UNIVERSIDADE FEDERAL DE GOIÁS (UFG)

INSTITUTO DE INFORMÁTICA (INF)

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO (PPGCC)

JURANDIR JUNIOR DE DEUS DA SILVA

Interpretabilidade de Modelos de Aprendizado de Máquina: Uma Abordagem baseada em Árvores de Decisão

Goiânia
2023

UNIVERSIDADE FEDERAL DE GOIÁS
INSTITUTO DE INFORMÁTICA

TERMO DE CIÊNCIA E DE AUTORIZAÇÃO (TECA) PARA DISPONIBILIZAR VERSÕES ELETRÔNICAS DE TESES

E DISSERTAÇÕES NA BIBLIOTECA DIGITAL DA UFG

Na qualidade de titular dos direitos de autor, autorizo a Universidade Federal de Goiás (UFG) a disponibilizar, gratuitamente, por meio da Biblioteca Digital de Teses e Dissertações (BDTD/UFG), regulamentada pela Resolução CEPEC nº 832/2007, sem ressarcimento dos direitos autorais, de acordo com a Lei 9.610/98, o documento conforme permissões assinaladas abaixo, para fins de leitura, impressão e/ou download, a título de divulgação da produção científica brasileira, a partir desta data.

O conteúdo das Teses e Dissertações disponibilizado na BDTD/UFG é de responsabilidade exclusiva do autor. Ao encaminhar o produto final, o autor(a) e o(a) orientador(a) firmam o compromisso de que o trabalho não contém nenhuma violação de quaisquer direitos autorais ou outro direito de terceiros.

1. Identificação do material bibliográfico

[X] Dissertação [ ] Tese [ ] Outro*:_____

*No caso de mestrado/doutorado profissional, indique o formato do Trabalho de Conclusão de Curso, permitido no documento de área, correspondente ao programa de pós-graduação, orientado pela legislação vigente da CAPES.

Exemplos: Estudo de caso ou Revisão sistemática ou outros formatos.

2. Nome completo do autor

Jurandir Junior de Deus da Silva

3. Título do trabalho

Interpretabilidade de Modelos de Aprendizado de Máquina: Uma Abordagem baseada em Árvores de Decisão

4. Informações de acesso ao documento (este campo deve ser preenchido pelo orientador)

Concorda com a liberação total do documento [X] SIM [ ] NÃO[1]

[1] Neste caso o documento será embargado por até um ano a partir da data de defesa. Após esse período, a possível disponibilização ocorrerá apenas mediante:
a) consulta ao(à) autor(a) e ao(à) orientador(a);
b) novo Termo de Ciência e de Autorização (TECA) assinado e inserido no arquivo da tese ou dissertação.
O documento não será disponibilizado durante o período de embargo.
Casos de embargo:
- Solicitação de registro de patente;
- Submissão de artigo em revista científica;
- Publicação como capítulo de livro;
- Publicação da dissertação/tese em livro.

Obs. Este termo deverá ser assinado no SEI pelo orientador e pelo autor.

Documento assinado eletronicamente por **Rogerio Lopes Salvini**, **Professor do Magistério Superior**, em 20/10/2023, às 15:09, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Jurandir Junior De Deus Da Silva**, **Discente**, em 20/10/2023, às 15:47, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

A autenticidade deste documento pode ser conferida no site https://sei.ufg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **4141339** e o código CRC **0560BB59**.

JURANDIR JUNIOR DE DEUS DA SILVA

Interpretabilidade de Modelos de Aprendizado de Máquina: Uma Abordagem baseada em Árvores de Decisão

Dissertação apresentada ao Programa de Pós–Graduação em Ciência da Computação do Instituto de Informática da Universidade Federal de Goiás, como requisito parcial para obtenção do título de Mestre em Ciência da Computação .

Área de concentração: Ciência da Computação .

Linha de pesquisação: Sistemas Inteligentes e Aplicações.

Orientador: Prof. Dr. Rogerio Lopes Salvini

Goiânia
2023

Ficha de identificação da obra elaborada pelo autor, através do
Programa de Geração Automática do Sistema de Bibliotecas da UFG.

Silva, Jurandir Junior de Deus da
Interpretabilidade de Modelos de Aprendizado de Máquina: Uma Abordagem baseada em Árvores de Decisão [manuscrito] / Jurandir Junior de Deus da Silva. - 2023.
83 f.: il.

Orientador: Prof. Dr. Rogerio Lopes Salvini.
Dissertação (Mestrado) - Universidade Federal de Goiás, Instituto de Informática (INF), Programa de Pós-Graduação em Ciência da Computação, Goiânia, 2023.
Bibliografia.
Inclui tabelas, lista de figuras, lista de tabelas.

1. Aprendizado de Máquina. 2. Interpretabilidade. 3. Árvores de Decisão. 4. Explicação Contrafactual. I. Salvini, Rogerio Lopes, orient. II. Título.

CDU 004

UNIVERSIDADE FEDERAL DE GOIÁS

INSTITUTO DE INFORMÁTICA

ATA DE DEFESA DE DISSERTAÇÃO

Ata nº **19/2023** da sessão de Defesa de Dissertação de **Jurandir Junior de Deus da Silva**, que confere o título de Mestre em **Ciência da Computação**, na área de concentração em **Ciência da Computação.**

Aos vinte e dois dias do mês de setembro de dois mil e vinte e três, a partir das catorze horas, na sala 150 do Instituto de Informática, realizou-se a sessão pública de Defesa de Dissertação intitulada "**Interpretabilidade e Prognóstico de Modelos de Aprendizado de Máquina: Uma Abordagem baseada em Árvores de Decisão**". Os trabalhos foram instalados pelo Orientador, Professor Doutor Rogerio Lopes Salvini (INF/UFG) com a participação dos demais membros da Banca Examinadora: Professora Doutora Nádia Félix Felipe da Silva (INF/UFG), membro titular interno; e Professor Doutor Eduardo José Aguilar Alonso (ICT/UNIFAL), membro titular externo; cuja participação ocorreu através de videoconferência. Durante a arguição os membros da banca não fizeram sugestão de alteração do título do trabalho. A Banca Examinadora reuniu-se em sessão secreta a fim de concluir o julgamento da Dissertação, tendo sido o candidato **aprovado** pelos seus membros. Proclamados os resultados pelo Professor Doutor Rogerio Lopes Salvini, Presidente da Banca Examinadora, foram encerrados os trabalhos e, para constar, lavrou-se a presente ata que é assinada pelos Membros da Banca Examinadora, aos vinte e dois dias do mês de setembro de dois mil e vinte e três.

TÍTULO SUGERIDO PELA BANCA

Interpretabilidade de Modelos de Aprendizado de Máquina: Uma Abordagem baseada em Árvores de Decisão

Documento assinado eletronicamente por **Rogerio Lopes Salvini**, **Professor do Magistério Superior**, em 22/09/2023, às 16:17, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Nadia Felix Felipe Da Silva**, **Professor do Magistério Superior**, em 22/09/2023, às 16:17, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Eduardo José Aguilar Alonso**, **Usuário Externo**, em 22/09/2023, às 16:22, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Jurandir Junior De Deus Da Silva**, **Discente**, em 22/09/2023, às 17:00, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

A autenticidade deste documento pode ser conferida no site https://sei.ufg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **4002210** e o código CRC **10457813**.

Referência: Processo nº 23070.047831/2023-11 SEI nº 4002210

Sumário

Lista de Figuras

Lista de Tabelas

Resumo

da Silva, Jurandir. **Interpretabilidade de Modelos de Aprendizado de Máquina: Uma Abordagem baseada em Árvores de Decisão**. Goiânia, 2023 . 84p. Dissertação de Mestrado. Instituto de Informática, Universidade Federal de Goiás.

A interpretabilidade é definida como a capacidade de um ser humano entender por que um modelo de IA toma determinadas decisões. A interpretabilidade pode ser alcançada por meio do uso de modelos interpretáveis, como regressão linear e árvores de decisão, e por métodos de interpretação agnósticos de modelo, que tratam qualquer modelo preditor como uma "caixa-preta". Outro conceito relacionado à interpretabilidade é o de Explicações Contrafactuais, que mostram as mudanças mínimas nas entradas que levariam a resultados diferentes, fornecendo uma compreensão mais profunda das decisões do modelo. A abordagem proposta neste trabalho explora o poder explicativo das Árvores de Decisão para criar um método que oferece explicações mais concisas e explicações contrafactuais. Os resultados do estudo indicam que as Árvores de Decisão não apenas explicam o “porquê” das decisões do modelo, mas também mostram como diferentes valores de atributos poderiam resultar em saídas alternativas.

Palavras–chave

Aprendizado de Máquina, Interpretabilidade, Árvores de Decisão, Explicação Contrafactual

Abstract

da Silva, Jurandir. **Interpretability of Machine Learning Models: An Approach Based on Decision Trees.**. Goiânia, 2023 . 84p. MSc. Dissertation. Instituto de Informática, Universidade Federal de Goiás.

Interpretability is defined as the ability of a human to understand why an AI model makes certain decisions. Interpretability can be achieved through the use of interpretable models, such as linear regression and decision trees, and through model-agnostic interpretation methods, which treat any predictive model as a "black box". Another concept related to interpretability is that of Counterfactual Explanations, which show the minimal changes in inputs that would lead to different results, providing a deeper understanding of the model's decisions. The approach proposed in this work exploits the explanatory power of Decision Trees to create a method that offers more concise explanations and counterfactual explanations. The results of the study indicate that Decision Trees not only explain the "why" of model decisions, but also show how different attribute values could result in alternative outputs.

Keywords

Machine Learning, Interpretability, Decision Trees, Counterfactual Explanation

CAPÍTULO 1

Introdução

Este capítulo apresenta os elementos essenciais para a definição do objeto de pesquisa, a saber: a Seção 1.1 contextualiza o assunto da pesquisa e o problema de pesquisa; a Seção 1.2 apresenta a justificativa da pesquisa; a Seção 1.3 expõe o objetivo geral e os objetivos específicos. Por fim, a Seção 1.5 compreende a estrutura deste documento.

