reduzir dimensões de modo que se possa encontrar a informação desejada.

Como argumentado na introdução, embora não sejam ainda muito usados na economia, há potencial para que sua utilização cresça, havendo pelo menos dois caminhos possíveis e não excludentes entre si para o uso desses modelos na econometria. O primeiro é seu emprego como um estágio da análise econômica. Métodos de agrupamento, por exemplo, podem ser usados para particionar as variáveis e estimar efeitos de tratamento em cada elemento da partição, no caso de efeitos heterogêneos (ATHEY, 2017) ou com o fim de segmentar variáveis adequadamente para posterior análise (ATHEY, IMBENS, 2018).

O segundo é a aplicação para a resolução de problemas econômicos, tais como o exemplo de definição de mercado relevante mencionado na introdução a esse texto. Conforme o Guia para Análise de Atos de Concentração Horizontal, "*a delimitação do MR [mercado relevante} é um processo de identificação do conjunto de agentes econômicos (consumidores e produtores) que efetivamente reagem e limitam as decisões referentes a estratégias de preços, quantidades, qualidade (entre outras) da empresa resultante da operação*" (CADE, 2016, p. 13). Trata-se,

portanto, de um processo que visa destacar os agentes cujas ações repercutem na empresa analisada e que, concomitantemente, são afetados por essa empresa. Como afirmado na introdução, pode-se compreender esse problema como um de agrupamento: cumpre identificar quais agentes fazem parte de um mesmo conjunto, no quais as ações afetam reciprocamente uns aos outros. A depender do caso, o problema pode ser visto, alternativamente, como um de classificação, no qual os ofertantes são compreendidos como concorrentes, mas há concorrentes mais ou menos próximos. Definidos parâmetros, é possível classificar os ofertantes e determinar quais fazem parte do mercado relevante.

Por fim, há um terceiro conjunto de modelos que não serão objeto deste estudo, que compõem um grupo chamado de aprendizagem por reforço. Esses modelos são compostos por agentes, que realizam ações em um determinado ambiente (tudo com o qual os agentes interagem) e são recompensados de acordo com essa ação. O agente deve alcançar o seu objetivo o mais rapidamente possível e deve aprender qual é a melhor função (*policy*) para isso.

3. APRENDIZAGEM DE MÁQUINA E DEFESA DA CONCORRÊNCIA

A economia digital tem sido motivo de preocupação de autoridades antitruste no mundo todo, pois alterou a dinâmica competitiva de mercados existentes e criou novos mercados com características singulares, tais como serem *multi-sided,* haver efeitos de rede e substanciais economias de escala e escopo, além de dependência dos dados dos usuários, custos de mudança, preço nulo ou muito baixo. Essas características podem resultar na concentração dos mercados, no surgimento de conglomerados digitais e em uma dinâmica de competição pelo mercado (OCDE, 2022). Com o fim de analisar casos nesse setor adequadamente, tem-se feito esforços de capacitar os técnicos das autoridades concorrenciais. Exemplos disso são o treinamento dos funcionários do Departamento de Justiça norte-americano (DELRAHIM, 2020) e as ações do próprio Cade, que contratou consultoria sobre o tema na qual foi incluído treinamento para a equipe. A capacitação para a análise de casos nesse setor não implica necessariamente o treinamento para a aplicação dos modelos e algoritmos empregados nas condutas.

A aplicação de modelos e algoritmos de aprendizado de máquina à defesa da concorrência requer tanto existência de pessoal treinado quanto disponibilidade de dados. No que tange ao primeiro ponto, há que se considerar que a aprendizagem de máquina ainda é um campo aberto. Muitos estatísticos, engenheiros e cientistas da computação trabalham na área. Na economia, o interesse pela análise de dados, além da econometria, tem crescido. No Massachusetts Institute of Technology (MIT), os departamentos de economia e de ciências da computação formaram parceria

para oferecer um novo curso (*major*) em ciências da computação, economia e ciência de dados. Tanto a procura de estudantes de economia por cursos de aprendizado de máquinas aumentou, quanto a procura de empresas de economia digital por economistas cresceu consideravelmente (ATHEY; LUCA, 2019).

Para as autoridades antitruste, isso significa ter uma equipe diversa, não só composta por advogados e economistas. Implica, ainda, treinar os economistas e advogados que já atuam nas agências – se já não tiverem o conhecimento necessário – para aplicar as técnicas mais adequadas aos problemas que enfrentam, considerando os dados que dispõem. Como mencionado anteriormente, esses dois movimentos estão sendo feitos por diversas autoridades antitruste. No Cade, a equipe do Departamento de Estudos Econômicos (DEE) tem cientistas da computação. Paralelamente, economistas e advogados são encorajados a ampliar sua formação na análise de dados.

A segunda questão a ser considerada para aplicação de modelos ou algoritmos de aprendizado de máquina no antitruste é a disponibilidade de dados. Usualmente, associa-se essa área ao que se convencionou chamar de *big data*, ou seja, conjuntos de dados enormes, que crescem exponencialmente e que não são passíveis de serem analisados com os métodos tradicionais. Os algoritmos de aprendizado de máquina são um meio de extrair-se conclusões a partir desses conjuntos de dados. Concomitantemente, há algoritmos de aprendizado de máquina que precisam de um conjunto substancial de observações para ser adequadamente treinado. Em linhas gerais, pode-se afirmar que, quanto mais complexo o algoritmo, mais dados devem ser usados no seu treinamento. A falta de dados, contudo, pode ser contornada pela escolha de um modelo adequado à base disponível ou por meio de algoritmos criados com o fim de superar essa lacuna.

Na academia, questões afeitas ao antitruste já estão sendo analisadas com o uso de *big data*. Por exemplo, em (ATHEY et al., 2018), um modelo de escolha do consumidor por restaurantes foi estimado com o uso de dados de milhares de donos de telefones celulares. Em empresas da economia digital, por sua vez, uma das tarefas dos economistas é analisar problemas concorrenciais, tais como a quantidade de concorrentes que um dado mercado suporta (ATHEY; LUCA, 2019).

Diversas agências antitruste no mundo estão procurando aproveitar a disponibilidade de dados e usar o potencial dessas técnicas para aprimorar a análise antitruste e monitorar mais efetivamente mercados. A CMA, no Reino Unido, criou uma unidade com, pelo menos, 15 cientistas de dados. Entre os projetos já efetivados estão o desenvolvimento de ferramentas de aprendizado

de máquina para identificação de possíveis infrações da lei do consumidor nas plataformas digitais e aplicação de técnicas de processamento de linguagem natural para peinerar e revisar 100.000 documentos de companhias diversas recebidos em diversos casos (HUNT, 2019).

A Autoridade antitruste grega (*Hellenic Competition Commission*) também criou uma unidade de ciências de dados, que desenvolveu uma plataforma para coleta e processamento de dados econômicos em tempo real. Segundo o presidente da HCC, Ioannis Lianos, algoritmos viabilizam oportunidades adicionais para que as autoridades antitruste detectem colusões ou outras práticas anticompetitivas com base em *big data*. Essas ferramentas complementam as correntemente utilizadas (HELLENIC COMPETITION COMMISSION, 2021).

Outras experiências no continente europeu são as das autoridades francesa e espanhola. Ambas investiram em pessoal e equipamento para aprimorar seus métodos de investigação e monitoramento de mercados com o uso de tecnologia da informação. A experiência espanhola tem muitas similaridades com a brasileira. A fim de detectar casos de cartel e diminuir a dependência de denúncias e leniências, a autoridade desenvolveu ferramentas de análise de dados. Inicialmente, o foco foram os cartéis em licitação, mas o projeto está-se expandindo (CAMPUZANO, 2021).

Para além da detecção de cartéis, modelos simples de aprendizado de máquina, tais como os apresentados neste trabalho, podem ser usados em análises de casos concorrenciais de diferentes formas. A primeira, discutida na seção anterior, é o aprimoramento dos modelos econométricos. Em casos nos quais se pretende estimar o impacto de um ato de concentração ou a possibilidade de uma conduta causar prejuízos à concorrência, podem ser usados procedimentos de aprendizado de máquina no processo de estimação de modelos estruturais. Como se verá na seção 5, o Lasso (*least absolute shrinkage and selection operator*), uma regressão regularizada, pode ser usado na escolha das variáveis de controle de um modelo. Em (CAJUEIRO, 2018), o lasso foi usado na estimação de demanda.

A segunda aplicação a ser considerada é na previsão de comportamentos de ofertantes e consumidores ou de alterações de estruturas competitivas. Na análise de efeitos unilaterais de atos de concentração horizontais, por exemplo, verifica-se se é provável a entrada de novos concorrentes, se uma eventual entrada seria tempestiva – uma nova empresa estaria funcionando de forma completa e adequada em até dois anos – e suficiente – os entrantes teriam condições de deter os efeitos anticompetitivos decorrentes do ato de concentração (CADE, 2016). Esse é um típico problema de previsão. À análise qualitativa e econométrica já feita pelo Cade, pode-se somar a utilização de algoritmos para previsão da entrada e da estimação da probabilidade de que ocorra em menos de dois anos. Considerando-se os mercados geográficos, podem-se utilizar algoritmos

diversos para identificar características que tornam um mercado mais propenso à entrada e quais os mercados com essas características.

Uma terceira possibilidade a se considerar é a utilização de algoritmos de aprendizagem de máquina para a execução de etapas da análise concorrencial que, atualmente, são feitos de forma padronizada, mas sem que se depreendam dos dados disponíveis todas as informações que eles podem fornecer, tal como a já citada definição de mercado relevante. Em muitos setores, o Cade utiliza procedimentos padronizados para identificar os mercados relevantes. Esses procedimentos foram concebidos considerando-se restrições que, eventualmente, podem ser superadas por meio de algoritmos de aprendizagem de máquina.

Há, assim, grande potencial para a utilização de algoritmos e modelos de aprendizagem de máquinas na defesa da concorrência. Ainda que algumas aplicações não pareçam factíveis no curto prazo, muitos procedimentos e modelos podem ser usados atualmente de forma a tornar mais robustas conclusões em processos antitruste. Os procedimentos, conceitos e modelos discutidos nas próximas seções são exemplos de técnicas facilmente aprendidas por economistas e outros profissionais e que podem ser úteis na defesa da concorrência.

4. VALIDAÇÃO CRUZADA

Como afirmado anteriormente, modelos de aprendizado de máquina são utilizados principalmente para previsão, classificação e agrupamento. Tendo esses objetivos em conta, a avaliação dos modelos baseia-se na análise de seu desempenho quando aplicados a observações fora da amostra usada para seu treinamento. Isso é feito porque usar os dados do conjunto de treinamento para avaliar o desempenho de um modelo tende a resultar em avaliação enviesada. Um modelo com muitos parâmetros estimados, por exemplo, tende a ter um erro muito pequeno quando avaliado no conjunto de treinamento. Contudo, seu desempenho tende a ser pior quando aplicado a dados fora daquele conjunto (THEODORIS, 2015).

Mencionou-se a prática de divisão dos dados em três conjuntos: treinamento, validação e teste, sendo que, por vezes, utiliza-se o mesmo conjunto de dados para validação e teste. O primeiro seria usado para ajustar o modelo, o segundo, para selecioná-lo e o terceiro para avaliar o erro do modelo quando aplicado em um conjunto de dados diferentes do usado no seu ajuste, ou seja, quando o modelo é generalizado. Quando a divisão dos dados em conjuntos tais como descrito não é factível, existem outros procedimentos para verificar a capacidade preditiva do modelo fora do conjunto de treinamento. Um dos procedimentos mais utilizados é a validação cruzada, que é um método de fácil compreensão e execução e que costuma resultar em modelos

menos viesados e otimistas que outros métodos. Dentre os procedimentos utilizados de validação cruzada, um dos mais conhecidos é a K-*fold cross validation*.

Conforme apresentado em Hastie, Tibshirani e Friedman (2009), nesse procedimento, o conjunto de dados é dividido em k partes, sendo k-1 partes usadas como conjunto de treinamento e a k-ésima parte como conjunto de teste. As etapas seguintes são:

1. Determine a função custo a ser usada. É comum a utilização do erro quadrático em modelos de regressão. O erro quadrático é definido como $EQ = \sum_{i=1}^{k}(y_i - \hat{\mu}^{-k(i)}(\boldsymbol{X}_i))^2$, onde y_i são os valores observados de y no conjunto de teste, $\hat{\mu}^{-k(i)}$ são os valores dos parâmetros estimados a partir do conjunto de treinamento e $\boldsymbol{X}_i$ são os valores observados de $\boldsymbol{X}$ no conjunto de teste;
2. Ajuste o modelo usando o conjunto de treinamento;
3. Faça a previsão a partir do modelo aplicado no conjunto de teste e calcule o custo associado, por exemplo, o erro quadrático médio (EQM), que é a média do quadrado da diferença entre os valores estimados de um parâmetro e os valores do parâmetro[6];
4. Repita as etapas a partir do item 2 até que todas as partes k tenham sido usadas como conjunto de teste;
5. Calcule o erro mediante o EQM para cada k: $EP = \frac{1}{k}\sum_{k=1}^{k} EQM_k$.

Ao final do processo, o modelo terá sido treinado, em toda a base de dados disponível. Pode-se comparar o erro de previsão de diferentes modelos a fim de se escolher aquele com menor custo.

Muitos modelos têm um parâmetro que deve ser escolhido, como é o caso do Lasso (*least absolute shrinkage and selection operator*), que será discutido na seção 4. A validação cruzada é usada, nesses casos, para escolha desses parâmetros, que são chamados de parâmetros de ajuste. O procedimento é similar ao descrito acima. O modelo será ajustado para n diferentes parâmetros de ajuste a cada rodada k. Os custos associados a cada um dos n, em cada rodada k, serão calculados e, ao final, pode-se fazer a média dos custos associados ou alguma outra ponderação para cada parâmetro de ajuste n. Escolhe-se o parâmetro de ajuste que gera o menor custo.

[6] Matematicamente, sendo θ, o parâmetro a ser estimado, e $\hat{\mu}$, seu valor estimado, o erro quadrátrico médio é dado por: $\frac{1}{n}\sum_{i=1}^{n}(\hat{\mu} - \mu)^2 = \mathrm{E}(\hat{\mu} - \mu)^2$.

Quando se utiliza a validação cruzada para comparação de dois métodos cujos resultados esperados são parecidos, o conjunto de teste deve ser grande o bastante para permitir captar as diferenças entre os métodos. Em termos gerais, a escolha de k é, como em outros procedimentos de aprendizado de máquina, um *tradeoff* entre viés e variância. Quanto menor o k, maior tende a ser o viés, pois o conjunto de treinamento é menor e mais diferente do conjunto de teste. A variância, contudo, é menor. Segundo Theodoris (2015), o viés, em geral, não é um problema na aplicação da validação cruzada porque, em geral, o número de observações do conjunto de treinamento é próximo ao número de observações do conjunto teste, mas a variância pode ser. Se for muito alta, o autor recomenda a aplicação de outros métodos, como o *bootstrap*.

Além da relação entre viés e variância, na escolha de k, é necessário considerar a dificuldade computacional. À medida em que k diminui, o procedimento torna-se mais custoso computacionalmente. Três valores de k são muito utilizados: 5, 10, e z, sendo z o tamanho da amostra (nesse caso, o procedimento é conhecido como *leave-one-out cross-validation*).

A fim de ilustrar os resultados da aplicação da *k-fold cross validation* para diferentes k, estimou-se uma regressão linear na qual os custos assistenciais de planos médico-hospitalares de operadora de plano de saúde, são a variável endógena. Como variáveis explicativas foram incluídas o total de beneficiários, a proporção dos custos totais realizados em prestadores de saúde próprios e outras variáveis descritas no Anexo 1. Sendo o objetivo apenas comparar os resultados de diferentes funções custos gerados a partir de estimativas com diferentes k, as variáveis foram normalizadas. A descrição da base de dados, bem como de cada variável incluída na regressão são apresentadas no Anexo 1. Na Tabela 1, abaixo, são apresentados os resultados das regressões estimadas usando-se a *k-fold cross validation* considerando-se três medidas e diferentes *k*:

- RMSE (root mean-square error) – a raiz quadrada do EQM;
- MAE (mean absolute error), definido como $\frac{1}{k}\sum_{i=1}^{k}|y_i - \hat{\mu}^{-k(i)}(\boldsymbol{X}_i)|$;
- R^2.

*Tabela 1 - Resultados dos procedimentos de validação cruzada - medidas de ajuste dos modelos finais**

	R-squared	RMSE	MAE
sem validação	0.755	0.501	0.127
cv (k=5)	0.855	0.591	0.150
cv (k=10)	0.809	0.482	0.147
cv (k=n)	0.549	0.687	0.142
repeated cv (k=5)	0.876	0.600	0.153
repeated cv (k=10)	0.838	0.491	0.147

* k é o número de partes no qual foram divididas as observações.

Na Tabela 1, acima, o resultado da regressão estimada com toda a base de dados é a linha "sem validação". CV significa validação cruzada e *repeated cv* significa que foram realizadas três validações cruzadas. Nesse caso, os resultados são as médias dos resultados das três validações. Os códigos das estimações são apresentados no Anexo 2.

Como esperado, tanto o erro quadrático médio quanto o erro absoluto das regressões com validação cruzada são maiores do que da regressão em que foi utilizada toda a base de dados, com exceção das validações cruzadas quando o k=10, casos nos quais o RMSE diminui em relação à amostra completa. Interessante também notar que o R^2 das estimativas com validação cruzada, exceto quando k=n, são maiores que do modelo estimado com todas as observações.

Os resultados da Tabela 1 apontam para um fato mencionado por diferentes autores de que os resultados obtidos por meio da aplicação da *k-fold cross validation* podem ser afetados consideravelmente pela escolha de k. O processo pode criar relações entre as estimações, que tornam difícil a realização de testes estatísticos. Por isso, o ideal é ter-se uma base de dados grande o suficiente para que seja possível dividi-la em, ao menos, um conjunto de treinamento e um de teste.

5. TRADE-OFF VIÉS E VARIÂNCIA

Ainda que economistas não costumem usar a expressão "função custo" da mesma forma como é utilizada no aprendizado de máquina, em diversos procedimentos econométricos, os modelos também são avaliados a partir de uma função que tem por objetivo medir seu desempenho. Esse procedimento é comum no estudo de séries temporais, no quais se comparam diferentes modelos com base em critérios diversos, tais quais o critério de Akaike e Schwarz. Em

aplicações com dados em *cross-section* (ou em painéis), em muitos casos, aceitam-se modelos com pior desempenho a fim de se garantir que o estimador não seja viesado. Ainda assim, o erro quadrático médio ou o R^2, uma medida de ajuste de um modelo estatístico[7], costumam ser reportados.

No aprendizado de máquinas, a presença de viés não invalida, necessariamente, um modelo, haja vista que o objetivo principal é que este tenha bom desempenho fora da amostra, o que pode ocorrer mesmo se houver a presença de viés[8]. Considere-se, por exemplo, o EQM como medida de desempenho de um estimador. Utilizando-se a mesma notação da seção anterior, $\hat{\mu}$ é uma estimativa do parâmetro, e o parâmetro verdadeiro é μ_0. O EQM será, então, $\frac{1}{n}\sum_{i=1}^{n}(\hat{\mu}-\mu_0)^2 = \mathrm{E}(\hat{\mu}-\mu_0)^2$, sendo n a quantidade de observações. Dividindo-se as observações de modo a construir um conjunto de treinamento e um de teste, pode-se estimar o modelo com o conjunto de treinamento e calcular o EQM para ambos os conjuntos, utilizando para cada qual as observações pertinentes.

