UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE CIÊNCIAS AGRÁRIAS
DEPARTAMENTO DE ECONOMIA AGRÍCOLA
PROGRAMA DE PÓS-GRADUAÇÃO EM ECONOMIA RURAL

ANGELICA CAITANO DA SILVA

PREVISÃO DA POBREZA DO ESTADO DO CEARÁ

FORTALEZA
2022

ANGELICA CAITANO DA SILVA

PREVISÃO DA POBREZA DO ESTADO DO CEARÁ

Dissertação apresentada ao Programa de Pós-Graduação em Economia Rural da Universidade Federal do Ceará (UFC), como requisito para obtenção do título de Mestre em Economia Rural. Área de concentração: Políticas Públicas e Desenvolvimento Rural.

Orientador: Profº. Drº. Jair Andrade de Araújo
Coorientador: Profº. Draº Guaracyane Lima Campêlo

FORTALEZA
2022

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

S578p Silva, Angélica Caitano da.
Previsão da pobreza do estado do Ceará / Angélica Caitano da Silva. – 2022.
76 f. : il. color.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Ciências Agrárias, Programa de Pós-Graduação em Economia Rural, Fortaleza, 2022.
Orientação: Prof. Dr. Jair Andrade de Araújo .
Coorientação: Profa. Dra. Guaracyane Lima Campêlo .

1. Pobreza. 2. Modelos de predição. 3. Machine learning. I. Título.

CDD 338.1

ANGELICA CAITANO DA SILVA

PREVISÃO DA POBREZA DO ESTADO DO CEARÁ

Dissertação apresentada ao Programa de Pós-Graduação em Economia Rural da Universidade Federal do Ceará (UFC), como requisito para obtenção do título de Mestre em Economia Rural. Área de concentração: Políticas Públicas e Desenvolvimento Rural.

Orientador: Profº. Drº. Jair Andrade de Araújo
Coorientador: Profº. Draº Guaracyane Lima Campêlo

Aprovada em: 03/10/2022.

BANCA EXAMINADORA

_____
Profº. Drº. Jair Andrade de Araújo (Orientador)
Universidade Federal do Ceará (UFC)

_____
Profº. Draº Guaracyane Lima Campêlo (Coorientadora)
Universidade Federal do Ceará (UFC)

_____
Profº. Draº. Andréa Ferreira da Silva
Universidade Federal da Paraíba (UFPB)

_____

Profº. Drº. Filipe Augusto Xavier Lima
Universidade Federal do Ceará (UFC)

A Deus.

Aos meus pais, Cicero Caitano e Maria das Graças.

AGRADECIMENTOS

Senhor, foste tu quem me ensinaste que nada é impossível perante a qualquer dificuldade. Quem acredita no teu amor encontrará o caminho da superação. Por isso, te agradeço por não me abandonares nos momentos mais difíceis.

Aos meus amados pais, Cicero Caitano e Maria das Graças, por acreditarem sempre na educação. Apesar da pouca instrução, nunca desistiram de incentivar a seus filhos a estudar. Por todo apoio, carinho, amor, orações e conselhos que me trouxeram até aqui.

Ao meu irmão Jorge Jefersom, que me deu força durante o período do mestrado.

Ao professor Guilherme Irffi, que, audaciosamente, me atrevo a chamar de amigo, pois o seu papel nessa caminha vai muito além de professor. Agradeço por ter me incentivado a ingressar no mestrado, pelas longas conversas, pelo apoio incondicional nessa caminhada até o título de Mestre.

Ao Isaac Brasil por seu companheirismo, por ser o meu melhor ouvinte, por acreditar e estar junto comigo em parte dessa caminhada até o título de Mestre.

A todos os colegas de turma, em especial à Barbara Braga e ao João Luís, apesar do contexto pandêmico vivido durante o mestrado, foram sempre presentes.

À minha amiga Aline Mendes, por todo apoio, companheirismo e conversas intermináveis.

À Andrea Silva, por todo conhecimento compartilhado e pela sua disponibilidade na hora das dúvidas.

À coorientadora Guaracyane Lima, pela paciência e pelos aprendizados.

Ao meu professor orientador Jair Araújo, por toda a paciência e orientação nesse processo.

À Universidade Federal do Ceará e, em especial, ao Programa de Pós-graduação em Economia Rural pela formação recebida.

A Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001, pelo apoio financeiro recebido com a concessão da bolsa de estudos.

A todos àqueles que direta, ou indiretamente, contribuíram para pesquisa e elaboração desta dissertação.

“A Pobreza não é só falta de dinheiro: é não ter a capacidade de realizar todo o potencial de um ser humano” (Amartya Sem)

RESUMO

Nas últimas décadas, é recorrente a análise da pobreza e dos seus determinantes com o principal intuito de entender o cenário dessa parcela da população que vive com uma renda insuficiente e até mesmo em condições de vida inaceitáveis. O acesso aos dados e às técnicas precisas e atualizadas sobre a pobreza é essencial para que os governos e formuladores de políticas identifiquem as áreas vulneráveis, permitindo-lhes obter conhecimento confiável por meio da ciência de dados. Este estudo utiliza a técnica de M*achine Learning* para fazer a previsão de pobreza com a base de dados da última Pesquisa de Orçamentos Familiares (POF) de 2017-2018, em um recorte para o estado do Ceará. Estimam-se diversos modelos (Regressão Logística, LASSO e Regressão Linear). Dentre os métodos, o que teve maior acurácia foi o método *LASSO*. A Regressão logística teve maior AUC ROC. Entre as conclusões, é possível prever uma taxa de pobres classificados, corretamente de 80,5 % para o modelo logístico, e para *LASSO* 80,8%. Pode-se afirmar que 80% dos indivíduos da base de teste são pobres no estado do Ceará. Ambos os modelos finais tiveram variáveis de impacto parecidas, são elas: lixo, número de pessoas, número de crianças, parede, instrução, telhado, tipo de situação, sexo e idade. Esse resultado é importante, pois sabendo quais variáveis impactam diretamente, pode-se direcionar os investimentos nessas variáveis devido à importância para prever a pobreza no Ceará.

Palavras-chave: pobreza; modelos de predição; *machine learning.*

ABSTRACT

In recent decades, poverty and its determinants have persisted in being analyzed, with the main aim of understanding the scenario of the portion of the population that lives with insufficient income and even in unacceptable living conditions. Access to accurate and up-to-date data and techniques on poverty is essential for governments and policymakers to identify vulnerable areas, allowing them to obtain reliable knowledge through data science. Anticipating poverty is essential so that governments can help in preventing the armed forces of poverty and promoting the reallocation of resources. This study uses the Machine Learning technique to make poverty forecasts based on data from the last Family Organization Survey (POF) from 2017-2018, in a record for the state of Ceará. Various models are estimated (Logistic Regressão, LASSO and Linear Regressão). Of these methods, the one that has the greatest accuracy was the method that was LASSO and the logistical regressão had the greatest AUC ROC. Among the conclusions, it is possible to foresee a correctly classified poor taxa of 80.5% for the logistic model, and for LASSO 80.8%. It can be affirmed that 80% of the individuals on the basis of the test will be poor in the State of Ceará. Both the final models have similar impact variables, they are: type, number of people, number of children, wall, instruction, roof, type of situation, sex and age. This assumption is important because knowing which variables have a direct impact, it is possible to direct the investments that vary because it is known how important they are to anticipate poverty in Ceará.

Keywords: *poverty; prediction models; machine learning.*

LISTA DE TABELAS

LISTA DE GRÁFICOS

LISTA DE QUADROS

LISTA DE ABREVIATURAS E SIGLAS

AUC	Area Under the Curve
AutoML	Aprendizado de Máquina Automatizado
BIRD	Banco Internacional para a Reconstrução e o Desenvolvimento
Cadúnico	Cadastro Único
CEPAL	Comissão Econômica para América Latina e o Caribe
COVID 19	Coronavírus Disease 2019
DHS	Demographic and Health Surveys
DMSP	Defense Meteorological Satellite Program
DNN	Deep Neural Network
FDA	Função de Distribuição Acumulada
FECOP	Fundo Estadual de Combate à Pobreza
FN	Falsos Negativos
FP	Falsos Positivos
GDI	Índice Geral de Privação
GPRBFK	Gaussian Process with Radial Basis
IBGE	Instituto Brasileiro de Geografia e Estatística
IDH	Índice de Desenvolvimento Humano
IETS	Instituto de Estudos do Trabalho e Sociedade
IPEA	Instituto de Pesquisa Econômica Aplicada
IPECE	Instituto de Pesquisa e Estratégia Econômica do Ceará
KNN	K-Nearest Neighbor
LASSO	Least Absolute Shrinkage and Selection Operator
LIME	Local Interpretable Model-agnostic Explanations
MDPI	Índice de pobreza de dados de fontes múltiplas
MI	Múltipla-Imputação
ML	Machine Learning
MPL	Modelo de Probabilidade Linear
MQO	Mínimos Quadrados Ordinários
MSE	Erro Quadrado Médio
NNFE	Neural Network with Feature Extraction
NTL	Luz Noturna

OLS	Operational Linescan System
ONU	Organização das Nações Unidas
PBF	Programa Bolsa Família
PIB	Produto Interno Bruto
PLSRGLM	Partial Least Squares Regression for Generalized Linear Models
PMT	Proxy Means Test
PMT	Proxy Means Test
PNAD	Pesquisa Nacional por Amostra de Domicilio
PNUD	Programa das Nações Unidas para o Desenvolvimento
POF	Pesquisa de Orçamentos Familiares
PPC	Paridade de Poder de Compra
RF	Random Forest
RFR	Regressão Floresta Aleatória
ROC	Receiver Operating Characteristic
RSS	Soma dos Quadrados dos Resíduos
RT	Random Forest
SGB	Stochastic Gradient Boosting
SIS	Síntese de Indicadores Sociais
SVM	Support Vector Machines
SVR	Support Vector Regression
VN	Verdadeiros Negativos
VP	Verdadeiros Positivos
WI	
XGBoost	Extreme Gradient Boosting

SUMÁRIO

1 INTRODUÇÃO

A erradicação da pobreza não só é uma meta global, como também um dos maiores desafios para os países em desenvolvimento. Por sua vez, a previsão da pobreza é essencial para que os governos construam políticas públicas para prevení-la e promover a realocação de recursos de forma eficiente. Destaca-se que um dos Objetivos do Desenvolvimento Sustentável (ODS) é a Erradicação da Pobreza, em que é estabelecida a garantia de que até 2030 todas as pessoas, especialmente os mais pobres e vulneráveis, tenham direitos iguais aos recursos econômicos, além de acesso a serviços básicos, propriedades, novas tecnologias, entre outros.

Nesses termos, o relatório do Programa das Nações Unidas para o Desenvolvimento (PNUD, 2021) aponta a gravidade da pobreza com surgimento da pandemia da Covid-19 acompanhado com baixo crescimento dos países da América Latina. O documento destaca que dentre os países latino-americanos, Brasil, Chile e México possuíram a maior concentração de renda em 2019: os 10% mais ricos de cada país são responsáveis por em torno de 57% da renda nacional. De fato, tal estudo aponta o aumento da pobreza e a preocupação dos organismos internacionais com a situação do Brasil e do mundo frente ao desafio de dar fim à pandemia em curso.

No mais, é salutar destacar que Brasil possui uma heterogeneidade entre as regiões geográficas em relação a diversos requisitos, tais como: renda, gênero, trabalho, desigualdade e pobreza. Esta última proporciona maior risco de exposição e gravidade do surto do vírus, afetando, desproporcionalmente, as populações menos favorecidas que habitam as regiões Nordeste e Norte do país.

Conforme a Síntese de Indicadores Sociais (SIS) do Instituto Brasileiro de Geografia e Estatística IBGE (SIS, 2020), com o uso de dados da Pesquisa Nacional por Amostra de Domicilio (PNAD) Contínua 2019, destaca-se que a região Nordeste representava 27,2% do total populacional do país. Todavia, no contexto das pessoas consideradas extremamente pobres pela linha de internacional do Banco Mundial de US$ 1,90 PPC por dia, mais da metade, ou 56,8%, reside na região.

Em comparação aos dados da PNAD Contínua 2020, o SIS apresenta dados atualizados, levando em consideração o contexto pandêmico e as medidas de proteção social, quando se constatou que a concessão dos benefícios de caráter emergencial durante a pandemia de Covid-19 teve um impacto expressivo no rendimento domiciliar. Consequentemente, sobre as medidas de pobreza, em especial na região Nordeste, a proporção de pessoas em extrema pobreza teve uma queda. Os rendimentos médios nas regiões Norte e Nordeste são, historicamente, inferiores

àqueles observados nas regiões Sul e Sudeste. Com o Auxílio Emergencial, concedido em um valor nacional único, o impacto sobre a renda foi mais expressivo nas regiões Norte e Nordeste, mesmo em um contexto de perda elevada de rendimentos do trabalho em função da pandemia de Covid-19. Em 2020, o percentual de pessoas consideradas extremamente pobres na região Nordeste foi de 49,4% pela linha de US$ 1,90 e 45,5% das consideradas pobres pela linha de US$ 5,50.

O estado do Ceará, localizado na região Nordeste do Brasil, enfrenta o desafio de redução da pobreza e desigualdade de renda ao longo de sua existência. São inúmeros trabalhos acadêmicos que se dedicaram a estudar esses fenômenos no estado, por exemplo: Rodrigues *et al.* (2019), Silva *et. al*., (2021), Araújo (2009). De forma geral, esses autores apontam uma redução da pobreza até meados de 2014.

Silva e Araújo (2021) analisaram os indicadores de pobreza e renda com os dados da PNAD COVID-19, entre maio e novembro de 2020. Os resultados encontrados para a média dos rendimentos reais mensais *per capita* por domicílio no mês de maio foram mais baixos quando comparados a 2019. Teve, porém, uma leve elevação entre julho e setembro, mas voltou a cair em outubro e novembro. Essa queda pode ser explicada, ao menos em parte, pela redução no valor do Auxílio Emergencial, tanto para o Nordeste quanto para o Ceará. Em relação à pobreza e extrema pobreza no Ceará, ocorreu um declínio entre maio e agosto e acresceu novamente após o mês de setembro junto à redução do valor do Auxílio Emergencial.

De acordo com dados do IBGE, o estado do Ceará tem uma população estimada para 2021 de 9.240.580 pessoas. Nesse contexto, houve um rendimento domiciliar *per capita* de 1.028 reais em 2020. O Produto Interno Bruto (PIB) da capital cearense no ano de 2019 foi de 163.575.327 reais.

É importante destacar que o estado possui um importante Fundo Estadual de Combate à Pobreza (FECOP). Por meio de evidências encontradas utilizando a metodologia de controle sintético os autores Silva *et. al.*, (2021), teve um impacto médio na redução da pobreza. Para os autores, os efeitos do FECOP são positivos sobre a diminuição da pobreza. Porém, poderiam ser maiores se houvessem projetos financiados com ações e público-alvo melhor definidos. Apesar da vasta literatura sobre a pobreza no estado do Ceará tanto na literatura nacional quanto internacional, existem poucos estudos sobre previsão da pobreza para a referida unidade da federação.

Um pré-requisito para erradicar a pobreza é identificar com precisão as famílias em situação de pobreza. Isso acontece em todos os países do mundo onde os formuladores de políticas e governos lutam para reduzir a taxa de pobreza de seus países. No entanto, as formas

existentes de encontrar o grupo empobrecido certo para fornecer ajuda econômica geralmente são falhas devido a vários problemas, como transparência de dados e dados incorretos, redundantes ou desequilibrados. Embora existam soluções disponíveis para os problemas, algumas delas são caras e exigem muitos esforços de intervenção humana. Assim, a ciência de dados com a abordagem de aprendizado de máquina é uma solução alternativa para prever e fornecer eficiência de custo e solução eficaz para lidar com problemas de pobreza.

Nesse sentido, quais métodos de *Machine Learning* (ML) podem ser utilizados na previsão de pobreza no Ceará? Este estudo tem como objetivo geral caracterizar a pobreza no estado do Ceará, além do intuito de querer construir modelos de previsão utilizando técnicas de (ML) nos dados obtidos da POF 2017-2018. Destaca-se que a pobreza aqui é definida como parte de um grupo de indivíduos que possuem renda familiar *per capita* insuficiente para atender as necessidades básicas de sobrevivência. Nesse contexto, a elaboração de políticas de redução da pobreza exige ações baseadas em evidências e previsões para proteger os grupos mais vulneráveis.

Para definição da linha de pobreza, foi utilizada como estratégia de identificação o salário mínimo vigente em 2019 no Brasil de R$998,00 reais. Assim sendo, a linha de pobreza deste estudo é de ½ salário mínimo, equivalente a R$499,00 reais. Portanto, os indivíduos com renda menor 499,00 reais são classificados como pobres e indivíduos com renda superior a 499,00 reais são classificados como não pobre.

O ML apresenta boa aplicabilidade em dados de alta dimensão, como classificação, regressão e *clustering*, podendo auxiliar em tomadas de decisões confiáveis e repetíveis. Por essa razão, ele vem sendo aplicado em diversas áreas, como detecção de fraudes, pontuação de crédito, análise da próxima melhor oferta, reconhecimento de voz e imagem ou processamento de linguagem natural (NLP) (JANIESCH; ZSCHECH; HEINRICH, 2021).

Os modelos de classificação e previsão de aprendizado de máquina, *Machine Learning* (ML), exercem um instrumental relevante ao fornecer estimativas precisas de indicadores econômicos como a taxa de pobreza. A aplicação de tecnologia de dados em grande escala e algoritmos de extração de dados para a redução da pobreza podem identificar famílias verdadeiramente pobres com mais rapidez e precisão.

A literatura econômica internacional discute a necessidade de definir e reduzir com precisão a população pobre. Além disso, destaca o uso crescente de algoritmos de ML sobre os problemas de classificação e previsão da pobreza. Os dados de sensoriamento remoto, de comércio eletrônico, de endividamento e mídia social, juntamente com métodos avançados de aprendizado de máquina, foram utilizados para estimar a pobreza das famílias (JEAN *et al.*,

2016; ZHAO *et al*., 2019; LI *et al*., 2019; CHEN E YUAN, 2020; WIJAYA *et al.*, 2020; GUANGZHOU, CHEN E YUAN, 2020; FERREIRA *et al*, 2021).

Adicionalmente, os métodos de aprendizado de máquina podem aprimorar o desempenho preditivo de ferramentas de focalização da pobreza, como o Teste de Elegibilidade Multidimensional (*Proxy Means Test* - PMT). (MCBRIDE E NICHOLS, 2018; KAMBUYA , 2020). Os estudos de Sohnesen and Stender (2017) e Thoplan (2014) destacam o algoritmo *Random Forest* (RF) como mais preciso e de maior acurácia na previsão da pobreza.

No contexto nacional, os trabalhos empíricos de redução de pobreza que abordam os métodos de ML aplicados para classificação e previsão da pobreza ainda são muito escassos. Destaca-se o trabalho de Silva e França (2021), que utilizou os métodos de ML associados à aplicação do teste de elegibilidade de *Proxy Means Test* (PMT) para a classificação da pobreza a partir dos microdados da PNAD contínua (2019). Os autores adotaram uma linha de pobreza baseada no valor estabelecido pelo Banco Mundial, o qual caracteriza indivíduos pobres àqueles cujo a renda domiciliar *per capita* é inferior a US$ 5,50 por dia.

Ademais, os objetivos específicos deste trabalho são: caracterizar, sucintamente, a pobreza no estado do Ceará; identificar na literatura recente os trabalhos sobre previsão de pobreza e os métodos de *Machine Learning*; comparar as melhores previsões encontradas nos modelos tradicionais de econometria, como também nos métodos de algoritmos do *Machine Learning*; por fim, prever, de maneira precoce, a quantidade dos indivíduos pobres para uma amostra da população cearense. Portanto, o presente trabalho tem a finalidade de contribuir com a literatura ao analisar modelos de predição aplicando 11(onze) algoritmos de *Machine Learning* para a previsão do status de pobreza com o uso de dados obtidos da POF 2017-2018 no estado do Ceará. Dentre os principais resultados obtidos, os modelos que tiveram as melhores precisões de previsão considerando os resultados da acurácia e AUC ROC são: *LASSO* e Regressão Logística.

O restante do trabalho está organizado em cinco seções. Na segunda seção, caracteriza-se a pobreza no estado do Ceará. Na terceira, faz-se uma revisão de literatura sobre os modelos de aprendizado de máquina e pobreza. A quarta seção apresenta a metodologia e a base de dados. A quinta seção apresenta as análises de resultados e discussões referentes à comparação, seleção e avaliação dos modelos preditivos. Na última seção, são apresentadas as principais considerações finais.

2 A POBREZA NO ESTADO DO CEARÁ

Esta seção dedica-se a apresentar de forma sucinta a pobreza no estado do Ceará. O estudo da pobreza fez com que muitos pesquisadores dedicassem esforços sobre o tema. Atualmente, um dos grandes problemas do desenvolvimento econômico é entender a razão pela qual um grande contingente de pessoas permanece em extrema pobreza e em atividades econômicas de baixíssimo rendimento. É necessário entender os cenários que essas pessoas sobrevivem e quais barreiras precisam ser superadas para que esses indivíduos saiam da extrema pobreza.

Muitas são as tentativas de entender e explicar como reduzir a pobreza. Alguns economistas apostam que os indivíduos permanecem pobres por problemas nutricionais. Outros acreditam faltar investimento em capital humano, em educação pública de qualidade, ou ainda políticas públicas de transferência de renda. Mas, o que é pobreza?

Entende-se que a pobreza pode ser definida como uma privação de uma renda inferior a um patamar preestabelecido. Para estudos e definição de pobreza, existem os trabalhos de Silva (2015), Vieira (2017), Castro (2011), Crespo e Gurovitz (2002), entre outros.