1.1 Contextualização

Molnar [24] define Aprendizado de Máquina (AM) como "um conjunto de métodos que os computadores usam para fazer e melhorar previsões ou comportamentos baseados em base de dados". No entanto, é natural que surjam questões importantes relacionadas à confiabilidade de modelos de AM, principalmente quando se trata de tarefas que têm impacto direto na sociedade.

Ainda segundo Molnar [24], os computadores geralmente não explicam suas previsões, isso acaba se tornando um barreira para a adoção de aprendizado de máquina. Nesse contexto, surgem as técnicas de interpretabilidade de modelos. Apesar de não existir uma definição matemática para o que é interpretabilidade, ela pode ser assumida como "o grau em que um ser humano pode compreender a causa de uma decisão", ou ainda "o grau em que um ser humano pode prever de forma consistente o resultado do modelo" [23].

A interpretabilidade de modelos pode ser alcançada de duas formas [24]: a partir da utilização de modelos interpretáveis, sendo essa a forma mais fácil, em que são utilizados um subconjunto de algoritmos que já criam modelos inerentemente interpretáveis, exemplos são a regressão linear, regressão logística, árvores de decisão, dentre outros; e os métodos de interpretação agnósticos de modelo , que trata qualquer modelo como um modelo de caixa-preta, sendo ele interpretável ou não.

Segundo Ribeiro [31], a grande vantagem de se utilizar métodos de interpretação agnósticos de modelo, foco deste trabalho, é sua flexibilidade.

Esses métodos utilizam modelos interpretáveis para explicar qualquer modelo de aprendizado de máquina, desde árvores de decisão a redes neurais profundas. O mesmo autor, em [30], propõe o método *Local Interpretable Model-Agnostic Explanations* (LIME), que é agnóstico de modelo e que vem a ser, até o presente momento, o estado da arte em interpretabilidade de modelos.

Todavia, os principais métodos de interpretabilidade de modelos, como LIME [30] e o SHAP (*SHapley Additive exPlanations*) [20], fazem uso de modelos lineares para ajustar o modelo a ser explicado, ou seja, reproduzir o comportamento do modelo a ser explicado. Tais modelos podem funcionar em grande parte dos casos mas também podem falhar quando a relação entre as *features* e o resultado é não linear. Já em trabalhos recentes [18] [29], modelos não lineares, como Árvores de Decisão, têm sido utilizados como uma alternativa aos modelos lineares e uma tentativa de se obter explicações mais concisas e fiéis.

A possibilidade de utilização de modelos não lineares, como as Árvores de Decisão, chega a ser considerada em [30]. No entanto, apesar de levantar tal possibilidade, os autores se limitam à utilização da Regressão Linear, além de não realizarem estudos sobre as vantagens e desvantagens de outros modelos.

Considerando a crescente necessidade, por parte dos usuários finais, de se entender melhor os resultados das previsões de modelos de AM, somado a os últimos avanços nos métodos de interpretabilidade de modelos, lança-se a possibilidade de estudos e exploração de modelos não lineares, como Àrvores de Decisão, a fim de se obter melhorias na explicação desses modelos.

Uma outra forma empregada na interpretabilidade de modelos é o conceito de Explicações Contrafactuais (do inglês, *Counterfactual Explanations*). Explicações Contrafactuais indicam a menor mudança nos valores das *features* que pode se traduzir em um resultado diferente. Tal técnica mostra-nos o que deve ser diferente em uma instância de entrada, para obter uma saída alternativa.

1.2 Justificativa

Os principais métodos de interpretabilidade de modelos, exemplificados pelo LIME [30] e pelo SHAP [20], frequentemente recorrem a modelos lineares para explicar as previsões dos modelos complexos. No entanto, esses métodos deixam de explorar alternativas que podem oferecer desempenho igual ou superior, ao mesmo tempo em que proporcionam diferentes níveis de interpretabilidade, como é o caso das Árvores de Decisão.

Além disso, as Árvores de Decisão podem ser exploradas como uma técnica de explicação contrafactual, ou seja, a menor alteração nos valores dos atri-

butos que aparecem na árvore que pode se traduzir em um resultado diferente. Isto acrescenta mais um nível para a interpretabilidade de previsão de modelos de Aprendizado de Máquina.

1.3 Objetivos

1.3.1 Objetivo geral

Este trabalho tem como objetivo a realização de um estudo acerca da Interpretabilidade de Modelos de Aprendizado de Máquina, com foco em dados tabulares, além do desenvolvimentos de técnicas e procedimentos capazes de se obter prognósticos a partir da interpretação de tais modelos.

1.3.2 Objetivos específicos

- mostrar que usar árvore de decisão como modelo interpretável pode ser mais acurado em relação ao modelo preditor do que a regressão linear
- mostrar que usar árvore de decisão como modelo interpretável apresenta os atributos mais importantes numa predição de forma mais concisa
- mostrar que a árvore de decisão pode ser usada para mostrar resultados alternativos

1.4 Trabalhos relacionados

Muitos trabalhos na literatura se utilizam de técnicas de interpretabilidade para atestar a fidelidades de modelos de Aprendizado de Máquina [30, 31, 17, 18, 34, 21, 16, 27, 7, 29]. No entanto, não foram encontrados trabalhos que realizassem a comparação entre modelos interpretáveis e suas vantagens/desvantagens. Os trabalhos com maior aderência ao escopo desta pesquisa são os de Li et al.[18] e Ranjbar [29].

O método apresentado em [29] trata-se também de uma modificação do LIME denominada ALIME. Enquanto o LIME gera novas instâncias em torno da instância a ser interpretada e treina um modelo linear, o ALIME usa um *autoencoder* para pesar os novos dados em torno da amostra. No trabalho, os autores também utilizam árvores de decisão como modelo interpretável. Assim como o trabalho anterior, os experimentos demonstram melhorias na fidelidade e aprimoramentos na interpretabilidade.

Em [18] os autores propõem o tree-LIME, um método baseado no LIME mas que realiza o ajuste local através de uma regressão para árvore de decisão, além de explicar como a métrica *mean absolute error* (MAE) pode ser utilizada para verificar a fidelidade das explicações. Uma desvantagem do tree-LIME é que, apesar de ser agnóstico de modelos, foi desenvolvido no contexto de um problema de regressão, não abarcando problemas de classificação. Os experimentos demonstram que a técnica leva a explicações mais precisas além de fornecer os resultados de forma mais intuitiva através de uma árvore. Além disso, a abordagem se mostra mais eficaz em tarefas de previsão de séries temporais.

Neste trabalho, os autores mostram que as mudanças no modelo explicável local provocam dois tipos de efeitos. Em primeiro lugar, usar um modelo de árvore não linear para substituir o modelo linear aumentou a fidelidade local. Em segundo lugar, em vez de usar um modelo linear como representação interpretável, a substituição leva à representação em formato de árvore.

Apesar de não explorar o conceito de "importância de *features*", os autores usam uma métrica, a MAE, como medida de fidelidade da interpretação. No trabalho, os autores discutem a importância da fidelidade na explicação de modelos, com ênfase nas diferenças entre problemas de classificação e regressão. Para a classificação, a fidelidade é definida como a porcentagem de exemplos do conjunto de teste em que o modelo explicável concorda com o modelo original, formalmente definida por:

$$Fidelidade_{classificacao} = \frac{N_{f=g}}{N} \tag{1-1}$$

No entanto, para a regressão, ainda não há uma definição estabelecida. Neste caso, a métrica comumente usada é o erro médio absoluto (MAE), que mede a diferença entre os resultados do modelo explicável e os resultados do modelo original, formalmente definida por:

$$Fidelidade_{regressao} = MAE_{g,f} = \frac{1}{n}\sum_{i=1}^{n}|g_i - f_i| \tag{1-2}$$

No trabalho de Li et al. [18] os autores apresentam 3 experimentos realizados com objetivo de avaliar o Tree-LIME em comparação com o LIME. Os resultados são apresentados na Tabela 1.1, através dos valores de fidelidade alcançados pelo Lime e pelo Tree-Lime, ao longo de 3 experimentos que se diferenciam pelo profundidade das árvores de explicação geradas, a citar: profundidade 3 no experimento 1, 4 no experimento 2 e 5 no experimento 5.

	Experimento 1	Experimento 2	Experimento 3
LIME	9,64	6,43	11,28
Tree-LIME	6,22	3,66	3,63

Tabela 1.1: *Comparação de fidelidade com MAE entre LIME e Tree-LIME.* ***Fonte:*** *Li et al. [18]*

Li e colegas [18] enfatizam a importância da profundidade da árvore como parâmetro crucial em sua abordagem. Essa profundidade impacta diretamente o equilíbrio entre a fidelidade do explicador e sua capacidade de interpretação. Através de experimentos que avaliaram profundidades de árvore de 3, 4 e 5 em grupos de 50 instâncias cada, o autor concluiu que uma profundidade de 5 resulta no melhor desempenho em termos de fidelidade. Esses experimentos validaram a hipótese do autor sobre a influência da profundidade da árvore. Em aplicações do mundo real, o autor sugere que esse parâmetro seja configurado como 4 ou 5, levando em consideração as preferências dos usuários.