Com uma pequena manipulação, evidencia-se que o EQM é composto pelo viés e pela variância:

$$EQM = \mathrm{E}(\hat{\mu} - \mathrm{E}(\hat{\mu}) + \mathrm{E}(\hat{\mu}) - \mu_0)^2$$

$$EQM = \mathrm{E}[\big(\hat{\mu} - \mathrm{E}(\hat{\mu})\big)^2] + (\mathrm{E}(\hat{\mu}) - \mu_0)^2$$

O primeiro termo é a variância, o segundo, o quadrado do viés. Não é possível diminuir os dois termos, concomitantemente, sem aumentar o número de observações. E, por vezes, mesmo o aumento nesse número não reduz o viés ou a variância. Considerando-se o número de observações constantes, o aumento da complexidade do modelo, definida como o número de parâmetros a serem estimados, diminui o viés, mas faz crescer a variância. Por conseguinte, o EQM referente ao conjunto de treinamento diminui, mas o EQM referente ao conjunto de teste ou, em outras palavras, fora da amostra, tende a crescer.

---

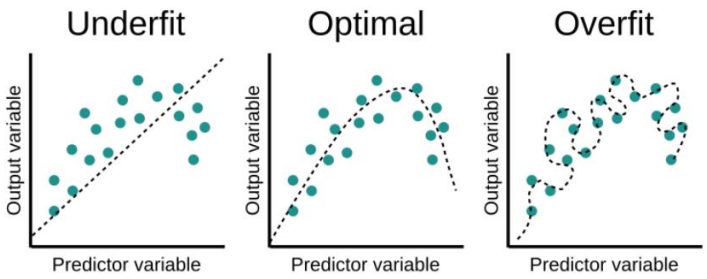
[7] O R^2 ou coeficiente de determinação é calculado a partir dos valores estimados de uma variável, a média de seus valores e seus valores observados. A soma do quadrado total (SQT) é dada pela soma do quadrado da diferença entre cada observação e a média de todas as observações ($SQT = \sum_{i=1}^{n}(y_i - \bar{y})^2$, onde y_i é a observação i e $\bar{y}$ é a média de todas as observações). A soma dos resíduos é a soma do quadrado da diferença entre o valor estimado de cada observação e o valor da observação ($SQE = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$, onde $\hat{y}_i$ é o valor estimado da observação y_i. O R^2 é dado por $\frac{1-SQE}{SQT}$.

[8] Um modelo pode alcançar seu fim, por exemplo, ter boa capacidade de predição ou classificação, mesmo tendo viés. Diminuir o viés exige, tudo o mais constante, a complexificação do modelo, o que tende a impactar na possibilidade de generalização. Isso dito, tem-se apontado para vieses em algoritmos que podem ter implicações éticas, como algoritmos que discriminam grupos específicos (MEHRABI, 2022).

Diz-se que há *overfitting* no modelo quando seu desempenho se considerando o conjunto de treinamento é muito bom, mas o desempenho em outros conjuntos não o é. Isso acontece quando, durante o processo de treinamento, os parâmetros estimados do modelo desconhecido aprendem demais das especificidades do conjunto de treinamento, e o modelo estimado tem desempenho ruim quando usado com outro conjunto de dados (THEODORIS, 2015). Existem diversas formas de tratar esse problema, todas baseiam-se no princípio de simplificar tanto quanto possível o modelo a ser estimado.

A Figura 1, abaixo, mostra três diferentes modelos ajustados ao mesmo conjunto de dados. No gráfico à direita, o modelo se ajusta a todos os pontos da base de dados, de modo que o EQM estimado no conjunto usado para estimação (conjunto de treinamento) é nulo. Contudo, é pouco provável que fora dessa amostra, os dados sejam exatamente iguais aos do conjunto de treinamento. O gráfico no centro da Figura, mostra um modelo que parece refletir a distribuição desconhecida dos dados, mas que não se ajusta a todos os dados. O EQM no conjunto de treinamento é maior do que o do modelo anterior, mas é provável que fora do conjunto amostral, o desempenho deste modelo seja melhor. Por fim, o gráfico à esquerda mostra um modelo que tem desempenho ruim quando avaliado com base no conjunto de treinamento e que, provavelmente, também tem desempenho insatisfatório quando avaliado no conjunto de teste.

Figura 1- Exemplo de modelos ajustados ao mesmo conjunto de dados



Fonte: https://livebook.manning.com/book/machine-learning-for-mortals-mere-and-otherwise/chapter-11/v-7/25, verificado em 2 de junho de 2020

Para mostrar com dados reais como o desempenho de um modelo pode variar, se avaliado no conjunto de treinamento ou no conjunto de teste, considere que se tenha por objetivo prever os custos assistenciais de uma operadora de planos de saúde. Além dos custos assistenciais em 2019, há na base dados a quantidade de beneficiários de cada operadora em dezembro de 2019 e a modalidade da operadora (seguradora especializada em saúde, medicina de grupo, cooperativa

médica, filantrópica ou autogestão). Estimaram-se regressões lineares, ajustadas por meio de mínimos quadrados ordinários. Foram usadas regressões desse tipo porque estas são muito conhecidas e familiares a boa parte dos leitores.

A base de dados foi dividida em dois conjuntos. O conjunto de treinamento incluiu 523 observações das 653 observações da base de dados. O restante das observações é o conjunto teste. Foram estimadas duas regressões:

$$\log(custo) = \beta_0 + \beta_1 \log(beneficiários) + \epsilon \quad \text{(reg.1)}$$

$$\log(custo) = \beta_0 + \beta_1 \log(beneficiários) + \beta_2 \log(população) + \beta_3 d_{seg} + \beta_4 d_{coop} + \beta_5 d_{fil} + \beta_6 d_{gest} + \epsilon \quad \text{(reg.2)}$$

Em ambas, β_0 é o intercepto, e β_1, o coeficiente associado à quantidade de beneficiários. As variáveis quantitativas foram transformadas logaritmiticamente. Na segunda regressão, foi incluída a população do município no qual a operadora tem a maior parte dos seus beneficiários e *dummies*[9] relacionadas à modalidade da operadora (d_{seg} se a operadora é uma seguradora especializada em saúde, d_{coop} se a operadora é uma cooperativa médica, d_{fil}, se é filantrópica e d_{gest}, se é uma autogestão. O erro é representado por ϵ.Os resultados das regressões são apresentados na Tabela 2, abaixo. Como se pode observar, salvo pela *dummy* relacionada ao fato de a operadora ser filantrópica, todas as variáveis são significativas. Na seção 5, será discutido o uso de regressões regularizadas para determinar quais variáveis devem fazer parte de um modelo de previsão. Por ora, note-se a significância dos coeficientes e o pequeno aumento do R^2, referente ao conjunto de treinamento, quando foram acrescidas as *dummies* referentes à modalidade e a população.

[9] Dummy é uma variável binária, utilizada para representar a presença de uma determinada característica.

Tabela 2 - Resultados do modelo de regressão linear - custo assistencial

	Dependent variable:	
	Custo assistencial	
	(1)	(2)
Beneficiários	1.041***	1.002***
	(0.024)	(0.024)
População		0.126***
		(0.026)
Filantropia		0.191
		(0.179)
Cooperativa		0.719***
		(0.094)
Seguradora		0.988***
		(0.343)
Autogestão		0.973***
		(0.106)
Constante	7.426***	5.665***
	(0.234)	(0.352)
Observations	523	523
R2	0.787	0.831
Adjusted R2	0.787	0.829
Residual Std. Error	0.897 (df = 521)	0.803 (df = 516)
F Statistic	1,928.830*** (df = 1; 521)	423.252*** (df = 6; 516)
Note:		*p<0.1; **p<0.05; ***p<0.01

Na Tabela 3, são apresentados os resultados da raiz do erro quadrático médio para as duas equações, medidos para o conjunto de treinamento e o conjunto de teste. Os coeficientes estimados na regressão 1, quando usados no conjunto teste, resultam em EQM maior do que no conjunto de treinamento. Já a regressão 2, os EQM calculados com base nos conjuntos de treinamento e de teste são similares, o que indica que o modelo pode ser generalizado com menor probabilidade de que perda poder de previsão. Nesse caso, a inclusão das variáveis, não gera *overfit* e melhora o ajuste do modelo, como pode ser verificado nos Gráficos 1 a 4, abaixo.

Tabela 3- Raiz do EQM - Regressões 1 e 2

	Treinamento	Teste
Regressão 1	0.89	0.99
Regressão 2	0.80	0.79

Gráfico 1 *- Regressão simples aplicada ao conjunto treinamento*

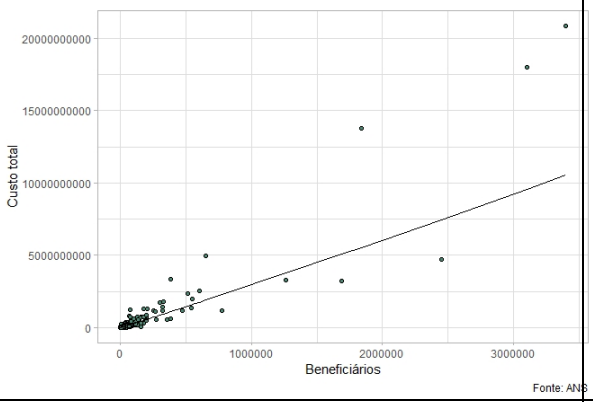


Gráfico 2 *- Regressão completa aplicado ao conjunto treinamento*

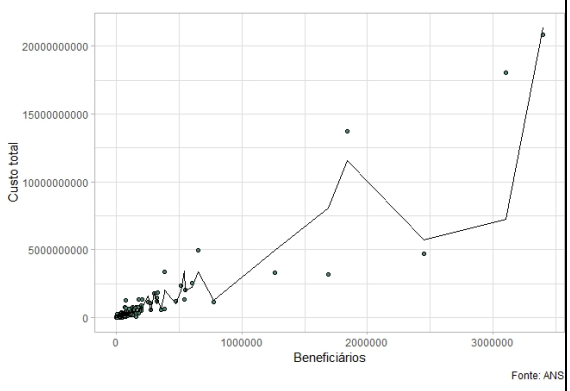


Gráfico 3 *- Regressão simples aplicada ao conjunto teste*

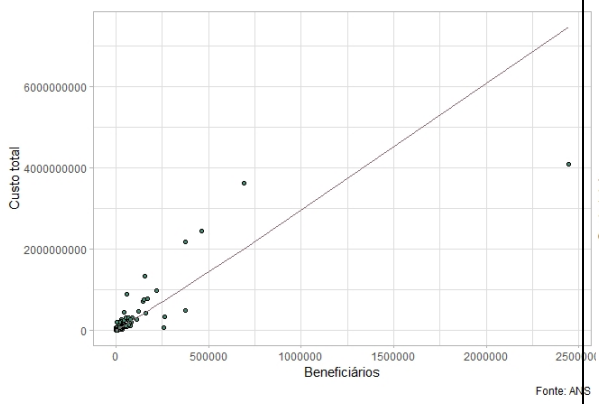
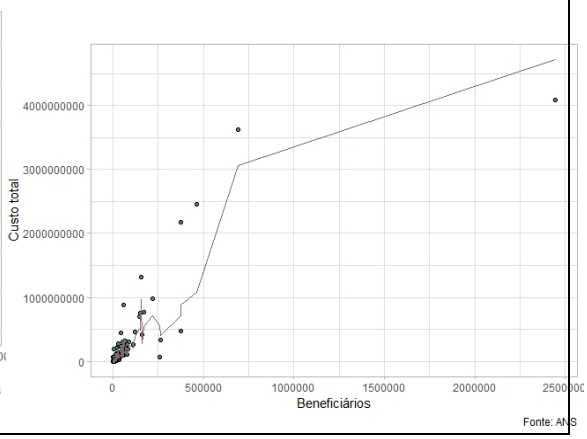


Gráfico 4 *- Regressão completa aplicada ao conjunto teste*



Além de ilustrar conceitos apresentados nesta seção, esse exemplo mostra que se pode usar procedimentos associados à aprendizagem de máquina em diferentes modelos estatísticos e econométricos a fim de aprimorar a seleção de modelos e, quando for o caso, sua capacidade de generalização. Nesse sentido, duas perguntas são suscitadas pelo que foi apresentado. A primeira refere-se à quantidade de dados no conjunto de treinamento necessária para que um modelo possa ser generalizado. No exemplo acima, a base de dados continha apenas 653 observações. Como o modelo estimado era simples, no entanto, a quantidade de observações era suficiente. Na seção

seguinte, será discutida o volume de dados necessário para estimação dos modelos discutidos neste trabalho.

A segunda pergunta que fica premente é como determinar as variáveis explicativas a serem incluídas em uma regressão. Na economia, costuma-se argumentar que a justificativa para inclusão ou exclusão de uma variável em um modelo econométrico baseia-se no modelo teórico que norteia a estimação. Em muitos casos, contudo, como se discutirá mais profundamente na seção seguinte, há variáveis no modelo econométrico que não necessariamente fazem parte do modelo teórico. Nesses casos, não há um procedimento padrão utilizado pelos economistas para decidir pela inclusão ou exclusão dessas variáveis. Modelos de regressão regularizada, que são usadas para diminuir o *overfitting*, podem ser usados também para se decidir sobre a inclusão de variáveis em modelos econométricos. Esses modelos serão apresentados na seção seguinte.

6. REGRESSÕES REGULARIZADAS

6. 1 Uma breve explicação sobre regressões regularizadas

Um dos principais métodos utilizados por economistas para encontrar os parâmetros de uma regressão linear é o mínimos quadrados ordinários (MQO). Se as variáveis explicativas forem exógenas, não houver multicolinearidade[10] e o erro da regressão for homocedástico[11] e não correlacionado com as variáveis explicativas, o estimador de mínimos quadrados é consistente e eficiente (tem a menor variância entre todos os estimadores não viesados).

Como visto acima, na maior parte das vezes, ao se estimar empiricamente um modelo econômico, não se deseja que as estimativas sejam viesadas. Essa é uma das razões pelas quais o estimador de MQO é tão utilizado. Outros estimadores, contudo, começam a ser cada vez mais usados em modelos empíricos econômicos. Regressões regularizadas têm-se popularizado, sendo três os seus principais usos na Economia: seleção de variáveis; estimação de componentes preditivos de modelos econômicos; estimação de modelos econômicos cujos dados não satisfazem as hipóteses do MQO. Cada um desses usos será discutido nesta seção. Antes disso, porém, cabe explicar o que são regressões regularizadas.

Diferentemente de outros métodos empregados no aprendizado de máquinas, as regressões regularizadas já são estudadas há bastante tempo por estatísticos. Suas propriedades são, por essa

[10] Multicolinearidade acontece quando duas ou mais variáveis explicativas são altamente correlacionadas

[11] Homocedasticidade quer dizer que a variância dos erros, condicionada às variáveis explicativas, é constante.

razão, bem conhecidas. Nessas regressões, a função custo é caracterizada pelo estimador de mínimos quadrados e um termo adicional, chamado de termo de regularização. Este é composto pela norma[12] *p* dos parâmetros a serem estimados, sendo a função custo de uma regressão regularizada escrita como

$$\left(\sum_{i=1}^{k}\left(y_i - \widehat{\boldsymbol{\mu}}(\boldsymbol{X}_i)\right)^2\right) + \lambda\left(\sum_{i=1}^{l}|\hat{\mu}_i|^p\right)^{1/p} \quad . \quad (1)$$

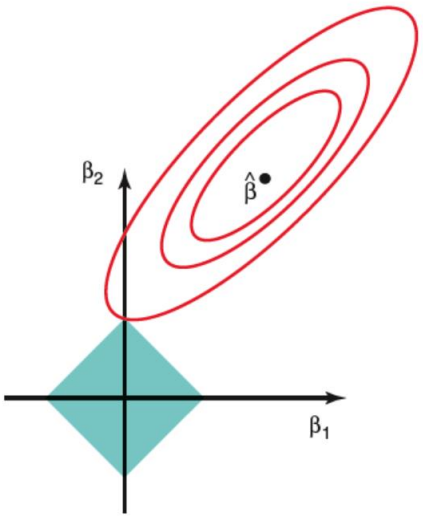
Nesta equação, *k* é a quantidade de observações, é y_i é a variável explicada ou dependente, $\boldsymbol{X}_i$ é um vetor composto por todas as variáveis explicativas ou independentes, $\widehat{\boldsymbol{\mu}}$ é o vetor de coeficientes que minimizam a função, *l* é a quantidade de coeficientes ($\widehat{\boldsymbol{\mu}}$), *p* é a norma, e λ é um parâmetro definido por quem está estimando a regressão e que controla a importância do segundo termo, que é o termo de regularização. O primeiro termo é o estimador de mínimos quadrados.

Na minimização do primeiro termo, busca-se reduzir o somatório do erro quadrático, ou seja, aproximar-se tanto quanto possível o valor estimado do valor observado. Eleva-se ao quadrado porque se evita que valores positivos contrabalancem valores negativos. Além disso, com essa operação, pequenas diferenças são minimizadas, e grandes diferenças são maximizadas. A minimização desse termo pode resultar em modelos mais complexos, ou seja, com muitos parâmetros. A otimização do segundo termo penaliza a complexidade. Desse modo, torna-se menos provável que haja *overfitting*. Como exemplo, na Figura 2, abaixo, apresenta-se, graficamente, o resultado da minimização da função custo para o estimador de MQO (ponto $\hat{\beta}$) e para o estimador regularizado com norma p=1 (conhecido como *Lasso*). A elipse mostra o contorno do MQO, e o quadrado, a fronteira do parâmetro de regularização. O resultado da estimação da regressão regularizada é o ponto no qual a elipse é tangente ao quadrado.

[12] Para que um objeto seja uma norma, é preciso que atenda quatro condições. Sendo $\| \quad \|_p$ o operador da norma p, definida como $\left(\sum_{i=1}^{l}|\hat{\mu}_i|^p\right)^{1/p}$, essas condições são :

1. $\|\boldsymbol{\beta}\|_p \geq 0$;
2. $\|\boldsymbol{\beta}\|_p = 0 \leftrightarrow \boldsymbol{\beta} = 0$;
3. $\|\alpha\boldsymbol{\beta}\|_p = |\alpha|\|\boldsymbol{\beta}\|_p \ \forall\ \alpha \in \mathcal{R}$;
4. $\|\boldsymbol{\beta_1} + \boldsymbol{\beta_2}\|_p \leq \|\boldsymbol{\beta_1}\| + \|\boldsymbol{\beta_2}\|$.

Figura 2 - Estimador de MQO e estimador Lasso



Fonte: HASTIE, TIBSHIRANI e FRIEDMAN, 2017, p. 71.

O termo de regularização tem, assim, papel central nos resultados do modelo. A depender da norma que se escolha (ou seja do *p*), o modelo assume diferentes características. Considerando-se a eq. (1), a Tabela 4, mostra os métodos, conforme a existência do termo de regularização e a norma *p*. Como se sabe, se não houver termo de regularização, o estimador é MQO. A rigor, não há norma definida quando $p = 0$ (basta notar que uma norma é definida por $\left(\sum_{i=1}^{k}|\hat{\mu}_i|^p\right)^{1/p}$, que não pode ser calculado), mas, por definição, a norma neste caso é tida como o número de $\hat{\mu}_i$s diferentes de zero. A regressão regularizada é chamada de *subset selection* e se caracteriza por ser um seletor de variáveis.