Para Townsend (1971), a pobreza não só significava não ter o nível mínimo de nutrição ou subsistência, mas também não atingir o padrão prevalecente numa dada sociedade. O estudo da privação das capacidades, que tem como grande referência em seu desenvolvimento o autor indiano Amartya Sen, avança no sentido de ampliar, aprofundar e incorporar outras dimensões na conceituação sobre o que é pobreza. Para Sen (1999), a pobreza pode ser definida como uma privação das capacidades básicas de um indivíduo e não apenas como uma renda inferior a um patamar pré-estabelecido. O autor faz críticas as abordagens que estabelecem a renda como o único critério de análise da condição de pobreza.

Os autores Loureiro, Suliano e Oliveira (2010) fizeram uma análise da pobreza no estado do Ceará com base em diferentes linhas de mensuração, em que a partir do conceito de pobreza se define a linha de pobreza e a linha de indigência. Para os autores, a linha de pobreza é baseada no consumo observado de uma população pobre com algumas de suas principais características, já a linha de indigência analisa os indivíduos que conseguem adquirir, com sua renda monetária, uma cesta de alimentos com quantidade de calorias mínimas para sobreviver, estabelecem um valor absoluto. Logo, as pessoas abaixo dela são consideradas indigentes ou extremamente pobres.

Os autores supracitados encontraram resultados bastante pertinentes em sua pesquisa para o estado do Ceará, levando em conta que uma pessoa é pobre se sua renda domiciliar *per*

capita for inferior a ½ salário mínimo. Indigente ou de extrema pobreza se a sua renda domiciliar *per capita* é inferior a ¼ de um salário mínimo. Logo, em 2008, aproximadamente, 49,9% da população cearense encontrava-se na condição de pobreza e a taxa de extrema pobreza no estado do Ceará era de 21,5% da população. Baseada na linha de pobreza calculada a partir da definição de uma cesta básica regional desenvolvida pela comissão IBGE-IPEA-CEPAL, a taxa de pobreza para o estado do Ceará foi de cerca de 32,7% em 2008.Os autores concluem que as taxas de pobreza e indigência calculadas com base nas frações do salário mínimo são, sistematicamente, maiores dos que as taxas obtidas pelo método da cesta de consumo.

Já os autores Assis, Medeiro e Nogueira (2017) estudaram a extrema pobreza total e infantil no estado do Ceará, referente ao período de 1991 a 2010. Alguns dos resultados encontrados foram que o percentual da população total em situação de miséria foi reduzido. Assim sendo, apresentaram um percentual de 39,76%, em 1991. Passou para 28,11%, no ano 2000, e 14,69%, em 2010, uma redução relativa de 63,05% entre 1991 e 2010. A extrema pobreza infantil saiu de 50,76%, em 1991, para 39,85%, em 2000, chegando a marca de 22,38%, no ano de 2010, uma diminuição relativa de 55,91%.

O autor Araújo (2009) faz uma análise da chamada relação triangular entre pobreza-crescimento-desigualdade. Analisando os indivíduos brasileiros, chegou a resultados parecido com o dos autores Assis, Medeiro e Nogueira (2017). Eles afirmam que regiões com baixo nível inicial de desenvolvimento e uma alta desigualdade inicial apresentam condições menos propícias à redução da pobreza por meio de crescimento da renda. Nesse sentido, é possível concluir que a elevada desigualdade e o baixo nível de desenvolvimento iniciais da maioria dos estados brasileiros são empecilhos para a reversão do quadro de pobreza, via crescimento da renda. Logo, o autor afirma que, independentemente de ser por crescimento econômico ou redução da desigualdade de renda, o baixo nível inicial de desenvolvimento dos brasileiros e a grande desigualdade inicial de renda são barreiras para a diminuição da pobreza.

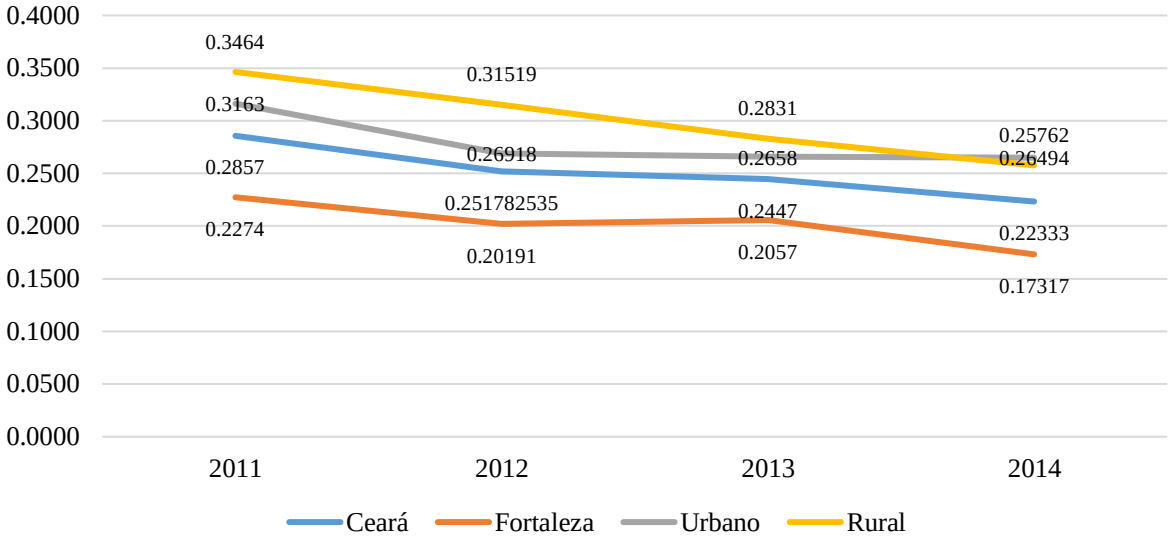
Pode-se salientar que o emprego de políticas de distribuição de renda para uma determinada população que sobrevive em pobreza é uma das soluções sugeridas por alguns autores. Araújo (2009) faz uma revisão literária sobre a relação triangular entre pobreza-crescimento-desigualdade e sobre as relações entre crescimento econômico e pobreza. As implicações apontam evidências de que políticas de combate à pobreza por meio do crescimento são mais eficientes quando combinadas à redistribuição de renda. Já na relação entre pobreza e desigualdade, acredita-se que a pobreza no Brasil é em sua maioria devido à desigualdade existente.

Recentemente, Silva *et. al* (2021) analisaram o impacto da criação do Fundo Estadual de Combate à Pobreza (FECOP) sobre indicadores de pobreza no estado do Ceará, usando o método de Controle Sintético Generalizado, no período de 1981 a 2014. Os resultados encontrados pelos autores admitem um impacto positivo do FECOP/CE com uma trajetória de declínio mais acentuada da pobreza e da pobreza extrema no estado. Os resultados passam a ser, estatisticamente, significantes a partir de 2008. Logo, a FECOP teve um impacto médio ao longo do período estudado de aproximadamente 9,26% sobre o indicador de pobreza e de 12,6% sobre o indicador de extrema pobreza. Os autores também se propõem a verificar se o Programa Bolsa Família (PBF) poderia estar interferindo nos resultados, e perceberam que os efeitos do FECOP encontrados em seu trabalho não estavam sendo afetados pela ausência da variável que representa o repasse do PBF, uma vez que todos os estados participam do PBF. Concluem que os efeitos da FECOP são positivos sobre o estado do Ceará, porém, poderiam ser maiores se houvessem os projetos financiados pelo FECOP com ações e público-alvo melhor definidos.

O Instituto de Estudos do Trabalho e Sociedade (IETS) divulga informações de indicadores de pobreza com os dados da Pesquisa Nacional por Amostra de Domicílios (PNAD). Pode-se observar os indicadores de pobreza para o Ceará por meio do Gráfico 1, a seguir. Ele apresenta a proporção de pobres no Ceará, na capital Fortaleza e nas regiões urbanas e rurais, dos anos de 2011 a 2014. Já no Gráfico 2, estão as quantidades de pobres dos mesmos anos e regiões.

O Gráfico 1 apresenta a proporção de pobres no estado do Ceará, Fortaleza e região Urbana e Rural. Pode-se dizer que ocorreu uma redução na proporção de pobres no Ceará, de 2011 a 2014, de 21, 84%. Em Fortaleza, houve uma redução, de 2011 a 2014, de 23,84%. Na região Urbana, uma redução de 16,23% e, na Rural, a redução foi de 25,62%. Logo, das quatro variáveis mostradas pelo Gráfico 1 a região Rural foi a que teve maior queda da proporção de pobres no período de 2011 a 2014.

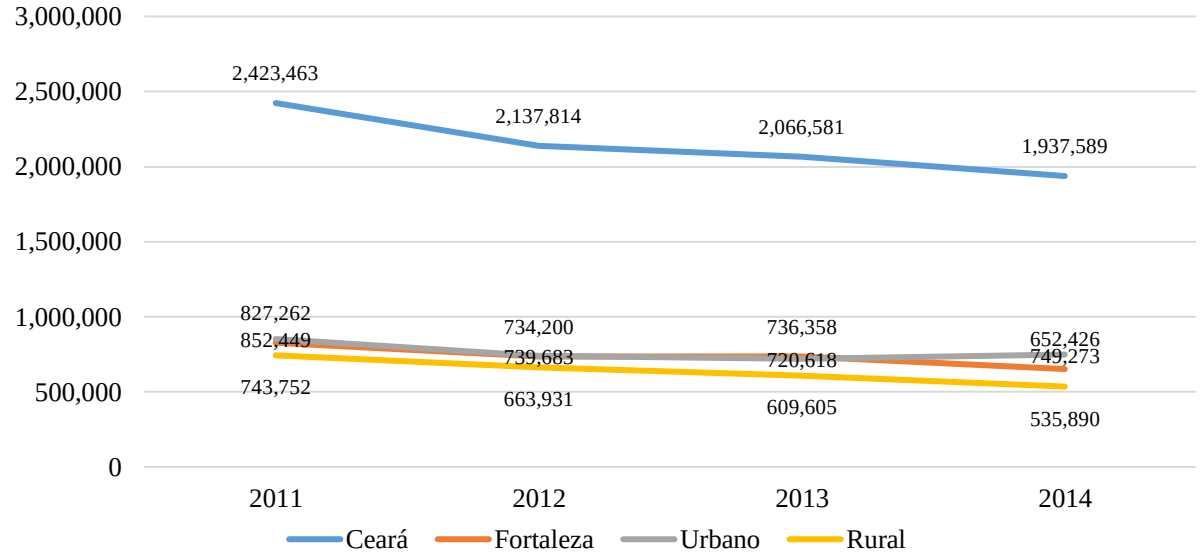
Gráfico 1- Proporção de Pobres no Ceará, no período 2011 a 2014



Fonte: Elaborado pela autora com base nos dados do IETS.

O Gráfico 2 mostra o número de pobres no Ceará, Fortaleza e nas regiões Urbana e Rural. Segundo o IBGE, a população estimada para 2014 foi de 8.842.791 indivíduos. Sendo assim, 21,91% da população cearense era de pobres nesse período. Em Fortaleza, 25,36% dos fortalezenses eram pobres.

Gráfico 2 - Número de pobres no Ceará, no período de 2011 a 2014



Fonte: Elaborado pela autora com base nos dados do IETS.

O Instituto de Pesquisa e Estratégia Econômica do Ceará (IPECE) divulgou a síntese dos indicadores sociais, com a série de dados atualizada para ano de 2019. Os resultados são

com base nos dados da PNAD Contínua e do IBGE, e compreende o período de 2012 a 2019. A partir de agora, apresenta-se os resultados relevantes para o Ceará encontrados nesse documento.

Considerando o rendimento domiciliar *per capita* médio do Ceará, os autores obtiveram uma taxa de crescimento da renda média de 16,7%, crescimento superior ao do Brasil, de 7,8%, e o do Nordeste, de 10,6%. É interessante ressaltar que o documento destaca que, em 2019, o rendimento domiciliar *per capita* médio em Fortaleza foi de R$1.617. Este valor chega a ser mais de 70% superior à média estadual de R$939. Em contrapartida, nas áreas rurais, o valor médio do rendimento domiciliar *per capita* foi de R$425, menos da metade do valor médio estadual (RODRIGUES *et al*., 2021).

Os referidos autores analisaram a taxa de crescimento da renda por décimos da distribuição de renda no primeiro subperíodo 2012-2014 e no segundo subperíodo 2014-2019.No primeiro subperíodo, observou-se um crescimento mais forte da renda nos primeiros decis da distribuição, variação que decresce para patamares de renda maiores e chega a ser negativa entre os 10% mais ricos. O segundo subperíodo apresentou uma variação negativa, dois extratos de renda mais baixos e o maior crescimento sendo observado entre os extratos de maior renda. A renda dos 10% mais ricos cresceu 4,2%, enquanto a renda dos mais pobres caiu 3,7%. Agora, considerando os 40% mais pobres no Ceará o rendimento cresceu a uma taxa maior do que para a média do estado, durante 2016 e 2017. Essa diferença reduz, e nos anos seguintes, ocorre a inversão desse crescimento até 2019. No Ceará, o Índice de Gini denuncia um crescimento da desigualdade de renda a partir de 2015 e segue uma tendência de crescimento até o último ano da série, em 2019 (RODRIGUES *et al.,* 2021).

Pode-se dizer, através das observações deste trabalho, que até final de 2014, antes dos anos da crise de 2015 e 2016, ocorreu uma diminuição das desigualdades de renda, já que nesse período houve um crescimento da renda dos 10% mais pobres e decresceu a renda dos 10% mais ricos. A partir de 2016, através do crescimento do rendimento domiciliar *per capita* médio, apresentou crescimento da desigualdade de renda, pois a renda real média dos mais pobres caiu enquanto a renda nos extratos mais altos apresentou aumento.

Quando os autores analisam o rendimento médio mensal domiciliar *per capita* entre os 10% mais ricos e os 40% mais pobres da população, razão 10/40, os autores Rodrigues *et al.,* (2021) chegam ao resultado de que a desigualdade de renda no Ceará teve o menor valor da razão 10/40 em 2014, com o valor de 14,8, e aumentou de forma acentuada até 2019, quando atingiu o valor de 18,6. Eles verificam a desigualdade de renda entre ricos e pobres através da tabulação da proporção de renda acumulada por grupos de renda. Eles ressaltam que, no Ceará,

em 2019, a parcela populacional correspondente aos 40% mais pobres acumulava o equivalente a 9,8% da renda total, e a parcela correspondente aos 1% mais ricos acumulava quase 15% da renda.

Rodrigues *et al.*, (2021) analisam a pobreza monetária, que mede o percentual de pessoas com rendimento domiciliar *per capita* inferior a um valor específico, chamado linha de pobreza. Os autores levam em consideração uma linha de pobreza de R$150 reais. Assim sendo, em 2019, 12% da população cearense estava vivendo em extrema pobreza, o que representava mais de 1,104 milhões de habitantes nessa condição. Os percentuais dos anos anteriores são de 12,7% em 2012, 12,7% em 2013, 11,9% em 2014, 10,1% em 2015, 10,7% em 2016, 12,4% em 2017, 12,3% em 2018 e 12,2% em 2019. Pode-se dizer que, em 2019, a extrema pobreza no Ceará indicava um patamar menor do que no início da série em 2012, ano que teve o maior percentual de pobreza extrema da população cearense. Vale ressaltar que os autores utilizam de outro critério para estabelecer a linha de pobreza extrema, com base no valor de corte do Bolsa Família, que em 2018 era de R$89 reais per capita.

No estudo, é identificado um percentual da população cearense em extrema pobreza, de 6,5% em 2019. De 2012 a 2015, ocorreu uma queda nos percentuais, mas, em 2016, ele volta a subir e chega a 6,5%, tendo um declínio em 2017 com 6,4%. Em 2018, tem seu maior valor percentual da população pobre no Ceará que foi de 6,7%. Em 2019, ele volta para o patamar de 6,5%. O autor chega a uma conclusão interessante quando discorre sobre a distribuição etária da população em extrema pobreza, quando incide com maiores percentuais entre crianças e adolescentes. Logo, as crianças são mais afetadas, independentemente do critério definido de linha de pobreza, o que torna este grupo prioritário nas estratégias de combate à pobreza no Ceará (RODRIGUES *et al.*, 2021).

3 A PREVISÃO DA POBREZA E OS MODELOS DE MACHINE LEANING

Nesta seção, faz-se uma breve revisão de estudos sobre algoritmos de aprendizado de máquina aplicados nas pesquisas recentes da pobreza. O acesso a dados precisos e atualizados sobre a pobreza é essencial para que os governos e formuladores de políticas identifiquem as áreas vulneráveis, permitindo-lhes obter conhecimento confiável através da ciência de dados e aliviar efetivamente a pobreza.

Os autores Ferreira *et al.,* (2021) examinaram como a inteligência artificial pode contribuir para uma melhor compreensão e para superar o superendividamento em contextos de alto risco de pobreza em Portugal. A metodologia adotada foi o Aprendizado de Máquina Automatizado (AutoML) desenvolvida por Feurer *et al.,* (2015), e aplicada em um banco de dados de 1.654 famílias superendividadas para os anos de 2016 e 2017, com o intuito de identificar clusters distinguíveis e prever seus fatores de risco. Os resultados *de Support Vector Machines* (SVM) indicaram que *Nu-Support Vector Clustering* teve a melhor precisão na previsão de fatores de risco de superendividamento das famílias (89,5%).

Em um estudo para Indonésia, Wijaya *et al.*, (2020) estimaram a taxa de pobreza com base em dados de comércio eletrônico, utilizando técnicas de aprendizado de máquina. O principal conjunto de dados usados foram anúncios de oito mercadorias (carros, motos, casas para venda, casas para alugar, apartamentos para venda, apartamentos para alugar, terreno para venda e terreno para aluguel) postados em 2016 em uma das maiores plataformas de comércio eletrônico do país, OLX (olx.com). Para construir o modelo preditivo foram empregados dois algoritmos: *Deep Neural Network* (DNN) e *Support Vector Regression* (SVR). De acordo com os resultados obtidos, o DNN poderia produzir um modelo de predição melhor do que o SVR quando a seleção de recursos fosse aplicada. Evidenciaram que carros e motos são os dois itens mais significativos para a previsão da pobreza no país.

Verme (2020) comparou o desempenho de modelos econométricos (OLS e logit) e de aprendizado de máquina (r*andom forest* e *LASSO*) na previsão da pobreza a nível familiar usando funções objetivas alternativas e a análise de dominância estocástica com base em curvas de cobertura. Conclui-se que a escolha de um modelo ótimo depende em grande parte da escolha da função objetivo, da distribuição de renda e da linha de pobreza.

Utilizando bases trimestrais da Pesquisa Domiciliar Permanente (EPH) da Argentina, para os anos de 2016, 2018 e 2019, Dabús (2020) utilizou diferentes métodos de aprendizado de máquina para a classificação entre famílias pobres e não pobres com base em preditores não monetários. Dentre os principais resultados obtidos, verificou-se que uma grande proporção das

observações pode ser classificada corretamente sem levar em consideração a variável renda. O modelo de melhor desempenho preditivo foi o r*andom forest*, em que este foi capaz de prever com precisão a condição de pobreza e não pobreza de 84,25% de todas as famílias. Adicionalmente, o método Conditional *random forest* selecionou como os preditores mais relevantes: a quantidade total de membros da família, o tipo de cobertura de saúde do chefe da família, o número de membros da família com mais de 10 anos de idade, a idade do chefe da família, a categoria de inatividade do mesmo e o número de membros com menos de 10 anos de idade.

De acordo com Li *et al.*, (2019), demonstraram que os municípios chineses de alta pobreza e características de classificação robustas podem ser identificados por abordagens de aprendizado de máquina usando apenas imagens noturnas *Operational Linescan System* (OLS), a bordo da constelação de satélites de cobertura global *Defense Meteorological Satellite Program* (DMSP), com dados disponíveis de 2010. As métricas resultantes, incluindo as precisões do usuário (> 63%), do produtor (> 66%) e geral (> 82%) da identificação do município pobre (probabilidade de pobreza maior que 0,6), indicaram que as sete abordagens de aprendizado de máquina usadas (*Gaussian Process with Radial Basis* (GPRBFK), *Stochastic Gradient Boosting* (SGB), *Partial Least Squares Regression for Generalized Linear Models* (PLSRGLM), *Random Forest* (RF), *Rotation Forest* (RoF), *Support vector machine* (SVM) e *Neural Network with Feature Extraction* (NNFE)) apresentaram um bom desempenho, embora existam algumas diferenças entre as abordagens.

Em um estudo de caso para Bangladesh, Zhao *et al.*, (2019), propuseram um modelo de Regressão Floresta Aleatória (RFR) para estimar a pobreza usando dados de luz noturna (NTL), imagens de satélite do Google e dados de cobertura dos solos, estradas e localização da sede da divisão. O Índice de Riqueza (WI) extraído do programa *Demographic and Health Surveys* (DHS) foi usado como a variável dependente do modelo de estimativa da pobreza.

Os principais resultados obtidos destacaram que as variáveis de acessibilidade foram as mais importantes para estimar WI com a importância total de 42,2%, seguida pelas variáveis socioeconômicas (32,6%). Constataram que o método RFR utilizado produziu uma boa acurácia com o uso de dados de várias fontes, além de boa capacidade de lidar com multicolinearidade.

Utilizando um conjunto de dados da Costa Rica, fornecidos pelo Banco Interamericano de Desenvolvimento, Mohamud e Gerek (2019) utilizam um método de previsão da pobreza baseado no conceito de pobreza multidimensional e identificaram os níveis de pobreza usando seleções de recursos dessas observações indiretas e técnicas de aprendizado de máquina.

Através da abordagem do *Proxy Means Test* (PMT) exploraram as características certas que definem cada classe de pobreza (pobreza extrema, moderada, vulnerável e não vulnerável). Para entender como tais características realmente fazem com que uma família seja pobre, empregaram a técnica Local *Interpretable Model-agnostic Explanations* (LIME) e ajustaram o classificador r*andom forest* aos dados. Observaram que, em vez de um conjunto unificado de características, diferentes conjuntos de características são necessários para descrever diferentes níveis de pobreza.