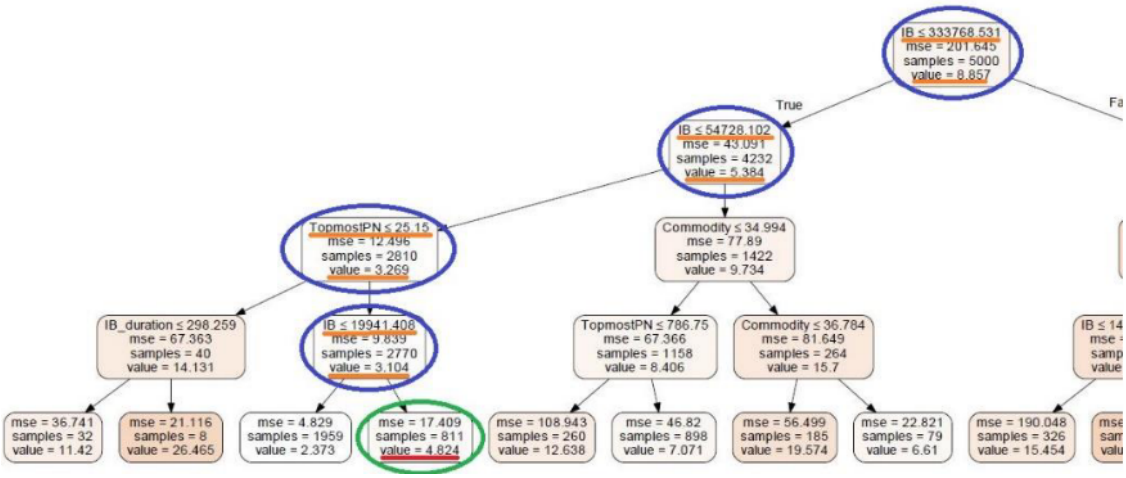

Figura 1.1: *Representação do Tree-LIME.* ***Fonte:*** *Li et al. [18]*

Na Figura 1.1, o tree-LIME resulta em uma representação interpretável baseada em uma árvore de decisão. A árvore de decisão é inerentemente explicável e, essencialmente, consiste em regras de decisão. Os autores demarcam, através dos círculos em azul, o caminho percorrido pela instância analisada e destacam, através do círculo em verde, o resultado da predição. As *features* que aparecem no caminho (“IB” e “TopmosPN”) são então ditas como as mais importantes.

1.5 Estrutura do Documento

O restante deste documento está organizado da seguinte maneira: o Capítulo 2 aborda o referencial teórico, apresentando os conceitos de modelos interpretáveis e o LIME; o Capítulo 3 descreve a proposta desta pesquisa; o Capítulo 4 apresenta os resultados dos experimentos realizados; o Capítulos 5 conclusão deste trabalho.

CAPÍTULO 2

Interpretabilidade de Modelos de Aprendizado de Máquina

Este capítulo apresenta o embasamento teórico necessário para o entendimento da pesquisa. A Seção 2.1 aborda conceitos relativos a modelos de Aprendizado de Máquina, com as definições para *dataset* e modelos; a Seção 2.2 discorre sobre Modelos interpretáveis; Na Seção 2.3 são apresentados conceitos e métodos agnósticos de modelo; Por fim, na Seção 2.4 o LIME é apresentado;

2.1 Modelos de Aprendizado de Máquina

Segundo Zhou [35], o **Aprendizado de Máquina** pode ser definido como "a técnica que melhora o desempenho de sistemas aprendendo através da experiência via métodos computacionais". Tais experiências se apresentam em estado de dados. Dessa forma, a principal tarefa do Aprendizado de Máquina é desenvolver **algoritmos** capazes de construir modelos a partir do conjunto de dados. Chamaremos o conjunto de dados no restante deste trabalho pelo seu termo equivalente, mais comumente utilizado, em língua inglesa, *dataset*.

2.1.1 Dados tabulares

Dado o problema a ser atacado, o ***dataset*** pode ser formado a partir da seleção de registros, também conhecidos como **instâncias** ou **objetos**, onde cada um contém a descrição de um evento. Formalmente, podemos definir [36]:

$$D = \{x_1, x_2, \ldots, x_m\} \tag{2-1}$$

onde, D é o *dataset* de tamanho m, formado pelas instâncias $x_{1\ldots m}$. Por sua vez, cada instância x_i é um vetor de r dimensões definido como:

$$x_i = (x_{i1}, x_{i2}, \ldots, x_{ir}) \in X \tag{2-2}$$

onde cada x_{ij} representa o valor da j-ésima **característica**, ou ***feature***, da instância x_i e X é o espaço de amostra. Em tarefas de Aprendizado de Máquina Supervisionado, assume-se a necessidade de realizar a predição de pelo menos uma dessas características, que passa a se chamar **variável alvo** ou ***target***. Assim, a equação 2-2 torna-se:

$$x_i = (x_{i1}, x_{i2}, \ldots, x_{is}, y_i) \tag{2-3}$$

onde $s = r - 1$ é a dimensão do problema e y_i é a variável alvo a ser predita para o objeto x_i, com $y_i \in Y$ o espaço de variáveis alvo. Quando o domínio de Y é discreto o problema definido por x_i é de classificação, quando o domínio é contínuo o problema é de regressão.

2.1.2 Modelo

De acordo com Ertel [9], a tarefa do Aprendizado de Máquina consiste em gerar uma função a partir dos dados coletados e classificados. Zhou [36] formaliza tal processo afirmando que um problema de predição pode ser definido pelo mapeamento

$$f : X \mapsto Y, \tag{2-4}$$

do espaço de entrada X para o espaço de saída Y, por meio de um *dataset* $\{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$ [36]. No contexto de Aprendizado de Máquina, damos o nome de **modelo** à função f.

Um modelo f treinado é capaz de realizar a predição de um exemplo qualquer $x \in X$, fazendo $\hat{y} = f(x)$, onde $\hat{y}$ é o resposta da predição. O processo de construção de um modelo a partir de um algoritmo de Aprendizado de Máquina é chamado de **treinamento**. Nessa fase, é comum que o *dataset* seja dividido em 2 subconjuntos disjuntos denominados conjunto de **treinamento** e conjunto de **teste**. O conjunto de treinamento é usado para ajustar o modelo, extraindo padrões e regras sobre os dados. Após o treinamento o modelo é então chamado de hipótese [35]. O conjunto de teste é usado para avaliar a generalização do modelo gerado, ou seja, qual é a expectativa do modelo predizer corretamente um novo caso ainda não visto.

2.2 Modelos Interpretáveis

De acordo com a perspectiva apresentada por Lipton [19], a interpretabilidade de um modelo está relacionada à capacidade de um ser humano compreender o modelo como um todo, em uma única análise abrangente. Isso implica a necessidade de uma visão global de como o modelo treinado toma suas decisões, levando em consideração não apenas as características envolvidas, mas também cada componente aprendido, como pesos, estrutura e outros parâmetros que definem o funcionamento do modelo.

Em termos práticos, a interpretabilidade de um modelo se traduz na capacidade de explicar de forma clara e intuitiva como as decisões são tomadas, de modo que um especialista ou usuário possa compreender o raciocínio por trás das previsões ou classificações do modelo.

Modelos que se enquadram nessa definição de interpretabilidade incluem a Regressão Linear, que é conhecida por sua simplicidade e transparência, bem como as Árvores de Decisão, que podem ser visualizadas de forma intuitiva como um conjunto de regras de decisão facilmente compreensíveis.

Aos modelos que não são interpretáveis dá-se o nome de **modelos caixa-preta**. Molnar [24] os definem como "um sistema que não revela seus mecanismos internos" e "modelos que não podem ser entendidos apenas observando seus parâmetros".

Portanto, a interpretabilidade não apenas torna os modelos mais acessíveis a especialistas não técnicos, mas também desempenha um papel crucial na confiança que podemos depositar em modelos de aprendizado de máquina em aplicações críticas.

2.2.1 Regressão Linear

Os modelos de Regressão Linear assumem que o problema a ser atacado é linear na entrada [13, 24]. Segundo Hastie [13], tais modelos são simples e na maioria das vezes entrega uma descrição adequada e interpretável sobre como a entrada afeta a saída do modelo. Tais modelos são capazes de realizar previsões a partir da soma ponderada das *features* de entrada. Além disso, permitem modelar a relação, ou dependência, entre o alvo do problema e as *features* . As relação podem então ser escritas através de uma única equação:

$$f(x) = \beta_0 + \sum_{j=1}^{p} x_j \beta_j \tag{2-5}$$

onde os β_j's são os parâmetros, ou coeficientes, da regressão e os x_j's referem-se aos valores dos atributos do exemplo x. β_0 é o erro ou a diferença entre a previsão e o resultado final. Dado um *dataset* qualquer, o conjunto de treino $\{(x_1,y_1),(x_2,y_2),\ldots,(x_m,y_m)\}$ é então usado para estimar os parâmetros β.

Segundo Hastie [13], o método de estimação mais popular é o método dos mínimos quadrados (em inglês, *least squares*), no qual escolhermos os coeficientes β que minimizam a soma residual dos quadrados (em inglês, *Residual Sum-of-Squares ou RSS*), com em:

$$\begin{aligned} RSS(\beta) &= \sum_{i=1}^{N}(y_i - f(x_i))^2 \\ &= \sum_{i=1}^{N}(y_i - \beta_0 - \sum_{i=j}^{p} x_{ij}\beta_j)^2 \end{aligned} \tag{2-6}$$

Interpretação da Regressão Linear

A interpretação da regressão linear é direta e depende do tipo das *features* correspondentes e pode ser feita da seguinte forma [24]:

- *features* numéricas: Um aumento de valor de uma *feature* x_k em uma unidade aumenta a previsão para y por β_k unidades quando todos os outros valores da *feature* permanecem fixos.
- *features* categóricas: Alterando a *feature* x_k da categoria de referência para a outra categoria aumenta a previsão para y em β_k quando todos as outras *features* permanecem fixas.

Dessa forma, podemos concluir que a interpretação de modelos de Regressão Linear se dá através da análise da variação dos valores de β_k, com o modelo previamente treinado.

2.2.2 Árvores de Decisão

Dentre os algoritmos para Aprendizado de Máquina, as Árvores de Decisão se destacam pelo alto poder de aprendizado, versatilidade e simplicidade. A versatilidade destes algoritmos se justifica pela gama de problemas em que podem ser aplicados, contando com algoritmos tanto para tarefas de regressão quanto para tarefas de classificação [9].

Existe uma grande variedade de algoritmos para construção de Árvores de Decisão, como o C4.5 [28], para problemas contínuos (regressão) e categóricos (classificação), e seu predecessor ID3 [28], para problemas categóricos, largamente utilizados. Além destes, enfatizaremos neste trabalho o CART (*Classification and Regression Trees*) [6] que, assim como o C4.5, também pode ser aplicado em problemas contínuos e categóricos.