Para implementação da *subset selection,* são comparados todos os modelos compostos por subconjuntos de $\hat{\boldsymbol{\mu}}$**.** Seleciona-se o modelo com o melhor desempenho, considerando-se a função custo. O problema é que a quantidade de análises cresce de forma combinatória em relação à quantidade de parâmetros. Mesmo quando o número total de parâmetros p é pequeno, o número de subconjuntos tende a ser grande. Por isso e por o Lasso (*least absolute shrinkage and selection operator*) também funcionar como um seletor de variáveis e ser computacionalmente mais fácil de aplicar, *subset selection regressions* não são comumente usadas.

Tabela 4 - Métodos de estimação

Método	Norma do termo de regularização
MQO	O termo é igual a 0.
Subset selection	p=0*
Lasso	p=1
Ridge	p=2
Elastic net	Combinação linear de p=1 e p=2

Fonte: Adaptado de ATHEY; IMBENS, 2018

Quando a norma é um, a regressão é o Lasso. A sua função custo é, portanto, $\left(\sum_{i=1}^{k}\left(y_i - \hat{\boldsymbol{\mu}}(\boldsymbol{X}_i)\right)^2\right) + \lambda \sum_{i=1}^{l} |\hat{\mu}_i|$. Pode-se escrever o problema de minimização também como

$$\left(\sum_{i=1}^{k}\left(y_i - \hat{\boldsymbol{\mu}}(\boldsymbol{X}_i)\right)^2\right) \qquad (2)$$

s. a

$$\lambda \sum_{i=1}^{l} |\hat{\mu}_i|$$

Nas equações acima, k é a quantidade de observações, é y_i é a variável explicada ou dependente, $\boldsymbol{X}_i$ é um vetor composto por todas as variáveis explicativas ou independentes, $\hat{\boldsymbol{\mu}}$ é o coeficiente que minimiza a função, p é a norma, e λ é um parâmetro definido por quem está estimando a regressão e que controla a importância do segundo termo, que é o termo de regularização. O primeiro termo é o estimador de mínimos quadrados.

A figura 2, acima, mostra a otimização de um Lasso. Quando a elipse que representa o contorno do estimador de MQO tangencia o termo de regularização em um de seus vértices. a otimização resulta no valor nulo de pelo menos um parâmetro. Nos outros casos, a solução tem coeficientes menores do que a solução de MQO. Como mencionado acima, esses coeficientes são viesados.

Em decorrência da possibilidade de ocorrência de soluções no qual o estimador de MQO tangencia o termo de regularização em um vértice, como mostrado na Figura, o Lasso é usado como seletor de variáveis. Nas aplicações econômicas, o procedimento mais comum de seu uso com o fim de selecionar variáveis tem dois passos:

1. Estima-se um Lasso com todas as variáveis do modelo;

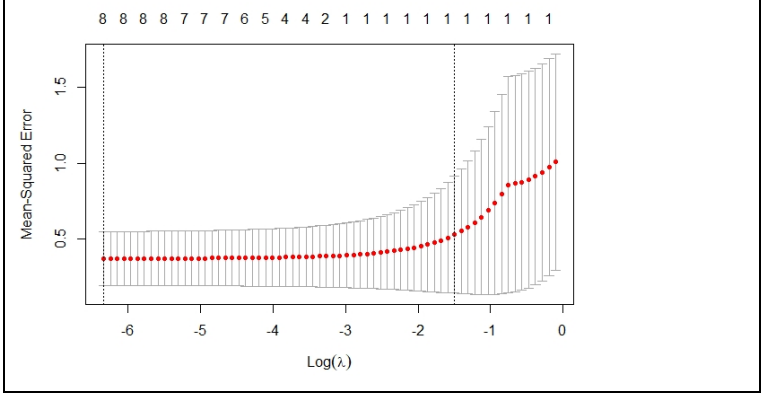
2. Utilizam-se as variáveis cujos coeficientes na estimativa resultante do modelo não sejam nulos em uma regressão por MQO.

Esse procedimento garante que (i) apenas os regressores relevantes permaneçam no modelo e que (ii) as estimativas finais sejam não viesadas. Ao adotá-lo, contudo, é preciso ter em conta que o Lasso seleciona as variáveis relevantes na predição da variável endógena, podendo haver entre as variáveis excluídas da regressão algumas que são correlacionadas com as variáveis mantidas. Nesses casos, na regressão final, em decorrência da aplicação do algoritmo, haveria um problema de variável omitida (WÜTHRICH; ZHU, 2020; ATHEY, IMBENS, 2018). Esse problema pode ser percebido à medida que se varia o λ. Se os coeficientes das variáveis mantidas variarem consideravelmente, é sinal de que há omissão de variáveis (ATHEY; IMBENS, 2018). Quanto a λ, seu valor é usualmente determinado por meio de validação cruzada. Escolhe-se o coeficiente que resulta no menor erro quadrático ou qualquer outra função custo.

6.2 Um exemplo de estimação de um Lasso

Para ilustrar o processo descrito acima, foi estimado um Lasso do custo assistencial normalizado das operadoras de planos de saúde, cujos regressores inicialmente estabelecidos foram: verticalização; total de beneficiários, população da cidade na qual está o maior percentual de beneficiários; HHI dos hospitais nessa cidade; percentual de idosos beneficiários da operadora; percentual de mulheres entre os beneficiários da operadora; percentual de beneficiários de planos individuais da operadora e percentual de beneficiários que moram na cidade com o maior número de beneficiários da operadora. Todas as variáveis explicativas foram normalizadas (a base de dados é descrita no Anexo 1). Essa estimativa foi feita usando validação cruzada, com n= 10, por meio do pacote do R glmnet. O código está no Anexo 3. O Gráfico 5, abaixo, mostra a variação do erro quadrático médio conforme a variação de λ.

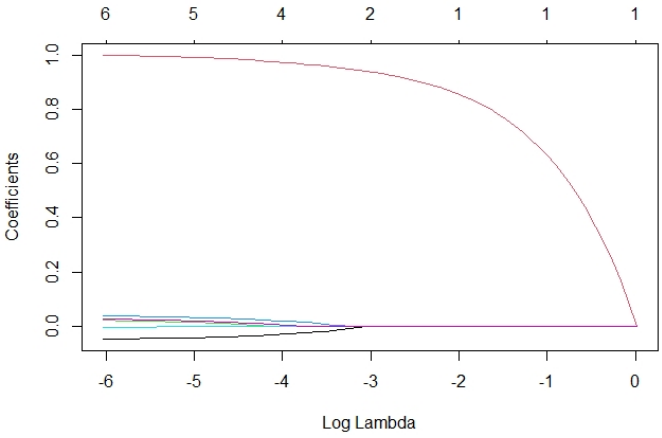
Gráfico 5 - Erro quadrático médio conforme λ



No Gráfico 5, os números na parte superior indicam a quantidade de parâmetros ($\widehat{\mu}_l$) mantidos na regressão conforme λ aumenta. As duas linhas verticais indicam o λ que minimiza o erro quadrático médio (mais à esquerda) e o que resulta no modelo mais regularizado, ou seja, cujo erro está dentro do intervalo de um erro padrão do mínimo. Na regressão estimada o λ que gerou o menor EQM é igual a 0,00177. Como se pode observar no Gráfico 5, são mantidas as oito variáveis explicativas nesse modelo, quais sejam: nível de verticalização; total de beneficiários; população da cidade na qual a operadora tem mais beneficiários; HHI da rede hospitalar na principal cidade de atuação; percentual de beneficiários idosos; percentual de beneficiários do sexo feminino; percentual de beneficiários de planos individuais e percentual de beneficiários que vivem na cidade onde a operadora tem mais beneficiários.

Considerando-se os diferentes λ, é interessante saber como os coeficientes da regressão variam com diferentes valores do parâmetro. Um gráfico muito utilizado para o primeiro objetivo é chamado de caminho do Lasso. Na ordenada, constam os valores dos coeficientes e, na abcissa, ou o valor da norma p=1 ou os próprios λs. No Gráfico 6, abaixo, foi plotado o caminho do Lasso para a mesma regressão anteriormente estimada. Dessa vez, contudo, os dados foram separados em um conjunto de treinamento, com 80% das observações e um conjunto teste, com 20% das observações. Quanto mais λ se aproxima de zero, menor a importância do termo regularizador, portanto, mais a regressão se aproxima de mínimos quadrados ordinários. Todos os coeficientes são estimados. À medida que λ cresce, a quantidade de coeficientes vai diminuindo. O coeficiente que sobressai no gráfico por ser maior do que os outros é o total de beneficiários.

Gráfico 6 - Caminho do Lasso



Para aplicar o procedimento descrito acima, no qual o Lasso é estimado inicialmente para selecionar as variáveis e, depois, a regressão final é estimada por meio de MQO, à estimação do custo assistencial das operadoras de planos de saúde, serão considerados os resultados da regressão que minimizou o erro quadrático médio na estimação com validação cruzada. Na Tabela 5, abaixo, são apresentadas as estimativas de MQO, tendo sido incluídas como variáveis explicativas o total de beneficiários, a população da cidade na qual a operadora tem mais beneficiários, o percentual de beneficiários que vivem na cidade onde a operadora tem mais beneficiários e o percentual de beneficiários idosos. Note-se que dois dos coeficientes selecionados são não significativos. O processo de seleção de variáveis, todavia, já foi feito anteriormente.

Tabela 5 - Resultados da regressão (Pós-Lasso)

	Dependent variable:
	Custo total
Total de Beneficiários	4.457,726***
	(93.875)
População	14,246*
	(7.851)
Verticalização	-260.947.458,000*
	(157.042.921,000)
HHI	13.408,130
	(8.794,753)
Percentual de Idosos	328.962.234,000*
	(184.256.401,000)
Percentual de mulheres	167.893.323,000
	(549.188.130,000)
Percentual em planos individuais	-90.400.153,000
	(99.518.197,000)
Percentual na Cidade	146.205.630,000
	(108.804.959,000)
Constant	-323.529.438,000
	(283.639.863,000)
Observations	653
R2	0.803
Adjusted R2	0.801
Residual Std. Error	575.763.766,000 (df = 644)
F Statistic	328,268*** (df = 8; 644)
Note:	*p<0.1; **p<0.05; ***p<0.01

6.3 Aplicação do Lasso em estudos econômicos

Quando o modelo econométrico é constituído por um componente de inferência causal e um componente preditivo, o primeiro componente deve ser incluído como preconiza o modelo teórico. Para análise e estimação do segundo, podem-se usar técnicas de aprendizado de máquina, como o Lasso. Isso tem sido feito em estudos com o objetivo de se avaliar os efeitos de uma política ou de outros tratamentos.

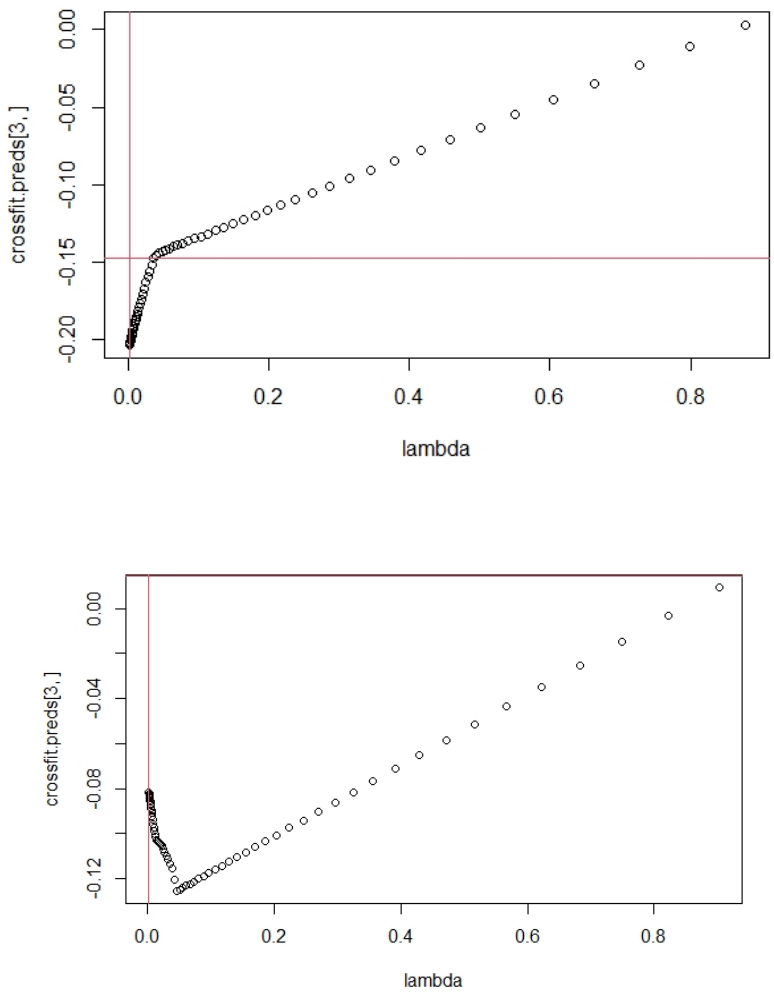
Nesses estudos, o objetivo é tão somente a avaliação do efeito de um tratamento em uma unidade qualquer (indivíduo, empresa, mercado etc.). Contudo, para evitar que os resultados captem mudanças na unidade associadas a outros fatores que não o tratamento, costumam-se incluir variáveis de controle, que têm o fim de garantir que o efeito medido está associado exclusivamente ao tratamento. Em geral, não há preocupação com as razões ou o modo como essas variáveis associam-se à unidade tratada. O importante é a sua relevância para a previsão da unidade tratada não havendo tratamento. Quanto melhor preditoras da unidade tratada elas forem, mais correta tende a ser a avaliação dos efeitos, tudo o mais constante. Considerando isso, pode-se usar o procedimento mencionado anteriormente, adaptado às especificidades da análise dos efeitos de tratamentos ou da aplicação em modelos com dois componentes (um de inferência causal e outro de predição), da seguinte forma:

1. Define-se qual efeito se pretende medir e como esse efeito se relaciona com os resultados;
2. Para a determinação das variáveis de controle a serem incluídas no modelo, utilizam-se técnicas de aprendizado de máquina, como o Lasso;
3. Se for utilizado o Lasso, Belloni et al. (2014) recomendam que sejam estimadas duas regressões regularizadas. A primeira teria como variável endógena o resultado, e a segunda o tratamento. Devem ser incluídas na regressão final, todas as variáveis selecionadas nas duas regressões;
4. Estimam-se os efeitos médios do tratamento. O procedimento do item anterior, evita que haja variáveis omitidas correlacionadas com o tratamento.

A segunda aplicação do Lasso na Economia é com o fim de fazer previsões. Como afirmado anteriormente, na maior parte das vezes, economistas buscam inferir causalidade, sendo essa uma diferença importante em relação ao aprendizado de máquina. Por vezes, contudo, é necessário prever fenômenos e os ferramentais do aprendizado de máquina podem ser bastante úteis. Varian (2014) considera que modelos preditivos podem mesmo ser usados para verificar os efeitos de um programa ou política. Se houver um modelo que preveja bem o comportamento da unidade na qual foi aplicada o tratamento, pode-se usá-lo para prever o comportamento da variável no momento após a aplicação do tratamento. Essa previsão seria feita apenas com base nos dados passados, desconsiderando o tratamento. Assim, a diferença entre o valor observado da variável e o valor predito pelo modelo seria efeito do tratamento.

No uso do Lasso para a realização de previsões, é importante notar que os resultados das estimações da variável endógena podem variar muito conforme o λ. O Gráfico 7 mostra essa questão. Nele são apresentadas as previsões para um ponto de y (a terceira observação do vetor) das regressões com todos os λ testados no processo de validação cruzada. A linha vermelha horizontal indica o valor observado na base de dados desse ponto, e a linha vermelha vertical indica o λ que minimizou o erro quadrático médio.

Gráfico 7 - Predição de Y conforme λ



Ainda em relação ao Lasso, importante ressaltar que o Cade e, especialmente, o DEE o tem utilizado bastante, seja no desenvolvimento de filtros para avaliação inicial de mercados e condutas competitivas, seja para outras estimações. Nesse sentido, a utilização desse modelo para seleção de variáveis de controle em regressões pode evitar discussões sobre a escolha dessas variáveis com o fim de se obter resultados específicos.

6.4 Aplicação de outras regressões regularizadas em estudos econômicos

A terceira aplicação que se delineou para utilização de regressões regularizadas na Economia é a estimação de modelos que não satisfazem as hipóteses de MQO, portanto, se usado este estimador, os resultados seriam viesados ou, em alguns casos, não poderiam ser estimados. Nesse caso, regressões regularizadas podem ser um meio de viabilizar a estimação ou de gerar resultados mais adequados, conforme determinado critério.

As regressões *ridges*, como mencionado anteriormente, são regressões regularizadas nas quais o termo de regularização é a norma p=2, ou seja, $\lambda\left(\sum_{i=1}^{l}|\hat{\mu}_i|^2\right)^{1/2}$. Graficamente, o contorno desse termo é uma elipse. Consequentemente, as soluções das regressões *ridge* são bem-comportadas. Não há soluções de canto, e há apenas uma solução.

Existem algumas situações nas quais essas regressões têm soluções, enquanto MQO não tem. Uma das mais citadas é quando há presença de multicolinearidade perfeita ou quase perfeita. Ainda que essa situação seja rara, pode ocorrer e pode não ser adequado retirar do conjunto de regressores alguma das variáveis correlacionadas. Um exemplo é a inclusão de variáveis associadas à escolaridade e à renda em regressões de demanda de um produto ou para avaliação do efeito de um programa. Essas variáveis tendem a ser correlacionadas e, por vezes, podem ser muito correlacionadas. Mesmo assim, considerando-se o objetivo das estimações, pode ser necessário conhecer o efeito de ambas na variável exógena, pois, apesar de relacionadas matematicamente, são variáveis que se referem a características distintas.

Quando a multicolinearidade não é perfeita, mas é alta, o estimador de MQO continua a ser não viesado, mas as estimativas tendem a ter grande variância. Os parâmetros estimados a partir de uma regressão *ridge*, ainda que viesados, tendem a ter menor variância devido à restrição imposta pelo termo de regularização. Contudo, como o Lasso, esse termo diminui os estimadores em relação ao estimador de MQO. O tamanho desse encolhimento é determinado pelo λ.

Há ainda um grande potencial de aplicação dessas regressões para a estimação de demanda quando a quantidade de regressores é maior que o número de observações. Nesses casos não é possível usar o estimador de MQO, que não é passível de ser estimado quando o número de coeficientes a serem estimado é maior que o número de observações, mas é possível usar uma regressão *ridge*. Situações desse tipo tendem a se tornar mais comuns graças ao aumento da disponibilidade de dados coletados na internet, em supermercados e outros centros de compras por agentes privados ou mesmo pelo governo. Existem possibilidades, tais como se estimar a demanda por um produto vendido em supermercado, considerando-se não apenas os substitutos

próximos, mas toda a cesta adquirida pelos consumidores e seus comportamentos de compra. Há estudos desse tipo, como (DONNELY et al., 2019). Nos estudos encontrados, foram usados, contudo, outros métodos de aprendizado de máquinas. Faltam, ainda, trabalhos que indiquem quais os pressupostos necessários ou como regressões regularizadas podem ser usadas nesses casos para inferir causalidade.