A fim de modelar a classificação da pobreza para a Indonésia com dados de março de 2018, Sihombing e Arsani (2018) aplicaram vários métodos de aprendizado de máquina, incluindo *Decision Tree, Naïve Bayes*, K-*Nearest Neighbor* (KNN) and *Rotation Forest*. Constataram que o modelo, usando a técnica de reamostragem, produz uma classificação melhor do que ao utilizar os dados originais (dados desequilibrados). Além disso, dos quatro modelos de classificação testados, o modelo KNN oferece o melhor desempenho com um valor da acurácia de 0,73%, quando visto do aspecto de sensibilidade, especificidade, valores de área sobre a curva ROC (*Receiver Operating Characteristic*) (AUC, *Area Under the Curve*, também conhecida por c-statistic t) e da estatística *Cohen's Kappa*.

Para prever a pobreza rural e urbana em seis países (Malawi, Ruanda, Etiópia, Uganda, Albânia e Tanzânia), Sohnesen and Stender (2017) usaram os dados de despesas de consumo e os métodos de r*andom forest* (RF), além de uma variante de regressão linear. Esses países não só têm dados de consumo comparáveis para pelo menos dois anos, como também representam uma boa variação no número de anos entre as pesquisas, nível e tendência da pobreza. O estudo descobriu que RF é mais preciso do que o método de Múltipla-Imputação (MI) e apresentou uma maior acurácia na previsão da pobreza para a maioria dos países analisados.

Em um estudo africano, esse algoritmo r*andom forest* também foi implementado por Thoplan (2014) com o propósito de melhorar a exatidão da classificação dos pobres na Mauritânia, usando dados do censo do ano 2000. Verificou-se que a variável mais importante para classificar a pobreza é o número de horas trabalhadas por semana. As três variáveis identificadoras do status de pobreza em ordem decrescente de importância foram: idade, educação e sexo. Além disso, destacou a existência de uma lacuna de gênero, em que as mulheres são mais propensas a serem classificadas como pobres na comparação aos homens.

Nessa mesma linha, com o uso de dados de pesquisas domiciliares (como tamanho da família, idade do chefe da família, materiais de construção da casa e bens materiais) de três países (Bolívia, para o ano de 2005, Timor-Leste, de 2001 e Malawi, de 2004-2005), McBride e Nichols (2018) verificaram que os métodos de *Matching Learning* (Floresta Aleatória e

Floresta Aleatória por Quantil) podem aprimorar a performance preditiva fora da amostra para o desenvolvimento de ferramentas de focalização da pobreza, sendo utilizada o Teste de Elegibilidade Multidimensional (*Proxy Means Test* - PMT).

De forma análoga, utilizando dados da pesquisa domiciliar tailandesa para o ano de 2016, Kambuya (2020) também buscou aperfeiçoar a eficiência e a acurácia do modelo PMT em termos de seleção de variáveis, além do erro de predição fora da amostra, através dos algoritmos *Least Absolute Shrinkage and Selection Operator* (*LASSO*) e *Random Forest* (RF). Dentre os principais resultados auferidos, tem-se que os PMTs com base nas variáveis selecionadas do RF reduziram o número de famílias pobres que são classificadas como não pobres (um erro de exclusão) e aumentaram a taxa de precisão da pobreza nos níveis nacional, urbano e rural. Todavia, o erro de inclusão ainda era alto. Concluíram que se o propósito do programa de bem-estar social é ajudar os pobres, então o PMT com base na seleção variável de Floresta Aleatória seria mais apropriado.

Em um estudo de caso chines realizado em Guangzhou, Chen e Yuan (2020) desenvolveram uma nova abordagem para medir a pobreza urbana usando big data de várias fontes, como dados de mídia social e imagens de sensoriamento remoto para representar as condições sociais e as características dos ambientes construídos. Esses dados foram integrados para construir um índice composto, ou seja, o Índice de pobreza de dados de fontes múltiplas (MDPI), baseado no algoritmo de Floresta Aleatória (RF) e o Índice Geral de Privação (GDI) derivado dos dados do censo como referência para facilitar o treinamento de RF. Os resultados mostraram alta consistência entre o MDPI e o GDI. Ao analisar os resultados do MDPI, apontaram uma autocorrelação espacial, significativamente, positiva na condição de pobreza no nível da comunidade. Combinando imagens de satélite e aprendizado de máquina para prever a pobreza em cinco países africanos - Nigéria, Tanzânia, Uganda, Malawi e Ruanda, Jean *et al*., (2016) mostraram como os algoritmos de Rede Neural Convolucional podem ser treinados para estimar resultados econômicos de imagens de satélite.

Dentre as principais conclusões obtidas, verificaram que o modelo é fortemente preditivo para ambos os gastos médios de consumo das famílias e riqueza de ativos medidos a nível de cluster na maioria dos países analisados. As previsões de Validação Cruzada, baseadas em modelos treinados separadamente para cada país, explicaram 37% a 55% da variação no consumo médio das famílias em quatro países e 55% a 75% da variação na riqueza média dos ativos domésticos em cinco países.

Em um estudo de caso para o Ceará, localizado no Nordeste do Brasil, Silva e França (2021) utilizaram os métodos de *Machine Learning* associados à aplicação do teste de

elegibilidade de *Proxy Means Test* (PMT) para a classificação da pobreza a partir dos microdados da PNAD contínua (2019), além de aperfeiçoar a seleção de beneficiários de políticas de combate à pobreza. A modelagem empírica ajusta um modelo de classificação de domicílios/famílias segundo seu status de pobreza, adotando um algoritmo *Extreme Gradient Boosting* (XGBoost). Outras técnicas também foram empregadas de forma complementar como os métodos de *LASSO* e de Floresta Aleatória (*Random Forest*).

Os referidos autores utilizaram como variáveis preditoras na modelagem: as características referentes ao chefe do domicilio e domicílio. Os resultados obtidos do modelo XGBoost indicaram um modelo de boa performance preditiva com acurácia total de 83%, AUC de 0,91% e precisão de 0,77%.

Quadro 1 - Variáveis utilizadas por outros autores para previsão de pobreza

Autor	**Titulo**	**Fonte**	**Variáveis**
Silva, V. H. M. C.; França, J. M. S.	Modelos de *machine learning* na classificação de pobreza: uma aplicação para o estado do Ceará (2021).	PNADC	Sexo do chefe de domicílio, Idade do chefe de domicílio, Presença de cônjuge do chefe de domicílio, Cor declarada pelo chefe de domicílio, Nível educacional mais elevado do chefe do domicílio, Situação de ocupação do chefe do domicílio no mercado de trabalho, Chefe de domicílio recebe aposentadoria, Localização em área urbana ou rural, Número de pessoas residentes, Número de crianças residentes, Número de adultos residentes, Número de cômodos, Abastecimento de água adequado, Banheiro exclusivo do domicílio, Esgotamento sanitário adequado, Tipo da residência, Propriedade do domicílio, Posse de aparelho de televisão, Posse de refrigerador, Posse de máquina de lavar roupas, Posse de computador, Posse de telefone celular/smartfone, Acesso à internet, Posse de automóvel e Posse de motocicleta.
McBride e Nichols	*Improved poverty targeting through Machine Learning*	Bolivia - Encuesta de Hogares (EH) de 2005	Tamanho da família, Tamanho da família ao quadrado, idade, idade ao quadrado, regiões, rural, sublocação, parede, de tijolos, parede de madeira, piso de terra, piso de cimento, geladeira, rádio, tv, dvd, ventilador, carro, número de camas, número de cozinhas, número computadores e ovelhas
McBride e Nichols	*Improved poverty targeting through Machine Learning: An application to the USAID Poverty Assessment Tools (2015).*	Malawi - Second Integrated Household Survey (IHS2) - 2004-2005	Tamanho da família, Tamanho da família ao quadrado, idade, idade ao quadrado, regiões, rural, nunca se casou, proporção de adultos sem educação, proporção de adultos que sabem ler, número de quartos, piso de cimento, eletricidade, banheiro com descarga, sabonete, cama, bicicleta, música jogador, mesa de centro, ferro, jardim e cabras
		Timor Leste - Timor Leste Living Standards Survey (TLSS) - 2001	Tamanho da Família, Tamanho da Família ao quadrado, idade, idade ao quadrado, regiões, parede de vime, telhado de folha, concreto ou telhado de telha, número de quartos, água privada, água compartilhada, banheiro é uma tigela ou balde, luz elétrica, luz privada, ventilador, número de adultos quem lê, campos agrícolas, número de eixos, número de cestos e número de galinhas.

Fonte: Elaborado pela autora.

Quadro 1 - Variáveis utilizadas por outros autores para previsão de pobreza

(conclusão)

Autor	Titulo	**Fonte**	**Variáveis**
Kambuya, P.	*Better Model Selection for Poverty Targeting through Machine Learning: A Case Study in Thailand (2020)*	SES	Número de membros da família, Chefe de família do sexo feminino, Idade do chefe de família (Ano), Chefe de família é casado, Número de membros ativos da família, chefe de família com ensino fundamental, chefe de família com ensino médio, chefe de família com ensino médio, chefe de família com ensino profissional, chefe de família com ensino superior, Proporção de membros da família com idade < 15 anos , Proporção de membros da família com idade >= 60 anos, Proporção de membros da família com deficiência, Número de quartos, Eletricidade na habitação, Habitação construída com material local, Arrendamento a terceiros, Água potável do poço ou subterrânea, Água potável do rio, vapor, água da chuva etc, Residência não tem banheiro, Usando agachamento, Bicicleta, Motocicleta, Carro, Van ou mini caminhão, Fogão a gás, Fogão de cozinha usando eletricidade, Forno micro-ondas, Panela elétrica, Geladeira, Ferro elétrico, Panela Elétrica, Ventilador Elétrico, Radio, TV LCD ou LED ou Plasma, Reprodutor de vídeo, Máquina de lavar, Ar condicionado, Caldeira de água, Computador, Telefone, Celular, Lâmpada Fluorescente e Fluorescente compacta.
Mohamud, J. H.; Gerek, O. N.	*Poverty Level Characterization via Feature Selection and Machine Learning (2019)*	Inter-American Development Bank (available from Kaggle)	Paredes Boas, Sem nível de educação, Aluguel de quarto, Pós-graduação, material do piso, número de adultos em casa, Paredes boas, Sem Nível de Educação, Aluguel por quarto, Pós-graduação, Educação, Piso é material natural, Patrimônio Não Natural (Privado), Número de adultos em a casa, Idade dos Indivíduos > 65, Desvio padrão de idade, O material da parede externa é de madeira, O material da parede externa é de zinco, Média de anos Educação, Número de crianças na família (0-19), Ensino Fundamental Incompleto, O material no chão é cimento, Média de anos de educação para adultos, não houver banheiro na residência, Paredes boas, Telefone por pessoa no domicílio, Televisão, telhado é bom, Taxa de Dependência, Média de anos de escolaridade e o piso for bom.
Verme, P.	*Which Model for Poverty Predictions? (2020)*	Não Informado	Renda, sexo, idade, estatuto, competências do chefe do agregado familiar, dimensão do agregdo familiar e localização urbano-rural
Dabús, A.	*Pobreza en Argentina : un análisis predictivo utilizando herramientas de Machine Learning (2020)*	a Encuesta Permanente de Hogares (EPH)	Região geográfica, domicilio com mais de 500 mil habitantes; tipo de habitação; material dos interiores da habitação; material do telhado; telhado da casa tiver forro ou forro interior; se a habitação tiver água canalizada; se habitação tiver banheiro; se o banheiro estiver dentro da habitação ou não; se o banheiro tiver vaso sanitário com descarga ou não; se o ralo do banheiro vai para rede pública ou não; se a casa tem cozinha; se a casa tem lavanderia; se tem garagem; regime de posse da habitação; combustível utilizado para cozinhar; o banheiro é de uso exclusivo do domicílio ou compartilhado; número total de cômodos da residência; número de membros do agregado familiar com menos de 10 anos de idade; sexo; estado civil; tipo de cobertura médica; Se você sabe ler e escrever; Se frequenta ou frequentou um estabelecimento de ensino; Se esse estabelecimento for público ou privado; Nível de escolaridade mais elevado alcançado; ocupação; entre outras variáveis.

Fonte: Elaborado pela autora.

No Quadro 1, estão alguns trabalhos que utilizam a metodologia de ML para previsão de pobreza, contém as variáveis utilizadas pelos autores. Esses trabalhos servirão de base para a seleção das variáveis desta dissertação. Salienta-se que o autor Dabus (2020) utiliza do método *LASSO* para reduzir o número de variáveis, e a importância das variáveis é analisada a partir da floresta aleatória e da floresta aleatória condicional. Também o autor Kambuya (2020) utiliza o *LASSO* e a floresta aleatória para melhor selecionar as variáveis do modelo.

4 METODOLOGIA

Esta seção apresenta a metodologia da pesquisa que compreende a definição da base de dados, a descrição das variáveis, além dos modelos econométricos e dos algoritmos de *Machine Learning* utilizados, e, por fim, é definido os critérios de avaliação dos modelos estudados nessa pesquisa.

4.1 Base de dados

Neste trabalho, os dados utilizados são da Pesquisa de Orçamentos Familiares 2017-2018 (POF), realizada pelo IBGE. Nela, estão contidas as informações sobre consumo, renda, gastos, condições de vida da população e abrange a percepção subjetiva da qualidade de vida como o perfil nutricional da população.

A partir da disponibilidade dos dados, os mesmos foram submetidos a um procedimento de organização e limpeza. Fez-se um recorte na base para indivíduos residentes no estado do Ceará. Assim, a base possui uma amostra de 2.674 indivíduos e um total de 79 variáveis.

4.2 Descrição das variáveis

Para definição da linha de pobreza deste estudo, considerou-se o salário mínimo vigente em 2019 no Brasil de R$998,00 reais. Logo, essa linha para este estudo foi de ½ salário mínimo, equivalente a R$499,00 reais. Portanto, os indivíduos com renda até 499,00 reais são classificados como pobres e indivíduos com renda superior a 499,00 reais são classificados como não pobre. A estratégia adotada por essa pesquisa foi criar uma variável binária chamada status que identifica os indivíduos pobres e não pobres. Tal variável é oriunda da variável original da base chamada renda monetária mensal.

No Quadro 2, são apresentadas as descrições das variáveis utilizadas nesta pesquisa. Algumas evidências iniciais também são descritas sobre as variáveis utilizadas no modelo, começando pela variável status que está dividida em indivíduos pobre e não pobres. Destes, no presente estudo, 1.682 são não pobres e 992 são identificados como pobres; 1410 indivíduos são do sexo feminino e 1264 são do sexo masculino; a idade mínima dos indivíduos nessa amostra foi de 12 anos e a máxima foi de 103 anos. A média de idade foi de 49 anos, mediana de 48 anos; 2.163 indivíduos sabem ler e escrever e 511 indivíduos não sabem ler e escrever.

O mínimo de pessoas residentes foi de 1 pessoa. O máximo foi de 16 pessoas e a média de quantidade de pessoas residentes foi de 3 pessoas; 167 indivíduos não possuem paredes em sua residência e 2.507 possuem paredes; o número mínimo de crianças na residência foi 0, o máximo foi de 6 crianças e a média foi de 5 crianças; 2.300 indivíduos têm coleta de lixo no seu bairro e 374 não têm; 2.154 indivíduos possuem energia elétrica e 520 não possuem. Por fim, identificou-se residências de um cômodo a 18 cômodos, sendo a média de 6 cômodos.

Quadro 2 - Descrição das Variáveis

Variáveis	**Descrição**
Dependente:	
Status	*Dummy:* assume 1 para pobre, e 0, para não pobre
Explicativas:	
Sexo	*Dummy*: assume 1 para homem, e 0, para mulher
Idade	De 12 a 103 anos
Ler e Escreve	*Dummy:* assume 1 se sabe ler, e 0, se não souber ler
Número de Pessoas	O número de pessoas residentes varia de 1 a 16 pessoas
Número de Adultos	O número de adultos varia de 0 a 9 adultos
Número de criança	O número de crianças varia de 0 a 6
Número de Idosos	O número de idosos varia de 0 a 5
Situação do Domicílio	*Dummy*: assume 1 se Urbana, e 0, se não
Domicilio Próprio	*Dummy*: assume 1 se sim, e 0, se não
Parede	*Dummy*: assume 1 se possui paredes adequadas, e 0, se não
Sem instrução	*Dummy*: assume se tiver instrução, e 0, se não tiver
Energia	*Dummy*: assume 1 se tiver energia elétrica, e 0, se não tiver
Lixo	*Dummy*: assume 1 se tiver coleta de lixo, e 0, se não
Cômodos	O número de cômodos varia de 1 a 18 cômodos
Instrução	Assume 1 se sem instrução; 2 ensino fundamental incompleto; 3 ensino fundamental completo; 4 ensino médio incompleto; 5 ensino médio completo; 6 ensino superior incompleto; 7 ensino superior completo.
Raça	*Dummy*: assume 1 se branco, e 0, se não
Telhado	*Dummy*: assume 1 se possui telhado adequado, e 0, se não

Fonte: Elaborado pela autora com base nos dados da POF 2017-2018.

4.3 Modelo econométrico

Esta seção será dedicada não só às abordagens empíricas dos modelos tradicionais de econometria, como também aos algoritmos do ML, aprendizado de máquina. A presente pesquisa propõe-se a trabalhar com modelo de regressão de resposta qualitativa, com uma variável de resposta binária, em que 0 é não pobre e 1 é pobre.

Vale destacar que, de acordo com James *et al.*, (2013), a econometria convencional é focalizada na construção do modelo. Refere-se à inferência, isto é, a entender a relação entre as variáveis. Os modelos de ML têm a preocupação focada no poder preditivo, ou seja, prever uma resposta de interesse (variável de saída) a partir das variáveis de entradas.

4.3.1 Probabilidade linear

Para estimar um modelo de regressão linear em que a variável de resposta Y é binária (não pobre e pobre), a expectativa condicional de Y dado o vetor X, $E(Y|X_i)$ deve ser interpretada como a probabilidade da pessoa ser pobre ou não, cujo vetor das variáveis explicativas é X. Logo, deve-se usar o Modelo de Probabilidade Linear (MPL), porque a probabilidade de resposta é linear nos parâmetros β_j. Ele mede as mudanças na probabilidade de sucesso quando Xj muda, mantendo fixo os fatores. Quando Y é uma variável binária, ou seja, quando a variável dependente assume somente dois valores, adota-se o Modelo de

Probabilidade Linear. De acordo com Woodridge (2011), para probabilidade linear, os interceptos não assumem mais as mudanças em Y devido ao aumento de uma unidade de X, pois Y muda somente de 0 para 1. Admite-se que a hipótese de média condicional 0 é válida. Logo, $E(X_i) = 0$. A função pode ser representada da seguinte maneira:

$$E(X) = \beta_1 + \beta_2 X \quad (1)$$

Quando Y é uma variável binária que assume valores de 0 e 1, é sempre verdade que P(Y=1|X) = E(Y|X): a probabilidade de sucesso, sendo Y = 1 é a mesma do valor esperado de Y. Tem-se a equação:

$$P(Y=1|X) = \beta_1 + \beta_2 X \quad (2)$$

Se P(X) = P(Y=1|X) é uma função linear de Xj e a soma das suas probabilidades deve ser igual a 1. Logo, P(Y=0|X) = 1 – P(Y=1|X) também é uma função linear em Xj.

4.3.2 Modelo logit

Pode-se usar a Função de Distribuição Acumulada (FDA) para modelar regressões em que a variável resposta é dicotômica, assumindo valores de 0 e 1. Normalmente, ela é a função escolhida para representar os modelos logit e probit. De acordo com Porter e Gujarati (2011), o MPL é Pi = β1 + β2Xi, quando os valores assumidos por X são as variáveis explicativas, e Pi = E(Yi = 1|X1) é a probabilidade de ser pobre ou não, podendo ser representada também por:

$$P_i = \frac{1}{1+e^{-(\beta_1+\beta_2 X_i)}} \quad (3)$$

Ao escrever a função acima de outra forma, em que Zi = β1 + β2Xi, tem-se a Função de Distribuição Logística (acumulada):

$$P_i = \frac{1}{1+ e^{-Z_i}} = \frac{e^Z}{1+ e^Z} \quad (4)$$

em que Zi varia de -∞ a +∞ e Pi varia entre 0 e 1, quando Pi está relacionado não linearmente a Zi.. No entanto, não se permite usar o procedimento de MQO, visto que, além de Pi ficar não linear em Zi,, também fica não linear em β. Se Pi é a probabilidade de ser pobre, (1-Pi) é a probabilidade de não ser pobre, logo:

$$1 - P_i = \frac{1}{1+ e^{Z_i}} \quad (5)$$

Ao reescrever a função, tem-se:

$$\frac{P_i}{1- P_i} = \frac{1+ e^{Z_i}}{1+ e^{-Z_i}} = e^{Z_i} \quad (6)$$

Assim, tem-se que $\frac{P_i}{1- P_i}$ é a razão de chances em favor de ser pobre, ou seja, a razão da probabilidade do indivíduo ser pobre contra a probabilidade de não ser pobre. Se adotar o logaritmo natural da equação (6), L será o logaritmo de chances e passa a ser não apenas linear em X, como também linear nos parâmetros. O L é o logit, logo, denomina-se como modelo logit a representação abaixo:

$$L_i = ln\left(\frac{P_i}{1- P_i}\right) = Z_i \quad (7)$$

4.3.3 Modelo probit

A Função de Distribuição Acumulada (FDA) é utilizada para explicar o comportamento de uma variável dependente dicotômica. Para o modelo logit, usa-se a função logística acumulada, ainda existe a FDA normal. O modelo de estimação que emerge da função de distribuição acumulada normal (FDA) é conhecido como modelo probit. O modelo probit levará

em conta a teoria da utilidade tendo por base o preceito de que os agentes fazem escolhas de forma racional. A decisão do i-esimo indivíduo pobre ou não pobre depende de um índice de utilidade não observável Ii, também conhecido como variável latente, que é determinado por uma ou mais variáveis explanatórias. De tal modo que quanto maior for o valor do índice Ii, maior a probabilidade de ocorrência de Y (PORTER e GUJARATI, 2011). Essa variável pode ser determinada pela equação abaixo:

$$I_i = \beta_1 + \beta_2 X_i \quad (8)$$

Pode-se dizer que Ii, conhecido como variável latente, é determinado por um ou mais variáveis explicativas Xi, de forma que quanto maior o valor de Ii, maior a probabilidade de ser pobre (PORTER; GUJARATI, 2011).