O CART é um algoritmo de Aprendizado de Máquina que gera Árvores de Decisão através de um processo de particionamento recursivo binário, capaz de processar variáveis contínuas e categóricas [6]. Partindo da raiz, os dados são particionados em dois nós filhos que, seguindo o processo recursivo, são também particionados.

Nesse processo, cada instância do problema é alocada em apenas um subconjunto. Cada divisão define um nó de uma árvore que é criado a cada passo iterativo onde os nós intermediários são chamados de nós internos e os nós finais, formados pelos subconjuntos resultantes, são chamados de nós folhas. O resultado médio, que pode ser o valor encontrado no nó folha ou a média dos valores encontrados nos nós vizinhos, dos dados de treinamento são usados para prever o resultado em cada nó folha. Os modelos baseados em árvores de decisão utilizam valores de cortes para dividir os dados em vários subconjuntos [24].

Apesar do resultado final do CART ser uma Árvore de Decisão, tal algoritmo não gera apenas uma árvore. Ao contrário, ele gera uma sequência de árvores podadas aninhadas, onde cada uma é candidata a ser a árvore ótima [6].

Diferente de outros algoritmos como o C4.5, o CART não usa uma medida de desempenho interna para seleção das árvores. Ao invés disso, o desempenho é sempre medido em um conjunto de teste independente, e a seleção das árvores continua apenas após a avaliação no conjunto de teste.

O CART particiona as regras sempre da forma [6]:

x_i vai para a ESQUERDA se a CONDIÇÃO for verdadeira, caso contrário vai para a direita

onde x_i é a instância e a condição é definida como $x_{ij} <= C$, sendo C o ponto de corte, para dados contínuos. A escolha do ponto de corte C se dá através de um processo de busca de todas as divisões possíveis em uma variável de entrada e a seleção da divisão que resulta na maior redução da impureza, conforme medida pelo Índice de Gini, ou outra medida de impureza adequada ao problema [6, 24].

Para dados categóricos, a condição avaliar a pertinência do atributo x_{ij} a uma lista de valores, como em:

x_i *vai para a ESQUERDA se* $x_{ij} \in \{a,b,c\}$*, caso contrário vai para a direita*

onde $\{a,b,c\}$ é um conjunto de valores categóricos.

Conforme observado por Breiman [6], no âmbito do algoritmo CART, existem várias abordagens e métricas disponíveis para realizar a divisão dos dados. Estas incluem o índice Gini, Gini Simétrico, *Twoing* e *Twoing* Ordenado, entre outras. No entanto, neste trabalho, concentraremos nossa atenção no índice Gini, que se destaca como a métrica mais amplamente adotada e implementada na construção de árvores de decisão.

Interpretação da Árvore de Decisão

Além de simples e eficiente para extrair conhecimento, as Árvores de Decisão contam com outra vantagem com relação a outros algoritmos de Aprendizado de Máquina. Como apontado por Ertel [9], a Árvore de Decisão não apenas adquire conhecimento e o disponibiliza na forma de uma função ou modelo de predição, como também o armazena em uma estrutura de fácil entendimento e interpretação por parte de um humano, na forma de uma árvore como a exibida na Figura 2.1.

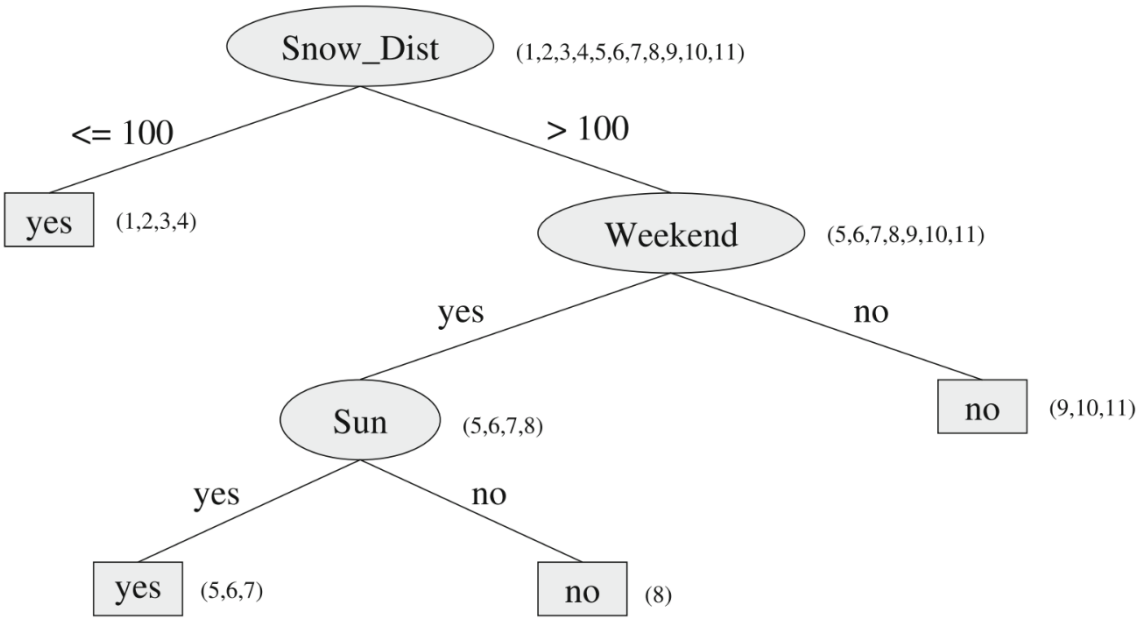

Figura 2.1: *Exemplo de Árvore de Decisão para o problema de classificação de condições para esquiar.* ***Fonte:*** *Ertel [9]*

A interpretação das árvores de decisão é relativamente simples e pode ser obtida seguindo os passos [24]:

1. Inicie no nó raiz;

2. Cada aresta é formada por um subconjunto, utilize os valores de referência desse subconjunto para avançar para o próximo nó;
3. Repita o passo anterior até alcançar um nó folha;
4. As arestas da árvore são conectadas pelo operador lógico 'AND'.
5. A ordem dos nós visitados segue uma lógico. Nós mais próximos da raiz tendem a ser mais importantes para a previsão do que os nós mais próximos do nó folha.

Ao final, o modelo gera uma sentença interpretável e que tem a forma:

Se o valor da *feature x* for [menor/maior] que o limite *c* AND... Então o resultado previsto é o valor médio de *y* das instâncias do nó folha.

Na Figura 2.1 é apresentado um exemplo de Árvore de Decisão treinada para prever se o clime será ou não propenso a esquiar. Os nós internos da árvore apresentada na figura representam as *features* do problema ("*Snow_Dist*", "*Weekend*" e "*Sun*"), nas folhas encontramos as duas possíveis classes, "yes" e "no". À direita de cada nó encontram-se as referências para os exemplos de treino correspondentes.

2.3 Interpretação agnóstica de modelos

Apesar de existirem modelos intrinsecamente interpretáveis, como os já citados Regressão Linear e Árvores de Decisão, existe também uma vasta gama de modelos que não são interpretáveis, como o *Support Vector Machines* (SVM) e as Redes Neurais Artificais [24]. Tais modelos, não interpretáveis, têm desempenho já comprovado e são largamente utilizados, principalmente para tratar problemas complexos.

Nesse contexto, métodos especializados em interpretação vêm sendo desenvolvidos com poder lidar com modelos treinados na forma de um modelo caixa-preta. A esses métodos dá-se o nome de "métodos agnósticos de modelo", pois seguem uma abordagem que independe das características do modelo, como estrutura e parâmetros [31].

Os métodos agnósticos de modelo extraem explicações através do treinamento de um modelo interpretável a partir das predições do modelo caixa-preta [8] [4]. Os métodos mais conhecidos são o LIME [30], que treina um modelo interpretável a partir de uma única observação, e o SHAP [20], que usa uma amostra

de mais de uma observação para calcular valores SHAP. Tais valores ajudam a determinar a contribuição relativa de cada *feature* para uma previsão específica e podem ser calculados a partir da criação de subconjuntos de incluem todas as combinações possíveis de *features*, excluindo a *feature* que se deseja explicar. Então o modelo é executado com o subconjunto criado e é calculada a diferença entre a os valores esperados e os previstos. Assim é possível calcular a contribuição de cada *feature* e então os valores SHAP, definido pela média ponderada das contribuições, com a ponderação dependendo do número de *feature* do subconjunto.

Ambos os métodos têm seus códigos-fonte abertos para uso e personalização. No entanto, além da possibilidade de se realizar a interpretação a partir de uma única observação, O LIME conta com a vantagem de ter sua implementação mais simples, necessitando apenas da função de predição do modelo a ser interpretado, enquanto o SHAP necessita de mais estruturas específicas do modelo. Por esses motivos, o LIME foi escolhido como método de estudo foco deste trabalho.

Além da interpretabilidade para uma única instância o LIME é também capaz de realizar a interpretação global por meio de uma amostra de instâncias. Diferente da interpretação para uma única instância, que tem por objetivo obter as *features* mais importantes para a predição desta, a interpretação global tem por objetivo obter um conhecimento mais profundo sobre as *features* mais importantes para o modelo de predição quando executado para uma instância qualquer, obtendo as *features* que serão importantes para a predição de qualquer instância, e não apenas uma. No entanto, o escopo deste trabalho se limita às interpretações locais, ou seja, de uma única instância.

2.4 LIME

O LIME (*Local Interpretable Model-Agnostic Explanations*) foi desenvolvido para explicar predições de qualquer tipo de classificador ou regressor, de forma fiel, aproximando-se localmente de um modelo linear interpretável, a citar, uma regressão linear [30].

A técnica consiste em utilizar um modelo preditor e uma instância do *dataset* para aproximar o comportamento do modelo preditor a um modelo mais simples e interpretável. Isso é realizado no LIME a partir de conceito de perturbação, onde o algoritmo realiza, de forma pontual, substituições *feature*-a-*feature* para valores do mesmo domínio, e utiliza essas saídas como entradas para o modelo original treinado.