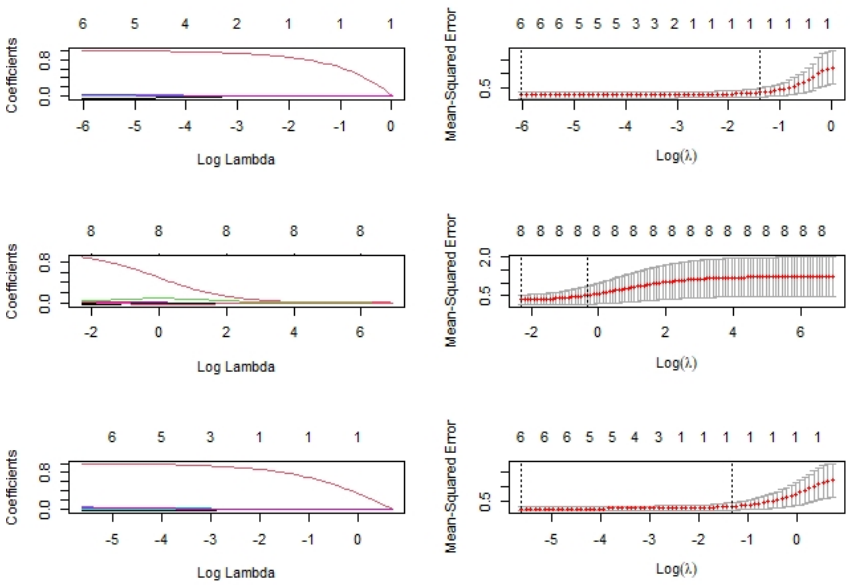
Por fim, a última regressão citada na Tabela 4 é a *elastic net*, cujo termo de regularização é uma combinação linear do Lasso e da regressão *ridge*. Assim, a *elastic net* é $\left(\sum_{i=1}^{k}\left(y_i - \widehat{\boldsymbol{\mu}}(\boldsymbol{X}_i)\right)^2\right) + \lambda[\alpha(\sum_{i=1}^{l}\widehat{|\mu_i|} + (1-\alpha)\ \left(\sum_{i=1}^{l}|\hat{\mu}_i|^2\right)^{1/2}]$, sendo $0 \leq \alpha \leq 1$.

Quando $\alpha = 1$, a regressão é um lasso, quando $\alpha = 0$, uma regressão *ridge*. Qualquer valor alternativo gera uma combinação dos dois modelos que, por um lado, seleciona variáveis e, por outro, permite a estimação mesmo quando o número de regressores é maior que o número de observações (quando se estima o Lasso sendo o número de regressores maior do que o de observações, este seleciona no máximo o mesmo número de regressores que o de observações).

Além disso, quando há presença de multicolinearidade, o Lasso, isoladamente, tende a selecionar apenas uma das variáveis. Não necessariamente, será selecionada a variável que, por razões diversas, *a priori,* um pesquisador consideraria a mais relevante. É possível que a exclusão em si das demais variáveis correlacionadas não seja adequada no problema em tela. Um exemplo seria a estimação da demanda por apartamentos em determinada localidade. O número de quartos e o número de banheiros em um apartamento tendem a ser altamente correlacionados, mas ambas as variáveis podem ser relevantes para a determinação dos preços e seria razoável incluir ambas como regressores.

Para ilustração dos três modelos (Lasso, *ridge* e *elastic net*), usando-se a base de dados descrita no Anexo 1, estimaram-se os custos assistenciais das operadoras, considerando-se valores para α que variaram de 0 a 1, com intervalo de 0,1 entre cada dois valores. O conjunto de treinamento e de teste foi o mesmo empregado para estimação anterior do Lasso. O código está descrito no Anexo 6. No Gráfico 8, abaixo, são apresentados os gráficos do Lasso, da regressão *ridge* e da *elastic net*, com $\alpha = 0.5$. Como se nota, principalmente no que tange ao coeficiente do total de beneficiários, a variação dos coeficientes é mais suave na regressão *ridge*. Na *elastic net*, o número de regressores que permanecem na regressão para um mesmo valor de λ é maior ou igual que no Lasso.

Gráfico 8 - Regressões regularizadas - Lasso (superior), ridge (meio) e elastic net (inferior)



A Tabela 6, abaixo mostra o erro quadrático médio conforme o α. O modelo com melhor desempenho para predição no conjunto teste, se forem consideradas as estimativas com os λ que minimizaram o EQM no conjunto de treinamento, é uma regressão ridge, ou seja o $\alpha = 0$.

Tabela 6 - Comparação dos modelos

Alfa	EQM - Lambda Mínimo	EQM - Lambda menor DP
0	0.238	0.082
0.100	0.312	0.131
0.200	0.289	0.118
0.300	0.313	0.138
0.400	0.313	0.154
0.500	0.314	0.154
0.600	0.314	0.161
0.700	0.314	0.143
0.800	0.314	0.144
0.900	0.315	0.154
1	0.315	0.141

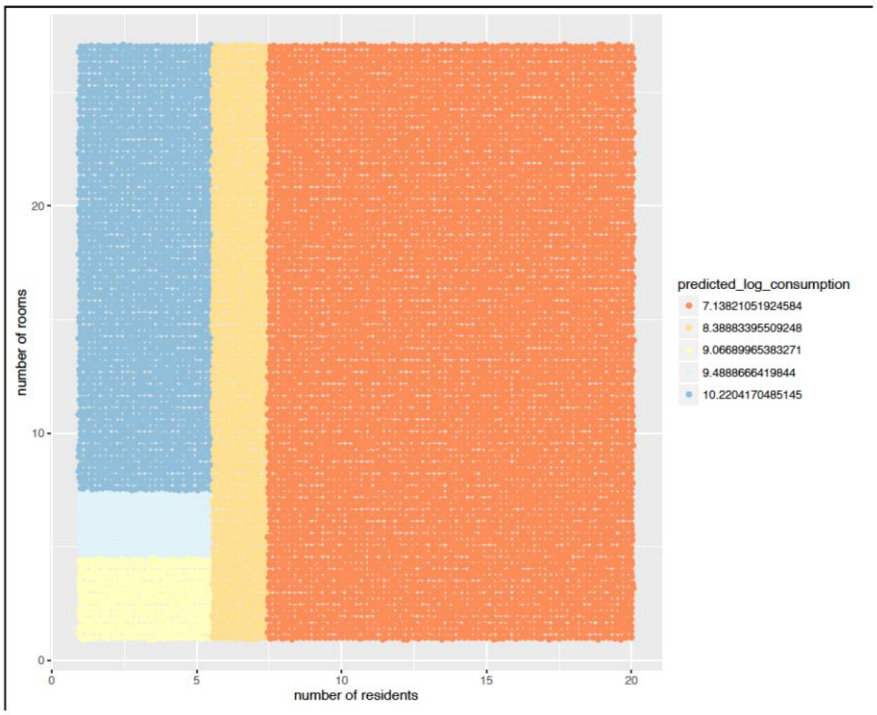
7. ÁRVORES DE REGRESSÃO E CLASSIFICAÇÃO (ÁRVORES DE DECISÃO)

Árvores de decisão são modelos de aprendizagem supervisionada usados para a previsão ou classificação. A partir de regras de decisão, classifica-se uma variável ou prevê-se o seu valor. Sua principal vantagem é a transparência: é fácil ver os critérios adotados, discuti-los e adequá-los. Outros modelos, contudo, tendem a apresentar melhores resultados, especialmente em problemas de previsão.

Quando as respostas (***y,*** mantida a notação anteriormente usada) são contínuas, a árvore é de regressão e, quando as respostas são discretas, nos quais o ***y*** assume valores associados a diferentes classes, a árvore é de classificação. Nos dois casos, as árvores são um meio de particionar o espaço das características ***X*** em regiões retangulares, sem sobreposições e paralelas aos eixos de entrada dos dados. Em cada uma dessas regiões, um modelo para a variável de resposta é estimado. Nas árvores de regressão, é comum a estimação de médias e, nas árvores de classificação, a proporção de observações em cada classe.

Na Figura abaixo por exemplo, são apresentados os resultados de uma árvore de regressão na qual a variável de resposta é o consumo (em logaritmo), e as variáveis usadas na construção da árvore são o número de residentes no domicílio (representado na abcissa) e o número de cômodos (representado na ordenada). O resultado apresentado é o particionamento do espaço de variáveis em cinco regiões.

Figura 3 - Representação do resultado de uma árvore de regressão no espaço de variáveis



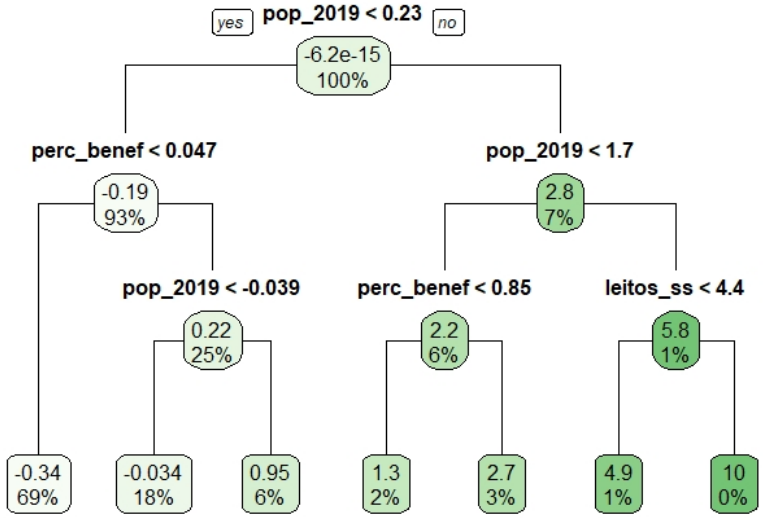
Fonte: ATHEY; IMBENS, 2018

As regiões são usualmente definidas com o fim de que seus elementos sejam os mais similares entre si possível ou que haja menos incerteza sobre o que há em cada área. Nas árvores de regressão essas áreas podem ser definidas a partir da minimização do somatório do erro quadrático. O algoritmo tem de encontrar tanto as variáveis que, quando utilizadas, geram o menor somatório do erro quadrático quanto os pontos de corte mais adequados. Para isso, a cada passo, são testadas as variáveis explicativas (***X***) e, uma vez definida a variável que mais influencia a resposta naquela etapa, o ponto de corte.

Além da figura acima, um outro modo de representar uma árvore de regressão são gráficos como o 9, abaixo. Figuras desse tipo permitem visualizar o processo de partição do espaço. Nele, é apresentado o resultado da estimação de uma árvore de regressão da quantidade de operadoras de saúde que tiveram a adesão de pelo menos um beneficiário recentemente, em cada município brasileiro. Para essa estimação, utilizou-se uma base de dados dos municípios brasileiros, usada anteriormente pelo DEE/Cade, na qual constavam dados referentes à estimativa da população residente em 2019, estimativa de percentual de idosos na população baseado no censo de 2010 realizado pelo Instituto Brasileiro de Geografia e Estatística (IBGE), produto interno bruto per capita municipal, referente a 2017, calculado pelo IBGE, em parceria com órgãos estaduais de estatística, secretarias estaduais de governo e a Superintendência da Zona Franca de Manaus, e o índice de

desenvolvimento humano do município em 2010, calculado a partir de informações do censo demográfico pelo PNUD Brasil, Instituto de Pesquisa Econômica Aplicada – IPEA e Fundação João Pinheiro. Além desses dados, compunham essa base a nota do município associado ao indicador de cobertura populacional de equipes básicas de saúde do Sistema Único de Saúde (SUS) em 2011, calculado pelo Ministério da Saúde e a quantidade de leitos em hospitais de natureza pública e leitos em hospitais de natureza privada, conforme o Cadastro Nacional de Estabelecimentos de Saúde (CNES) (fevereiro de 2020). Por fim, a base continha dois dados referentes à saúde suplementar, ambos provenientes da Agência Nacional de Saúde Suplementar (ANS): beneficiários de planos de saúde em junho de 2019; quantidade de operadoras que tiveram adesão de beneficiários em dezembro de 2019.

Gráfico 9 - Árvore de regressão da quantidade de operadoras que tiveram a adesão de pelo menos um beneficiário, em cada município brasileiro (normalizado) (dezembro de 2019)



Na árvore de decisão representada no Gráfico 9, acima de cada retângulo, consta o critério utilizado para partição e, nos retângulos, a média da quantidade de operadoras com beneficiários recentemente vinculados nos municípios que fazem parte daquele grupo e a proporção de município que fazem parte do grupo. Nota-se que duas variáveis são chave para as partições: a população do município e a cobertura da saúde suplementar no município (percentual de beneficiários na população). Além dessas duas variáveis, a quantidade de leitos em hospitais privados também foi usada em uma das partições (leitos_ss).

A aplicação do critério de partição no conjunto de treinamento para determinação da extensão da árvore, pode resultar em *overfitting*. O algoritmo gerará previsões personalizadas ou

quase personalizadas para as observações do grupo de treinamento, mas sua capacidade de previsão pode não ser alta fora desse conjunto. Por essa razão, duas recomendações comuns na estimação de árvores de regressão são para que se faça avaliação cruzada ou, se houver dados suficientes, para que as observações sejam divididas em conjuntos de treinamento e de teste e que sejam estabelecidos critérios com o fim de limitar o crescimento da árvore e podá-la. Assim, o algoritmo para a estimação de uma árvore de regressão pode ser descrito como:

1. A partir do conjunto de todos os dados, escolha uma característica (variável j) que determinará a divisão do conjunto em dois grupos. É preciso também, estabelecer um ponto de corte s. Por exemplo, se o conjunto for um grupo de pessoas, pode-se escolher a idade como variável que define a qual grupo uma pessoa pertence. Tem-se que definir também o ponto s, ou seja qual a idade que distingue os dois grupos. Feito isso, pode-se definir dois grupos ou planos:

$$R_1(j,s) = \{x | x_j \leq s\}$$

$$R_2(j,s) = \{x | x_j > s\}$$

   Uma pessoa (x) está no grupo R_1 se sua idade (x_j) é menor que s.

2. Da escolha da característica j e do ponto de corte s dependerá o resultado do algoritmo. Considerando-se o objetivo de haver a menor incerteza possível sobre a composição de cada grupo, um critério possível é escolher j e s que minimizem o erro quadrático médio em cada grupo e do modelo em si, de forma que: $min_{j,s}[min_{c1} \sum_{x_{i \in R_1(j,s)}} (y_i - c_1)^2 + min_{c2} \sum_{x_{i \in R_2(j,s)}} (y_i - c_2)^2]$, sendo y_i a variável observada, $\hat{c}_1 = média(y_i | x_i \in R_1(j,s))$ e $\hat{c}_2 = média(y_i | x_i \in R_2(j,s))$. Note-se que quanto menor a diferença entre cada observação é a média, menor a dispersão, portanto, menor a incerteza;

3. Determine o menor tamanho que um nó pode ter (cada ponto de decisão da árvore) conforme a quantidade de observações ou outra medida e somente pare o crescimento da árvore quando esse tamanho for atingido;

4. Estabeleça uma medida de custo-complexidade e pode a árvore com base nessa medida.

Uma medida de custo complexidade pode ser obtida considerando-se:

T é qualquer árvore resultante da poda de T_o;

m é um índice para os nós terminais (também chamados de folhas);

$|T|$ é o número de nós terminais de T;

N_m é o número de de observações em R_m;

$\hat{c}_m = \frac{1}{N_m}\sum_{x_i \in R_m} y_i$;

$Q_{m(T)} = \frac{1}{N_m}\sum_{x_{i \in R_m}} (y_i - \hat{c}_m)^2$.

A partir dessas definições, o critério pode ser estabelecido como $C_\alpha(T) = \sum_{m=1}^{|T|} N_m Q_{m(T)} + \alpha|T|$, no qual α é um parâmetro de ajuste, tal como o λ das regressões regularizadas. Na sua definição é preciso considerar que uma árvore muito grande pode resultar em *overfitting*, mas uma árvore muito pequena pode não capturar estruturas importantes.

As árvores de classificação têm a mesma estrutura das árvores de regressão. O tipo de resposta, contudo, não permite que seja utilizada a mesma função custo usada nas árvores de regressão. São definidas funções que têm por objetivo tornar os nós finais das árvores (chamados na literatura de folhas), os mais diferentes entre si e as observações que compõem cada folha os mais semelhantes entre si. As principais funções usadas são o erro de classificação, o índice de Gini e a entropia. O primeiro é a proporção de observações classificadas incorretamente. O segundo é calculado com base na multiplicação entre a proporção de observações classificadas corretamente e a proporção de observações classificadas incorretamente. Por fim, a entropia é calculada com base na multiplicação das observações classificadas corretamente com o logaritmo dessas observações.

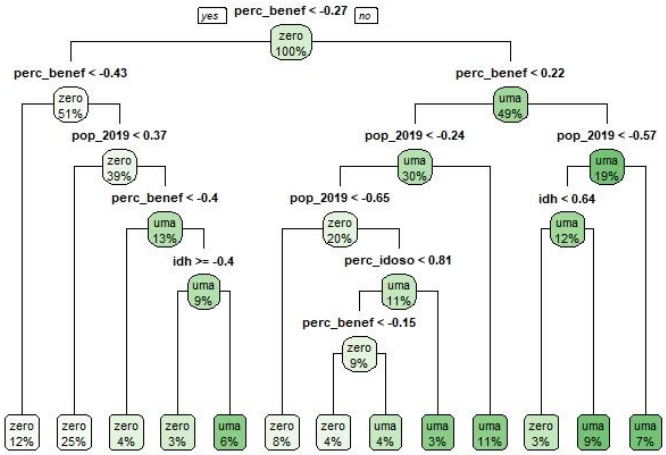
Esses critérios, em especial o índice de Gini e a entropia, costumam gerar árvores semelhantes, mas isso pode não acontecer. O índice de Gini tende a ser calculado mais rapidamente que a entropia e pode ser visto como uma medida de erro. A entropia baseia-se na teoria da informação[13] e pode ser entendida como a informação esperada em um conjunto de resultados possíveis. As três medidas possibilitam mensurar o ganho de informação decorrente de uma partição comparando-se o resultado antes e depois da partição.

No Gráfico 10, é apresentada uma árvore de classificação cuja resposta é se há no município alguma operadora com ao menos um novo vínculo de beneficiário ou não. Foi usada a mesma base de dados utilizada na estimação da árvore constante no Gráfico 9, mas limitaram—se os municípios a aqueles em que não havia operadoras com novos beneficiários e aqueles em que só havia uma

[13] Teoria da informação pode ser definida como o estudo da quantificação, armazenamento e comunicação de informação digital. Para um resumo do desenvolvimento desse campo, ver: https://web.mit.edu/6.933/www/Fall2001/Shannon2.pdf. Acesso em 07/07/2022.

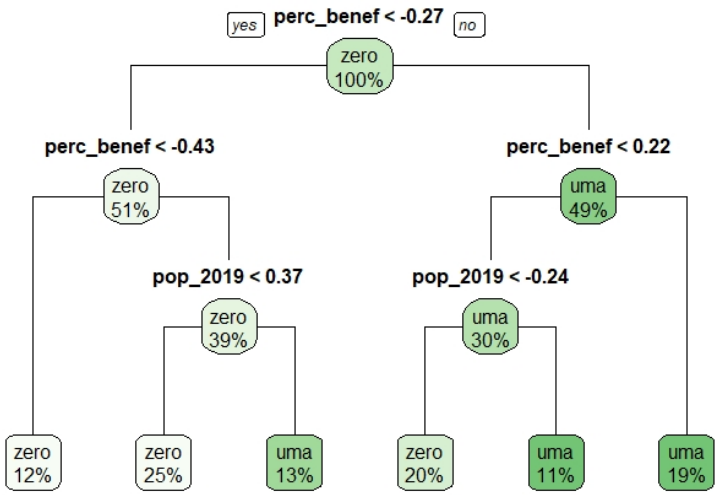
operadora com novos beneficiários. Acima dos retângulos, são indicados os parâmetros usados nas partições e, no retângulo, a classe predominante naquele grupo e o percentual de observações que estão naquele grupo em relação ao total de observações. No algoritmo, estabeleceu-se como limite que nenhum nó tivesse menos de 50 observações (Anexo 6) e como função custo o índice de Gini. Como se pode observar, a porcentagem de beneficiários em relação à população do município, o tamanho da população são as variáveis mais importantes na definição das partições. O IDH e a porcentagem de idosos também são variáveis utilizadas na árvore.