De acordo com Potter e Gujarati (2011), seja Y = 1 se o indivíduo é pobre e Y = 0 se não pobre, pode-se supor que existe um nível crítico ou limiar do índice, chamado de I_i^*, tal que, se Ii exceder I_i^*, o indivíduo será pobre, caso contrário, não será pobre. O limiar I_i^*, como Ii, não é observável. Suponha-se que ele distribui normalmente com a mesma média e variância, é aceitável não apenas estimar os parâmetros do índice dado, mas obter algumas informações sobre o próprio índice não observável. Seguindo a hipótese da normalidade, a probabilidade de que I_i^* seja menor ou igual a Ii pode ser calculada pela FDA normal padronizada da seguinte forma:

$$P_i = P(Y = 1|X) = P\ (I_i^* \leq\ I_i) = P(Z_i\ \leq\ \beta_1 + \beta_2 X_i) = F(\beta_1 + \beta_2 X_i) \quad (9)$$

Logo, $P(Y = 1|X)$ indica a probabilidade de um evento ocorrer dado os valores das variáveis explicativas X, e em que Zi é a variável normal padrão, assim sendo, $Z \sim N(\ 0, \sigma^2)$. E F é a FDA normal padrão, escrita como:

$$F(Ii) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{Ii} e^{-Z^2/2}\, dz \quad (10)$$

Para se obter as informações sobre o índice de utilidade Ii, bem como sobre os $\beta_1\ e\ \beta_2$, toma-se o inverso da Equação (10) para conseguir:

$$I_i = F^{-1}(I_i) = F^{-1}(P_i) \quad (11)$$
$$= \beta_1 + \beta_2 X_i$$

onde F^{-1} é o inverso da FDA normal (PORTER e GUJARATI, 2011).

4.4 Algoritmo de machine learming

O algoritmo de *Machine Learning* (ML) é área de estudo interessada no desenvolvimento de algoritmos computacionais para transformar dados em ações inteligentes,

conhecido como aprendizado de máquina. Este campo originou-se em um ambiente onde os dados disponíveis, métodos estatísticos e poder de computação evoluem rapidamente e simultaneamente. O crescimento nos dados exigiu mais poder de computação, que por sua vez, estimulou o desenvolvimento de métodos estatísticos para analisar grandes conjuntos de dados. Isso criou um ciclo de avanço, permitindo dados ainda maiores e mais interessantes a serem coletados. O ML destina-se a ensinar computadores como usar dados para resolver um problema (LANTZ, 2015).

Para Mahesh (2020), o ML sendo utilizado constantemente por conta da abundância de dados disponíveis e por ser bastante eficiente. Além de contar com diferentes algoritmos para resolver problemas de dados, o tipo de algoritmo empregado está amarrado ao tipo de problema que se espera resolver, do número de variáveis, do tipo de modelo que melhor se adequa a ele e assim por diante. De acordo com Lantz (2015), para aplicar o aprendizado, precisa-se de um processo de cinco etapas. Independentemente da tarefa, qualquer algoritmo de aprendizado de máquina pode ser implantado seguindo estas etapas: coleta de dados; exploração e preparação de dados; treinamento e avaliação do modelo.

O ML tem como objetivo aplicar um método de aprendizagem estatística aos dados de treinamento, a fim de estimar a função desconhecida *f*. Em outras palavras, espera-se encontrar uma função $\hat{f}$ de tal modo que $Y \approx \hat{f}(X)$ para qualquer observação (X, Y). De um modo geral, a maioria dos métodos de aprendizagem estatística pode ser caracterizada como paramétrico ou não paramétrico (JAMES *et al.*, 2013).

A aprendizagem estatística faz referência a um conjunto de abordagens para estimar *f*. Existem duas razões pelas quais se espera estimar *f*: predição e inferência. Em predição, há algumas situações em que um conjunto de entradas *X* está prontamente disponível, mas a saída *Y* não pode ser obtida facilmente. Sendo assim, uma vez que a média do termo de erro é 0, deve-se prever *Y* usando $\hat{Y} = \hat{f}(X)$, quando $\hat{f}$ representa a estimativa para *f* e $\hat{Y}$ representa a previsão resultante para *Y*. Nesse cenário, $\hat{f}$ é, frequentemente, tratado como uma caixa preta, no sentido de que normalmente não se está preocupado com a forma exata de $\hat{f}$ desde que produza previsões precisas para *Y*. A precisão $\hat{Y}$ como uma previsão para *Y* depende de duas quantidades, que pode ser chamada de erro redutível e erro irredutível. Em geral, $\hat{f}$ não será uma estimativa perfeita para *f*, e essa imprecisão irá introduzir alguns erros, erro redutível e erro irredutível. O erro é redutível quando se pode melhorar potencialmente a precisão de $\hat{f}$ usando a técnica de aprendizagem estatística mais apropriada para estimativa *f*. Porém, *Y* também está relacionado ao termo de erro, ε, que, por definição, não pode ser previsto usando

X. Portanto, a variabilidade associada a ε também afeta a precisão das previsões. Isso é conhecido como erro irredutível, porque não importa o quão bem se estima *f*, não é possível reduzir o erro introduzido por ε (JAMES *et al.*, 2013).

Em inferência, o objetivo não é necessariamente fazer previsões para *Y*, é preciso entender o relacionamento entre X e Y, ou mais especificamente, para entender como *Y* muda em função de $X_1,...,X_p$. Assim, $\hat{f}$ não pode ser tratada como uma caixa preta, porque é essencial saber sua forma exata (JAMES *et al.*, 2013). O ML pode ser classificado como aprendizagem supervisionado e não supervisionado. Na aprendizagem supervisionada, tem-se o papel de aprender uma função que mapeia uma entrada para uma saída com base em pares de entrada-saída exemplificados, em que o conjunto de dados de entrada é dividido em conjunto de dados de treinamento e teste. Ela permite que todos os algoritmos aprendam algum tipo de padrão a partir de dados de treinamento e os apliquem em dados testes desconhecidos para previsão ou classificação (MAHESH, 2020).

Muitos métodos clássicos de aprendizagem estatística, como regressão linear e regressão logística, bem como abordagens mais modernas, como GAM, *boosting* e máquinas de vetores de suporte (SVM), operam no domínio de aprendizagem supervisionada. Essa pesquisa também é focada em aprendizagem supervisionada. Para cada observação da (s) medição (ões) do preditor X_i, $i = 1, ..., n$, existe uma medida de resposta associada Y_i. Espera-se ajustar um modelo que relacione a resposta aos preditores com o objetivo de prever com precisão a resposta para observações futuras ou melhor compreender a relação entre a resposta e os preditores (inferência) (JAMES *et al.*, 2013).

Nos modelos preditivos, deseja-se receber instruções claras sobre o que precisam aprender e como devem ser aprendidos. O processo de treinamento de um modelo preditivo é conhecido como aprendizado supervisionado. A aprendizagem não supervisionada é uma situação um pouco mais desafiadora em que, para cada observação $i = 1, ..., n$, é observado um vetor de medida X_i, porém com nenhuma resposta associada Y_i (LANTZ, 2015).

A aprendizado de máquina supervisionada é frequentemente empregada para prever em qual categoria um modelo pertence, ela é conhecida como classificação. Na classificação, a característica alvo a ser prevista é uma característica categórica conhecida como a classe, e é dividida em categorias chamadas de níveis. Uma classe pode ter dois ou mais níveis, e os níveis podem ser ou não ser ordinais. Devido vasta utilização da classificação no aprendizado de máquina, existem muitos tipos de algoritmos de classificação, com pontos fortes e fracos que se adequam para diferentes tipos de dados de entrada (LANTZ, 2015).

De acordo com James *et al.,* (2015), as decisões sobre a precisão do modelo têm se concentrado na configuração da regressão. No entanto, muitos dos conceitos encontrados, como o trade-off de bias-variância, são transferidos para a configuração de classificação com apenas algumas modificações devido ao fato de que y_i não é mais numérico. Veja a equação abaixo:

$$\frac{1}{n}\sum_{i=1}^{n} I\,(y_i \neq \hat{y}_i) \qquad (12)$$

A equação acima é conhecida como taxa de erro de treinamento, porque é calculada com base nos dados que foram usados para treinar o classificador, interessando as taxas de erro que resultam da aplicação do classificador para testar observações que não foram usadas no treinamento (JAMES *et al.*, 2013). A taxa de erro de teste associada a um conjunto de observações de teste da forma (x_0, y_0) é dada por:

$$Ave(I\,((y_0 \neq \hat{y}_0)) \qquad (13)$$

em que $\widehat{y_0}$ é o rótulo de classe previsto que resulta da aplicação do classificador à observação de teste com o preditor x_0. Um bom classificador é aquele para o qual o erro de teste é o menor (JAMES *et al.*, 2013).

Lantz (2015) lista alguns tipos de aprendizado de máquina, divididos em algoritmos de aprendizagem supervisionada e não supervisionada, as supervisionadas são*: naive bayes, nearest neighbor, decision trees, classification rule learners, linear regression, regression trees, model trees, neural networks* e *support vector machines*. Os algoritmos de aprendizagem não supervisionados são: *association rules* e *k-means clustering*. E, por último, os algoritmos de meta-aprendizagem, que são: *bagging, boosting* e *random forests.*

Na etapa de aprendizado do algoritmo de ML, Silva (2019) afirma que ele pode ser dividido nas seguintes categorias: lineares (regressão linear e regressão logística); não lineares (*K – nearest neighbors, naïve bayes classifier, neural network e support vector machines*); e modelos baseados em árvores de decisão (*regression trees, classification trees, bagging, random forest e gradiente boosting*). Para esta pesquisa, foca-se no estudo do método linear com a regressão linear e a regressão logística; o modelo não linear K – *nearest neighbors* (KNN*), support vector machines* (SVM) e os modelos de árvore de decisão de *regression trees, classification tree*.

4.4.1 Modelos lineares

Nesta seção, é apresentado uma descritiva dos conceitos de regressão linear e da regressão logística, que são métodos utilizados neste estudo.

4.4.1.1 Regressão linear

A regressão linear é a abordagem mais simples para aprendizagem supervisionada. É um ponto inicial, para abordagens mais recentes, usada para prever uma resposta quantitativa para Y com base na variável explanatória X. Matematicamente, tem-se:

$$Y \approx \beta_0 + \beta_1 X \quad (14)$$

Os β_0 e β_1 representam o interceptar e a inclinação dos termos no modelo, conhecidos como parâmetros. Quando usado os dados de treinamento para produzir estimativas de $\widehat{\beta_0}$ e $\widehat{\beta_1}$, pode-se prever $\hat{y}$ com base em x. Onde $\hat{y}$ indica uma previsão de Y com base em X = x. Logo, $\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$.

Estima-se a equação (14) pelo Método dos Mínimos Quadrados Ordinários. Assim, supondo-se que $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$ seja a previsão para Y baseado no i valor de X, então ei $= y_i$ - $\hat{y}_i$, que representa o resíduo i, é a diferença entre o i-ésimo valor de resposta observado e o i-ésimo valor de resposta que é previsto por nosso modelo linear.

Conforme detalhado em JAMES *et al.*, (2013), a Soma dos Quadrados dos Resíduos (RSS) é definida por:

$RSS = e_1^2 + e_2^2 + \cdots + e_n^2$, substituindo ei $= y_i$ - $\hat{y}_i$, logo:

$$RSS = \left(y_1 - \hat{\beta}_0 + \hat{\beta}_1 x_1\right)^2 + \left(y_2 - \hat{\beta}_0 + \hat{\beta}_2 x_2\right)^2 + \ldots + \left(y_n - \hat{\beta}_0 + \hat{\beta}_n x_n\right)^2. \quad (15)$$

O Método de Mínimos Quadrados escolhe $\widehat{\beta_0}$ e $\widehat{\beta_1}$ para minimizar o RSS, assim sendo:

$$\widehat{\beta_1} = \frac{\sum_{i=1}^{n}(x_1 - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \text{ e } \widehat{\beta_0} = \bar{y} - \hat{\beta}_1 \bar{x} \quad (16)$$

em que as médias são denotadas por: $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$, $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ (17)

Quando se rejeita a hipótese nula em favor da hipótese alternativa, é natural querer quantificar até que ponto o modelo se ajusta aos dados. A qualidade de um ajuste de regressão linear é, normalmente, avaliada usando duas quantidades relacionadas: o erro padrão residual (RSE) e o R2. Associa-se a cada observação um termo de erro. Devido à presença desses termos de erro, mesmo que se conheça os β_0 e β_1 , não seria possível prever perfeitamente Y a partir de X. O RSE é uma estimativa do desvio padrão de ε, ou seja, é a quantidade média que a resposta irá desviar da linha de regressão verdadeira (JAMES *et al.*, 2013). Calculada da seguinte maneira:

$$RSE = \sqrt{\frac{1}{n-2} RSS} \text{ ou ainda, } RSE = \sqrt{\frac{1}{n-2}\sum_{i=1}^{n}(y_i - \bar{y})^2} \quad (18)$$

Se as previsões obtidas usando o modelo estiverem muito próximas dos verdadeiros valores do resultado, então RSE será pequeno e se pode concluir que o modelo se ajusta aos dados muito bem. Sendo assim, se $\hat{y}_i$ estiver muito longe de y_i para uma ou mais observações, o RSE pode ser muito grande, indicando que o modelo não se ajusta bem aos dados (JAMES *et al.*, 2013).

Já a estatística R2 fornece uma medida alternativa de ajuste. Tem a forma de uma proporção, a proporção da variância explicada, que sempre assume um valor entre 0 e 1, e é independente da escala de Y (JAMES *et al.*, 2013).

$$R^2 = \frac{\text{TSS}-\text{RSS}}{\text{TSS}}, \quad (19)$$

Em que $TSS = \sum(y_i - \bar{y})^2$ mede a variância total na resposta Y e pode ser considerado como a quantidade de variabilidade inerente à resposta antes que a regressão seja realizada. O TSS − RSS mede a quantidade de variabilidade na resposta que é explicada (ou removida) realizando a regressão, e R2 mede a proporção de variabilidade em Y que pode ser explicada usando X. Uma estatística R2 próxima de 1 indica que uma grande proporção da variabilidade na resposta foi explicada pela regressão (JAMES *et al*., 2013).

Em modelos com muitas covariáveis, o método dos mínimos quadrados não conduz a bons resultados devido ao super-ajuste e à variância muito alta, quando existe muitas covariáveis há muitos parâmetros a serem estimados e, portanto, a função de regressão estimada em geral possui baixo poder preditivo, ou seja, a variância do estimador resultante é alta pois muitos parâmetros devem ser estimados. Uma saída para tal problema é a de retirar algumas das variáveis da regressão, diminuindo a variância da função de predição estimada. Um modo de encontrar um estimador linear da regressão quando se tem muitas covariáveis são os Métodos de s*hrinkage* (IZBICKI; SANTOS, 2019).

4.4.1.1.1 Método de shrinkage

Normalmente, ao se ajustar o modelo de regressão linear usando Mínimos Quadrados, há algumas maneiras pelas quais o modelo linear simples pode ser melhorado, substituindo o ajuste de mínimos quadrados por alguns procedimentos de ajustes alternativos. Os procedimentos de ajustes alternativos podem render melhor precisão de previsão e interpretabilidade do modelo (JAMES *et al*., 2013). De acordo com Hastie *et al.*, (2009), existem duas razões pelas quais muitas vezes não se satisfaz as estimativas de Mínimos

Quadrados, sendo elas: i) a precisão da previsão, pois as estimativas de mínimos quadrados geralmente têm baixa polarização, mas grande variação.

A precisão da previsão, às vezes, pode ser melhorada diminuindo ou definindo alguns coeficientes como zero. Ao fazer isso, sacrifica-se um pouco de viés para reduzir a variância dos valores previstos e, portanto, pode-se melhorar a precisão geral da previsão. ii) a segunda razão é a interpretação. Com um grande número de preditores, geralmente, determina-se um subconjunto menor que exibe os efeitos mais fortes. Para obter o "quadro geral", dispõe-se a sacrificar alguns dos pequenos detalhes.

Existem muitas alternativas, clássicas e modernas, para ajustar os Mínimos Quadrados. Em James *et al.*, (2013), os autores discutem três classes importantes de métodos, o *subset selection, shrinkage* e o *dimension reduction*, mas aqui se evidencia *shrinkage.* Esta abordagem envolve o ajuste de um modelo englobando todos p preditores. No entanto, os coeficientes estimados são reduzidos a zero em relação às estimativas de Mínimos Quadrados. Este *shrinkage* (também conhecido como regularização) tem o efeito de reduzir a variância. Dependendo do tipo de encolhimento realizado, alguns dos coeficientes podem ser estimados como exatamente zero. Portanto, os métodos de *shrinkage* também podem realizar seleção de variável (JAMES *et al.,* 2013).

Os métodos de seleção de subconjunto envolvem o uso de mínimos quadrados para ajustar um modelo linear que contém um subconjunto de preditores. Como alternativa, ajusta-se um modelo contendo todos os preditores p usando uma técnica que restringe ou regulariza as estimativas de coeficiente. Ou, de forma equivalente, reduz as estimativas de coeficiente para zero. Pode não ser imediatamente óbvio, porque essa restrição deve aperfeiçoar o ajuste, mas se verifica que reduzir as estimativas dos coeficientes pode reduzir significativamente a sua variância. As duas técnicas mais conhecidas para reduzir os coeficientes de regressão até zero são a regressão de *ridge* e o *LASSO* (JAMES *et al.*, 2013).

Hastie *et al.,* (2009) discorrem que a regressão linear para previsões, como já dito, pode produzir alta variância. Ao reter um subconjunto dos preditores e descartar o resto, com método seleção do subconjunto, produz-se um modelo que é interpretável e tem possivelmente menos erro de predição do que o modelo completo. No entanto, por ser um processo discreto, as variáveis são mantidas ou descartadas, muitas vezes exibe alta variância e, portanto, não reduz o erro de previsão do modelo completo. Os métodos de *shrinkage* são mais contínuos e não sofrem tanto com a alta variabilidade, assim, esses são mais indicados.

A regressão de *ridge* é muito parecida com os mínimos quadrados, exceto que os coeficientes são estimados minimizando uma quantidade ligeiramente diferente. Em particular,

as estimativas de coeficiente de regressão de *ridge*, $\hat{\beta}^R$, são os valores que minimizam, em que $\lambda \geq 0$ é um parâmetro de ajuste a ser determinado separadamente.

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}\beta_j^2 = RSS + \lambda\sum_{j=1}^{p}\beta_j^2 \quad (20)$$

Assim como acontece com os mínimos quadrados (equação 18), a regressão de *ridge* procura estimativas de coeficientes que se ajustem bem aos dados, tornando o RSS pequeno. No entanto, o segundo termo, $\lambda\sum_{j=1}^{p}\beta_j^2$, também chamado de *shrinkage Penalty*, é pequena quando $\beta_1 \ldots, \beta_p$ são próximos de zero e, logo, tem o efeito de reduzir as estimativas de β_j para zero (JAMES *et al.*, 2013).

O parâmetro de ajuste λ serve para controlar o impacto relativo desses dois termos nas estimativas do coeficiente de regressão. Quando $\lambda = 0$, o termo de penalidade não tem efeito, e regressão de *ridge* irá produzir as estimativas de mínimos quadrados. No entanto, à medida em que $\lambda \to \infty$, o impacto da *shrinkage penalty* aumenta, e as estimativas do coeficiente de regressão de *ridge* se aproximam de zero (Dabus, 2020). Ao contrário dos mínimos quadrados, que geram apenas um conjunto de estimativas de coeficiente, a regressão de *ridge* irá produzir um conjunto diferente de estimativas de coeficiente, $\hat{\beta}_\lambda^R$, para cada valor de λ. Observa-se que na equação (20) a *shrinkage penalty* é aplicada a $\beta_1 \ldots, \beta_p$, mas não para o intercepto β_0 (JAMES *et al.*, 2013).

É esperado que haja redução da associação estimada de cada variável com a resposta. No entanto, não se quer diminuir a interceptação, que é simplesmente uma medida do valor médio da resposta quando $x_{i1} = x_{i2} = \cdots = x_{ip} = 0$. Adotando que as colunas da matriz de dados X foram centralizadas para ter média zero antes que a regressão de *ridge* seja realizada, então a interceptação estimada assumirá a forma $\widehat{\beta_0} = \bar{y} = \sum_i^n y_i/n$ (JAMES *et al.*, 2013).

Em Hastie *et al.*, (2009), a penalização da interceptação faria com que o procedimento dependesse da origem escolhida para Y. Isto é, adicionar uma constante c a cada um dos alvos yi não resultaria simplesmente em um deslocamento das previsões na mesma quantidade c. A solução para a equação (22) pode ser separada em duas partes, após reparametrização usando entradas centradas: cada x_{ij} é substituído por $x_{ij} - \bar{x_j}$. Estima-se β_0 por $\bar{y} = 1/n\ \sum_i^n y_i$. Os coeficientes restantes são estimados por uma regressão de *ridge* sem interceptação, usando o x_{ij}centrado. Supondo que essa centralização foi feita, de forma que a matriz de entrada X tem p (ao invés de p + 1) colunas, pode ser escrita na forma de matriz, como:

$$RSS(\lambda) = (y - X\beta)^T(y - X\beta) + \lambda\beta^T\beta \quad (21)$$

Logo, as soluções de regressão de *ridge* são com facilidade vistas como:

$$\hat{\beta}^R = (X^T X + \lambda I)^{-1} X^T y \quad (22)$$

O *LASSO* é uma alternativa relativamente recente à regressão do *ridge* que supera as desvantagens de *ridge* (JAMES *et al*., 2013). Os coeficientes de $\hat{\beta}_\lambda^L$ minimizem a quantidade:

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda \sum_{j=1}^{p}|\beta_j| = RSS + \lambda \sum_{j=1}^{p}|\beta_j| \quad (23)$$

Comparando, pode-se ver que a regressão *LASSO* é parecida com a de *ridge*. Elas têm formulações semelhantes. A diferença está no β_j^2, o termo da regressão de *ridge* foi substituído por $|\beta_j|$ na penalidade de *LASSO*. Em linguagem estatística, o *LASSO* é uma a penalidade L_1 e a penalidade de *ridge* é a L_2. A normal L_1 de um vetor coeficientes β é dada por $\|\beta\|_1 = \sum|\beta_j|$ (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

O *LASSO* reduz as estimativas de coeficiente em direção a zero. No entanto, no caso do *LASSO*, a penalidade L_1 tem o efeito de forçar algumas das estimativas dos coeficientes a serem exatamente iguais a zero quando o parâmetro de ajuste λ é suficientemente grande.