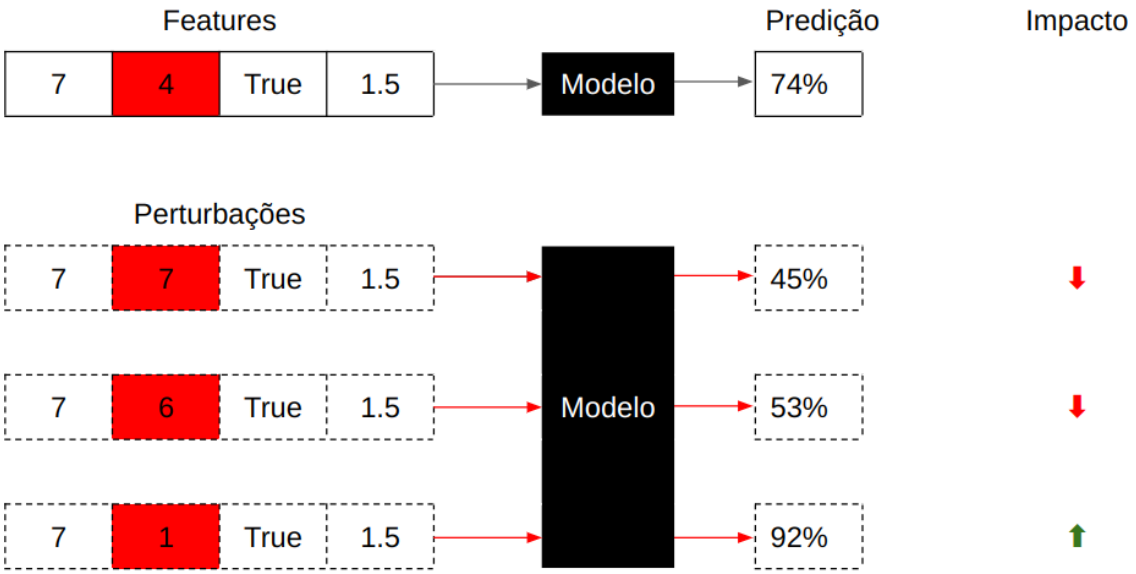

Figura 2.2: *Ilustração das perturbações geradas a partir de uma instância.* ***Fonte:*** *O autor*

Na Figura 2.2 podemos verificar a forma como as perturbações são usadas. No exemplo, quando o valor da *feature* destacada em vermelho aumenta o resultado da predição piora, o que caracteriza um impacto negativo desta perturbação. Já quando o valor diminui o resultado da predição melhora, caracterizando um impacto positivo.

É importante destacar que o método consiste em realizar as perturbações para todas as *features*, e não apenas uma. Isso possibilita uma análise geral acerca do comportamento do modelo em relação às variações em torno da instância e, ao mesmo tempo, medir o impacto das *features* na predição.

Apesar do LIME funcionar para dados de qualquer domínio (imagem, texto e dados tabulares), neste trabalho abordaremos apenas sua utilização para dados tabulares.

Nesses problemas, o LIME gera novas instâncias amostrando através da perturbação de cada *feature* individualmente, selecionando novos valores aleatórios extraídos de uma distribuição normal com média e desvio padrão retiradas do conjunto de valores da *feature*. Para as *features* categóricas as novas instâncias são geradas a partir da seleção aleatória de um novo valor a partir da distribuição da *feature*.

O conjunto de novas instâncias com pertubação e suas respectivas predições são então utilizadas para treinar um modelo interpretável e, dessa forma, a interpretação desse modelo corresponderá à interpretação da predição da instância original pelo modelo original. Como resultado, um analista poderá verificar qual a importância de cada *feature* e como ela colaborou para o resultado da pre-

visão.

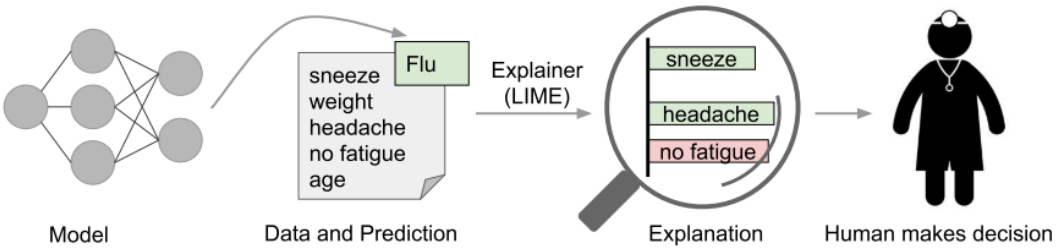

Figura 2.3: *Ilustração do fluxo de execução do LIME para explicação de uma predição.* ***Fonte:*** *Ribeiro et al. [30]*

Seguindo o fluxo da figura 2.3, o modelo classifica o estado do paciente como “resfriado” (*flu*, em inglês), de acordo com os valores de um determinado conjunto de *features*. Cada *feature* representa um sintoma ou um dado sobre o paciente. O LIME então identifica quais sintomas ou dados levaram o modelo a classificar o paciente como “resfriado”.

2.4.1 Representações de Dados Interpretáveis

O primeiro passo para compreender o funcionamento interno do LIME é discernir entre "*features*"(características) e "representações de dados interpretáveis". Em termos de interpretabilidade, é essencial usar uma representação compreensível por seres humanos, independentemente das features originais usadas pelo modelo que está sendo explicado, como mencionado em [30].

Por exemplo, ao explicar um classificador de texto, pode-se adotar uma representação interpretável, como um vetor binário que indica a presença ou ausência de uma palavra, mesmo que o modelo subjacente use uma representação mais complexa, como *embeddings* de palavras.

Em resumo, o LIME enfoca a tradução das complexas *features* ou representações usadas pelo modelo original em algo mais compreensível, tornando as explicações acessíveis para as pessoas, o que é fundamental para a interpretabilidade do modelo.

2.4.2 Escolha por Fidelidade e Interpretabilidade

O LIME tem como premissa a utilização de um segundo modelo para explicar o modelo a ser explicado. Dessa forma, pode-se definir um modelo $g \in G$, em que G representa o conjunto de possíveis modelos interpretáveis. Exemplos de modelos que podem ser utilizados são regressão linear e árvores de decisão [30].

Uma das limitações das técnicas de interpretabilidade está relacionada à dimensão das *features*. Basicamente, a capacidade de explicação é inversamente proporcional à quantidade de *features* que contribuem para a predição [30]. Por conta disso, até mesmo modelos tidos como interpretáveis podem se tornar não interpretáveis por seres humanos.

Dessa forma, como nem todo modelo $g \in G$ pode ser interpretável em todos os casos, os autores do LIME [30] adotam uma medida de complexidade, denotada por $\Omega(g)$, da explicação $g \in G$. Por exemplo, para árvores de decisão, $\Omega(g)$ pode ser a profundidade da árvore.

Na tarefa de interpretabilidade de um modelo podemos então denotar a instância a ser explicada como x, o modelo como $f : \mathbb{R}^d \longrightarrow \mathbb{R}$, com d igual à dimensão do problema, e uma medida de proximidade $\pi_x(z)$ que define a vizinhança em torno de x. Esses são os parâmetros para $\mathcal{L}(f, g, \pi_x)$, a medida do quanto impreciso g é para se aproximar de f na localidade de π_x.

Podemos então definir o processo de explicação usado pelo LIME como sendo o de minimizar $\mathcal{L}(f, g, \pi_x)$, mantendo uma **fidelidade** local, somado a $\Omega(g)$ baixo o suficiente para ser **interpretado** por humanos, ou seja:

$$\xi(x) = \underset{g \in G}{\operatorname{argmin}}\, \mathcal{L}(f, g, \pi_x) + \Omega(g) \tag{2-7}$$

Essa é a formulação utilizada e os autores optaram por modelos lineares esparsos como modelos interpretáveis g e usando perturbações para realizar a busca que define π_x.

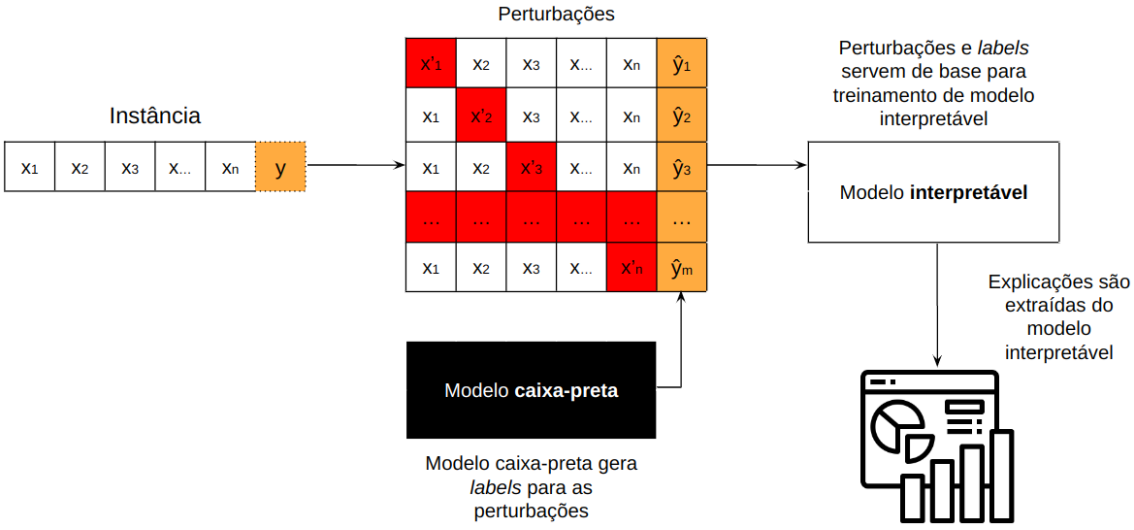

Figura 2.4: *Fluxo de execução do LIME para explicação de uma predição.* ***Fonte:*** *O autor.*

Na Figura 2.4 podemos acompanhar o fluxo de execução do LIME. Dada

uma instância x a ser explicada, o método gera as perturbações *feature*-a-*feature*, mantendo a maior parte dos valores inalterados. Os *targets* das perturbações são inferidas usando-se o modelo caixa-preta f. O novo *dataset* criado é então utilizado como base de treinamento para um modelo interpretável g. A partir dos parâmetros aprendidos pelo modelo interpretável g são extraídas as explicações. A ideia das perturbações consiste em mostrar que as *features* com valores perturbados e que afetam o resultado da predição são ditas como as mais importantes.