Gráfico 10 - Árvore de classificação - Municípios - Adesões de beneficiários



Quando se utiliza validação cruzada, o parâmetro de custo complexidade ótimo, considerando-se o valor mínimo do xerror, é 0.012. Usando-se esse valor a árvore final tem seis folhas, como mostra o Gráfico 11, abaixo, e apenas a população do município e o percentual de beneficiários são usados nas partições da árvore.

Gráfico 11 - Árvore de classificação podada - Adesões de beneficiários



Nas Tabelas 7 e 8, abaixo, são apresentadas as matrizes de confusão para a árvore completa (Gráfico 10) e a árvore podada (Gráfico 11) considerando-se as previsões para o conjunto de teste. Matriz de confusão é como se chama, no aprendizado de máquina, a matriz na qual as colunas representam os valores preditos e as linhas representam os valores realizados (ou vice-versa). Observa-se que a árvore podada tem melhor desempenho.

É interessante notar que a avaliação do modelo se relaciona ao objetivo pelo qual ele foi estimado. Pode-se dar maior ou menor peso a um tipo de erro específico na avaliação dependendo de sua importância. Por exemplo, na avaliação de um exame de diagnóstico inicial, pode ser mais importante que o exame preveja corretamente quais são os pacientes sadios. Se houver entre os considerados doentes, pacientes sadios, no momento seguinte, estes serão identificados e não serão tratados. É menos provável, contudo, que os pacientes considerados sadios realizem novos exames.

Tabela 7 - Matriz de confusão - Árvore Completa

	Previsto	
	Zero	Uma
Zero	200	62
Uma	99	129

Tabela 8 - Matriz de confusão - Árvore podada

	Previsto	
	Zero	Uma
Zero	204	58
Uma	92	136

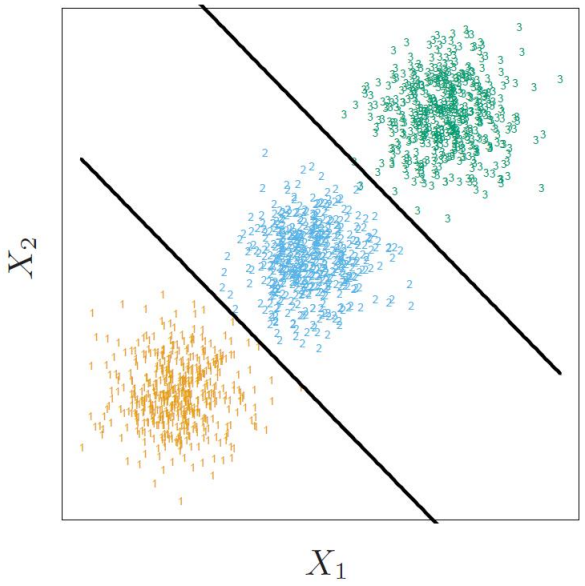
Além de serem facilmente interpretadas, o que favorece sua utilização, tanto árvores de regressão quanto árvores de classificação têm sido usadas na Economia para a avaliação de efeitos heterogêneos de políticas e programas. Elas são usadas para identificar e agrupar observações semelhantes entre si. Em uma árvore ótima de classificação, por exemplo, cada folha seria composta por observações bastante homogêneas entre si, e diferentes folhas seriam significativamente diferentes. A divisão das observações permitiria avaliar os efeitos nos diferentes grupos. Uma outra possibilidade é incluir no espaço das variáveis explicativas o programa ou política. Se essa variável não for selecionada para particionar o espaço, é indicativo de que o programa não teve efeito. Há trabalhos que conjugam esses modelos com a avaliação de efeitos e, diferentemente do uso do Lasso, tanto são usados como um meio para se responder novas perguntas quanto para aprimorar procedimentos já utilizados.

8. CLASSIFICAÇÃO

Modelos de classificação são modelos de aprendizagem supervisionada que têm variáveis de resposta discretas. Chamando-se cada resposta possível diferente de classe, o objetivo desses modelos é classificar, a partir das variáveis endógenas, qual a classe da resposta de uma observação. Como nas demais aplicações de aprendizado de máquina, buscam-se desenvolver modelos que consigam prever adequadamente as classes de observações fora da amostra usada nas estimações.

Geometricamente, considera-se que o modelo será tanto melhor quanto melhor particionar o espaço formado pelas variáveis exógenas de forma que cada classe fique em uma região diferente. Na Figura 4, por exemplo, observam-se três classes e duas variáveis explicativas (X_1 e X_2). As três classes são perfeitamente separáveis no espaço formado pelas duas variáveis explicativas. Nesse exemplo, verificar a separação das classes é fácil. Em dimensões maiores, tanto é mais difícil verificar visualmente se um modelo de classificação alcança seu objetivo quanto é mais difícil generalizá-lo adequadamente (DOMINGOS, 2012). Assim, modelos que funcionam bem em dimensões pequenas podem não funcionar tão bem em altas dimensões.

Figura 4 - Exemplo de um espaço no qual as observações pertencem a três classes diferentes



vFonte: HASTIE; TIBSHIRANI e FRIEDMAN,2008, p. 105

8.1 Regressões com resposta discreta

Modelos de classificação são conhecidos dos economistas. O principal objetivo da maior parte dos trabalhos econômicos que usam esses modelos, contudo, é compreender quais variáveis influenciam a resposta, sendo comum que modelos de resposta discreta sejam a expressão econométrica de um modelo teórico de escolha. Em outros casos, o objetivo da estimação é avaliar qual a probabilidade de um agente ou objeto ser classificado de uma dada forma[14]. Por exemplo, qual a probabilidade de um solicitante de empréstimo ser adimplente. Diferentemente do aprendizado de máquina, todavia, mesmo quando o objetivo é a classificação, raramente há preocupação com a validade externa dos modelos estimados ou com sua capacidade de classificar fora da amostra.

Assim, enquanto no aprendizado de máquina, um modelo é tanto melhor quanto melhor for generalizado, na economia, modelos de classificação não costumam ser avaliados pelo seu desempenho fora da amostra ainda que seu uso se destine à previsão. Apropriar-se das técnicas e métodos do aprendizado de máquina para aplicação em problemas de classificação pode melhorar a qualidade das estimativas e tornar o modelo estimado mais aderente ao seu objetivo, quando

[14] Nesse caso, ainda que o procedimento seja o mesmo, o problema não mais seria de classificação (determinar a que grupo pertence um objeto), mas de previsão.

este for fazer previsões fora da amostra. Isso porque a adoção de procedimentos que meçam o desempenho do modelo considerando sua capacidade de generalização tende a levar a escolha de modelos mais adequados a esse propósito.

Apresenta-se, a seguir, um algoritmo proposto por Carvalho, Cajueiro e Camargo (2015) para o uso de regressões com resposta binária para classificação. Esse procedimento pode ser facilmente adaptado para regressões de resposta discreta com mais de duas alternativas.

Parte-se do pressuposto de que a resposta é uma variável aleatória que tem distribuição de Bernoulli, de forma que possa assumir os valores de 1 ou 0. Por exemplo, se houver, no município, uma operadora a qual se vinculou algum beneficiário recentemente a variável assume o valor de 1. Se não houver, o valor da variável resposta é 0.

Supondo-se que a probabilidade de haver no município uma operadora a qual se vinculou ao menos um beneficiário recentemente depende de variáveis pertencentes a ***X,*** é possível descrevê-la como $P(y_i = 1|x_i) = F(x_i'\beta)$, ou seja, a probabilidade de a probabilidade de ter havido a adesão de um beneficiário a operadora depende das variáveis em ***X.*** F é uma função de distribuição acumulada. São duas as funções mais usadas na estimação de $P(y_i = 1|x_i)$. A primeira é $F(z) = \Phi(z)$, na qual Φ é a função acumulada normal, que é o modelo probit. A segunda é quando $F(z) = \Lambda(z) = \frac{\exp(z)}{1+\exp(z)}$. Esse é o modelo logit. Ambos, bem como suas variações são muito utilizados na pesquisa econômica. Para usá-los como classificadores, o algoritmo proposto por Carvalho, Cajueiro e Camargo (2015) é:

1. A partir de uma amostra A com *n* elementos, selecione uma parcela B de A com *m* elementos ($n < m$);

2. Avalie se cada elemento de *B* é um evento (é um município no qual houve a aquisição de plano de uma operadora) ou não;

3. Estime os modelos de resposta binária para uma parte dos elementos de *B* (parcela *C*);

4. Faça uma escolha da probabilidade de corte p_c. Observações cuja probabilidade seja menor que p_c serão consideradas como não sendo um evento (no caso, municípios no quais não houve a aquisição de plano de uma operadora) e observações cuja probabilidade for maior que p_c, serão consideradas um evento. Para escolher p_c, teste diferentes valores nos dados $B - C$ e verifique a taxa de acerto;

5. Utilize seu modelo de escolha discreta para classificar a amostra $A - B$, usando p_c como probabilidade de corte.

É interessante notar alguns elementos desse algoritmo. Como o objetivo é prever fora da amostra, o conjunto de dados é dividido em treinamento (parcela *C*), validação ($B - C$) e teste ($A - B$). Alternativamente, poderia ser utilizada a validação cruzada. Um segundo elemento a ser ressaltado é a avaliação da taxa de acerto. Tal como afirmado na discussão das árvores de classificação, é possível que um erro seja mais grave do que outro na avaliação de um problema concreto. Pode-se e deve-se atribuir pesos ou adaptar o critério de avaliação do modelo para o seu objetivo.

Além disso, é importante considerar que se uma resposta prevalecer em relação a outra, é mais fácil prever a resposta que prevalece. Consequentemente, um critério de acerto que atribua os mesmos pesos para os acertos de ambas as variáveis pode ser pouco informativo. Uma alternativa nesses casos é usar a porcentagem predita correta ponderada, apresentada por Woodridge (2003) e citada em Carvalho, Cajueiro e Camargo (2015), que é

$$p\% = (1 - p_1) \times q_0 + p_1 \times q_1,$$

Sendo p_1 a proporção da resposta 1 na amostra, q_0 é a porcentagem corretamente predita da resposta 0 e q_1, a porcentagem corretamente predita da resposta 1.

A fim de ilustrar o algoritmo apresentado acima, esse procedimento foi adotado para o mesmo exemplo usado para ilustrar a aplicação das árvores de classificação. Usando a mesma base de dados e os mesmos conjuntos de treinamento e teste usados na estimação da árvore de classificação na seção anterior, foram estimados um logit e um probit, cujas variáveis explicativas foram: população em 2019; percentual de idosos no município; PIB per capita municipal; IDH do município; avaliação da assistência de saúde básica pública do município; quantidade de leitos em hospitais privados; quantidade de leitos em hospitais públicos e percentual de beneficiários de planos de saúde em relação à população. As fontes de todas as variáveis são as mesmas citadas anteriormente. Tanto para o logit quanto para o probit, o ponto de corte escolhido (p_c) foi 50%.

A Tabela 9 mostra os coeficientes estimados das duas regressões. Primeiramente, vale notar que, apesar de os valores serem diferentes, em ambas, as mesmas variáveis apresentam coeficientes significativos (população em 2019, IDH, nota da assistência básica no SUS, leitos em hospitais privados, percentual de beneficiários em relação à população). Salvo pelas caudas, as funções logística e probit têm gráficos parecidos. Em comparação com as variáveis utilizadas na construção das árvores de classificação, o IDH, a nota da assistência básica no SUS, os leitos em hospitais privados não foram usados nas partições da árvore podada, e apenas o IDH, dessas variáveis, estava na árvore sem poda.

Tabela 9 - Regressões binárias

	Dependent variable:	
	Municípios onde houve adesão de beneficiários	
	logistic	probit
	(1)	(2)
pop_2019	0.00004***	0.00002***
	(0.00001)	(0.00001)
perc_idosos	2.626	1.318
	(1.608)	(0.989)
pib_capita	-0.00000	-0.00000
	(0.00000)	(0.00000)
idh	4.702***	2.639***
	(1.348)	(0.827)
nota_cob_atencao	0.064*	0.037*
	(0.033)	(0.020)
leitos_ss	0.025***	0.016***
	(0.008)	(0.005)
leitos_pub	0.0001	0.001
	(0.005)	(0.003)
perc_benef	12.663***	9.198***
	(2.269)	(1.397)
Constant	-4.889***	-2.762***
	(0.772)	(0.468)
Observations	1,763	1,763
Log Likelihood	-1,125.939	-1,141.106
Akaike Inf. Crit.	2,269.877	2,300.212

Nas Tabelas 10 e 11, abaixo, são apresentadas as matrizes de confusão para esses modelos. Comparando-se com os resultados das árvores de classificação, se o critério utilizado for o total previsto corretamente, não há diferenças significativas nos resultados. Enquanto o percentual previsto corretamente no conjunto teste, na árvore podada, é de 69,39%, no logit é de 66,94% e no probit, de 66,73%.

Tabela 10 - Matriz de confusão - Regressão binária - Logit

	Previsto	
	Zero	Uma
Zero	219	43
Uma	119	109

Tabela 11 - Matriz de confusão - Regressão binária - Probit

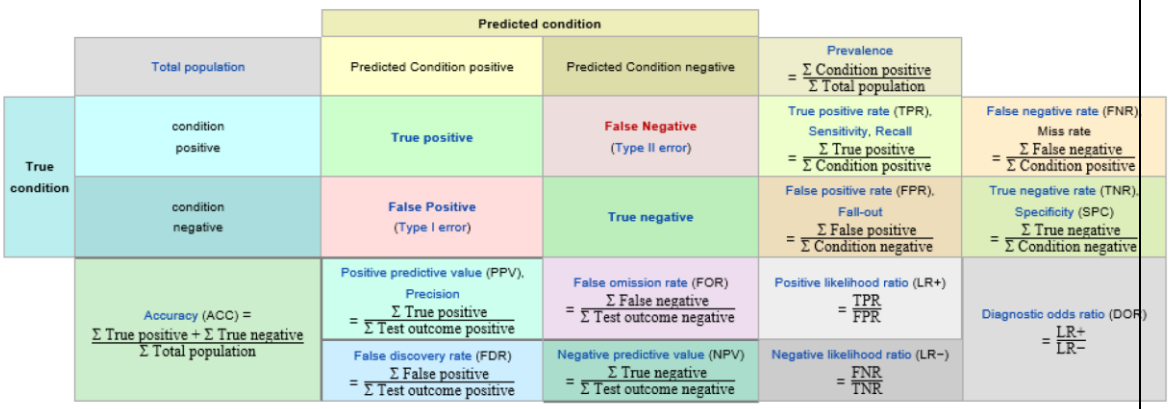
	Previsto	
	Zero	Uma
Zero	224	38
Uma	125	103

8.2 Avaliação de classificadores

A avaliação de modelos de classificação é um ponto chave no aprendizado de máquina. Essa avaliação deve ser adaptada ao objetivo pelo qual o modelo foi estimado e levar em conta que diferentes erros de classificação podem ocorrer. A Figura 5, abaixo, mostra critérios bastante utilizados, dos quais destacam-se:

- Acurácia: (total de verdadeiros positivos + verdadeiros negativos) / população total;
- Sensibilidade (*recall*): total de verdadeiros positivos/total de positivos;
- Especificidade: total de verdadeiros negativos/total de negativos;
- Precisão: total de verdadeiros positivos/ total de positivos previsto.

Figura 5 - Critérios de avaliação de classificadores



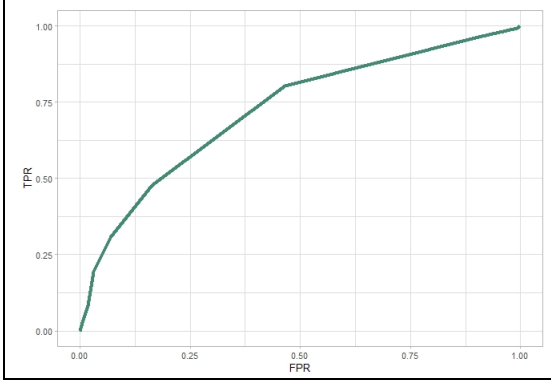
Fonte: ATHEY; IMBENS,2018

Verdadeiros positivos e verdadeiros negativos são as observações positivas ou negativas que foram classificadas corretamente.

A acurácia é um indicador bastante utilizado por economistas. Contudo, a sensibilidade pode ser mais útil se o objetivo for garantir que, se uma observação for classificada como positiva (por exemplo, municípios nos quais há adesão de novos beneficiários em uma operadora), ela de fato o é. Da mesma forma, a especificidade pode ser mais relevante se for importante garantir que, se uma observação for considerada negativa, ela de fato o é. Um modelo pode ser muito bom para classificar observações verdadeiras (ou falsas) e não ser tão bom para classificar observações falsas (ou verdadeiras). Ainda assim, pode ser melhor do que um modelo que, em geral, classifique melhor as observações (a acurácia seja maior). De forma geral, considerar a conexão da medida com o problema e utilizar mais de uma medida tendem a garantir a validação adequada do modelo. Um Gráfico muito usado para verificação dos resultados de um modelo de classificação quando se alteram o parâmetro é a curva ROC (em inglês, *receiver operating characteristic curve*). A abcissa do gráfico é a taxa de positivos verdadeiros (TPR), e a ordenada é a taxa de falsos positivos (FPR). Em uma regressão binária, por exemplo, o gráfico é usado para verificar os resultados de um modelo com a variação de p_c. À medida que p_c diminui, aumentam as observações classificadas como positivas. Consequentemente, aumentam tanto os positivos verdadeiros quanto os falsos positivos. Para o logit estimado nessa seção, considerando-se os municípios em que não houve adesão de beneficiários como negativos, por exemplo, a curva ROC é apresentada no gráfico 12, abaixo.

A área sob a curva ROC, chamada de AUC (*area under the ROC curve*), também é uma medida do desempenho do modelo considerando-se todos os pontos de corte (no caso do exemplo, p_c). Se todas as previsões do modelo forem erradas, seu valor é zero. No outro extremo, se todas as previsões forem corretas, sua área é 1. Por conseguinte, quanto maior a AUC, melhor o desempenho do modelo. Essa área tem ainda duas características relevantes. A primeira é que não varia com a escala das medidas, e a segunda é que é uma medida de desempenho que independe do ponto de corte adotado. Note-se, contudo, que se for interessante ponderar diferentemente falsos positivos de falsos negativos, a AUC pode não ser o melhor indicador do desempenho do modelo.

Gráfico 12 - Curva ROC - Modelo Logit



8.3 Outros classificadores

Como muitos dos problemas enfrentados no aprendizado de máquina são problemas de classificação, foram desenvolvidos inúmeros modelos e algoritmos para lidar com questões desse tipo. Ainda que conhecê-los permitam aos economistas ampliar seu repertório e, possivelmente, adotar técnicas mais eficientes quando confrontados com problemas desse tipo, não se tem o objetivo de fazer uma apresentação exaustiva desses modelos. Optou-se por descrever os dois principais tipos de classificadores usados no aprendizado de máquina, ilustrando-se com um exemplo de cada um dos tipos.