Consequentemente, assim como a seleção do melhor subconjunto, o *LASSO* realiza a seleção de variável. Como resultado, os modelos gerados a partir do *LASSO* são geralmente muito mais fáceis de interpretar do que aqueles produzidos por regressão de *ridge*. Logo, pode-se dizer que o *LASSO* produz modelos esparsos, isto é, modelos que envolvem apenas um subconjunto das variáveis (JAMES *et al.*, 2013).

Uma generalização do modelo *LASSO* é a *elastic net*, um algoritmo que envolve as penalidades L1 e L2. A vantagem desse modelo é que ele permite a regularização efetiva por meio da penalidade do tipo *ridge* com a qualidade de seleção de recursos da penalidade de *LASSO*. O método fornece estimação de soluções esparsas como também para a restrição das estimativas dos parâmetros. Esse modelo combina dois tipos de penalidade (KUHN; JOHNSON, 2013):

$$RSS_E = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda_1 \sum_{j=1}^{P}\beta_j^2 + \lambda_2 \sum_{j=1}^{P}|\beta_j| \quad (24)$$

4.4.1.2 Regressão logística

A regressão logística é comumente empregada para classificar as observações em duas categorias. Nesse sentido, resulta feramente comum em outros métodos estatísticos, que pode ser utilizada como classificador. Em especial, é um modelo para a probabilidade condicional de ocorrência de Y dado um conjunto de preditores X (DABUS, 2020).

Supondo um conjunto de dados em que a resposta padrão tem duas categorias, sim ou não, a regressão logística modela a probabilidade para que Y pertença a uma categoria. Deve-

se modelar a relação de $p(x) = Pr(Y = 1|X)$, precisando modelar p (X) usando uma função que fornece resultados entre 0 e 1 para todos os valores de X. Usa-se a função logística para estimar:

$$p(X) = \frac{e^{\beta_0 + \beta_1 X}}{1 + e^{\beta_0 + \beta_1 X}} \quad (25)$$

Depois de um pouco de manipulação da função logística, tem-se que:

$$\frac{p(X)}{1-p(X)} = e^{\beta_0 + \beta_1 X}. \quad (26)$$

Essa parte da equação $\frac{p(X)}{1-p(X)}$ é chamada de odds e pode assumir qualquer valor entre 0 e ∞. Os valores perto de 0 e ∞ indicam probabilidade muito baixa e muito alta, respectivamente (JAMES *et al.*, 2013).

Aplicando o logaritmo em ambos os lados, encontra-se:

$$\log\left(\frac{p(X)}{1-p(X)} = \beta_0 + \beta_1 X\right). \quad (27)$$

O lado esquerdo é chamado de *log-odds* ou logit, em um modelo de regressão logística, aumentando X em uma unidade muda o log-odds de β_1 (27), ou de forma equivalente, multiplica as probabilidades por e^{β_1} (26) (JAMES *et al.*, 2013). Para estimar os coeficientes β_0 e β_1 que são desconhecidos, e devem ser estimados com base nos dados de treinamento disponíveis, o modelo mais utilizado para estimar um modelo de regressão logística é o método da máxima verossimilhança. O entendimento básico por trás do uso de probabilidade máxima para ajustar um modelo de regressão logística é a seguinte: buscar estimativas para β_0 e β_1 de modo que a probabilidade prevista $\hat{p}(x_i)$ corresponda ao mais próximo possível do status verdadeiro observado do indivíduo (JAMES *et al.*, 2013).

Para Dabus (2020), essa intuição pode ser formalizada empregando os valores de $\hat{\beta}$ de modo que maximizem a função de verossimilhança:

$$\ell(\beta_0, \beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i':y_{i'}} (1 - p(x_{i'})) \quad (28)$$

Os β_0 e β_1 são escolhidos para maximizar a função de verossimilhança acima.

E, por fim, de acordo com Silva (2019), existe uma relação entre a fronteira de decisão linear no método de regressão logística e a escolha de um ponto de corte para $p(x)$. Logo, a determinação de um ponto de corte irá definir uma fronteira de decisão linear para o modelo de regressão logística. Conforme a regressão linear, as penalidades *ridge, LASSO e elastic net* podem ser também aplicadas em conjunto com a regressão logística (JAMES *et al.*, 2013).

4.4.2 Modelos não – lineares

Nesta seção, é apresentado uma descritiva dos conceitos de duas abordagens clássicas de métodos de ML, o *K-Nearest Neighbors (KNN)* e o *Support Vector Machines (SVM),* que são métodos utilizados neste estudo.

4.4.2.1 K-nearest neighbors (KNN)

O método não paramétrico mais simples e mais conhecidos na comunidade de aprendizagem de máquina é o *K-Nearest Neighbors (KNN)*, ou K-vizinhos mais próximos (IZBICKI; SANTOS, 2019). De modo simples, Sarker (2021) explica que o método de KNN tanto pode ser usado para regressão como para classificação. O KNN não se emprega na construção de um modelo interno geral, ele armazena todas as instâncias correspondentes aos dados de treinamento no espaço n-dimensional. Utiliza-se dos dados e classifica novos pontos de dados com base em medidas de similaridade. A classificação é calculada a partir de uma votação, predominantemente, simples dos k vizinhos mais próximos de cada ponto.

O algoritmo KNN é assim nomeado por utilizar dados sobre os k-vizinhos mais próximos de um exemplo para classificar exemplos não rotulados. A letra k é um termo variável que sugere que qualquer número de vizinhos mais próximos pode ser empregado. Após escolher o k, o algoritmo requer um conjunto de dados de treinamento composto de exemplos que foram classificados em várias categorias, rotulados por uma variável nominal. Então, para cada registro não rotulado no conjunto de dados de teste, KNN identifica K registros nos dados de treinamento que são os "mais próximos" e semelhantes. A instância de teste não rotulada é atribuída à classe da maioria dos k vizinhos mais próximos (LANTZ, 2015).

Dado um valor para K e um ponto de predição x0, a regressão KNN primeiro identifica as observações de treinamento K que estão mais próximas de x0, representado por N_0. Em seguida, estima $f\ (x_0)$ usando a média de todas as respostas de treinamento em N_0. Pode-se escrever dessa forma:

$$\hat{f}(x_0) = \frac{1}{K} \sum_{x_i \in N_0} y_i, \tag{29}$$

O valor ótimo para K dependerá da compensação de viés-variância, um valor pequeno para K fornece o ajuste mais flexível, que terá baixa tendência, mas alta variância. Essa variação deve-se ao fato de que a previsão em uma determinada região é inteiramente dependente de apenas uma observação. Em contrapartida, os valores maiores de K fornecem um ajuste mais

suave e menos variável. A previsão em uma região é uma média de vários pontos e, portanto, alterar uma observação tem um efeito menor. No entanto, a suavização pode causar viés, mascarando parte da estrutura em $f\ (x)$ (LANTZ,2015).

O tuning parameter K pode ser selecionado via validação cruzada. Um valor alto de K pode levar a um modelo muito simples (uma constante quando $k \longrightarrow \infty$) e, assim um viés alto, mas uma variância baixa. E um valor baixo para K leva a um estimador com variância alta, mas viés baixo (IZBICKI; SANTOS, 2019).

Agora, a discussão abordará sobre métodos baseados em árvore para regressão e classificação. A fim de fazer uma previsão para uma dada observação, normalmente, usa-se a média ou o modo das observações de treinamento na região a que pertence. Como o conjunto de regras de divisão usado para segmentar, o espaço do preditor pode ser resumido em uma árvore. Esses tipos de abordagens são conhecidos como métodos de árvore de decisão.

4.4.2.2 Support vector machines (SVM)

Support vector machines (SVM) trata-se de uma abordagem para classificação que foi desenvolvida na comunidade da ciência da computação na década de 1990 e que cresceu em popularidade desde então. Os SVMs têm demonstrado um bom desempenho em uma variedade de configurações e costumam ser analisados como um dos melhores classificadores (JAMES *et al*.,2013).

Embora a matemática básica que empregada a SVMs já exista há muitos anos, ela ficou popular recentemente. Sua popularidade pode ser explicada pelo seu desempenho de última geração, mas, talvez, também devido ao fato de que algoritmos SVM premiados foram implementados em várias bibliotecas populares e bem suportadas em muitas linguagens de programação, incluindo R. O SVM. Este foi assim adotado por um público muito mais amplo, caso contrário, poderia ter sido incapaz de aplicar a matemática um tanto complexa necessária para implementar um SVM. Assim sendo, ainda que com uma matemática um tanto complexa, os conceitos básicos são compreensíveis. Logo, os SVMs podem ser adaptados para o uso com quase qualquer tipo de tarefa de aprendizagem, incluindo a classificação e previsão numérica (LANTZ, 2015).

A SVM é uma generalização de um classificador simples e intuitivo denominado maximal margin classifier. Ainda que seja elegante e simples, observar-se que esse classificador, infelizmente, não pode ser aplicado à maioria dos conjuntos de dados, uma vez que requer que as classes sejam separáveis por um limite linear, o support vector classifier.

Nesse sentido, é uma extensão do maximal margin classifier, que pode ser aplicado em uma gama mais ampla de casos. A Support Vector Machines, que é uma extensão adicional do support vector classifier, serve para acomodar limites de classe não lineares. As Support Vector Machines são destinadas à configuração de classificação binária na qual existem duas classes.

Existem, também, as extensões das máquinas de vetores de suporte para o caso de mais de duas classes, e as conexões estreitas entre as máquinas de vetores de suporte e os outros métodos estatísticos, como regressão logística (JAMES *et al*., 2013).

O estudo está focado em SVM. A máquina de vetores de suporte (SVM) admite expandir o espaço de recursos usados pelo *support vector classifier* de uma forma que leva a cálculos mais eficientes. A *Support Vector Machines* (SVM) é uma extensão do vetor de suporte classificador que resulta da ampliação do espaço de recursos de uma maneira específica, usando Kernels. A ideia é querer ampliar nosso espaço de recursos para acomodar um limite não linear entre as classes. A abordagem do Kernel descrita aqui é simplesmente uma abordagem computacional eficiente para concretizar essa ideia. (JAMES *et al*., 2013).

Lantz *et al*. (2013) descreve SVM, em que ele pode ser imaginado como uma superfície que cria um limite entre pontos de dados plotados em multidimensionais que representam exemplos e seus valores de recursos. A finalidade de um SVM é criar um limite plano chamado hiperplano, que divide o espaço para criar partições bastante homogêneas em ambos os lados. James *et al.* (2013) destaca que o SVM é uma extensão do classificador do vetor suporte. A solução para resolver o problema do classificador envolve apenas os produtos internos das observações (em oposição às próprias observações). O produto interno de dois vetores a e b é definido como $\langle a, b\rangle = \sum_{i=1}^{r} a_i b_i$, assim, o produto interno de duas observações x_i , $x_{i'}$, é dado por:

$$\langle x_i , x_{i'}\rangle = \sum_{j=1}^{p} x_i , x_{i'} \tag{30}$$

O classificador de vetor de suporte linear pode ser representado como:

$$f(x) = \beta_0 + \sum_{i=1}^{n} \alpha_i \langle x, x_i\rangle, \tag{31}$$

onde existem n parâmetros α_i, i=1,...,n, um por observação de treinamento (JAMES *et al*., 2013).

E para estimar os parâmetros $\alpha_i, \dots, \alpha_n$, e β_0, é preciso que o $\binom{n}{2}$ produtos internos de $\langle x_i , x_{i'}\rangle$ entre todos os pares de observações de treinamento. A notação $\binom{n}{2}$ meio $n(n-1)/2$, e dá o número de pares entre um conjunto de n item (JAMES *et al*., 2013).

Em resumo, ao representar o classificador linear f(x) e ao calcular seus coeficientes, os produtos internos são imprescindíveis. Assim, supõe-se que cada vez que o produto interno (30)

aparece na representação (31), ou em um cálculo da solução para o classificador de vetor de suporte, pode-se substituir por uma generalização do produto interno da forma:

$$K(x_i , x_{i'}), \quad (32)$$

em que K é alguma função à qual referido como Kernel (JAMES *et al*., 2013).

De acordo com James *et al.*, (2013), Kernel é uma função que quantifica a similaridade de duas observações. Por exemplo, pode-se dizer que:

$$K(x_i , x_{i'}) = \sum_{j=1}^{p} x_{ij}\, x_{i'j}, \quad (33)$$

Logo, a equação (33) é conhecida como um Kernel linear, porque o classificador de vetor de suporte é linear nos recursos. O Kernel linear essencialmente quantifica a semelhança de um par de observações usando correlação de Pearson (padrão). Contudo, é possível escolher outra forma para equação 34. Podendo substituir $\sum_{j=1}^{p} x_{ij}\, x_{i'j}$, e, dessa maneira, ter a função de quantidade:

$$K(x_i , x_{i'}) = \left(1 + \sum_{j=1}^{p} x_{ij}\, x_{i'j}\right)^d \quad (34)$$

Essa equação 34 é conhecida como Polinômio de Kernel de grau d, onde d é um número inteiro positivo. Empregando Kernel com d > 1, em vez do Kernel linear padrão (33), no algoritmo do classificador de vetor de suporte leva a um limite de decisão muito mais flexível.

Fundamentalmente, isso equivale a ajustar um classificador de vetor de suporte em um espaço de dimensão superior, envolvendo o polinômio de grau d, em vez de ajustar no espaço de recurso original. Quando o classificador de vetores de suporte é combinado com um Kernel não linear como (34), o classificador resultante é conhecido como uma máquina de vetores de suporte (SVM) (JAMES *et al.*, 2013).

Observe que, nesse caso, a função não linear, tem a forma de:

$$f(x) = \beta_0 + \sum_{i \in \mathcal{S}} \alpha_i\, K(x, x_i) \quad (35)$$

Conforme James *et al.*, (2013), o ajuste é um aprimoramento substancial em relação ao classificador de vetor de suporte linear. Quando d = 1, então, o SVM reduz-se ao classificador de vetor de suporte, visto anteriormente.

4.4.3 Modelos de árvore de decisão

De acordo com Lantz (2015), os métodos de aprendizes de árvore de decisão são classificadores poderosos, que utilizam uma estrutura de árvore para modelar as relações entre os recursos e os resultados potenciais. Pode-se dizer que as árvores de decisão são as técnicas

de aprendizado de máquina mais usada e podem ser aplicadas para modelar quase qualquer tipo de dados.

Uma das vantagens da árvore de decisão é que a estrutura de árvore do tipo fluxograma não é exclusivamente para uso interno do aluno. Depois que o modelo é criado, muitos algoritmos de árvore de decisão produzem uma estrutura resultante em um formato bastante legível. Isso fornece uma visão tremenda de como e por que o modelo funciona ou não funciona bem para uma tarefa específica (LANTZ, 2015).

Apesar de sua grande aplicabilidade, existem alguns cenários em que as árvores podem não ser o ajuste ideal. Exemplificando um desses cenários, é uma tarefa em que os dados têm um grande número de recursos nominais com muitos níveis ou tem um grande número de recursos numéricos. Esses casos talvez resultem em um número muito grande de decisões e em uma árvore, demasiadamente, complexa. Isso pode também colaborar para a tendência das árvores de decisão de superajustar os dados (LANTZ, 2015).

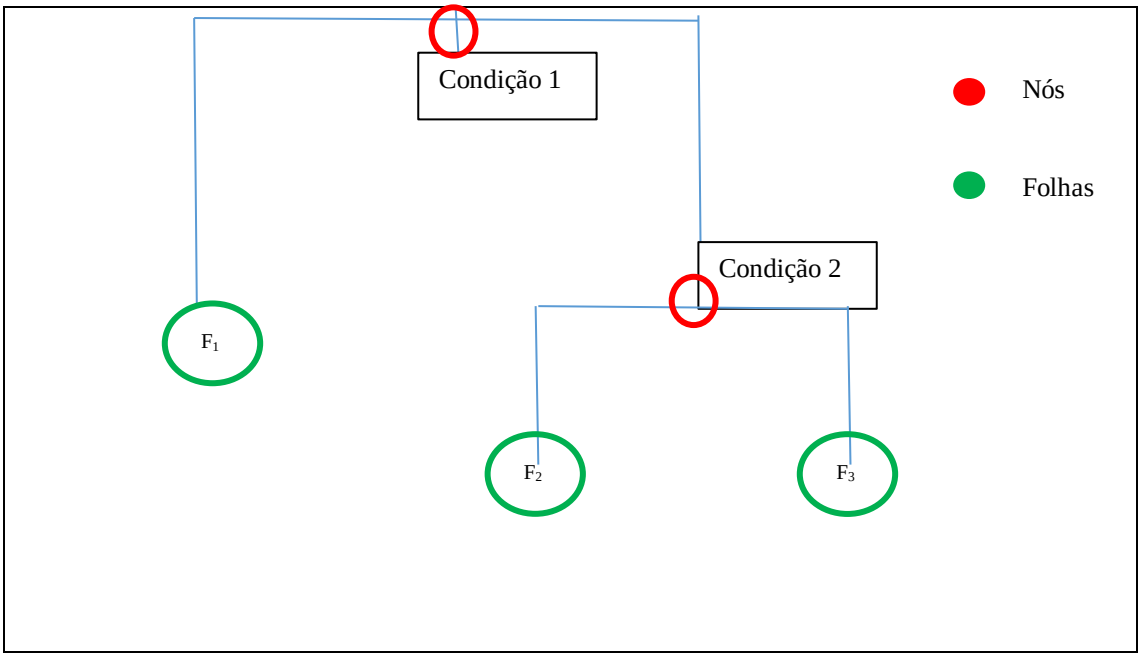
Para James *et al*., (2013), as árvores de decisão podem ser aplicadas a problemas de regressão e classificação. Os métodos de árvore de decisão são simples e bastante fáceis de interpretar. Eles envolvem estratificar ou segmentar o espaço do preditor em várias regiões simples. A fim de fazer uma previsão para uma dada observação, normalmente, utiliza-se a média ou a moda das observações de treinamento na região a qual ela pertence. Como o conjunto de regras de divisão usadas para segmentar o espaço do preditor, pode ser resumido em uma árvore, esses tipos de abordagens são popularmente conhecidos como métodos de árvore de decisão.

4.4.3.1 Regression tree

De acordo com Izbicki e Santos (2019), a árvore de regressão é uma metodologia não paramétrica que induz a resultados extremamente interpretáveis. Uma árvore é construída por particionamentos, recursivos no espaço das covariáveis. Cada particionamento recebe o nome de nó e cada resultado final recebe o nome de folha, como se pode observar na Figura 01. Os autores referidos afirmam que o emprego da árvore para prever uma nova observação pode ser feito da seguinte maneira: começa-se pelo topo e constata-se se a condição descrita no topo (primeiro nó) é satisfeita. Caso seja, segue-se à esquerda. Caso contrário, segue-se à direita. Assim, prossegue-se até atingir uma folha. Como pode ser visto na figura 01, se a condição 1 for satisfeita, a predição é dada por F1. Caso não seja satisfatória, segue-se à direita, e assim,

encontrando outra condição. Caso a mesma seja satisfeita, a observação é prevista como F2 e, caso contrário, é prevista como F3.

Figura 1 - Exemplo de estrutura de árvore de decisão



Fonte: elaborado pela autora com base na estrutura de árvore dos autores Izbicki e Santos (2019).

James *et al.*, (2013) afirmam que para a construção da árvore de regressão existem duas etapas: na primeira etapa, o espaço do preditor é dividido, ou seja, o conjunto de valores possíveis para X1, X2, ..., Xp em J regiões distintas e não sobrepostas, R1, R2, ..., RJ. Na segunda etapa, para cada observação que cai na região Rj, a mesma previsão é feita, que é simplesmente a média dos valores de resposta para as observações de treinamento em Rj.

Para construir a região da etapa 1, R1, R2, ..., RJ, em teoria, as regiões podem ter qualquer forma, mas se opta por dividir o espaço do preditor em retângulos de alta dimensão, ou caixas, para simplicidade e facilidade de interpretação do modelo preditivo resultante. O objetivo é encontrar as caixas R1, R2, ..., RJ, que minimizam o RSS, fornecido por:

$$\sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \widehat{y_{R_j}})^2 \qquad (36)$$

onde $\widehat{y_{R_j}}$ é a resposta média para as observações de treinamento dentro da j-ésima caixa.

Infelizmente, é computacionalmente inviável analisar todas as partições possíveis do espaço de recursos em J caixas. Logo, uma abordagem gananciosa de cima para baixo, conhecida como divisão binária recursiva, é adotada (JAMES *et al.*, 2013).

A abordagem é de cima para baixo, porque começa no topo da árvore e então divide sucessivamente o espaço do preditor. Cada divisão é apontada por meio de dois novos ramos

mais abaixo na árvore. É ganancioso, porque em cada etapa do método de construção da árvore, a melhor divisão é feita naquela etapa específica, em vez de olhar para frente e escolher uma divisão que levará a uma árvore melhor em alguma etapa futura (JAMES *et al.*, 2013).