2.4.3 Amostragem para Exploração Local

A fim de aprender o comportamento de f enquanto as entradas variam, o LIME aproxima $\mathcal{L}(f, g, \pi_x)$ através de amostras ponderadas por $\pi_x(z)$. Instâncias em torno de x são escolhidas de forma uniformemente aleatória, obtendo uma nova amostra perturbada de dados z' e, então, $f(z)$, que é usado como *target* para o modelo interpretável. Dado esse novo conjunto $\mathcal{Z}$ a equação 2-7 é otimizada para obter uma explicação $\xi(x)$.

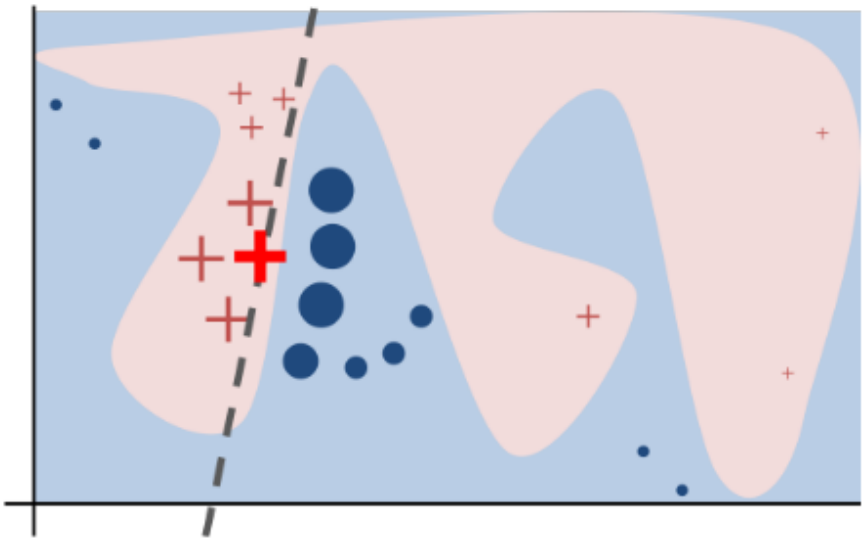

Figura 2.5: *Ilustração da aproximação local de um modelo linear a uma determinada instância.* ***Fonte:*** *Ribeiro et al. [30]*

Na Figura 2.5, a função f é representada pelas cores de fundo azul e rosa, tal função é desconhecida pelo LIME e não pode ser aproximada por um modelo linear. O símbolo "+" em destaque representa a instância a ser explicada. A linha tracejada representa a explicação obtida pela técnica.

O funcionamento básico da exploração local é apresentado na Figura 2.5. A técnica consiste em realizar uma amostragem em torno de x, gerando objetos próximos (alto valor para π_x) e distantes (baixo valor para π_x).

2.5 Explicações Contrafactuais

As explicações contrafactuais são representações hipotéticas que fornecem *insights* sobre como uma decisão ou resultado poderia ter sido diferente sob circunstâncias alternativas [11]. No contexto da IA, essas explicações têm o potencial de ajudar os usuários a compreender por que uma decisão específica foi tomada e quais alternativas estavam disponíveis. Isso é especialmente relevante em situações em que os sistemas de AM desempenham um papel crítico em decisões que afetam a vida das pessoas, como aprovações de empréstimos ou avaliações de candidaturas a empregos.

Segundo Molnar [24], uma explicação contrafactual de uma previsão descreve a menor alteração nos valores da *feature* que altera a previsão para uma saída predefinida. Tal definição é formalizada no trabalho de Guidotti [11] como:

> Dado um classificador f que realiza a predição $y = b(x)$ para uma instância x, uma explicação contrafactual consiste em uma instância x' de tal forma que a predição para f em x' é diferente para y, ou seja, $f(x') \neq y$, de tal forma que a diferença entre x e x' seja a mínima.

Além disso, as explicações contrafactuais também têm o potencial de empoderar os usuários, concedendo-lhes um senso maior de controle e agência sobre as decisões que os afetam. Quando os usuários compreendem os fatores que influenciaram uma decisão de IA, podem tomar decisões mais informadas e adotar ações apropriadas em resposta. Por exemplo, se um candidato a emprego receber uma rejeição de um sistema de recrutamento baseado em IA, uma explicação contrafactual pode destacar as qualificações ou experiências específicas que faltaram para uma contratação bem-sucedida. Com esse entendimento, o candidato pode buscar oportunidades para adquirir as habilidades necessárias ou procurar emprego em outros lugares. Isso não apenas promove a inclusividade, mas também fortalece a sensação de justiça e confiança nas decisões de IA.

Além de contribuir para a inclusividade e a justiça, as explicações contrafactuais de IA também podem aprimorar a responsabilidade e o cumprimento das regulamentações. Em muitos setores críticos, como saúde e finanças, os sistemas de IA estão sujeitos a requisitos legais e éticos rigorosos. As explicações contrafactuais desempenham um papel fundamental em garantir que esses sistemas cumpram essas regulamentações, proporcionando uma trilha clara do processo de tomada de decisão. Isso é particularmente importante quando as decisões tomadas pelos sistemas de IA podem ter consequências substanciais, como diagnósticos médicos ou determinações de pontuação de crédito.

Entretanto, a implementação efetiva de explicações contrafactuais de IA não é isenta de desafios. Um dos principais desafios é encontrar o equilíbrio certo entre transparência e desempenho. Sistemas de IA altamente transparentes podem sacrificar o desempenho, enquanto sistemas que priorizam o desempenho podem carecer de transparência. Encontrar essa harmonia é crucial para garantir que os sistemas de IA sejam tanto precisos quanto explicáveis.

Para superar esses desafios, os pesquisadores e desenvolvedores estão explorando várias abordagens. Uma delas envolve o uso de modelos de aprendizado de máquina mais interpretáveis, que possam fornecer explicações mais claras de suas decisões. Além disso, está havendo um esforço significativo para integrar explicações contrafactuais em sistemas de IA existentes, de modo a torná-los mais acessíveis e inclusivos.

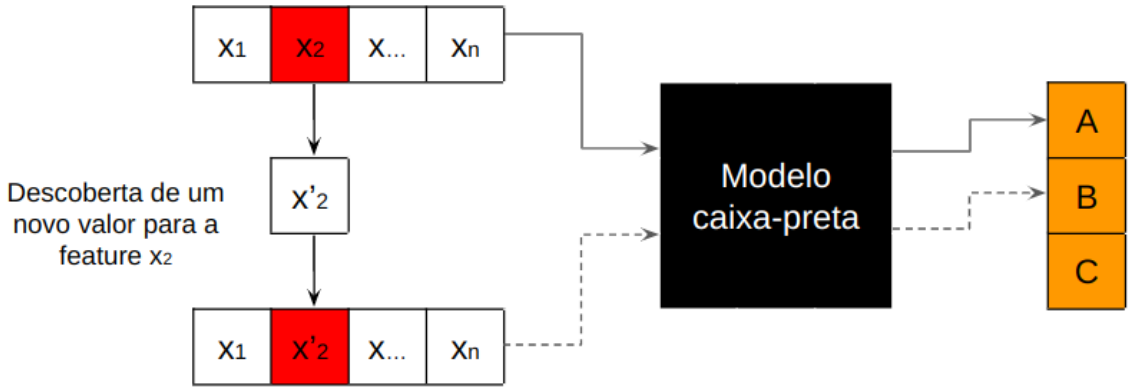

Figura 2.6: *Esquematização da explicações contrafactuais.* ***Fonte:*** *O autor*

A Figura 2.6 exemplifica a explicações contrafactual. Dada uma instância x, a explicações contrafactual é realizado encontrando-se pelo menos uma *feature* x_i que, uma vez alterado o seu valor, obtém um resultado diferente na previsão através do modelo caixa-preta.

Em resumo, as explicações contrafactuais de IA têm o potencial de abordar a falta de transparência e promover a inclusão em sistemas de IA. Elas revelam vieses, capacitam os usuários e fortalecem a responsabilidade, contribuindo para sistemas de IA mais justos e confiáveis. Embora desafios permaneçam, a pesquisa contínua e a colaboração entre acadêmicos, indústria e reguladores são essenciais para alcançar todo o potencial das explicações contrafactuais de IA.

CAPÍTULO 3

Método Proposto

A proposta deste trabalho baseia-se na hipótese, já levantada por Ribeiro [30], de que as Árvores de Decisão, como modelos interpretáveis, podem vir a "substituir" a já empregada Regressão Linear.

Molnar [24], além também levantar tal hipótese, lança luz ao tema ao demonstrar, teoricamente, como as Árvores de Decisão podem ser interpretadas a partir da sua estrutura e das importâncias das *features* que, a princípio, podem ser calculadas a partir do índice *Gini* para os problemas de classificação. Para os problemas de regressão consideraremos a MSE, como indicada por Archer [3].

Dessa forma, este trabalho foi realizado tendo como base o trabalho de Li e colegas [18], replicando configurações como o uso do algoritmo CART de Árvore de Decisão como modelo interpretável, com profundidade máxima da árvore igual a 5, como sugerido pelos autores para obtenção do melhor desempenho em termos de fidelidade.

3.1 Descrição do método proposto

A partir da metodologia empregada pelo LIME e a ideia de utilização de Árvores de Decisão propostas por Molnar [24] e Li et al. [18], foi-se desenvolvido uma nova abordagem de interpretabilidade e que expande as possibilidades de interpretação.