Semelhante às regressões com respostas discretas, há um conjunto de outros classificadores também probabilísticos, mas que se baseiam no teorema de Bayes. Como se sabe, o teorema de Bayes é derivado a partir da definição de probabilidade condicional e das propriedades da função de probabilidade e estabelece que

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}.$$

A equação acima estabelecer que a probabilidade de A dado B ($P(A|B)$) é igual a probabilidade de B dado A ($P(B|A)$) multiplicada pela probabilidade de A ($P(A)$) e dividida pela probabilidade de B ($P(B)$). A partir desse teorema, definindo-se w_j como a classe j e M como a quantidade de classes ($w_j, j = 1,2,\dots,M$), a probabilidade $P(w_j|\boldsymbol{x})$ pode ser escrita como $P(w_j|\boldsymbol{x}) = \frac{P(\boldsymbol{x}|w_j)P(w_j)}{P(\boldsymbol{x})}$. O vetor $\boldsymbol{x}$ é o vetor das variáveis explicativas e $P(\boldsymbol{x}|w_j)$ é a probabilidade de aparecer uma observação com as caractéristicas $\boldsymbol{x}$ na classe w_j. Um classificador bayseano assinala a classe w_j a uma dada observação de modo a maximizar a probabilidade $P(w_j|\boldsymbol{x})$, ou seja,

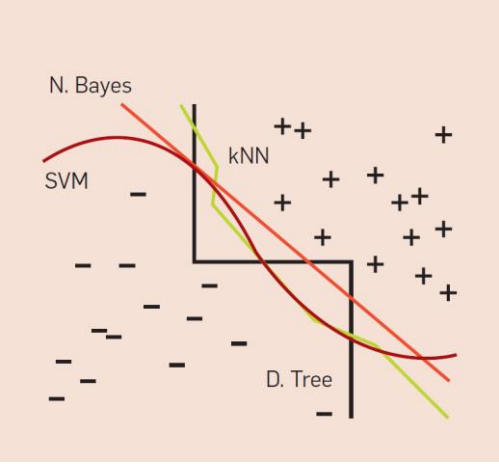
$$arg\max_{w_j} P(x|w_j)P(w_j).$$

O denominador $P(\boldsymbol{x})$ não entra na maximização porque é uma quantidade positiva que independe de w_j. Havendo no conjunto de treinamento N observações, sendo N_j o número de observações na classe *j*, a probabilidade *a priori* da classe *j* pode ser aproximada por $P(w_j) \approx \frac{N_j}{N}$. Pode-se escolher o método mais conveniente para estimar a probabilidade condicional $P(x|w_j)$ e assim estimar a probabilidade de uma classe dado $\boldsymbol{x}$.

Um classificador que não é ótimo, mas que, por vezes, tem resultados melhores que o classificador bayseano ótimo, obtido a partir da maximização de $P(w_j|\boldsymbol{x})$, é o modelo bayesiano naïve. Assume-se que os componentes de $\boldsymbol{x}$ são independentes. Consequentemente a distribuição conjunta de $\boldsymbol{x}$ pode ser descrita como um conjunto de *M* marginais $p(x|w_i) = \prod_{k=1}^{l} p(x_l|w_j)$. Adotada a hipótese de uma normal, cada distribuição marginal pode ser descrita pela sua média e variância. Esses dois pressupostos facilitam consideravelmente a estimação.

O segundo conjunto de classificadores que será apresentado distancia-se dos métodos econométricos mais conhecidos. São métodos não probabilísticos que buscam particionar o espaço das variáveis de forma a obter a divisão que melhor separe as classes. Na Figura 4, acima, por exemplo, duas retas separam perfeitamente as classes Indicadores muito diferentes podem gerar os mesmos resultados, como mostra a Figura 6, na qual são apresentados os resultados de um classificador de Bayes naïve (N.Bayes), de uma árvore de classificação (D. Tree), de um *K-nearest neighbor* (kNN) e de uma *support vector machine* (SVM). Os dois últimos métodos são classificadores não probabilísticos que serão apresentados a seguir.

Figura 6 - Diferentes classificadores para uma mesma previsão



Fonte: DOMINGOS, 2012

O K-nearest neighbor é um dos algoritmos de classificação mais simples que existem. K é o número de vizinhos mais próximos que o algoritmo considera e deve ser definido pelo usuário, não devendo ser múltiplo do número de classes. O usuário deve definir também uma métrica para a mensuração da distância, por exemplo a distância euclidiana. Com base nessas duas definições, as observações são classificadas a partir dos seus k vizinhos mais próximos já classificados. A classe mais comum entre esses vizinhos será atribuída também à observação.

SVM é um conjunto de métodos para classificação que definem um ou mais hiperplanos em espaços, em geral, com muitas dimensões, de forma a separar as classes das observações[15]. As linhas que separam os hiperplanos são traçadas de forma a maximizar a distância entre as próprias linhas e os pontos mais próximos de cada uma das classes. Esse método é utilizado porque pode ser generalizado para casos não lineares. No "R", o pacote e1071 permite que se estime o classificador usando quatro kernels (funções): linear, polinomial, radial e sigmóide. Em linhas gerais, contudo, esse método consome muito processamento no treinamento sem que gere, em muitos casos, resultados superiores a outros algoritmos, como as florestas aleatórias. Estas são uma combinação de várias árvores de decisão que produzem um resultado.

Como afirmado na introdução a esta seção, problemas de classificação são uma parte relevante do aprendizado de máquina, razão pela qual diversos métodos foram e continuam a ser desenvolvidos. Embora os economistas tenham dado menos atenção a esse tipo de problema, como afirmam Athey e Imbens (2019), eles podem ser usados para responder perguntas que ainda não são usuais na Economia, mas que podem ajudar na compreensão de questões relevantes. Além disso, os procedimentos adotados a fim de possibilitar a generalização dos classificadores podem e devem ser adotados por economistas quando resolvendo problemas de classificação.

No exemplo usado ao longo da seção, o método com melhor desempenho foi a árvore de classificação. Conhecer diferentes métodos possibilita que se use o mais adequado ao problema. Embora não tenha sido tratado neste trabalho, além dos diversos métodos existentes, é possível combinar classificadores a fim de se obter melhor desempenho, caso das florestas aleatórias citadas nesta seção. Existem diferentes formas de realizar essas combinações e, para economistas que enfrentam problemas de classificação, saber como combiná-los também pode ser muito útil.

Deve-se notar que todos os exercícios de estimação feitos nesse trabalho têm o objetivo apenas de ilustrar a aplicação dos métodos apresentados. O processo de escolha das variáveis explicativas e mesmo do classificador pode e deve ser feita de forma mais rigorosa quando da sua

[15] SVM podem ser usadas também em problemas de regressão e para detecção de outliers.

aplicação em um problema. Estudar as relações entre as variáveis, a distribuição dos dados é importante tanto para escolher as variáveis exógenas quanto o próprio classificador.

9. APRENDIZAGEM NÃO SUPERVISIONADA

Métodos de aprendizagem não supervisionada distanciam-se dos métodos comumente usados na Economia, pois caracterizam-se pela inexistência de variáveis resposta. Nos termos da literatura de aprendizagem de máquina, são dados não rotulados. O objetivo desses métodos é encontrar automaticamente padrões de similaridade entre os objetos estudados. Quando um conjunto (X designa este conjunto) é composto por objetos que têm poucas características (variáveis) que os distinguem ou, em outras palavras, tem dimensão baixa, existe uma variedade de métodos não paramétricos para estimar a sua função densidade de probabilidade e para representar o conjunto de observações graficamente. Esses métodos, contudo, costumam falhar quando X tem altas dimensões. A solução é a estimação de métodos que não estimam a função densidade de probabilidade em si, mas estatísticas descritivas ou outros elementos que possam caracterizar essas funções.

De um lado, métodos como dos componentes principais tentam identificar se há características dos dados em baixas dimensões que possam ser usadas para representar esses dados. Dessa forma, as dimensões de X poderiam ser reduzidas, o que facilitaria a interpretação e apresentação dos resultados. Note-se que o objetivo é similar ao da estimação do LASSO de selecionar variáveis. A diferença é que, neste caso, não há ou não se utiliza a variável resposta. A análise de componentes principais tem sido, assim, usada também como um passo anterior à estimação de regressões.

Do outro lado, a análise de *clusters* tenta identificar múltiplas regiões convexas do espaço formado por X. O objetivo é que as observações em cada região sejam o mais próximos ou similares entre si e que cada conjunto de observações formados diferenciem-se tanto quanto possível entre si. Esses métodos podem ser instrumentos úteis tanto como procedimentos preparatórios para outros tipos de análises quanto para a identificação de relações entre variáveis.

Como se pode perceber, métodos de agrupamento assemelham-se com os métodos de classificação, já que ambos visam atribuir um tipo ou classe a cada observação. Diferentemente dos métodos de classificação, todavia, não há variáveis resposta para orientar a formação dos *clusters* e permitir a avaliação do desempenho dos métodos.

Essa avaliação é uma das principais dificuldades da aplicação desses métodos. É possível comparar os grupos formados a partir da aplicação de diferentes modelos, mas não é possível identificar os que tiveram melhor desempenho, como foi feito com os métodos de classificação, por exemplo. Uma forma de contornar essa dificuldade é, havendo objetos cuja classificação ou grupo são conhecidos, omitir a variável correspondente à classificação, executar os modelos e contabilizar os resultados. Nesta seção, será apresentado um método de agrupamento, chamado *k-means*. Para que observações possam ser agrupadas, é necessária a utilização de uma métrica de similaridade. A métrica utilizada no *k-means* é a distância euclidiana. Por essa razão, esse método somente pode ser aplicado a variáveis quantitativas e os grupos formados tendem a ser circulares. Eventualmente, é possível adaptar variáveis qualitativas de modo a aplicá-lo. Por exemplo, cria-se uma escala para as variáveis (variáveis ordinais), tal qual ruim, bom, ótimo e uma função que meça o esforço de passar de uma classe para outra. Como no método *k- nearest neighbor*, cada observação é assinalada ao grupo ao qual é mais próxima. Para isso, é preciso definir, inicialmente, o número de grupos que serão formados (k) e um ponto de partida inicial. Não há um procedimento estabelecido para a escolha do número de agrupamentos a serem formados. Pode-se defini-los a partir do problema ou das observações dos dados. Um procedimento baseado nos dados que pode ser usado é chamado de Calinski-Harabasz (CH). Ele se baseia na variação intra e entre grupos (DEY, 2022). A variação dentro do grupo é medida por $W = \sum_{k=1}^{k} \sum_{c(i)=k} \|x_i - \bar{x}_k\|$, na qual k designa o agrupamento, x_i é a observação i e $\bar{x}_k$ é a média do agrupamento k. Já a variação entre grupos é dada por $B = \sum_{k=1}^{k} n_k \|\bar{x}_k - \bar{x}\|$, na qual n_k é o número de observações em k e $\bar{x}$ é a média de todas as observações. O índice CH é dado por

$$CH = \frac{B(k)/(k-1)}{W(k)/(n-k)}.$$

Em síntese, o índice de CH é uma medida do quão similar um objeto é ao restante do seu grupo, o que é captado por W (quanto menor W, mais coeso o grupo), e o quão díspares são os grupos entre si, o que é captado por B (quanto maior, mais díspares). Entendendo-se que o objetivo do modelo é formar grupos coesos, mas que se distingam entre si, quanto maior o CH, melhor o seu desempenho.

Diferentemente de métodos anteriormente estudados, não necessariamente deve-se normalizar os dados para aplicar o *k-means*. A normalização resulta em se atribuir o mesmo peso a todas as variáveis usadas no algoritmo, ignorando-se que variáveis diferentes podem influenciar diferentemente nos agrupamentos. Se esse for o intuito, pode-se normalizar as observações; mas, se não for, é preciso estar ciente dos possíveis impactos. As Figuras 7 e 8, abaixo, mostram um conjunto de observações no qual foi aplicado o *k-means* com o fim de separar as observações em

dois conjuntos. Na Figura 7, são apresentados os dados não tratados. Distinguem-se dois conjuntos. Na Figura 8, os dados foram normalizados. Os grupos não são mais facilmente distinguíveis.

Figura 7 - Dados não normalizados agrupados por meio do K-means

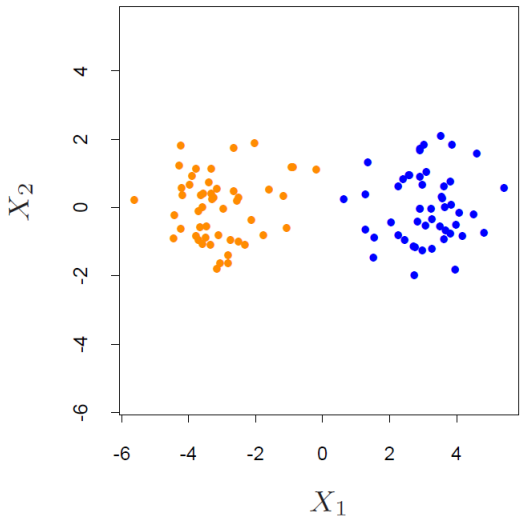
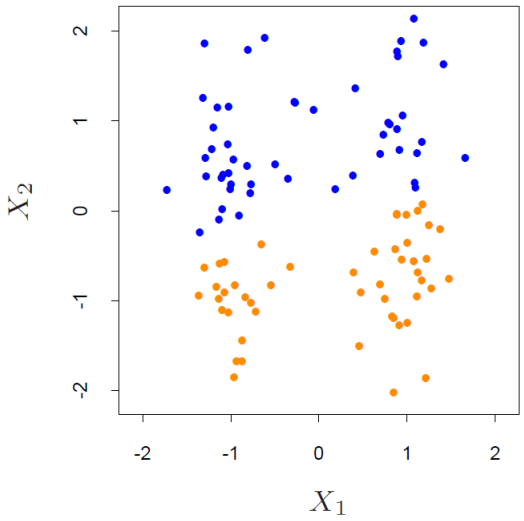


Figura 8 - Dados normalizados agrupados por meio do K-means



Fonte: HASTIE, TIBSHIRANI e FRIEDMAN, 2009, p. 506

Após a definição de *k,* estabelecem-se b_k centróides, suficientemente espalhados entre as observações. Dados os centróides, assinalam-se cada observação ao grupo do centróide mais próximo, considerando-se $C_i = \underset{c\in\{1,\dots,K\}}{\operatorname{argmin}} \|x_i - b_k\|^2$. Depois que todas as observações forem atribuídas a um grupo, atualiza-se o centróide de cada grupo, de forma que este seja a média do grupo formado: $b_k = \sum_{c(i)=k} x_i / n_k$. O processo recomeça até que todos os grupos estejam estáveis.

10. CONCLUSÃO

Neste estudo, procurou-se apresentar modelos e procedimentos de aprendizagem de máquina que podem ser utilizados na análise antitruste. Como parte da comunidade que trabalha com o tema tem familiaridade com econometria, discutiu-se, inicialmente, as relações entre esta e aquele. Foi visto que, apesar de estudos econométricos e de aprendizado de máquina terem objetivos diferentes, as diversas técnicas deste podem ser utilizadas por economistas tanto para aprimorar a análise de problemas econométricos quanto para avançar no estudo de problemas anteriormente não tratados na Economia. Ainda que o principal objetivo de trabalhos

econométricos seja inferir causalidade, muitos têm componentes de previsão, que podem ser aprimorados por meio do estudo do aprendizado de máquina. Trabalhos recentes, ademais, têm sido feitos com o objetivo de estabelecer as condições e formas que permitem que métodos de previsão sejam usados em problemas de causalidade. Na defesa da concorrência, esses métodos podem ajudar a aumentar a robustez das análises e orientar o trabalho das autoridades de concorrência e de outras entidades.

Procurou-se apresentar alguns métodos de aprendizado de máquinas com o fim de demonstrar a sua aplicabilidade. Como se pode perceber, os principais métodos são facilmente implementados, havendo, nos *softwares* estatísticos mais populares, meios estabelecidos para sua aplicação. A literatura na área também é crescente e, além de livros textos e artigos científicos, há uma ampla disponibilidade de informações na internet. Assim, o leitor interessado não terá dificuldade em obter novos conhecimentos. Nesse sentido, a bibliografia dos textos usados como referência neste trabalho pode ser um bom começo.

BIBLIOGRAFIA

ABADIE, A.; KASY, M. Choosing among regularized estimators in empirical Economics: The Risk of Machine Learning. **The Review of Economics and Statistics**, v. 101, n. 5, p. 743–762, 21 dez. 2018.

ABADIE, A.; KASY, M. Choosing among regularized estimators in empirical economics: The risk of machine learning. **The Review of Economics and Statistics**, vol. 101, 2019.

AGRAWAL, A.; GANS, J.; GOLDFARB, A. (EDS.). **The Economics of Artificial Intelligence: An Agenda**. First edition ed. Chicago: University of Chicago Press, 2019.

AI, S. Reinforcement learning algorithms — an intuitive overview. Disponível em: <https://medium.com/@SmartLabAI/reinforcement-learning-algorithms-an-intuitive-overview-904e2dff5bbc>. Acesso em: 16 jun. 2020.

ANGRIST, J. D.; KRUEGER, A. B. Instrumental variables and the search for identification: From Supply and Demand to Natural Experiments. **Journal of Economic Perspectives**, vol. 15, 2001.

ATHEY, S. Beyond prediction: Using big data for policy problems. **Science**, v. 355, n. 6324, p. 483–485, 3 fev. 2017.

ATHEY, S; BLEI, D.; DONNELLY, R.; RUIZ, F.; SCHIMDT, T. Estimating heterogenous consumer preferences for restaurants and travel time using mobile location data. **AEA Papers and Procedures**, v. 108, maio de 2018.

ATHEY, S.; HORVITZ, E. Fireside chat with Susan Athey. 7 de agosto de 2018. Disponível em https://www.microsoft.com/en-us/research/video/fireside-chat-with-susan-athey/. Acesso em 10/07/2022.

ATHEY, S.; IMBENS, G. A Measure of robustness to misspecification. **American Economic Review**, v. 105, n. 5, p. 476–480, 1 maio 2015.

ATHEY, S.; IMBENS, G. The state of applied econometrics: Causality and policy evaluation. **Journal of Economic Perspectives**, v. 31, n. 2, p. 3–32, 1 maio 2017.

ATHEY, S.; IMBENS, G. Machine learning and econometrics. Curso oferecido no Encontro Anual da Associação norte-americana de economistas, 2018. Gravação das aulas e material disponível em https://www.aeaweb.org/conference/cont-ed/2018-webcasts. Acesso em 06/08/2020.

ATHEY, S.; IMBENS, G. Machine learning methods economists should know about. **Annual Review of Economics, vol. 11, 2019**.

ATHEY, S.; IMBENS, G.; WAGER, S. Approximate residual balancing: De-biased inference of average treatment effects in high dimensions. **arXiv:1604.07125 [econ, math, stat]**, 31 jan. 2018.

ATHEY, S.; LUCA, M. Economists (and Economics) in tech companies. **Journal of Economic Perspectives**, v.33, n. 1, 2019.

ATHEY, S.; STERN, S. The impact of information technology on emergency health care outcomes. **The RAND Journal of Economics**, v. 33, n. 3, p. 399–432, 2002.

AUTORITÉ DE LA CONCURRENCE. Rapport annuel 2019. 27 de maio de 2020. Disponível em https://www.autoritedelaconcurrence.fr/sites/default/files/2020-07/rapport-annuel-2019.pdf. Acesso em 10/07/2022.