Para realizar a divisão binária recursiva, primeiro é selecionado o preditor Xj, e o ponto de corte s de modo que dividir o espaço do preditor nas regiões $\{X \mid Xj < s\}$ e $\{X \mid Xj \geq s\}$ leva à maior redução possível em RSS. (A notação $\{X \mid Xj < s\}$ significa a região do espaço do preditor em que Xj assume um valor menor que s). Assim, é considerado todos os preditores X1, ..., Xp e todos os valores possíveis dos pontos de cortes para cada um dos preditores e, em seguida, escolha o preditor e o ponto de corte de forma que a árvore resultante tenha o RSS mais baixo (JAMES *et al.*, 2013). Mais detalhadamente, para qualquer j e s, assim determina o par de regiões:

$$R_1(j,s) = \{X \mid Xj < s\} \; e \; R_2(j,s) = \{X \mid Xj \geq s\}, \tag{37}$$

logo, é procurado o j e s que minimiza a equação (38), abaixo:

$$\sum_{i:\, x_i \in R_1(j,s)} \left(y_i - \widehat{y_{R_1}}\right)^2 + \sum_{i:\, x_i \in R_2(j,s)} \left(y_i - \widehat{y_{R_2}}\right)^2 , \tag{38}$$

onde $\widehat{y_{R_1}}$ e $\widehat{y_{R_2}}$ é a resposta média para as observações de treinamento em $R_1(j,s)$ e $R_2(j,s)$.

Encontrar os valores de j e s que minimizam (38) pode ser feito rapidamente, em especial, quando o número de características p não é muito grande. Depois, repetir o processo, procurando o melhor preditor e o melhor ponto de corte, a fim não só de dividir os dados ainda mais, como também de minimizar o RSS em cada uma das regiões resultantes. No entanto, em vez de dividir todo o espaço do preditor, divide-se uma das duas regiões identificadas anteriormente. Agora há três regiões. Outra vez, procura-se dividir ainda mais uma dessas três regiões, de modo a minimizar o RSS. O processo continua até que um critério de parada seja alcançado, por exemplo, poderia continuar até que nenhuma região contenha mais de cinco observações. Uma vez que as regiões R1,..., Rj, prevê a resposta para uma determinada observação de teste, usando a média das observações de treinamento na região a qual essa observação de teste pertence (JAMES *et al.*, 2013).

A estratégia descrita resultará em árvores menores, pois uma divisão aparentemente sem valor no início da árvore pode ser seguida por uma divisão muito boa, ou seja, uma divisão que leva a uma grande redução no RSS, posteriormente (JAMES *et al.*, 2013). Uma melhor estratégia é cultivar uma árvore muito grande, T0 e, em seguida, podá-la de volta para obter uma subárvore. A finalidade é selecionar uma subárvore que leve a menor taxa de erro de teste.

Dada uma subárvore, admite-se estimar seu erro de teste, usando validação cruzada ou a abordagem de conjunto de validação. No entanto, estimar o erro de validação cruzada para

cada subárvore possível seria muito complicado, uma vez que há um número, extremamente, grande de subárvores possíveis. Em vez disso, é necessário encontrar uma maneira de selecionar um pequeno conjunto de subárvores para consideração. A poda de complexidade de custo, também conhecida como poda de elo mais fraco, oferece uma maneira de fazer exatamente isso.

Em vez de considerar todas as subárvores possíveis, é considerado uma sequência de árvores indexadas por um parâmetro de ajuste não negativo α (JAMES *et al.,* 2013).

Para cada valor de α corresponde uma subárvore T ⊂ T0 tal que:

$$\sum_{m=1}^{|T|} \sum_{x_i \in R_m} \left(y_i - \hat{y}_{R_m}\right)^2 + \alpha|T| \quad (39)$$

é o menor possível. Aqui |T| indica o número de nós terminais da árvore T. Rm é o retângulo (ou seja, o subconjunto do espaço preditor) correspondente ao m-ésimo nó terminal, e $\hat{y}_{R_m}$ é a resposta prevista associada a Rm, assim sendo, a média das observações de treinamento em Rm (JAMES *et al.,* 2013).

O parâmetro de ajuste α controla uma compensação entre a complexidade da subárvore e o seu ajuste aos dados de treinamento. Quando α = 0, então a subárvore T é simplesmente igual a T0, porque a equação (39) apenas mede o erro de treinamento (JAMES *et al*., 2013).

Mas, à medida que α aumenta, há um preço a pagar por ter uma árvore com muitos nós terminais, e assim a quantidade tenderá a ser minimizada para uma subárvore menor (JAMES *et al.,* 2013).

4.4.3.2 Classification trees

A árvore de classificação é muito parecida a uma árvore de regressão, exceto quando é usada para prever uma resposta qualitativa em vez de quantitativa. Porém, para uma árvore de regressão, a resposta prevista para uma observação é dada pela resposta média das observações de treinamento que pertencem ao mesmo nó terminal. Em contrapartida, para uma árvore de classificação, prevê-se que cada observação se refere à classe de observação de treinamento que ocorre mais comumente na região a que pertence. Quando se interpreta os resultados de uma árvore de classificação, normalmente, se está interessado não apenas na previsão de classe correspondente a uma região de nó terminal particular, mas também nas proporções de classe entre as observações de treinamento que caem nessa região (JAMES *et al.,* 2013).

Hastie, Thibshirani e Friedman (2008) afirmam que para um resultado de classificação assumindo os valores 1, 2, ..., K, as únicas mudanças necessárias no algoritmo da árvore

pertencem aos critérios para divisão de nós e poda da árvore. Para a regressão, aplica-se a medida de impureza do nó de erro quadrático, mas isso não é adequado para classificação.

A tarefa de fazer crescer uma árvore de classificação é bastante semelhante à tarefa de fazer crescer uma árvore de regressão. Como na configuração de regressão, é utilizada a divisão binária recursiva para desenvolver uma árvore de classificação. No entanto, como já explanado acima, na configuração de classificação, RSS não pode ser usado como um critério para fazer as divisões binárias, a opção é a taxa de erro de classificação (HASTIE; THIBSHIRANI; FRIEDMAN, 2019).

Visto que se projeta atribuir uma observação em uma determinada região à classe de observação de treinamento que ocorre mais comumente naquela região, a taxa de erro de classificação é simplesmente a fração das observações de treinamento nessa região que não pertencem à classe mais comum:

$$E = 1 - (\hat{p}_{mk}) \quad (40)$$

Como descrito por James *et al.* (2013), o $\hat{p}_{mk}$ representa a proporção de observações de treinamento na m-ésima região que são da k-ésima classe. No entanto, verifica-se que o erro de classificação não é suficientemente sensível para o cultivo de árvores e, na prática, duas outras medidas são preferíveis, o índice de Gin é um deles, definido por:

$$G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk}) \quad (41)$$

Não é difícil ver que o índice de Gini assume um valor pequeno se todos $\hat{p}_{mk}$ estiverem próximos de zero ou um. Por isso, o índice de Gini é conhecido como uma medida de pureza do nó. Então, um valor pequeno indica que um nó contém predominantemente observações de uma única classe (JAMES *et al*., 2013).

De acordo com Hastie, Thibshirani e Friedman (2008), a entropia cruzada e o índice de Gini são diferenciáveis e, portanto, mais receptivos à otimização numérica. Além disso, a entropia cruzada e o índice de Gini são mais sensíveis às mudanças nas probabilidades dos nós do que a taxa de erro de classificação. A entropia cruzada é dada por:

$$D = -\sum_{k=1}^{K} \hat{p}_{mk} \log \hat{p}_{mk} \quad (42)$$

em que, $0 \leq \hat{p}_{mk} \leq 1$, segue-se que $0 \leq \hat{p}_{mk}\ log\ log\ \hat{p}_{mk}$. De tal modo que pode observar que a entropia cruzada assumirá um valor próximo de zero se os $\hat{p}_{mk}$ estiverem todos próximos de zero ou próximos de um. Assim sendo, como o índice de Gini, a entropia cruzada terá um valor pequeno se o m-ésimo nó for puro. Constata-se que o índice de Gini e a entropia cruzada são bastante semelhantes, numericamente.

Lembrando que as árvores de decisão podem ser construídas com variáveis preditoras de valores contínuos, mas também pode ser construída na presença de variáveis preditoras qualitativas. James *et al.*(2013) salienta que, ao construir uma árvore de classificação, o índice de Gini ou a entropia cruzada são normalmente usados para avaliar a qualidade de uma divisão específica, já que essas duas abordagens são mais sensíveis à pureza do nó do que a taxa de erro de classificação. Mas, qualquer uma dessas três abordagens pode ser usada ao podar a árvore, porém a taxa de erro de classificação é preferível se a precisão da previsão da árvore podada final for o objetivo.

4.4.3.3 Bagging

O *bagging* é um procedimento de propósito geral para reduzir a variância de um método de aprendizagem estatística, é particularmente útil e frequentemente usado no contexto de árvores de decisão. Sabe-se que, dado um conjunto de n observações independentes $Z_1, \dots, Z_n$, cada um com variância σ^2, a variância da média $\bar{Z}$ das observações é dada por $\frac{\sigma^2}{n}$. Pode-se dizer que a média de um conjunto de observações reduz a variância (JAMES *et al*., 2013).

De acordo com James *et al.*, (2013), um modo natural de reduzir a variância e, portanto, aumentar a precisão de predição de um método de aprendizado estatístico é pegar muitos conjuntos de treinamento da população. Nesse sentido, construir um modelo de predição separado, usando cada conjunto de treinamento e calcular a média das previsões resultantes, que é calculada $\hat{f}^1(x), \hat{f}^2(x), \dots, \hat{f}^B(x)$, usando B conjuntos de treinamento separados para calcular a média deles a fim de obter um único modelo de aprendizagem estatística de baixa variância, dado por:

$$\hat{f}_{avg}(x) = \frac{1}{B} \sum_{b=1}^{B} \hat{f}^b(x) \quad (43)$$

Mas, não é tão simples assim, James *et al.* (2013) explicam que, geralmente, não se têm acesso a vários conjuntos de treinamento. Assim sendo, deverá inicializar tomando amostras repetidas do (único) conjunto de dados de treinamento. Nessa abordagem, é gerado B conjuntos de dados de treinamento bootstrapped diferentes. Em seguida, treina-se nosso método no bth conjunto de treinamento bootstrapped para obter $\hat{f}^{*b}(x)$ e, finalmente, calcular a média de todas as previsões. Isso é chamado de *bagging*:

$$\hat{f}_{bag}(x) = \frac{1}{B} \sum_{b=1}^{B} \hat{f}^b(x) \quad (44)$$

O procedimento de *bagging* foi abordado no contexto de regressão para prever um resultado quantitativo Y. Para Y qualitativo, existem algumas abordagens possíveis, mas a mais simples é a seguinte. Para uma determinada observação de teste, pode ser registrada a classe prevista por cada uma das árvores B e obter a maioria dos votos: a previsão geral é a que ocorre mais comumente entre as previsões B. Um valor de B suficientemente grande é usado para que o erro se estabeleça (JAMES *et al*., 2013).

O *bagging*, normalmente, resulta em maior precisão quando comparado à previsão, usando uma única árvore. Infelizmente, pode ser difícil interpretar o modelo resultante, já que as vantagens das árvores de decisão implicam o diagrama, atraente e facilmente interpretado. No entanto, quando se tem um grande número de árvores, não é mais possível representar o procedimento de aprendizagem estatística resultante usando uma única árvore. Além disso, não é mais claro quais variáveis são as mais importantes para o procedimento. Logo, o *bagging* melhora a precisão da previsão em detrimento da interpretação (JAMES *et al.*, 2013).

Embora a coleção de árvores *bagging* seja muito mais difícil de interpretar do que uma única árvore, pode-se obter um resumo geral da importância de cada preditor usando o RSS (para *bagging* e *regression trees*) ou o índice de Gini (para *bagging* e *classification trees*). No caso de árvores de regressão de *bagging*, registra-se a quantidade total em que o RSS é diminuído devido a divisões em um determinado preditor, calculando a média de todas as árvores B. Um grande valor indica um preditor importante. Da mesma forma, no contexto de árvores de classificação de *bagging*, soma-se a quantidade total do índice de Gini e diminui-se as divisões em um determinado preditor, calculando todas as árvores B (JAMES *et al.*, 2013).

Os autores Kuhn e Johnson (2013) afirmam que o algoritmo *bagging* apresenta uma desvantagem no que se refere ao fato das B árvores agregadas demonstrarem alta correlação devido à utilização de todas as covariadas como candidatas em todas as etapas da divisão das B árvores de decisão. Assim, pode-se utilizar o algoritmo *random forest* como opção para reduzir a correlação supracitada.

4.4.3.4 Random forests

É um método baseado em um conjunto, denominado *random forests* (ou florestas de árvore de decisão). Concentra-se, apenas, em conjuntos de árvores de decisão. Este método foi defendido por Leo Breiman e Adele Cutler, em 2001, e combina os princípios básicos de *bagging* com seleção aleatória de recursos para adicionar diversidade ao modelo de *decision*

tree. Depois que o conjunto de árvores (a floresta) é gerado, o modelo usa uma votação para combinar as previsões das árvores (LANDZ, 2013).

Random forests combina versatilidade e poder em uma única abordagem de aprendizado de máquina. Como o conjunto usa apenas uma pequena porção aleatória de todo o conjunto de recursos, as *random forests* podem lidar com conjuntos de dados extremamente grandes, em que a chamada "maldição de dimensionalidade" pode fazer com que outros modelos falhem. Simultaneamente, suas taxas de erro para a maioria das tarefas de aprendizagem estão no mesmo nível de quase qualquer outro método (LANDZ, 2013).

Random forests oferece uma melhoria em relação às árvores *bagged* por meio de um pequeno ajuste aleatório. Tal como no *bagging*, é construída uma série de árvores de decisão em amostras de treinamento inicializadas. Contudo, ao construir essas árvores de decisão, cada vez que uma divisão em uma árvore é considerada uma amostra aleatória de m preditores é escolhida como candidata à divisão do conjunto completo de p preditores. A divisão pode usar apenas um desses m preditores. Uma nova amostra de m preditores é obtida em cada divisão e, normalmente, se escolhe $m \approx \sqrt{p}$, ou seja, o número de preditores considerados em cada divisão é, aproximadamente, igual à raiz quadrada do número total de preditores (JAMES *et al*., 2013).

Na construção de uma *random forest*, a cada divisão da árvore, o algoritmo não pode nem mesmo considerar a maioria dos preditores existentes. Supondo que existaum preditor muito forte no conjunto de dados, junto com vários outros preditores moderadamente fortes,então, na coleção de árvores *bagged*, a maioria ou todas as árvores usarão esse forte preditor na divisão superior. Portanto, todas as árvores *bagged* parecerão bastante semelhantes entre si. As previsões das árvores *bagged* serão, altamente, correlacionadas. A média de muitas quantidades, altamente correlacionadas, não leva a uma redução tão grande na variância quanto a média de muitas quantidades não correlacionadas. Em particular, isso significa que *bagging* não levará a uma redução substancial na variância em uma única árvore nesse cenário (JAMES *et al*., 2013).

Para James *et al.,* (2013), a principal diferença entre *bagging* e o *random forest* é a escolha do tamanho do subconjunto do preditor m. Por exemplo, se uma *random forest* é construída usando m = p, então isso equivale, simplesmente, ao *bagging*. O *random forest*, usando $m = \sqrt{p}$, leva a uma redução no erro de teste e no erro *out-of-bag* (OOB), sobre o *bagging*. Usar um pequeno valor de m na construção de uma *random forest* normalmente será útil quando se tem um grande número de preditores correlacionados.

Landz (2013) lista os pontos fortes e fracos do *random forest*. Os pontos fortes são: i. um modelo multifuncional que funciona bem na maioria dos problemas; ii. pode lidar com dados ruidosos ou ausentes, bem como recursos categóricos ou contínuos; iii. seleciona apenas os recursos mais importantes; iv. pode ser usado em dados com um número extremamente grande de recursos ou exemplos. Os pontos fracos são: i. Ao contrário de uma *decision tree,* o modelo não é facilmente interpretável, e; ii. pode exigir algum trabalho para ajustar o modelo aos dados.

4.4.3.5 Boosting

É uma abordagem para melhorar as previsões resultantes de uma árvore de decisão. Assim como o *bagging, o boosting* é uma abordagem geral que pode ser aplicada a muitos métodos de aprendizagem máquina para regressão ou classificação. O *bagging* envolve a criação de várias cópias do conjunto de dados de treinamento original usando o *bootstrap*, ajustando uma árvore de decisão separada para cada cópia e, em seguida, combinando todas as árvores para criar um único modelo preditivo. Assim sendo, cada árvore é estabelecida em um conjunto de dados de *bootstrap*, independentemente das outras árvores. O *boosting* funciona de maneira semelhante, exceto quando as árvores são cultivadas sequencialmente: cada árvore é cultivada usando informações de árvores previamente cultivadas. O *boosting* não envolve amostragem de *bootstrap*. Em vez disso, cada árvore ajusta-se a uma versão alterada do conjunto de dados original (JAMES *et al*., 2013).

O método foi, originalmente, desenvolvido para problemas de classificação, em que muitos classificadores fracos (por exemplo, um classificador que prevê, marginalmente, melhor do que aleatório) foram combinados em um classificador forte. Na década de 1990, vários algoritmos de *boosting* apareceram para completar a teoria original (KUHN ; JOHNSON, 2013).

O AdaBoost gera uma sequência de classificadores fracos, em que a cada iteração o algoritmo encontra o melhor classificador com base nos pesos de amostra atuais (KUHN ; JOHNSON, 2013). As amostras classificadas incorretamente na k-ésima iteração ganham mais peso na iteração (k + 1), enquanto as amostras classificadas corretamente recebem menos peso na iteração subsequente. Isto é, as amostras que são difíceis de classificar recebem pesos cada vez maiores até que o algoritmo identifique um modelo que as classifique corretamente. Assim sendo, cada iteração do algoritmo é indispensável para aprender um aspecto diferente dos dados, com foco em regiões que contêm amostras difíceis de classificar. Em cada iteração, um

peso de estágio é calculado com base na taxa de erro nessa iteração. Logo, modelos mais precisos têm valores positivos mais altos, e modelos menos precisos têm valores negativos mais baixos. A sequência geral de classificadores ponderados é então combinada em um conjunto e tem um forte potencial para classificar melhor do que qualquer um dos classificadores individuais (KUHN; JOHNSON, 2013).

Em James *et al.,* (2013), os autores discorrem que o *boosting* é uma abordagem que pode ser aplicada a muitos métodos de aprendizagem estatística para regressão ou classificação. Considera-se, primeiro, a configuração de regressão, assim como o *bagging*, o *boosting* envolve a combinação de um grande número de árvores de decisão, $\hat{f}^1, \ldots, \hat{f}^B$. O *boosting* é descrito como:

Se $\hat{f}(x) = 0$ e $r_i = y_i$, para todos os *i* no conjunto de treinamento.

Para *b = 1,2, ..., B,* necessita que:

a. Ajustar uma árvore $\hat{f}^B$ com d divisões (*d + 1 nós terminais*), para o treinamento de dados (*X, r*);

b. Atualize $\hat{f}$ adicionando uma versão reduzida da nova árvore:

$$\hat{f}(x) \leftarrow \hat{f}(x) + \lambda \hat{f}^B(x) \tag{45}$$

c. Atualizar os resíduos;

$$r_i \leftarrow r_i - \lambda \hat{f}^B(x_i) \tag{46}$$

d. Produz o modelo *boosting*,

$$\hat{f}(x) = \sum_{b=1}^{B} \lambda \hat{f}^B(x) \tag{47}$$

Dado o modelo atual, é ajustada uma árvore de decisão aos resíduos do modelo. Ou seja, ajusta-se uma árvore usando os resíduos atuais, em vez do resultado *Y,* como a resposta. Em seguida, é adicionado essa nova árvore de decisão à função ajustada para atualizar os resíduos. Cada uma dessas árvores pode ser bastante pequena, com apenas alguns nós terminais, determinadas pelo parâmetro *d* no algoritmo. Ajustando pequenas árvores aos resíduos, melhora-se lentamente $\hat{f}$ em áreas em que ele não tem um bom desempenho. O parâmetro de encolhimento λ retarda o processo ainda mais, permitindo que mais árvores com formatos diferentes ataquem os resíduos. Em geral, as abordagens de aprendizagem estatística que aprendem lentamente tendem a executar bem. Observe que, no *boosting*, ao contrário do *bagging*, a construção de cada árvore depende fortemente das árvores que já foram cultivadas (JAMES *et al., 2013*).

As configurações para classificação ocorrem de maneira semelhante à regressão, mas um pouco mais complexa, e os detalhes são omitidos aqui (JAMES *et al., 2013*). *Boosting* tem três parâmetros de ajuste:

1. O número de árvores *B*. Ao contrário das *bagging* e o *random forest*, o *boosting* pode ser *overfitting* (sobre-ajustado, quando o modelo aprende demais sobre os dados) se *B* for muito grande, embora esse *overfitting* tenda a ocorrer lentamente, se é que ocorre. Usa-se validação cruzada para selecionar *B*.
2. O parâmetro de contração λ, é um pequeno número positivo. Isso controla a taxa em que o *boosting* aprende. Os valores típicos são 0,01 ou 0,001, e a escolha certa pode depender do problema. O λ muito pequeno pode exigir o uso de um valor muito grande de *B* para obter um bom desempenho.
3. O número *d* de divisões em cada árvore, que controla a complexidade do conjunto *boosted*. Frequentemente, $d = 1$ funciona bem: nesse caso, cada árvore é um toco, consistindo em uma única divisão. Nesse caso, o conjunto potencializado ajusta-se a um modelo aditivo, pois cada termo envolve apenas uma única variável. Geralmente, *d* é a profundidade de interação e controla a ordem de interação do modelo *boosted*, uma vez que *d* divisões podem envolver no máximo *d* variáveis.

4.5 Critérios de avaliação do modelo

Nesta seção, pretende-se conceituar a metodologia de avaliação dos modelos de ML, além de apresentar as métricas que podem ser estimas através da matriz de confusão.

4.5.1 Matriz de confusão

Um método comum para descrever o desempenho ou a avaliação de um modelo de classificação é a matriz de confusão. Esta é uma tabulação cruzada simples das classes observadas e previstas para os dados. As células diagonais denotam casos em que as classes são previstas corretamente, enquanto as fora da diagonal ilustram o número de erros para cada caso possível (KUHN; JOHNSON, 2013).