Na Figura 3.1 é apresentado um diagrama de todo o processo da nova abordagem.

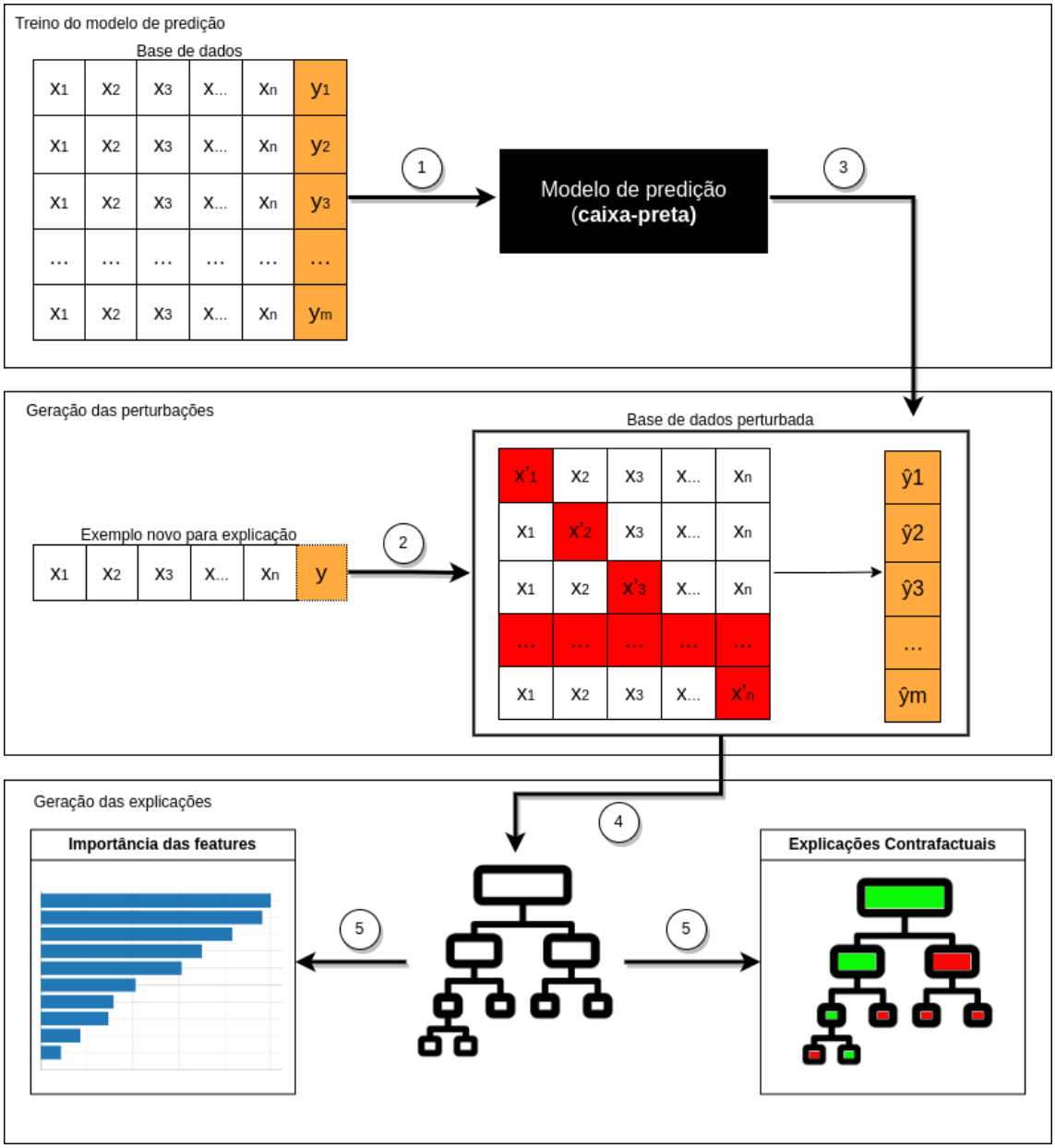

Figura 3.1: *Diagrama do processo completo do método proposto.*
***Fonte**: O autor.*

(1) Geração do modelo de predição: A primeira etapa do processo consiste em se treinar (ou obter) um modelo de predição a partir da base de dados subjacente ao problema. Independente de qual modelo de predição é gerado, ele será sempre considerado como caixa-preta, de forma que o método não depende informações ou mecanismos específicos por parte do modelo a ser explicado.

(2) Geração das Perturbações: A partir de um novo exemplo do domínio do problema realiza-se as perturbações, como explicadas na sessão 2.4, criando-se um novo *dataset* (sem variável alvo).Este será o exemplo a ser explicado.

(3) Predição do *dataset* perturbado: Utiliza-se o modelo obtido no passo 1 para

predição do *dataset* gerado na etapa 2. Dessa forma, obtém-se todas as informações necessárias, $\{(x_{1...n}, y_1), \ldots, (x_{1...n}, y_m)\}$ para treino de um novo modelo.

(4) Treinamento de uma Árvore de Decisão: Nessa etapa, um modelo de Árvore de Decisão é treinado a partir do *dataset* criado nas etapas anteriores.

(5) Resultado final: Ao final do processo, o método proposto gera duas saídas: as importâncias das *features* extraída a partir das métricas da Árvore de Decisão; e a Explicação Contrafactual.

3.2 Importância das *features* na Árvore de Decisão

A importância das *features* nas Árvores de Decisão, de uma forma geral, consiste em se verificar o quanto cada *features* colabora na tarefa de predição. Para isso, é calculado o ganho proporcionado pelas *features* ao ser escolhida para compor a condição de um nó da árvore. O cálculo do ganho pode ser realizado através do Índice *Gini* para problemas de classificação, e da MSE, para problemas de regressão.

Índice *Gini*

O Índice *Gini* é uma medida de pureza ou impureza que determina o quão puro é um nó na Árvore de Decisão com relação a uma classe específica após a divisão, levando-se em consideração uma *feature* em particular [22, 24]. Um nó t é dito maximamente **puro** quando é formado por elementos de uma única classe e **impuro** quando todas as classes que formam o *dataset* têm a mesma frequência relativa no nó t. A melhor divisão é aquela que aumenta a pureza dos conjuntos resultantes da divisão.

Dado um problema formado por um conjunto K de classes, com k valores distintos, o Índice *Gini* pode ser calculado, para um nó t qualquer, através da seguinte fórmula [5]:

$$Gini(t) = 1 - \sum_{i=1}^{k} p_i^2 \tag{3-1}$$

onde p_i é a frequência relativa da classe i no nó t. O valor **mínimo** é 0 e é alcançado quando o nó t atribui exclusivamente uma classe (nó puro). Já o valor **máximo** é $1 - 1/K$, alcançado quando as classes são uniformemente distribuídas (nó impuro).

O ganho gerado por um particionamento de um nó pai P em seus filhos à esquerda (E) e direita (D) é dado por:

$$G_{Gini}(P) = Gini(P) - qGini(E) - (1-q)Gini(D) \tag{3-2}$$

onde q é a fração de instâncias indo para a esquerda.

Segundo Breiman [6] e Molnar [24], a importância *Gini* de uma *feature* é calculada como o ganho total (normalizado) do critério trazido por essa *feature*, ou seja, o somatório dos seus ganhos dividido pelo número de ocorrências da *feature* na árvore. Dessa forma, podemos formular:

$$Gini\ Importance(f) = \frac{1}{|t_f|} \sum_{j \in t_f} G_{Gini}(j) \tag{3-3}$$

onde f é a *feature* a ser analisada e t_f o conjunto de todos os nós em que f aparece como divisor e, assim, $|t_f|$ é a quantidade de vezes que a *feature* f aparece na árvore.

Mean Squared Error - MSE

Como é possível verificar na seção anterior, o Índice *Gini* funciona a partir da frequência relativas das classes. Isso inviabiliza a métrica para problemas de regressão. Dessa forma, uma das métricas que podem sem empregadas em seu lugar é a MSE (*Mean Squared Error*). A MSE é uma média do erro quadrático entre os valores observados e os valores previstos, para dados contínuos, e pode ser definida como [2]:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{3-4}$$

onde n representa o número de elementos, y é o valor observado (variável alvo do conjunto de treino) e $\hat{y}$ é o valor previsto pelo modelo.

De acordo com Molnar [24], em problemas de classificação é possível medir a importância de uma *feature* na árvore de decisão através da soma do ganho em todos os nós que têm essa *feature* como divisor, ponderado pela fração dos dados de treino em cada nó de particionamento. Em outras palavras, medindo-se a diminuição na impureza causada pela divisão de um nó da árvore em dois nós filhos.

Anysz et at. [2] argumenta que a MSE pode ser utilizada para o cálculo da importância para variáveis independentes. A importância das *features* pode ser calculada percorrendo-se a árvore para cada *feature*.

Quando uma decisão é baseada em uma *feature* em um nó específico, a importância daquela *feature* é determinada somando-se a redução de erro, multiplicada pelo número de amostras que passaram por aquele nó. A redução de erro é calculada como a diferença na medida de impureza (MSE, neste caso) das amostras direcionadas para o nó, subtraindo as impurezas de seus nós filhos.

Podemos então formular o ganho gerado por um particionamento de um nó pai P em seus filhos à esquerda (E) e direita (D) como:

$$G_{mse}(P) = MSE(P) - qMSE(E) - (1-q)MSE(D) \tag{3-5}$$

onde q é a fração de instâncias indo para a esquerda.

O MSE é usado como medida de impureza para avaliar a qualidade da separação das amostras. Esse processo resulta em uma pontuação de importância para cada *feature*, indicando seu impacto nas previsões do modelo.

3.3 Explicações contrafactuais

O algoritmo CART realiza subdivisões binárias e a consequência disto é que todos os nós internos da Árvore de Decisão terão, obrigatoriamente, 2 nós filhos. Logo, para todos os casos, haverá uma explicação contrafactual que encontra um novo valor (ou classe) para a predição. Isso decorre do fato de que, nas Árvores de Decisão, podemos obter um novo valor de predição ao negarmos a condição do último nó interno, obtendo-se, assim, uma explicação contrafactual.