AYUYA, C. Entropy and information gain to build decision trees in machine learning. Jul. 2021. Disponível em https://www.section.io/engineering-education/entropy-information-gain-machine-learning/#:~:text=Entropy%20is%20an%20information%20theory,tree%20chooses%20to%20split%20data.. Acesso em 08/07/2022.

BAJARI, P. et al. Machine learning methods for demand estimation. **American Economic Review**, v. 105, n. 5, p. 481–485, 1 maio 2015.

BELLONI, A.; CHERNOZHUKOV, V.; HANSEN, C. High-dimensional methods and inference on structural and treatment effects. **Journal of Economic Perspectives**, v. 28, n. 2, p. 29–50, maio 2014.

BROWNLEE, J. Lessons for machine learning from econometrics. Machine Learning Mastery, 18 maio 2014. Disponível em: <https://machinelearningmastery.com/lessons-for-machine-learning-from-econometrics/>. Acesso em: 16 jun. 2020

CADE. Guia para análise de atos de concentração horizontal. Brasília, 2016. Disponível em https://cdn.cade.gov.br/Portal/centrais-de-conteudo/publicacoes/guias-do-cade/guia-para-analise-de-atos-de-concentracao-horizontal.pdf. Acesso em 10/07/2022.

CAJUEIRO, D. Técnicas de estimação de demanda usando aprendizagem de máquina. Nota técnica no 29/2018/DEE/CADE. 2018. CAMERON, A. C. Machine learning for microeconometrics. Disponível em http://cameron.econ.ucdavis.edu/e240f/trmachinelearningseminar.pdf. Acesso em 10/07/2022.

CAMPUZANO, Susana. Apresentação no painel “The use of data science tools in competition law enforcement – the prospects of artificial intelligence in the future” na conferência “The Promisse of computational competition law and Economics”. Co-organizada pela Hellenic Competition

Commission e o BRICS Competition Law and Policy Centre, 24/05/2021.

CARVALHO, A; CAJUEIRO, D.; CAMARGO, R. **Introdução aos Métodos Estatísticos para Economia e Finanças**. Brasília: Ed. UNB. 2015

CHERNOZHUKOV, V. et al. Double/debiased/neyman machine learning of treatment Effects. **The American Economic Review**, v. 107, 2017.

DELRAHIM, M. Never break the chain: Pursuing antifragility in antitrust enforcement. 27 de agosto de 2020. Disponível em https://www.justice.gov/opa/speech/assistant-attorney-general-makan-delrahim-delivers-remarks-thirteenth-annual-conference. Acesso em 10/07/2020.

DEY, D. Calinski-Harabasz Index – Cluster validaty indices | Set 3. Disponível em https://www.geeksforgeeks.org/calinski-harabasz-index-cluster-validity-indices-set-3/. Acesso em 07/07/2022.

DOMINGOS, P. A few useful things to know about machine learning. **Communications of the ACM**, v. 55, n. 10, p. 78–87, out. 2012.

DONNELLY, R. et al. Counterfactual inference for consumer choice across many product categories. Quantitative Marketing and Economics, v. 19, 2021.

DOUDCHENKO, N.; IMBENS, G. **Balancing, Regression, Difference-In-Differences and Synthetic Control Methods: A Synthesis**. Cambridge, MA: National Bureau of Economic Research, out. 2016. Disponível em: <http://www.nber.org/papers/w22791.pdf>. Acesso em: 3 jun. 2020.

GROSSMAN, L. CADE adota inteligência artificial para agilizar combate aos cartéis. 06/08/2018. Disponível em https://www.convergenciadigital.com.br/Inovacao/CADE-adota-inteligencia-artificial-para-agilizar-combate-aos-carteis-48613.html?UserActiveTemplate=mobile%2Csite. Acesso em 05/07/2022.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. **The Elements of Stastical Learning**. Springer, 2009.

HELLENIC COMPETITION COMMISSION. Press-release – Presentation of the HCC Data Analytics and Economic Intelligence Platform. 13/04/2021. Disponível em https://www.epant.gr/en/enimerosi/press-releases/item/1373-press-release-presentation-of-the-hcc-data-analytics-and-economic-intelligence-platform.html. Acesso em 30/06/2021.

HUNT, Stefan. CMA's new DaTA Unit: exciting opportunities for data scientists. Disponível em https://competitionandmarkets.blog.gov.uk/2018/10/24/cmas-new-data-unit-exciting-opportunities-for-data-scientists/. Acesso em 30/06/2021.

HUNT, Stefan. The CMA DaTA Unit – We're are growing. Disponível em https://competitionandmarkets.blog.gov.uk/2019/05/28/the-cma-data-unit-were-growing/. Acesso em 30/06/2021.

IMAI, K.; RATKOVIC, M. Estimating treatment effect heterogeneity in randomized program evaluation. **The Annals of Applied Statistics**, v. 7, n. 1, p. 443–470, mar. 2013.

IMAI, K.; RATKOVIC, M. Covariate balancing propensity score. **Journal of the Royal Statistical Society: Series B (Statistical Methodology)**, v. 76, n. 1, p. 243–263, jan. 2014.

IMBENS, G. W.; RUBIN, D. B. **Causal Inference for Statistics, Social, and Biomedical Sciences: An Introduction**. New York: Cambridge University Press, 2015.

IZENMAN, A. **Modern Multivariate Statistical Techniques – Regression, Classification and Manifold Learning**. Springer, 2008.

KOHAVI, R. et al. Controlled experiments on the web: survey and practical guide. **Data Mining and Knowledge Discovery**, v. 18, n. 1, p. 140–181, fev. 2009.

MCELHERAN, K; BRYNJOLFSSON, E. The rapid adoption of data-driven decision making. **American Economic Review**, v. 106, maio de 2016.

MEHRABI, N. Et al. A survey on bias and fairness in machine learning. ACM Computing Surveys, v. 54, n.6, p. 115:1-115:35, jul. 2022.

MULLAINATHAN, S.; SPIESS, J. Machine learning: An applied econometric approach. **Journal of Economic Perspectives**, v. 31, n. 2, p. 87–106, maio 2017.

OCDE. OECD Handbook on Competition Policy in Digital Age. Disponível em https://www.oecd.org/daf/competition/oecd-handbook-on-competition-policy-in-the-digital-age.pdf. Acesso em 07/07/2022.

PIMENTA, Guilherme. Projeto Cérebro: Cade usa inteligência artificial no combate a cartéis. Jota. Brasília, 29/10/2019. Disponível em https://www.jota.info/coberturas-especiais/inova-e-acao/projeto-cerebro-cade-usa-inteligencia-artificial-no-combate-a-carteis-29102019. Acesso em 05/07/2022.

ROSENBAUM, P. R.; RUBIN, D. B. The Central role of the propensity score in observational studies for causal effects. **Biometrika**, v. 70, n. 1, p. 41–55, 1983.

SAMUEL, ARTHUR. Some Studies in Machine Learning Using the Game of Checkers. IBM Journal, 1959.

SCHREPEL, T.; GROZA, T. (ed.). The adoption of computational antitrust by agencies: 2021 report. Stanford Computational Antitrust – Implementation Survey. Disponível em schrepel-groza-computational-antitrust (ssrn.com). Acesso em 05/07/2022.

THE ECONOMIST. Data, data everywhere – Special report. The Economics, 27/02/2010. Disponível em https://www.economist.com/special-report/2010/02/27/data-data-everywhere. Acesso em 05/07/2022.

THEODORIS, S. **Machine Learning: A Bayesian and Optimization Perspective**. Elsevier. 2015.

VARIAN, H. R. Big Data: New Tricks for Econometrics. **Journal of Economic Perspectives**, v. 28, n. 2, p. 3–28, maio 2014.

WUTHRICH, K.; ZHU, Y. Omitted variable bias of Lasso-based inference methods: A finite sample analysis. **arXiv:1903.08704 [math, stat]**, 12 jan. 2020.

ANEXO 1 – DESCRIÇÃO DA BASE DE DADOS – CUSTOS DE OPERADORAS DE PLANOS DE SAÚDE E OUTRAS VARIÁVEIS FINANCEIRAS

Para estimação dos métodos e modelos apresentados neste trabalho, foi construída uma base de dados a partir de informações públicas referentes a custos de operadoras de planos de saúde, beneficiários dessas operadoras, população dos municípios brasileiros e hospitais que têm leitos disponíveis para beneficiários de planos de saúde.

Os dados econômico-financeiros referem-se a 2019 e foram obtidos no portal brasileiro de dados, no conjunto referente às demonstrações contábeis das operadoras (http://dados.gov.br/dataset/http-www-ans-gov-br-perfil-do-setor-dados-abertos-dados-abertos-disponiveis-n3, acessado em 01/07/2020). Foram incluídas na base os valores referentes às seguintes contas:

Codificação da conta	Descrição
31111	Contraprestação emitida/ Prêmio emitido de assistência médico-hospitalar
41111	Eventos/sinistros conhecidos ou avisados na modalidade pagamento por procedimento, de assistência à saúde médico-hospitalar
41121	Eventos/sinistros conhecidos ou avisados na modalidade capitation, de assistência à saúde médico-hospitalar
41131	Eventos/sinistros conhecidos ou avisados na modalidade orçamento global, de assistência à saúde médico-hospitalar
41141	Eventos/sinistros conhecidos ou avisados na modalidade pacote, de assistência à saúde médico-hospitalar
41151	Eventos/sinistros conhecidos ou avisados na modalidade rateio de custos de recursos próprios, de assistência à saúde médico-hospitalar
41171	Eventos/sinistros conhecidos ou avisados de reembolso, de assistência à saúde médico-hospitalar
41181	Eventos/sinistros conhecidos ou avisados no sistema único de saúde, de assistência à saúde médico-hospitalar
41191	Eventos/sinistros conhecidos ou avisados em outras formas de pagamento, de assistência à saúde médico-hospitalar

Os eventos, nas diferentes formas, foram somados e o resultado foi considerado o custo assistencial da operadora em 2019. Foram calculadas as proporções de pagamento por procedimento (41111) e de despesas em prestadores próprios (41151) para cada operadora.

Os dados de beneficiários também vieram do portal brasileiro de dados, do conjunto intitulado "Informações consolidadas de beneficiários" (http://dados.gov.br/dataset/informacoes-consolidadas-de-beneficiarios, acessado em 01/07/2020). Foram usados os dados referentes a dezembro de 2020, de todos os estados brasileiros e a tabela com os beneficiários para os quais não há informações sobre o estado de moradia. Retirados todos os beneficiários de planos odontológicos, esses foram agrupados por operadora. Calculou-se, para cada operadora, a proporção de beneficiários do sexo feminino, idosos (acima de 59 anos), de planos contratados individualmente e de beneficiários que residiam no municipio no qual havia mais beneficiários da operadora.

Incluiu-se na base de dados a população de cada município no qual cada operadora tinha mais beneficiários. Esses dados foram retirados da tabela 6579 do Banco de tabelas estatísticas do Instituto Brasileiro de Geografia e Estatística (Sidra/IBGE).

Por fim, foi incluído o HHI (Índice de Herfindal-Hirschman) dos hospitais que têm leitos destinados à saúde suplementar do município que tem a maior quantidade de beneficiários por operadora. O índice foi calculado com base no município e nos leitos disponíveis para a saúde suplementar. Os dados foram coletados no site da Agência Nacional de Saúde Suplementar (http://www.ans.gov.br/perfil-do-setor/dados-e-indicadores-do-setor. Tabela "Estabelecimentos de saúde cadastrados (CNES – Ministério da Saúde). Acesso em 07/02/2020).

A tabela final tem 527 observações. Foram excluídas observações que não tinham todos os dados (15 observações.

ANEXO 2 – BASE DE DADOS DE MUNICÍPIOS

A segunda base de dados utilizada neste trabalho foi construída com dados provindos de diferentes fontes.

Os dados de beneficiários que foram utilizados para a determinação da quantidade de operadoras fazem parte do conjunto de dados "Informações Consolidadas de Beneficiários" do portal brasileiro de dados. Foi utilizada a coluna "QT_BENEFICIARIO_ADERIDO". Assim, a quantidade de operadoras que têm pelo menos um beneficiário aderido em dezembro de 2018 é a proxy da quantidade de operadoras em atuação, a variável dependente do modelo estimado. Dessa base de dados, também foram extraídos dados referentes à quantidade de beneficiários no município, independente da data em que se vincularam às operadoras (QT_BENEFICIARIO_ATIVO).

Foram coletados os dados de população residente estimada pelo IBGE para o ano de 2019 (tabela 6579, no sistema Sidra do IBGE). Também do IBGE, foram usados os dados coletados pelo Instituto, no Censo de 2010, para calcular a proporção de pessoas mais velhas na população (tabela 1378 do sistema Sidra). Somaram-se o total de pessoas com mais de 50 anos. A escolha do corte em 50 anos e não 60 ou 65, quando, usualmente, são considerados idosos, foi motivada pelo fato de que os dados referem-se a 2010. Se não houvesse havido nenhuma morte e nenhum nascimento nos municípios, essa seria a proporção de idosos atual. Independentemente disso, a faixa etária entre 50 e 60 anos também utiliza mais recursos de saúde do que os mais jovens.

Os dados de PIB per capita, referentes a 2017, calculados pelo IBGE, em parceria com órgãos estaduais de estatística, secretarias estaduais de governo e a Superintendência da Zona Franca de Manaus foram incluídos na base.

O IDH-M (ou IDH) é um índice composto por três dimensões: longevidade, educação e renda. Embora não tenha saúde, explicitamente, em sua composição, a logenvidade relaciona-se com esta. Além disso, é provável que haja correlação entre a capacidade do governo local prestar bons serviços a seus cidadãos e a o desempenho do IDH-M. O índice utilizado refere-se a 2010 e foi calculado a partir de informações do censo demográfico pelo PNUD Brasil, Instituto de Pesquisa Econômica Aplicada – IPEA e Fundação João Pinheiro.

O Índice de Desempenho do SUS (IDSUS) visava aferir o desempenho do SUS quanto ao cumprimento de seus princípios e diretrizes em cada município. Para isso, considerava-se a universalidade do acesso, integralidade, igualdade, resolubilidade e equidade de atenção, entre outros fatores. Apesar do objetivo, o índice foi calculado apenas uma vez, em 2011. Em decorrência de polêmica entre os prefeitos e de resistência política, o cálculo do índice foi descontinuado. Entre

os indicadores que compunham o índice, selecionou-se o indicador de cobertura populacional estimada pelas equipes básicas de saúde.

Tanto os dados de leitos em prestadores hospitalares de natureza pública quanto os dados de prestadores hospitalares de natureza privada foram coletados no site da ANS16. Foram considerados de natureza privada, os prestadores com as seguintes naturezas: associação privada, cooperativa, empresa individual de responsabilidade limitada (de natureza empresária), empresa individual de responsabilidade limitada (de natureza simples), empresário (individual), entidade sindical, fundação privada, organização social (OS), serviço social autônomo, sociedade anônima aberta, sociedade anônima fechada, sociedade empresária em comandita simples, sociedade empresária limitada, sociedade simples em nome coletivo, sociedade simples limitada e sociedade pura. Os prestadores que não tinham classificada sua natureza foram considerados privados.

[16] http://www.ans.gov.br/perfil-do-setor/dados-e-indicadores-do-setor. Tabela "Estabelecimentos de saúde cadastrados (CNES – Ministério da Saúde). Acesso em 07/02/2020.

ANEXO 3 – CÓDIGO DO TESTE DA K-FOLD CROSS VALIDATION

Base de dados

```
load("C:/Users/t_lim/Desktop/artigos/verticalizacao/dados_r/planilha_final.RData")

plan_final <- plan_final[complete.cases(plan_final), ]

plan_final<-plan_final %>% mutate(custo_benef=eventos/total_benef)
```

Matriz para a comparação de resultados das estimações com cross validation e sem

```
zero<-rep(0, times=18)
resultado<-matrix(zero, nrow=6, ncol=3)
rownames(resultado)<-c("sem validação", "cv (k=5)", "cv (k=10)","cv (k=n)", "repeated cv (k=5)",
                       "repeated cv (k=10)")
colnames(resultado)<-c("R-squared", "RMSE", "MAE")
```

Regressões

```
modelo_0<-lm(norm_eventos ~ d_filant + d_coop + norm_vert + norm_total_benef +
  norm_populacao + norm_hhi + norm_perc_idoso + norm_perc_mulher + norm_perc_ind + norm_perc_cidade,
data = plan_final)

summary(modelo_0)

resultado[1,1]<-0.7554

## Função para calculara o RMSE da regressão
rmse <- function(error)
{
  sqrt(mean(error^2))
}

rl_rmse<-rmse(modelo_0$residuals)

resultado[1,2]<-rl_rmse

## Função para calcular o MAE da regressão
mae <- function(error)
{
  mean(abs(error))
}


rl_mae<-mae(modelo_0$residuals)

resultado[1,3]<-rl_mae

## Modelos com validação cruzada
```

```r
## K-fold

# n=5

set.seed(123)

## k-fold com k=5
train.control_1 <- trainControl(method = "cv", number=5)

# Train the model
modelo_1 <- train(norm_eventos ~ d_filant + d_coop + norm_vert + norm_to
tal_benef +
                    norm_populacao + norm_hhi + norm_perc_idoso + norm_p
erc_mulher + norm_perc_ind + norm_perc_cidade,
                  data = plan_final, method = "lm",
                 trControl = train.control_1)

print(modelo_1)

resultado [2,]<-c(0.8546, 0.5906829, 0.1500213)

# k = 10

set.seed(123)
train.control_2 <- trainControl(method = "cv", number = 10)

# Train the model

modelo_2 <- train(norm_eventos ~ d_filant + d_coop + norm_vert + norm_to
tal_benef +
                    norm_populacao + norm_hhi + norm_perc_idoso + norm_p
erc_mulher + norm_perc_ind + norm_perc_cidade,
                  data = plan_final, method = "lm",
                trControl = train.control_2)

print(modelo_2)

resultado[3,]<-c(0.8095, 0.482271, 0.1472521)

## k-fold com k=n

set.seed(123)
train.control_3 <- trainControl(method = "LOOCV")

# Train the model

modelo_3 <- train(norm_eventos ~ d_filant + d_coop + norm_vert + norm_to
tal_benef +
                    norm_populacao + norm_hhi + norm_perc_idoso + norm_p
erc_mulher + norm_perc_ind + norm_perc_cidade,
                  data = plan_final, method = "lm",
                trControl = train.control_3)
```

```r
print(modelo_3)

resultado[4,]<-c(0.5490, 0.6867129, 0.1424234)

## Repeated k-fold com k=5

set.seed(123)
train.control_4 <- trainControl(method = "repeatedcv",
                                number = 5, repeats = 3)
# Train the model

modelo_4 <- train(norm_eventos ~ d_filant + d_coop + norm_vert + norm_to
tal_benef +
                    norm_populacao + norm_hhi + norm_perc_idoso + norm_p
erc_mulher + norm_perc_ind + norm_perc_cidade,
                  data = plan_final, method = "lm",
                  trControl = train.control_4)

print(modelo_4)

resultado[5,]<-c(0.8765, 0.5995473, 0.1528672)

## Repeated k-fold com k=10

set.seed(123)
train.control_5 <- trainControl(method = "repeatedcv",
                                 number = 10, repeats = 3)
# Train the model

modelo_5 <- train(norm_eventos ~ d_filant + d_coop + norm_vert + norm_to
tal_benef +
                    norm_populacao + norm_hhi + norm_perc_idoso + norm_p
erc_mulher + norm_perc_ind + norm_perc_cidade,
                  data = plan_final, method = "lm",
                  trControl = train.control_5)

print(modelo_5)

resultado[6,]<-c(0.8380, 0.4908809, 0.1472744)
```