A avaliação do modelo é um procedimento de avaliação para escolher entre diferentes tipos de modelos, parâmetros de ajuste e recursos. Melhores processos de avaliação levam a modelos melhores e mais precisos. A matriz de confusão é essencial para esse processo, pois pode ser definida livremente como uma tabela que exibe o desempenho de um modelo de

classificação em um conjunto de dados de teste para os quais os valores verdadeiros são conhecidos. Uma matriz de confusão é de fácil interpretação e pode ser usada para estimar várias outras métricas (SOUZA, 2019). Para Silva (2019), uma matriz de confusão é indispensável para analisar os valores previstos e reais pertencentes a cada uma das classes.

A matriz de confusão para o problema de duas classes ("eventos" e “não eventos”) é demostrada na Tabela 01. As células da tabela indicam o número de verdadeiros positivos (VP), falsos positivos (FP), verdadeiros negativos (VN) e falsos negativos (FN). A linha superior da tabela corresponde às amostras previstas para serem eventos. Alguns são previstos corretamente (os verdadeiros positivos ou VP), enquanto outros são classificados de forma incorreta (falsos positivos ou FP). Da mesma forma, a segunda linha contém os negativos previstos com verdadeiros negativos (VN) e falsos negativos (FN) (KUHN; JOHNSON, 2013).

Tabela 1 - Matriz de confusão

	Observado	
	Eventos	Não Evento
Evento	VP	FP
Não Evento	FN	VN

Elaboração própria a partir do Livro de Kuhn e Johnson (2013).

As métricas que podem ser estimadas pela matriz de confusão são, acurácia, sensibilidade e especificidade. A partir da sensibilidade e a especificidade, é possível obter outra métrica de desempenho, a AUC (Area Under Curve) (UNZA, 2020).

4.5.2 Accuracy, sensitivity, specificity e AUC

A métrica mais simples é a *accuracy*. Isso reflete a concordância entre as classes observadas e previstas e tem a interpretação mais direta. No entanto, existem algumas desvantagens em usar essa estatística, uma delas é que as contagens gerais de precisão não fazem distinção sobre o tipo de erros cometidos, ou sobre a importância (KUHN; JOHNSON, 2013).

$$Accuracy = \frac{verdadeiro\ positivo\ (VP)}{Verdadeiro\ positivo\ (VP) + falso\ positivo\ (FP) + falso\ negativo\ (FN) + verdadeiro\ negativo\ (VN)} \quad (48)$$

De acordo com Kuhn e Johnson (2013), para duas classes, existem estatísticas adicionais que podem ser relevantes quando uma classe é interpretada como o evento de interesse. A *sensitivity* do modelo é a taxa em que o evento de interesse é previsto corretamente para todas as amostras com o evento, ou:

$$Sensitivity = \frac{verdadeiros\ positivos (VP)}{verdadeiros\ positivos (VP) + falsos\ negativos\ (FN)} \quad (49)$$

A *sensitivity* pode ser considerada a taxa positiva verdadeira, uma vez que mede a *accuray* na população de eventos. A *specificity* é definida como a taxa em que as amostras de não-eventos são previstas como não-eventos, ou:

$$Specificity = \frac{verdadeiros\ negativos\ (VN)}{falsos\ positivos\ (FP) + verdadeiros\ negativos\ (VN)} \quad (50)$$

A taxa de falsos positivos é definida como 1- *specificity*. Assumindo um nível fixo de *accuracy* para o modelo, normalmente, há uma compensação a ser feita entre a *sensitivity* e a *specificity*. Intuitivamente, aumentar a *sensitivity* de um modelo pode resultar em uma perda de *specificity*, uma vez que mais amostras estão sendo previstas como eventos. O possível trade-offs entre *sensitivity* e *specificity* pode ser apropriado quando há diferentes penalidades associadas a cada tipo de erro. A curva de características operacionais do receptor/*receiver operating characteristic* (ROC) é uma técnica para avaliar essa compensação, o mais comum método para combinar *sensitivity* e *specificity* em um único valor (KUHN; JOHNSON, 2013).

As curvas ROC é projetada como um método geral que, dada uma coleção de pontos de dados contínuos, determinam um limite efetivo de modo que os valores acima do limite são indicativos de um evento específico. Isso pode ser usado para determinar pontos de corte alternativos para probabilidades de classe. A curva ROC é criada avaliando as probabilidades de classe para o modelo em vários possíveis de limiares. Para cada limite de candidato, a taxa de verdadeiro-positivo resultante (isto é, a *sensitivity*) e a taxa de falso-positivo (um menos a *specificity*) são representados um contra a outra.

Silva (2019) discorre que uma curva ROC ótima vai abraçar o canto superior esquerdo, então quanto maior a *AUC* (*Area Under the* ROC Curve), melhor o classificador. A *AUC* é a derivada da curva ROC, de maneira que essa busca sintetizar a curva ROC em um único valor, que varia de 0,0 a 1,0, tendo como o limiar de 0,5 entre elas.

Logo, utiliza-se curva ROC para comparação dos modelos. De acordo com Kuhn e Johson (2013), ela é bastante útil por maximizar o trade-off entre sensibilidade e especificidade. Uma curva ROC ideal abrangerá o canto superior esquerdo, portanto, o modelo com a maior

área sob a curva, AUC ROC, seria o mais eficaz, quanto maior o AUC, melhor será o classificador. A curva ROC considera todos os possíveis limiares, uma vez que para cada variação do limiar do classificador são alteradas as taxas de VP e FP. A Taxa de Verdadeiro Positivo (TVP) representa a sensibilidade e a Taxa de Falso Positivo é 1, - especificidade, que constitui as contagens reais dos indivíduos em cada classe. Levando em consideração a matriz de confusão como base, as TVP e a TFP são dadas por:

$$TVP = \frac{VP}{VP+FN} \text{ e } TFP = \frac{FP}{VN+FP} \quad (51)$$

4.5.3 Teste de McNemar

Para comparar o desempenho de modelos com previsões semelhantes, pode-se usar o teste de McNemar. De acordo com Ciechalski *et al*., (2002), é um teste qui-quadrado que emprega os dados correlacionados. Ou seja, dados dependentes em vez de dados independentes, podendo ser em um conjunto de dados e amostra de distribuição não normal. Silva, Almeida e Ramalho (2020) utilizam o teste de McNemar de maneira individual nos modelos de ML depois de ter feito uma pré-seleção, identificando que os mesmos possuem medidas de desempenho com valores muito próximos. Nesse sentido, os autores utilizam o teste para averiguar se os modelos são ou não estatisticamente semelhantes.

O autor Hawass (1997) afirma que esse teste pode ser usado para testar as diferenças entre os desempenhos de dois procedimentos para decisão final, na Tabela 2. Pode-se dizer que A e D são pares empatados, pois não apresentam diferenças no desempenho diagnóstico dos dois procedimentos. Porém, o teste de hipóteses para o Teste de McNemar usa dados das duas células discordantes B e C, as previsões falsas.

Visualizando a matriz de confusão deste estudo, é possível afirmar que o teste de McNemar é utilizado para obter a probabilidade da diferença entre os falsos negativos e os falsos positivos, ou seja, as classificações incorretas. Logo, o teste estatístico não corrigido para o procedimento de McNemar é dado pela equação 52 e o teste de McNemar pela equação corrigida 53, detalhadas abaixo (BURGESS E ADEDOKUN, 2012).

$$X^2 = \frac{(B-C)^2}{(B+C)} \quad (52)$$

$$X^2 = \frac{(|B-C|-1)^2}{(B+C)} \quad (53)$$

Tabela 2 - Tabela do Teste de McNemar

	Previsões	
	Verdadeiro	Falso
Verdadeiro	A	B
Falso	C	D

Elaboração própria a partir do artigo de Burgess e Adedokun (2012).

Assim, para uma distribuição X^2 e 1 grau de liberdade, nesse estudo a hipótese nula (H_0) do Teste de McNemar é que os algoritmos aplicados para prever pobreza têm precisões iguais, e a hipótese alternativa (H_1) é que os algoritmos aplicados para prever pobreza têm precisões diferentes.

5 RESULTADOS E DISCUSSÕES

Esta seção dedica-se a comparar os resultados encontrados não só nos modelos tradicionais de econometria, como também nos métodos de algoritmos do ML. A base contém uma amostra de 2674 indivíduos do estado do Ceará. Foram escolhidas da base original 17 variáveis para estimar os modelos.

Optou-se por dividir a base em um conjunto de treinamento e teste, em que 70% da base de dados será para treinamento e 30% para teste. A variável dependente status é dividida em pobre e não pobre. É identificado neste estudo uma proporção de 62,9% de não pobres e 37,1% de pobres.

Na Tabela 3, estão as estimações dos modelos tracionais de regressão linear, logit e probit, além das Odds Radio para cada variável e seus respectivos modelos, todos adequados para variável de resposta binária status, com o objetivo de prever a pobreza.

Tabela 3 - Estimações dos modelos tradicionais para prever pobre e não pobres

	Status					
	Prob. Linear	Odds Radio	Logit	Odds Radio	Probit	Odds Radio
Sexo	-0.0600***	0.94	-0.4298***	0.65	-0.2485***	0.77
	(0.0154)		(0.1026)		(0.0590)	
Idade	-0.0049***	0.99	0.0299***	0.97	0.0170***	0.98
	(0.0007)		(0.0050)		(0.0028)	
Ler e escreve	-0.0602**	0.94	-0.4874**	0.61	-0.2736**	0.76
	(0.0291)		(0.1913)		(0.1109)	
Número de Pessoas	0.1360***	1.14	0.8779***	2.40	0.5076***	1.66
Número de Crianças	0.0154	1.01	0.1095	1.11	0.0526	1.05
	(0.0161)		(0.1080)		(0.0623)	
Número de Adultos	-0.1003***	0.90	-0.6224***	0.53	-0.3644***	0.69
	(0.0147)		(0.0994)		(0.0570)	
Número de Idosos	-0.2196***	0.80	-1.5886***	0.20	-0.9153***	0.40
	(0.0209)		(0.1648)		(0.0908)	
Energia	-0.0172	0.98	-0.0440	0.95	-0.0364	0.96
	(0.0248)		(0.1631)		(0.0944)	

Fonte: Elaboração própria com base nos dados da POF 2017-2018.

Tabela 3 - Estimações dos modelos tradicionais para prever pobre e não pobres

(Conclusão)

	Status					
	Prob. Linear	Odds Radio	Logit	Odds Radio	Probit	Odds Radio
Lixo	-0.0878***	0.91	-0.6446***	0.52	-0.3626***	0.69
	(0.0335)		(0.2167)		(0.1259)	
Cômodos	-0.0136***	0.98	-0.1232***	0.88	-0.0603***	0.94
	(0.0046)		(0.0336)		(0.0189)	
Grau de Instrução	-0.0623***	0.93	-0.4117***	0.66	-0.2329***	0.79
	(0.0052)		(0.0377)		(0.0212)	
Raça	-0.0151	0.98	-0.0811	0.92	-0.0521	0.94
	(0.0146)		(0.1003)		(0.0571)	
Telhado	-0.0610***	0.94	-0.4480***	0.63	-0.2533***	0.77
	(0.0211)		(0.1487)		(0.0842)	
Tipo de Região	0.0689**	1.07	0.4083**	1.50	0.2269**	1.25
	(0.0291)		(0.1844)		(0.1074)	
Domicilio Próprio	-0.0238	0.97	-0.1664	0.84	-0.1134*	0.89
	(0.0177)		(0.1144)		(0.0662)	
Parede adequada	-0.0455	0.95	-0.2390	0.78	-0.1569	0.85
	(0.0323)		(0.2058)		(0.1201)	
Sem Instrução	-0.0800**	0.92	-0.4986**	0.60	-0.2898**	0.74
	(0.0318)		(0.2077)		(0.1204)	
Observações	2674		2674		2674	
R2	0.3390					
Log Likelihood			-1,209.5150		-1,214.8870	
Akaike. Inf. Crit			2,455.0300		2,465.7730	
Note:	*p<0,1	**p<0,5	***p<0,01			

Fonte: Elaboração própria com base nos dados da POF 2017-2018.

De acordo com James *et al.*, (2013), o ponto de corte para definir a fronteira de decisão linear é de 0,5. Ou seja, para prever se o indivíduo é pobre ou não, é necessário calcular se a probabilidade de ser pobre é maior ou menor do que 0,5. Produzidas as previsões, apresenta-se a matriz de confusão do logit, construída para determinar quantas observações foram classificadas correta e incorretamente. Os elementos na diagonal da matriz representam

indivíduos cujos *status* padrão foram previstos corretamente, enquanto os elementos fora da diagonal, representam indivíduos que foram classificados incorretamente. Pode-se dizer que o modelo previu corretamente 1471 não pobres e 636 pobres. Nesse caso, a regressão logística previu corretamente 78,79%.

Tabela 4 - Matriz de Confusão da regressão logit, para modelo tradicional

		Status padrão verdadeiro		
		Não (0)	Sim (1)	Total
Status padrão previsto	Não (0)	1471	356	1827
	Sim (1)	211	636	847
	Total	1682	992	2674

Fonte: Elaboração própria com base nos dados da POF 2017-2018.

Na prática, um classificador binário como este pode cometer dois tipos de erros: ele pode atribuir incorretamente um indivíduo que não está na categoria padrão ou pode atribuir incorretamente um indivíduo que está na categoria padrão. Frequentemente, é interessante determinar qual desses dois tipos de erros está sendo cometido. A matriz de confusão, demonstrada para os dados padrão na Tabela 3, é uma maneira apropriada de exibir essas informações (JAMES *et al*.,2013).

Silva (2019) discorre em sua pesquisa que os resultados dessa matriz podem ser enganosos, já que o algoritmo foi treinado e testado no mesmo conjunto de observações. Destaca-se ainda que, ao aplicar um modelo de econometria, em situações de previsão como essa, a taxa de erro de treinamento pode ser, por muitas vezes, demasiadamente otimista, tendendo a subestimar a taxa de erro de teste. Logo, se a acurácia é de 78,79%, a taxa de erro de treinamento é de 21,21% (100% - 78,79%).

Para melhorar a acurácia do modelo de regressão logit, foi dividido os dados em treinamento e teste, e, em seguida, o ajustamento foi feito usando os dados de treinamento, analisando as previsões no conjunto de dados teste. Em seguida, foi feito novamente a regressão logit e a matriz de confusão. Agora, o modelo de regressão logística é ajustado usando apenas o subconjunto das observações que correspondem à base de treinamento, e as previsões são calculadas usando a base teste. Novamente, dada às previsões obtidas a partir da fronteira de decisão linear com um limiar de 0,5, agora para uma base de teste contendo 802 indivíduos, a

matriz de confusão da Tabela 05. A Tabela 04 irá apresentar a regressão logística estimada com a base de treinamento.

Tabela 5 - Estimação do algoritmo de regressão logística, com *Machine Learning* - Base de treinamento

Variáveis	**Coeficientes**	**Erro-Padrão**	**Odds Radio**
Intercepto	3.102298***	0.638900	22.2490214
Sexo	- 0.278491*	0.121669	0.7569252
Idade	-0.031689***	0.005959	0.9688075
Ler escreve	-0.646303**	0.228810	0.5239795
Número de pessoas	0.947222***	0.102059	2.5785375
Número de Crianças	-0.028823	0.128403	0.9715886
Número de Adultos	-0.715384***	0.118933	0.4890045
Número de Idosos	-1.674924***	0.197017	0.1873225
Energia	-0.135835	0.188833	0.8729867
Lixo	-0.597032*	0.253910	0.5504431
Cômodos	-0.109072**	0.039685	0.8966662
Instrução	-0.381133***	0.044221	0.6830868
Raça	-0.073682	0.112559	0.9289667
Telhado	-0.600942***	0.179895	0.5482950
Tipo de Situação	0.338631	0.219027	1.4030257
Domicilio Próprio	-0.234748 .	0.135803	0.7907703
Parede	-0.206251	0.241217	0.8136289
Sem Instrução	-0.546597*	0.246092	0.5789168

Fonte: Elaboração própria com base nos dados da POF 2017-2018.
Nota (1) nível de significância: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’

Tabela 6 - Matriz de Confusão da regressão logit, para *Machine Learning*

		Status padrão verdadeiro		
		Não (0)	Sim (1)	Total
Status padrão previsto	Não (0)	448	100	548
	Sim (1)	62	192	253
	Total	510	292	802

Fonte: Elaboração própria com base nos dados da POF.

Portanto, o modelo previu corretamente 192 indivíduos pobres e 448 indivíduos não pobres, um total de 640 previsões corretas. Ou seja, previu corretamente 79,8% e 20,19% incorreto. Pode-se perceber que a taxa de erro foi um pouco menor agora com a base teste.

Tendo isso em conta, são comparados os algoritmos, e selecionado aquele com a melhor previsão relacionada à etapa de predição do processo dos modelos de ML. Os critérios adotados para selecionar o algoritmo com a melhor análise preditiva são: *accuracy*, *sensitivity*, *specifity* e AUC ROC. A Tabela 6 sumariza os resultados dos indicadores de desempenho.

De acordo com a Tabela 6, inicialmente, observa-se que as acurácias dos modelos de penalização tiveram melhores acurácias, destacando a melhor performance foi de *LASSO, elastic net e ridge*, seguidos da Regressão logit. De acordo com James *et al.*, (2013), o desempenho geral de um classificador, resumido em todos os limites possíveis, é dado pela área sob a curva ROC (AUC).

Conforme Kuhn e Johson (2013), a curva ROC também pode ser empregada para uma avaliação quantitativa do modelo. Um modelo perfeito que separasse completamente as duas classes teria 100% de sensibilidade e especificidade. Graficamente, a curva ROC seria um único passo entre (0, 0) e (0, 1) e permaneceria constante de (0, 1) a (1, 1). A AUC ROC para tal modelo seria um.

Avaliam-se a AUC ROC e a *accuracy* dos modelos para seleção dos melhores algoritmos, e as melhores precisões são os métodos de penalização de *LASSO*, *elastic net*, *ridge*; regressão logistica; SVM; métodos de árvore de decisão: *random forest* e *bagging.*

Tabela 7 - Estimações dos algoritmos de *Machine Learning*

Algoritmo		**Critérios de avaliação do desempenho**				
		Accuracy	Sensitivity	Specificity	AUC ROC	IC 95% (AUC)
Regressão Linear		0.8042394	0.8191126	0.7838900	0.8617	0.835-0.8884
Penalized Methods	$Ridge^1$	0.8067332	0.8122867	0.7917485	0.8619	0.8352-0.8886
	$Lasso^2$	0.8104738	0.8088737	0.7917485	0.857	0.8298-0.8842
	$Elastic\ Net^3$	0.8092269	0.8122867	0.7897839	0.8575	0.8303-0.8846
Regressão Logistica		0.8042394	0.8054608	0.7956778	0.8629	0.8361-0.8896
KNN		0.6346633	0.5665529	0.3202358	0.5481	0.5202-0.5759
SVM		0.7942643	0.6048110	0.9021526	0.7535	0.7225-0.7844
Decision Tree Based Methods	*C. Tree*	0.7406484	0.5979730	0.8241107	0.711	0.6785-0.7436
	Bagging	0.7643392	0.7905405	0.7351779	0.8231	0.7935-0.8528
	Random Forest	0.7755611	0.7770270	0.7608696	0.8342	0.8057-0.8626
	Boosting	0.7568579	0.7697595	0.7769080	0.8437	0.8153-0.8721

Fonte: Elaboração própria com base nos dados da POF 2017-2018.
Nota: $1\lambda_{Ridge} = 0{,}01\ e\ MSE = 0{,}159293;\ 2\ \lambda_{Lasso} = 0{,}01\ e\ MSE = 0{,}1622937;\ 3\ \lambda_{EN} = 0{,}01\ e\ MSE = 0{,}1606532, \alpha = 0{,}5; K = 2$

Ressalta-se que grande parte dos algoritmos tiveram performances parecidas. Kuhn e Johson (2013) aconselham que seja feita inicialmente a comparação dos modelos baseados em termos de performance e sejam ponderados alguns benefícios como a interpretabilidade do

algoritmo, a complexidade computacional e a facilidade de implementação. Silva (2019) salienta que, em um exemplo de escolha de modelo final, os pesquisadores necessitam avaliar, primeiro, os modelos mais flexíveis e menos interpretáveis como o SVM, o *boosting* e o *random forest*. Em seguida, investigar os métodos mais simples como o *LASSO* e o *ridge*. Caso ambos os modelos sejam equivalentes em termos de performance, o pesquisador deve optar pelo algoritmo mais simples que se assemelha aos modelos mais complexos. Com relação a maior *accuracy*, o *LASSO* previu corretamente 81 % dos indivíduos da base teste (pobres e não pobres), já a Regressão Logística aponta para uma precisão geral do modelo de 80,4% dos indivíduos pobres.

As performances obtidas pelos 11 métodos, desde os mais tradicionais (regressão linear e logística) aos mais específicos de ML (SVM e Métodos baseados em árvores) foram semelhantes. Contudo, os modelos de penalização tiveram melhores *accuracy*, e os modelos tradicionais tiveram melhor desempenho na AUC ROC. Levando em consideração esses resultados, apenas cinco deles apresentaram, em conjunto, os melhores desempenhos de previsão: os métodos de penalização *LASSO*, *ridge*, *elastic net*, Regressão logística e SVM.

Das três melhores *accuracy* dos métodos de penalização, o que melhor previu pobreza foi *LASSO*, Analisando a AUC ROC, quem melhor previu pobreza foi a Regressão Logística. Para *LASSO*, pode-se dizer que dos 802 indivíduos que compõem o conjunto de base de teste em ambos status (pobre e não pobre), foi previsto corretamente através da medida da *accuracy* de 81%.A taxa de falso positivo (1 - *specifity*) é de 20,9%(1 – 79,1%), a taxa de verdadeiro positiva, medida pela *sensitivity*, é de 80,8%, assim sendo, 80,8% dos indivíduos pobres foram previstos corretamente.