Step-regression com repeated cv

```r
# Instalar pacote leaps

# install.packages("leaps")

# Train the model

modelo_6 <- train(eventos ~ d_filant + d_coop + vert + total_benef  +
                    populacao + hhi + perc_idoso + perc_mulher + perc_in
d + perc_cidade,
                  data = plan_final, method = "leapBackward",
                  tuneGrid = data.frame(nvmax = 1:10),
```

```
                    trControl = train.control_5)

print(modelo_6)
```

ANEXO 4 – LASSO

```
options(stringsAsFactors = FALSE, scipen = 999)
library(dplyr)

library(ggplot2)
library(readr)
library(tidyr)
library(glmnet)
```

Base de dados

```
load("")

plan_final <- plan_final[complete.cases(plan_final), ]

# Para estimação do lasso, devemos usar os dados normalizados
# Os regressores devem estar em uma matriz

x<-as.matrix(plan_final[,(28:35)])
y<-as.matrix(plan_final[,27])
```

Estimação do lasso

```
lasso.fit<-cv.glmnet(x, y, keep=TRUE)

# Lasso com o lambda com menor EQM

lasso.fit$lambda.min
```

Resultados

```
# Gráfico do EQM x Lambda
plot(lasso.fit)



# Número de coeficientes por lambda

lambda<-lasso.fit$lambda
ncoef<-lasso.fit$nzero
res<-data.frame(lambda,ncoef)

# Coeficientes da regressão ótima

coef(lasso.fit, s="lambda.min")

# Matriz de dimensão n x { numéro de lambdas}, com as predições no conju
nto de treinamento
```

```
crossfit.preds = lasso.fit$fit.preval[,1:length(lambda)]

# Plot the dependence of the cross-fitted prediction on lambda,
# for the 3rd training point.
plot(lambda, crossfit.preds[3,])
abline(h = y[3], col=2)
abline(v = lasso.fit$lambda.min, col=2)
```

Lasso sem validação cruzada

```
# Vai gerar diversos lambda para se escolher

# Estabeleço conjunto de treinamento e de teste


set.seed(123)
treinamento<-plan_final[sample(nrow(plan_final), size=523, replace=FALSE
),]
teste<-anti_join(plan_final, treinamento, by=NULL)

x.treinamento<-as.matrix(treinamento[,(28:35)])
y.treinamento<-as.matrix(treinamento[27])
x.teste<-as.matrix(teste[,(28:35)])
y.teste<-as.matrix(teste[27])

# Estimo um lasso com o conjunto de treinamento

lasso<-glmnet(x.treinamento, y.treinamento)

plot(lasso, xvar="lambda")
```

Pós-lasso

```
# Estimar MQO com as variáveis selecionadas pelo Lasso


pos.lasso<-lm(eventos~total_benef + populacao+ vert + hhi + perc_idoso+
perc_mulher + perc_ind +
                perc_cidade, data=plan_final)


stargazer(pos.lasso, type="text",
          dep.var.labels=c("Custo total"),
          covariate.labels=c("Total de Beneficiários", "População", "Ver
ticalização", "HHI",
                   "Percentual de Idosos", "Percentual de mulheres", "Per
centual em planos individuais",
                             "Percentual na Cidade"), out="pos_lasso.txt
```

```
")
```

Lasso para a variável associada à verticalização

```
# Primeiramente, vamos criar uma dummy associada à verticalização

plan_final <- plan_final %>% mutate(d_vert=ifelse(vert==0, 0, 1))
plan_final <- plan_final %>% mutate(d_vert=factor(d_vert))


# Matrizes com as variáveis explicativas e a endógena

x.vert<-as.matrix(plan_final[,(29:35)])
y.vert<-as.matrix(plan_final[,27])


# Estimação do lasso

lasso.vert<-cv.glmnet(x.vert, y.vert, keep=TRUE)


# Lasso com o lambda com menor EQM

lasso.vert$lambda.min

# Coeficientes da regressão ótima

coef(lasso.vert, s="lambda.min")

# Regressão com o fim de verificar os efeitos do verticalização nos cust
os



pos.vert.1<-lm(log(eventos) ~ log(total_benef) + log(populacao) + d_vert
+ hhi + + perc_idoso +
                 perc_mulher + perc_ind + perc_cidade, data=plan_final)

summary(pos.vert.1)

stargazer(pos.vert.1, type="text",
          dep.var.labels=c("Custo eventos"),
          covariate.labels=c("Total de Beneficiários", "População", "Ver
ticalização", "HHI",
                             "Percentual de Idosos", "Percentual de mulh
eres",
                             "Percentual em planos individuais", "Percen
tual na Cidade"),
          out="regressao3.txt")
```

ANEXO 5 - COMPARAÇÃO DE MODELOS – EQUAÇÕES REGULARIZADAS

* Baseado no Código apresentado em https://www4.stat.ncsu.edu/~post/josh/LASSO_Ridge_Elastic_Net_-_Examples.html (acessado em 05/07/2020)

```
options(stringsAsFactors = FALSE, scipen = 999)
library(dplyr)
library(ggplot2)
library(readr)
library(tidyr)
library(glmnet)
library(stargazer)

## Baixar a planilha com todos os dados

load("C:/Users/t_lim/Dropbox/DEE/econometria_machine_learning/dados_tratados/planilha_final.RData")

## Tirar observações que não estão completas

plan_final <- plan_final[complete.cases(plan_final), ]


# Para estimação das regressões, devemos usar os dados normalizados
# Os regressores devem estar em uma matriz

## Separar o conjunto de treinamento e o de teste
set.seed(123)
treinamento<-plan_final[sample(nrow(plan_final), size=523, replace=FALSE),]
teste<-setdiff(plan_final, treinamento, by=NULL)

x.treinamento<-as.matrix(treinamento[,(28:35)])
y.treinamento<-as.matrix(treinamento[27])
x.teste<-as.matrix(teste[,(28:35)])
y.teste<-as.matrix(teste[27])

## Estimação do lasso, do ridge e da elastic.net

fit.lasso <- glmnet(x.treinamento, y.treinamento, family="gaussian", alpha=1)
fit.ridge <- glmnet(x.treinamento, y.treinamento, family="gaussian", alpha=0)
fit.elnet <- glmnet(x.treinamento, y.treinamento, family="gaussian", alpha=.5)


## Validação cruzada para diferentes alfas

for (i in 0:10) {
  assign(paste("fit", i, sep=""), cv.glmnet(x.treinamento, y.treinamento, type.measure="mse",
                                            alpha=i/10,family="gaussian"))
}

## Gráficos
```

```
par(mfrow=c(3,2))


plot(fit.lasso, xvar="lambda")
plot(fit10)

plot(fit.ridge, xvar="lambda")
plot(fit0)

plot(fit.elnet, xvar="lambda")
plot(fit5)


## EQM no conjunto teste

# Lambda mínimo

yhat0 <- predict(fit0, s=fit0$lambda.min, newx=x.teste)
yhat1 <- predict(fit1, s=fit1$lambda.min, newx=x.teste)
yhat2 <- predict(fit2, s=fit2$lambda.min, newx=x.teste)
yhat3 <- predict(fit3, s=fit3$lambda.min, newx=x.teste)
yhat4 <- predict(fit4, s=fit4$lambda.min, newx=x.teste)
yhat5 <- predict(fit5, s=fit5$lambda.min, newx=x.teste)
yhat6 <- predict(fit6, s=fit6$lambda.min, newx=x.teste)
yhat7 <- predict(fit7, s=fit7$lambda.min, newx=x.teste)
yhat8 <- predict(fit8, s=fit8$lambda.min, newx=x.teste)
yhat9 <- predict(fit9, s=fit9$lambda.min, newx=x.teste)
yhat10 <- predict(fit10, s=fit10$lambda.min, newx=x.teste)

mse0 <- mean((y.teste - yhat0)^2)
mse1 <- mean((y.teste - yhat1)^2)
mse2 <- mean((y.teste - yhat2)^2)
mse3 <- mean((y.teste - yhat3)^2)
mse4 <- mean((y.teste - yhat4)^2)
mse5 <- mean((y.teste - yhat5)^2)
mse6 <- mean((y.teste - yhat6)^2)
mse7 <- mean((y.teste - yhat7)^2)
mse8 <- mean((y.teste - yhat8)^2)
mse9 <- mean((y.teste - yhat9)^2)
mse10 <- mean((y.teste - yhat10)^2)

# Lambda com menor desvio

yhat11 <- predict(fit0, s=fit0$lambda.1se, newx=x.teste)
yhat12 <- predict(fit1, s=fit1$lambda.1se, newx=x.teste)
yhat13 <- predict(fit2, s=fit2$lambda.1se, newx=x.teste)
yhat14 <- predict(fit3, s=fit3$lambda.1se, newx=x.teste)
yhat15 <- predict(fit4, s=fit4$lambda.1se, newx=x.teste)
yhat16 <- predict(fit5, s=fit5$lambda.1se, newx=x.teste)
yhat17 <- predict(fit6, s=fit6$lambda.1se, newx=x.teste)
yhat18 <- predict(fit7, s=fit7$lambda.1se, newx=x.teste)
```

```
yhat19 <- predict(fit8, s=fit8$lambda.1se, newx=x.teste)
yhat20 <- predict(fit9, s=fit9$lambda.1se, newx=x.teste)
yhat21 <- predict(fit10, s=fit10$lambda.1se, newx=x.teste)

mse11 <- mean((y.teste - yhat11)^2)
mse12 <- mean((y.teste - yhat12)^2)
mse13 <- mean((y.teste - yhat13)^2)
mse14 <- mean((y.teste - yhat14)^2)
mse15 <- mean((y.teste - yhat15)^2)
mse16 <- mean((y.teste - yhat16)^2)
mse17 <- mean((y.teste - yhat17)^2)
mse18 <- mean((y.teste - yhat18)^2)
mse19 <- mean((y.teste - yhat19)^2)
mse20 <- mean((y.teste - yhat20)^2)
mse21 <- mean((y.teste - yhat21)^2)


eqm<-c(0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1,
       mse0, mse1, mse2, mse3, mse4, mse5, mse6, mse7, mse8, mse9, mse10, mse11, mse12,
mse13,
     mse14, mse15, mse16, mse17, mse18, mse19, mse20, mse21)
REQM<-matrix(eqm, nrow=11, ncol=3)
colnames(REQM)<-c("Alfa", "EQM - Lambda Mínimo", "EQM - Lambda menor DP")

setwd("C:/Users/t_lim/Dropbox/DEE/econometria_machine_learning/dados_r")

stargazer(REQM, type = "text", title="Tabela 5: Comparação dos modelos",out="table4.txt")
```

ANEXO 6 – ÁRVORES DE REGRESSÃO E DE CLASSIFICAÇÃO

Base de dados

```
load("C:/Users/t_lim/Desktop/artigos/verticalizacao/dados_r/planilha_fin
al.RData")


load("C:/Users/t_lim/Dropbox/DEE/Bresnahan/dados_trabalhados/dados_probi
t_ops/final.RData")

final <- final[complete.cases(final), ]

final_1 <- final %>% select(cod_muni, ops_aderido, pop_2019,
perc_idosos,
                            pib_capita, idh, nota_cob_atencao,
leitos_ss, benef_total, leitos_pub) %>%
  mutate(perc_benef=benef_total/pop_2019)

## Normalizar variáveis que vamos usar

medias <- apply(final_1[,3:12], 2, mean)
```

```
dp <- apply(final_1[,3:12],2,sd)

final_1 <- final_1 %>% mutate(norm_ops_aderido=(ops_aderido-
medias[1])/dp[1],
                              norm_pop_2019=(pop_2019-medias[2])/dp[2],
                    norm_perc_idoso=(perc_idosos-medias[3])/dp[3],
norm_pib_capita=(pib_capita/medias[4])/dp[4],
                    norm_idh=(idh-medias[5])/dp[5],
norm_nota_atencao=(nota_cob_atencao-medias[6])/dp[6],
                    norm_leitos_ss=(leitos_ss-medias[7])/dp[7],
norm_leitos_pub=(leitos_pub-medias[9])/dp[9],
                    norm_perc_benef=(perc_benef-medias[10])/dp[10])


final_2 <- final_1[,13:21]
names(final_2) <- c("qtde_ops", "pop_2019", "perc_idosos", "pib_capita",
"idh", "nota_sus", "leitos_ss",
                    "leitos_pub", "perc_benef")
```

Árvores de regressão

```
## Criar uma árvore de regressão



model <- rpart(qtde_ops ~., data = final_2)

rpart.plot(model,
           type = 1,
           leaf.round=1,
           extra=100,
           box.palette="Greens")

previsao <- predict(model, final_2)
previsao <- data.frame(previsao)
final_2 <- bind_cols(final_2, previsao)
final_2 <- final_2 %>% mutate(dif=previsao-qtde_ops, dif2=dif^2,
dif_sq=(dif2)^(1/2))
```

Árvore de classificação

```
## Manter apenas municípios onde não há operadoras e aqueles nas quais
há 1 ou 2 ops


final_3 <- final %>% filter(fator=="zero" | fator=="uma")

## Selecionar variáveis e normalizar

final_3 <- final_3 %>% select(cod_muni, fator, pop_2019, perc_idosos,
                              pib_capita, idh, nota_cob_atencao,
leitos_ss, benef_total, leitos_pub) %>%
  mutate(perc_benef=benef_total/pop_2019)
```

```
medias_3 <- apply(final_3[,4:12], 2, mean)
dp_3 <- apply(final_3[,4:12],2,sd)

final_3 <- final_3 %>% mutate(norm_pop_2019=(pop_2019-
medias_3[1])/dp_3[1],
            norm_perc_idoso=(perc_idosos-medias_3[2])/dp_3[2],
            norm_pib_capita=(pib_capita/medias_3[3])/dp_3[3],
            norm_idh=(idh-medias_3[4])/dp_3[4],
norm_nota_atencao=(nota_cob_atencao-medias_3[5])/dp_3[5],
            norm_leitos_ss=(leitos_ss-medias_3[6])/dp_3[6],
            norm_leitos_pub=(leitos_pub-medias_3[8])/dp_3[8],
            norm_perc_benef=(perc_benef-medias_3[9])/dp_3[9])

final_4 <- final_3[,c(3, 13:20)]
names(final_4) <- c("ops", "pop_2019", "perc_idoso", "pib_capita",
"idh", "nota_sus", "leitos_ss",
                    "leitos_pub", "perc_benef")

## Dividir em conjunto de treinamento e de teste

set.seed(123)
treinamento<-final_4[sample(nrow(final_4), size=1963, replace=FALSE),]
teste<-anti_join(final_4, treinamento, by=NULL)

# Árvore de classificação
```{r}
## Fazer árvore de classificação com o conjunto de treinamento
classif <- rpart(ops ~ ., data=treinamento, method="class",
                 control = rpart.control(xval = 10, minbucket = 50, cp =
0))

# minibucket= número mínimo de obserações em um nó
# cp= parâmetro de complexidade

## Tabela com a proporção de corretos

# Árvore sem poda

rpart.plot(classif,
           type = 1,
           leaf.round=1,
           extra=100,
           box.palette="Greens")

printcp(classif)

poda <- prune(classif, cp = 0.012)

rpart.plot(poda,
           type = 1,
           leaf.round=1,
           extra=100,
```
```

```
           box.palette="Greens")

## Previsão no treinamento

previsao <- predict(classif, treinamento, type="class")
previsao <- data.frame(previsao)
treinamento <- bind_cols(treinamento, previsao)
previsao_1 <- predict(poda, treinamento, type="class")
previsao_1 <- data.frame(previsao_1)
names(previsao_1) <- "prev_poda"
treinamento <- bind_cols(treinamento, previsao_1)

## Previsão no teste

previsao_2 <- predict(classif, teste, type="class")
previsao_2 <- data.frame(previsao_2)
teste <- bind_cols(teste, previsao_2)
previsao_3 <- predict(poda, teste, type="class")
previsao_3 <- data.frame(previsao_3)
names(previsao_3) <- "prev_poda"
teste <- bind_cols(teste, previsao_3)

## Matriz de confusão

# Treinamento sem poda

treinamento <- treinamento %>% mutate(class_0_cor=ifelse(ops=="zero" &
previsao=="zero", 1, 0),
                                      class_0_fal=ifelse(ops=="zero" &
previsao=="uma", 1, 0),
                                      class_1_fal=ifelse(ops=="uma" &
previsao=="zero",1,0),
                                      class_1_cor=ifelse(ops=="uma" &
previsao=="uma",1,0))


res_trei_comp <- apply(treinamento[,12:15], 2, sum)

# Colunas indicam as previsões
res_trei_comp <- matrix(data=res_trei_comp, nrow=2,ncol=2, byrow=TRUE,
                        dimnames=list(c("Zero", "Uma"),c("Zero",
"Uma")))

# Treinamento com poda

treinamento <- treinamento %>% mutate(class_0_cor_p=ifelse(ops=="zero" &
prev_poda=="zero", 1, 0),
                                      class_0_fal_p=ifelse(ops=="zero" &
prev_poda=="uma", 1, 0),
                                      class_1_fal_p=ifelse(ops=="uma" &
prev_poda=="zero",1,0),
                                      class_1_cor_p=ifelse(ops=="uma" &
prev_poda=="uma",1,0))
```

```
res_trei_poda <- apply(treinamento[,16:19], 2, sum)

# Colunas indicam as previsões
res_trei_poda <- matrix(data=res_trei_poda, nrow=2,ncol=2, byrow=TRUE,
                        dimnames=list(c("Zero", "Uma"),c("Zero",
"Uma")))

## Teste completo

teste <- teste %>% mutate(class_0_cor=ifelse(ops=="zero" &
previsao_2=="zero", 1, 0),
                                        class_0_fal=ifelse(ops=="zero" &
previsao_2=="uma", 1, 0),
                                        class_1_fal=ifelse(ops=="uma" &
previsao_2=="zero",1,0),
                                        class_1_cor=ifelse(ops=="uma" &
previsao_2=="uma",1,0))


res_teste_comp <- apply(teste[,12:15], 2, sum)

# Colunas indicam as previsões
res_teste_comp <- matrix(data=res_teste_comp, nrow=2,ncol=2, byrow=TRUE,
                        dimnames=list(c("Zero", "Uma"),c("Zero",
"Uma")))

## Teste com poda

teste <- teste %>% mutate(class_0_cor_p=ifelse(ops=="zero" &
prev_poda=="zero", 1, 0),
                                        class_0_fal_p=ifelse(ops=="zero" &
prev_poda=="uma", 1, 0),
                                        class_1_fal_p=ifelse(ops=="uma" &
prev_poda=="zero",1,0),
                                        class_1_cor_p=ifelse(ops=="uma" &
prev_poda=="uma",1,0))


res_teste_poda <- apply(teste[,16:19], 2, sum)

# Colunas indicam as previsões
res_teste_poda <- matrix(data=res_teste_poda, nrow=2,ncol=2, byrow=TRUE,
                        dimnames=list(c("Zero", "Uma"),c("Zero",
"Uma")))

## Matriz bonitinha
setwd("C:/Users/t_lim/Dropbox/DEE/econometria_machine_learning/dados_r")

stargazer(res_teste_comp, summary=FALSE, rownames=TRUE, title="Tabela 6
- Matriz de confusão - Árvore Completa", out="table6.txt")
 ------------------------- FIM-------------------------------------
```