Comparado com o estudo mais recente feito por Silva e França (2021), os autores obtiveram uma *accuracy* de 83% e *sensitivity* de 78%, resultados bem próximos dos encontrados nessa pesquisa. Para a Regressão Logística, conclui-se que foram classificados corretamente 80,4% das previsões, e que 80,5% dos indivíduos pobres foram previstos corretamente (*sensitivity*).

Por apresentarem valores muito próximos, pode-se definir que 80% dos indivíduos serão pobres, ou seja, dos 802 indivíduos da base teste, 641 serão pobres. Silva, Almeida e Ramalho (2020) utilizam o teste de *McNemar* de maneira individual nos modelos de ML depois de terem feito uma pré-seleção, identificando que os mesmos possuem medidas de desempenho com valores muito próximos. Assim, os autores utilizam o teste para averiguar se os modelos são ou não estatisticamente semelhantes.

Observando que os cinco modelos com melhores resultados tiveram medidas de desempenhos semelhantes, é imprescindível verificar a significância estatística das diferenças entre os algoritmos de ML. Assim, optou-se por utilizar o teste de *McNamer* e por avaliar quais dos onze modelos são ou não estatisticamente semelhantes. Como apresentado na Tabela 8, ao nível de significância de 5%, pode-se dizer que todos os modelos rejeitam a hipótese nula, dado que todos os p-valores são menores que 0,05.

Tabela 8 - Teste de *McNemar*

Teste *McNemar*			
Algoritmos		Teste *Chi-squared*	p-valor
Penalized Methods	$Ridge$	13,192	0,002811
	$Lasso$	15,429	8,568e-05
	$Elastic\ Net$	15,266	9,337e-05
Regressão Logistica		4,05	0,04417
KNN		111,72	<2,2e-16
SVM		24,824	6,28e-07
Decision Tree Based Methods	*C. Tree*	16,905	3,931e-05
	Bagging	5,418	0,01993
	Random Forest	8,2902	0,003986
	Boosting	115,38	<2,2e-16

Fonte: Elaboração própria com base nos dados da POF 2017-2018.

Para este estudo, a hipótese nula (H_0) do Teste de *McNemar* é que os algoritmos aplicados para prever pobreza têm precisões iguais. A hipótese alternativa (H_1) é que os algoritmos aplicados para prever pobreza têm precisões diferentes. Portanto, existem evidencias para rejeitar a hipótese nula de que as distribuições das previsões são iguais para todos os algoritmos empregados.

5.1 Importância das variáveis

Nesta seção, pretende-se ranquear as variáveis mais importantes para os quatro modelos que tiveram os melhores resultados. A técnica foi aplicada em todos os modelos. Porém, na Tabela 9, estão os quatro modelos que tiveram melhor desempenho. Esse teste tem a finalidade de buscar evidências das variáveis que foram determinantes para a previsão de pobreza.

A importância das variáveis pode ser resumida como uma métrica que mensura o impacto das variáveis individualmente para cada método de aprendizagem. Ela é específica para o algoritmo individual e o seu valor comumente não tem uma interpretação causal ou mesmo estatística. Em síntese, a medida, na maioria das vezes, pode ser pensada como uma ordem de classificação de quais variáveis são as mais importantes para o modelo ajustado (GOLDSTEIN; NAVAR; CARTER, 2016).

Tabela 9 - Ranking da Importância das variáveis

	Ridge	***Lasso***	***Elastic Net***	***Logit***
1	Energia	Lixo	Lixo	Instrução
2	Lixo	N. pessoas	N. pessoas	N. pessoas
3	Parede	N. Crianças	N. Crianças	N. Idosos
4	Sem Instrução	Instrução	Instrução	Idade
5	N. adultos	Idade	Idade	N. adultos
6	Instrução	Telhado	Telhado	Sexo
7	Telhado	Tipo de Situação	Tipo de Situação	Cômodos
8	Ler e Escreve	Parede	Ler e escreve	Telhado
9	Tipo de Situação	Sexo	Parede	Lixo
10	Idade	Raça	Raça	Tipo de Situação
11	Sexo	Ler e escreve	Sexo	N. crianças
12	N. crianças	N. adultos	N. adultos	Ler e Escreve
13	N. Idosos	Cômodos	Cômodos	Parede
14	Domicilio Próprio	N. de Idosos	N. de Idosos	Energia
15	Raça	Sem Instrução	Sem Instrução	Domicilio próprio
16	Comodo	Energia	Energia	Sem Instrução
17	N. Pessoas	Domicilio Próprio	Domicilio Próprio	Raça

Fonte: Elaboração própria com base nos dados da POF 2017-2018.

A Tabela 9 mostra os rankings de importância das variáveis para diferentes algoritmos de predição de pobreza. Assim sendo, diferentes métodos produzem classificações semelhantes, mas também diferentes.

Para Goldstein, Navar e Carter (2016) os algoritmos de ML têm vantagens comparativas em relação aos métodos tradicionais de regressão, pois produzem previsões com resultado mais precisos. Nesse sentido, podem ser empregados para auxiliar o enfrentamento de problemas de preditores múltiplos e correlacionados, relacionamentos não lineares e interações entre preditores e endpoints, em grandes conjuntos de dados. Assim sendo, vale salientar que este

estudo propõe-se a adotar métodos de ML que possam gerar boas previsões, dadas as restrições, levando em consideração até que ponto os algoritmos conseguem prever o comportamento observado em uma nova base de dados relacionados ao problema de retenção.

Destaca-se que os dois modelos (*LASSO* e Regressão logística) que tiveram melhores resultados de previsão de pobreza para a amostra de dados da população cearense.

6 CONSIDERAÇÕES FINAIS

Este trabalho propôs construir modelos de predição utilizando técnicas de *Machine Learning* nos dados obtidos da POF 2017-2018 a fim de prever a pobreza. Com a ajuda do modelo de aprendizado de máquina, procurou prever a situação de pobreza dos domicílios com base nos dados, reconhecendo quais características são preditivas para a pobreza. Quanto melhor for a capacidade de identificar os pobres, mais provável será o sucesso dos programas de combate à pobreza.

A princípio, foi feito uma revisão dos indicadores de pobreza do estado do Ceará. Em seguida, uma comparação dos modelos de clássicos de probabilidade linear, probit e logit. Depois, optou-se por dividir a base em treinamento e teste, quando foi estimada a regressão logit com a base de dados treinamento e feito a matriz de confusão com a base teste. Em seguida, comparou-se as *accuracy*. Assim sendo, foi percebido que a *accuracy* teve melhor resultado quando usada a técnica de *Machine Learning* com menores níveis de previsões incorretas.

Posteriormente, *accuracy, sensitivity, specifity* e AUC ROC dos onze algoritmos de *Machine Learning* foram comparadas como critérios para avaliar e selecionar os algoritmos com a melhor performance preditiva. Para o problema de classificação, utilizou-se o total de previsões classificadas corretamente (*accuracy),* as taxas de pobres classificados (previstos) corretamente (*sensitivity*) e as taxas de não pobres classificados (previstos) corretamente *(specificity*), todas provenientes da *confusion matrix.*

Entende-se que os modelos que melhores previram a pobreza foram os modelos de *LASSO* e regressão logística. As medidas como *specifity* para *LASSO* foram de 79,1%. Logo, a Taxa de Falso Positivo (TFP) (1-79,1%) foi de 20,9%, e para regressão logística, a *specifity* foi de 79,5% e uma taxa de falso positivo de 20,5%. Finalmente, observa-se a *sensitivity*. É possível prever uma taxa de pobres classificados corretamente de 80,5 % (*sensitivity*) para o modelo logístico, e para *LASSO* 80,8% (*sensitivity*).

Conclui-se que, se considerar a proximidade dos valores das *sensitivity* dos modelos de regressão logística e *LASSO*, resulta pertinente dizer que 80% dos indivíduos observados na base da POF serão pobres no estado do Ceará. Para os dois modelos finais que tiveram melhores resultados, podemos listar as dez variáveis que mais impactaram. Nesse sentido, foram: lixo, número de pessoas, número de crianças, parede, instrução, idade, telhado, tipo de situação, sexo e idade. A proposição é relevante, pois, sabendo das variáveis de importância, que impactam diretamente, pode-se direcionar os investimentos nessas variáveis para prever a pobreza no Ceará.

Entende-se que coleta de lixo, quantidade de residentes em um domicílio, número de crianças residentes e grau de instrução podem impactar no status dos indivíduos serem pobres. Por meio dos estudos de previsão de pobreza no Ceará, pretende-se dar maior suporte para políticas que atuem diretamente na pobreza. Além disso, trata-se de um estudo com comprovações científicas de previsões de pobreza no Ceará, o que pode incentivar os governos e formuladores de políticas nessa área a atuarem efetivamente para a redução dos números de indivíduos pobres no estado.

Dentre as limitações deste estudo, destaca-se a pouca quantidade de trabalhos com previsão de pobreza através de métodos de ML tanto no Brasil quanto para o estado Ceará comparando algoritmos de ML. Por isso, acredita-se também que, no geral, o presente estudo é capaz de servir como contribuição à literatura da área, sobretudo, com o objetivo de estabelecer estratégias de redução dos níveis de pobreza. Assim, recomenda-se que futuros trabalhos ampliem o que foi até aqui apresentado através de informações de outros estados brasileiros ou mesmo pensando um recorte pelas diferentes regiões do país.

REFERÊNCIAS

ADEDOKUN, Omolola; BURGESS, Wilella. Analysis of Paired Dichotomous Data: A Gentle Introduction to the McNemar Test in SPSS. **Journal Of Multidisciplinary Evaluation**, Estados Unido, v. 8, n. 17, p. 125-131, jan. 2012. Disponível em: https://www.researchgate.net/publication/288523843_Analysis_of_paired_dichotomous_data_A_gentle_introduction_to_the_McNemar_TestinSPSS. Acesso em: 14 set. 2022.

DABÚS, Andrés. **Pobreza en Argentina : un análisis predictivo utilizando herramientas de machine learning**. (2020). Tese (Mestrado em Economia) - Departamento de Economia, Universidad de San Andrés, Bahía Blanca, 2020. Disponível em: https://repositorio.udesa.edu.ar/jspui/bitstream/10908/18489/1/%5bP%5d%5bW%5d%20T.M.%20Eco.%20Dab%c3%bas%2c%20Andr%c3%a9s.pdf. Acesso em: 05 ago. 2021.

FERREIRA, Mário Boto *et al*. Using artificial intelligence to overcome over-indebtedness and fight poverty. **Journal of Business Research**, Portugal, v. 131, p. 411-425, 2021. Disponível em: https://www.sciencedirect.com/science/article/pii/S0148296320306949. Acesso em: 06 set. 2022.

FEURER, Matthias *et al*. Efficient and Robust Automated Machine Learning. *In*: NEURAL INFORMATION PROCESSING SYSTEMS, MONTREAL, 28, 2015, Montreal. **Anais** […]. Montreal: Universidade de Freiburg, 2015. p. 1-9. Disponível em: https://papers.nips.cc/paper/2015/file/11d0e6287202fced83f79975ec59a3a6-Paper.pdf. Acesso em: 12 set. 2022.

GOLDSTEIN, Benjamin A.; NAVAR, Ann Marie; CARTER, Rickey E.. Moving beyond regression techniques in cardiovascular risk prediction: applying machine learning to address analytic challenges. **European Heart Journal**, [S.L.], p. 1805-1814, 19 jul. 2016. Oxford University Press (OUP). http://dx.doi.org/10.1093/eurheartj/ehw302. Disponível em: https://academic.oup.com/eurheartj/article/38/23/1805/3056931. Acesso em: 06 set. 2022.

GUJARATI, Damodar N.; PORTER, Dawn C. Econometria básica. 5. ed. Porto Alegre: Amgh Editora, 2011.

HASTIE, Trevor; TIBSHIRANI, Robert; FRIEDMAN, Jerome. The elements of statistical learning: data mining, inference, and prediction. 2. ed. New York: Springer, 2009.

HAWASS, N e. Comparing the sensitivities and specificities of two diagnostic procedures performed on the same group of patients. **The British Journal Of Radiology**, [S.L.], v. 70, n. 832, p. 360-366, abr. 1997. British Institute of Radiology. Disponível em: https://pubmed.ncbi.nlm.nih.gov/9166071/. Acesso em:12 set. 2022.

IZBICKI, Rafael; SANTOS, Tiago Mendonça. Machine learning sob a ótica estatística - Uma abordagem preditivista para a estatística com exemplos em R. 2019. Disponivel em: http://www.rizbicki.ufscar.br/sml.pdf. Acesso em: 30 ago. 2021.

JAMES, Gareth; WITTEN, Daniele; HASTIE, Trevor; TIBSHIRANI, Robert. An Introduction to statistical learning – with applications in R. New York: Springer, 2013.

JANIESCH, Christian; ZSCHECH, Patrick; HEINRICH, Kai. Machine learning and deep learning. **Electronic Markets**, v. 31, n. 3, p. 685-695, 2021. Springer Science and Business Media LLC. Disponível em: https://link.springer.com/content/pdf/10.1007/s12525-021-00475-2.pdf. Acesso em: 08 jul. 2022.

JEAN, Neal *et al*. Combining satellite imagery and machine learning to predict poverty. **Science,** v. 353, n. 6301, p. 790-794, 2016. Disponível em: https://pubmed.ncbi.nlm.nih.gov/27540167/. Acesso em: 12 jan. 2022.

KAMBUYA, Pisacha. Better Model Selection for Poverty Targeting through Machine Learning: A Case Study in Thailand. **Thailand and The World Economy**, v. 38, n. 1, p. 91-116, 2020. Disponível em: https://so05.tci-thaijo.org/index.php/TER/article/view/183260. Acesso em: 21 jul. 2021.

KUHN, Max; JOHNSON, Kjell. Applied predictive modeling. 1. Ed. New York: Springer, 2013.

LANTZ, B. Machine learning with R. 2. ed. Birmingham: Packt Publishing, 2013.

LI, Guie *et al*. A comparison of machine learning approaches for identifying high-poverty counties: Robust features of DMSP/OLS night-time light imagery. **International journal of remote sensing**, v. 40, n. 15, p. 5716-5736, 2019. Disponível em: https://www.tandfonline.com/doi/abs/10.1080/01431161.2019.1580820. Acesso em: 26 jun. 2021.

MAHESH, Batta. Machine Learning Algorithms - A Review: 1. **International Journal Of Science And Research (Ijsr)**, Chhattisgarh, v. 9, n. 1, p. 380-386, 2020. Disponível em: https://www.researchgate.net/publication/344717762_Machine_Learning_Algorithms_-A_Review. Acesso em: 8 jul. 2022.

MCBRIDE, Linden; NICHOLS, Austin. Retooling poverty targeting using out-of-sample validation and machine learning. **The World Bank Economic Review**, v. 32, n. 3, p. 531-550, 2018. Disponível em: https://openknowledge.worldbank.org/handle/10986/33525. Acesso em: 08 jan. 2022.

MOHAMUD, Jama Hussein; GEREK, Omer Nazih. Poverty level characterization via feature selection and machine learning. *In*: SIGNAL PROCESSING AND COMMUNICATIONS APPLICATIONS CONFERENCE (SIU*)*. 27, 2019, Sivas. **Anais** […]. Sivas: IEEE, 2019. p. 1-4. Disponível em: https://ieeexplore.ieee.org/abstract/document/8806548. Acesso em: 24 ago. 2021.

NIU, Tong; CHEN, Yimin; YUAN, Yuan. Measuring urban poverty using multi-source data and a random forest algorithm: A case study in Guangzhou. **Sustainable Cities and Society**, v. 54, p. 102014, 2020. Disponível em: https://www.sciencedirect.com/science/article/abs/pii/S2210670720300019?via%3Dihub. Acesso em: 09 ago 2021.

PODSIADLO, Mariusz; RYBINSKI, Henryk. Financial time series forecasting using rough sets with time-weighted rule voting. **Expert Systems With Applications**, [S.L.], v. 66, p. 219-233, dez. 2016. Elsevier BV. Disponível em:

https://www.sciencedirect.com/science/article/pii/S0957417416304651?casa_token=MWICP8w6FakAAAAA:yYlF5WNAhsPFnVdRu7y1OcM8FP7Fu0qvWyF_NkPAM3nCjwmSAX7SQh3tcgfGs0BycKLOkd9beoGC. Acesso em: 08 set. 2022.

Programa das Nações Unidas para o Desenvolvimento. Regional Human Development Report 2021. Trapped: High Inequality and Low Growth in Latin America and the Caribbean. Nova Iorque: PNUD, 2021. Disponível em: https://www.undp.org/latin-america/publications/regional-human-development-report-2021-trapped-high-inequality-and-low-growth-latin-america-and-caribbean. Acesso em: 08 Jun. 2021.

RODRIGUES, Luciana de Oliveira; OLIVEIRA, Jimmy Lima de; SALES, Raquel da Silva; BADAGNAN, Thaisa França; OLIVEIRA, Victor Hugo de; SILVA, Vitor Hugo Miro Couto. **Indicadores Sociais do Ceará 2019**. Fortaleza: Instituto de Pesquisa e Estratégia Econômica do Ceará – Ipece, 2021. Disponível em: file:///C:/Users/User/Desktop/Qualifica%C3%A7%C3%A3o/artigos/IPECE_Rodrigo,%20Oliveira,%20Sales%20et%20al.pdf. Acesso em: 24 jan. 2022.

SIHOMBING, Pardomuan Robinson; ARSANI, Ade Marsinta. Comparison of Machine Learning Methods in Classifying Poverty in Indonesia in 2018**. Jurnal Teknik Informatika (JUTIF)**, v. 2, n. 1, p. 51-56, 2021. Disponível em: https://www.neliti.com/publications/495390/comparison-of-machine-learning-methods-in-classifying-poverty-in-indonesia-in-20. Acesso em: 08 ago 2022.

SILVA, Andrea Ferreira. **Ensaios sobre Economia Aplicada: Doações eleitorais, compras públicas, análise de políticas afirmativas e reprovação no Ensino Superior.** (2019). Tese (Doutorado em Economia) – Programa de Pós-Graduação em Economia, Universidade Federal da Paraíba, 2018. Disponível em: https://repositorio.ufpb.br/jspui/handle/123456789/20028?locale=pt_BR. Acesso em: 24 ago. 2021.

SILVA, Andréa Ferreira da; ALMEIDA, Aléssio Tony Cavalcanti de; RAMALHO, Hilton Martins de Brito. Predição do Risco de Reprovação no Ensino Superior Usando Algoritmos de Machine Le-arning. **Teoria e Prática em Administração**, [S.L.], v. 10, n. 2, p. 58-80, 7 ago. 2020. Portal de Periodicos UFPB. http://dx.doi.org/10.21714/2238-104x2020v10i2-51124. Disponível em: https://periodicos.ufpb.br/index.php/tpa/article/view/51124. Acesso em: 07 set. 2022.

SILVA, Vitor Hugo Miro C.; ARAUJO, Natália Carvalho. Indicadores de renda e pobreza no Ceará em 2020: o que dizem os dados da PNAD Covid-19. **Desenvolvimento Econômico em Foco.** Fortaleza, p. 1-5. mar. 2021. Disponível em: https://lepcaen.ufc.br/wp-content/uploads/2021/03/lep-deemfoco-31mar2021.pdf. Acesso em: 24 out. 2022.

SILVA, Vitor Hugo Miro Couto; FRANÇA, João Mario Santos. Modelos de machine learning na classificação de pobreza: uma aplicação para o estado do Ceará. *In:* Encontro Nacional de Economia, 49, 2021. **Anais** [...]. 2021. Disponível em: https://www.anpec.org.br/encontro/2021/submissao/files_I/i12-e0b926132a1ee744b58e387c542588ef.pdf. Acesso em: 22 dez. 2021.

SOHNESEN, Thomas Pave; STENDER, Niels. Is random forest a superior methodology for predicting poverty? An empirical assessment. **Poverty & Public Policy**, v. 9, n. 1, p. 118-133, 2017.

SOUZA, Alex. **Algoritmo SVM (Máquina de Vetores de Suporte) a partir de exemplos e código (Python e R) ALEX SOUZA**. 2019. Disponível em: https://pessoalex.wordpress.com/2019/04/10/algoritmo-svm-maquina-de-vetores-de-suporte-a-partir-de-exemplos-e-codigo-python-e-r. Acesso em: 26 jun. 2021.

THOPLAN, Ruben. Random forests for poverty classification. **International Journal of Sciences: Basic and Applied Research (IJSBAR)**, North America, v. 17, n. 2, p. 252-259, 2014. Disponível em: https://www.gssrr.org/index.php/JournalOfBasicAndApplied/article/view/2574/1855. Acesso em: 24 jan. 2022.

UNZER, Maicon Basilio Teixeira. ***Machine learning*** **para o apoio ao diagnóstico de depressão.** 2020. Trabalho de conclusão do curso - Bacharel em Ciência da Computação, Universidade Feevale. 2020. Disponível em: https://tconline.feevale.br/tc/files/0001_4981.pdf. Acesso em: 02 nov. 2021.

VERME, Paolo. Which Model for Poverty Predictions?. *In:* GLO DISCUSSION PAPER, 486, 2020, Hamburgo. **Anais** […]. Hamburgo: Global Labor Organization (GLO), 2020. p. 1-14. Disponível em: https://www.econstor.eu/bitstream/10419/213811/1/GLO-DP-0468.pdf. Acesso em: 24 ago. 2021.

WIJAYA, Dedy Rahman *et al*. Estimating city-level poverty rate based on e-commerce data with machine learning. **Electronic Commerce Research**, Chicago, v. 22, n. 1, p. 195-221, 2020. Disponível em: https://link.springer.com/content/pdf/10.1007/s10660-020-09424-1.pdf?pdf=button%20sticky. Acesso em: 08 jun. 2021.

WOOLDRIDGE, Jeffrey M. Introdução à econometria: uma abordagem moderna. 4. Ed. São Paulo: Thomson Learning, 2007.

ZHAO, Xizhi *et al*. Estimation of poverty using random forest regression with multi-source data: A case study in Bangladesh. **Remote Sensing**, v. 11, n. 4, p. 375-393, 2019. Disponível em: https://www.mdpi.com/2072-4292/11/4/375#metrics. Acesso em: 11 fev. 2021.