Além disso, o algoritmo CART permite que nós folha irmãos atribuam o mesmo valor (ou classe) para a predição. Nesses casos, consideramos que o nó pai pode se tornar um nó folha, pois seus dois filhos levam ao mesmo valor (ou classe). A busca por um resultado alternativo, então, se dará por "subir" um nível na árvore e visitar o nó folha mais próximo.

Seja, então, uma Árvore de Decisão g, com profundidade h (h níveis), gerada a partir das perturbações de uma instância x qualquer. A fim de se verificar qual *feature* deve ter seu valor atualizado de modo a alterar o caminho seguido na árvore g para se alcançar um outro resultado de predição da instância x, os passos são:

1. A partir do nó raiz, para cada nó da árvore, utilize os valores de referência para avançar para o próximo nó, considerando os valores de x;

2. Ao alcançar o nó folha F', tome a condição C encontrada no nó P, pai de F', como referência;

3. Seja F'' o nó irmão de F', seu valor (ou classe) será alcançado caso a *feature* executada em P seja alterada, negando (invertendo o valor de verdadeiro ou falso) a condição C;

4. Se o valor (ou classe) de F' for igual a F'', significa que ambos os nós atribuem o mesmo valor (ou classe). Considere P um nó folha e repita os passos anteriores; senão

5. A explicação contrafactual será dada pela identificação da ***feature*** encontrada no passo 3, juntamente com seu **novo valor** (que inverte o valor da condição C).

3.4 Implementação da abordagem proposta

A implementação da proposta e dos experimentos foi realizada tomando como base a metodologia empregada pelo LIME, aproveitando-se para isso a menor quantidade de códigos pré-existenes possível. Para fins de comparação com o LIME original, apenas a base de dados perturbada foi gerada pelo algoritmo do LIME. A geração das Árvores de Decisão, bem como o cálculo das importâncias das *features* e o cálculo da fidelidade da árvore foram implementados e realizados de forma independente e exterior ao LIME.

Para o desenvolvimento dos experimentos, foi utilizado um computador com as seguintes configurações: processador Intel core i7, 16GB de memória RAM, sistema operacional Ubuntu 22.04 de 64 bits. A linguagem de programação utilizada é a linguagem Python 3.9.

As bibliotecas que serão utilizadas neste experimento são:

- *NumPy* [12]: responsável pelo processamento de grandes matrizes, com implementação de métricas e métodos para transformação de dados
- *Pandas* [33]: ferramenta de análise de dados e que fornece estruturas de dados flexíveis e eficientes;
- *Matplotlib* [14]: biblioteca de baixo nível para criação de diagramas e gráficos bidimensionais;
- *Scikit-learn* [26]: biblioteca que fornece algoritmos para muitas tarefas padrão de aprendizado de máquina e mineração de dados.
- *LightGBM* [15]: O LightGBM é um framework de código aberto para aprendizado de máquina (machine learning) que é altamente otimizado para tarefas de classificação e regressão. (modelo usado para treinamento dos *datasets* ou seja, o modelo preditor a ser explicado)

CAPÍTULO 4

Resultados

Neste capítulo são apresentados os resultados obtidos através de experimentos realizados seguindo a abordagem apresentada no Capítulo 3. A Seção 4.1 discorre sobre os modelos e os *datasets* usados como estudos de casos nos experimentos. A Seção 4.2 descreve a metodologia empregada para se avaliar os resultados. A Seção 4.3 apresenta os resultados dos estudos de caso. A Seção 4.4 apresenta uma comparação entre as abordagens proposta neste trabalho e o LIME sob a perspectiva da fidelidade. A Seção 4.5 discorre sobre a importância das *features* e faz um comparativo entre as abordagens deste trablho e o LIME. Por fim, na Seção 4.6 são apresentados os principais avanços para interpretabilidade de modelos de Aprendizado de Máquina alcançados por este trabalho, as explicações contrafactuais.

4.1 Modelo de predição e *datasets* usados nos experimentos

A fim de atestar a validade da hipótese levantada neste trabalho, sobre o uso de Árvore de Decisão para interpretabilidade e explicação contrafactual de modelos de Aprendizado de Máquina, o algoritmo LightGBM [15] foi escolhido como modelo de predição a ser interpretado.

O método foi testado para tarefas de classificação e regressão com modelos LightGBM treinados em 4 *datasets* diferentes, a saber: "Íris" [10] e "*Wine*" [1] como problemas de classificação; "*Abalone*" [25] e "*Parkinsons Telemonitoring*" [32] como problemas de regressão. A escolha desses *datasets* se deu pela suas disponibilidades e larga adoção em *benchmarkings* e trabalhos da área de Aprendizado de Máquina e Inteligência Artificial. Na Tabela 4.1 são apresentadas as principais características dos *datasets* selecionadas.

	Datasets			
	Iris	(Wine)	(Abalone)	Parkinsons Telemonitoring
Tarefa	classificação	classificação	regressão	regressão
#Instâncias	150	178	4177	5875
#Features	4	13	8	19
Tipo das features	Real	Inteiro, Real	Categórico, Inteiro, Real	Inteiro, Real
#Classes	3	3	-	-
Classes balanceadas	Sim	Não	-	-

Tabela 4.1: *Informações básicas dos datasets utilizados nos experimentos de interpretabilidade.*

Conforme apontado por Ribeiro [31], a interpretabilidade de modelos de Aprendizado de Máquina não se preocupa em avaliar o desempenho do modelo preditor a ser explicado. No entanto, o autor ressalta que, para se alcançar explicações concisas, é necessário partir do pressuposto que o modelo de predição seja o mais preciso possível. Dessa forma, neste trabalho, foi utilizada validação cruzada para aferir o desempenho do modelo de predição nos *datasets* selecionados.

A validação cruzada foi realizada dividindo o *dataset* em 10 *folds*, de forma estratificada. Nas tabelas 4.2 e 4.3 são apresentados os resultados obtidos pelo modelo preditor nos *datasets*. As métricas apresentadas nas tabelas são as médias de acurácia e medida F1 macro para os problemas de classificação, e MAE (do inglês, erro absoluto médio) para os problemas de regressão, todas acompanhadas com seus respectivos desvios padrões. Uma vez obtidos os desempenhos dos modelos preditivos, os modelos foram retreinados utilizando a totalidade dos dados dos *datasets*, reproduzindo a situação de modelos em produção.

Dataset	Acurácia	Desvio padrão	F1	Desvio padrão
Wine	97,77	2,86	97,80	2,83
Iris	96,00	5,62	95,87	5,83

Tabela 4.2: *Resultado da validação cruzada para os modelos de* ***classificação****.*

Dataset	MAE	Desvio padrão
Abalone	1,50	0,06
Parkinsons Telemonitoring	1,28	0,05

Tabela 4.3: *Resultado da validação cruzada para os modelos de* ***regressão****.*

4.2 Avaliação do método proposto

Três tipos de avaliações são realizadas sobre os experimentos apresentados. A primeira irá comparar a fidelidade dos modelos interpretáveis com relação ao modelo preditor. A métrica de fidelidade será usada para verificar o quanto a regressão linear, usada pelo LIME, e a árvore de decisão, usada no método proposto, se assemelham às predições do modelo preditor. Para cada exemplo da base de dados será calculada a fidelidade das predições feitas pela regressão linear e pela árvore de decisão sobre a mesma base de dados perturbada gerada pelos exemplos. No caso da regressão linear, a base perturbada ainda será transformada por meio das operações de normalização e discretização dos dados da mesma forma que é feito no LIME. No final, será calculado a média de fidelidade dos exemplos da regressão linear e da árvore de decisão para ser comparada em cada experimento realizado.

A segunda avaliação dos experimentos será a comparação das variáveis tidas como as mais importantes para a explicação da predição de um exemplo, dadas pelos modelos interpretáveis (regressão linear e árvore de decisão). Dois exemplos serão selecionados aleatoriamente de cada base de dados e serão preditos pelo modelo preditor. Então, os modelos interpretáveis irão ranquear as variáveis que foram importantes para a predição dos exemplos. No caso da regressão linear, as primeiras 10 variáveis serão apresentadas, enquanto que na árvore de decisão todas as variáveis que forem nós da árvore serão apresentadas. As medidas de importância destas variáveis serão: o coeficiente da regressão linear, no caso do LIME, e o índice Gini ou o MSE, conforme a tarefa do problema, no caso do método proposto.

Como abordado por Ribeiro [31], a avaliação qualitativa da interpretabilidade depende da análise crítica de especialistas do domínio a fim de validar ou invalidar a explicação. Dessa forma, os resultados apresentados neste trabalho baseiam-se na comparação com as técnicas já existentes quando executadas com as mesmas instâncias e, além disso, a análise dos ganhos e vantagens obtidos.

Por fim, a terceira avaliação, exclusiva do método proposto, será realizada a partir dos grafos das árvores de interpretabilidade obtidas. Por meio da

estrutura das árvores será possível avaliar os caminhos possíveis que a predição poderá tomar, bem como quais condições devem ser satisfeitas para que se tenha resultados diferentes. Para os exemplos selecionados anteriormente, serão extraídos os caminhos do nó raiz até o nó folha da predição. Em seguida, será feita a busca pela *feature* mais próxima ao nó folha da predição que, ao ter o seu valor alterado, levará para um caminho que leva a um outro nó folha que altera o resultado da predição. Para mostrar que este caminho é possível, será buscado na base de dados um exemplo que tenha as características com os mesmos valores do exemplo original, exceto a *feature* com valor alterado, e cuja resposta seja a mesma da previsão alternativa dada pela árvore.

4.3 Estudos de casos

Nesta seção são apresentados os resultados dos experimentos realizados para os problemas de classificação e regressão. Os experimentos foram realizados com amostras de 150 exemplos obtidos aleatoriamente de cada *dataset*. No caso do *dataset Iris*, que já contém um total de 150 exemplos, foi-se considerado em sua totalidade. Para cada *dataset* foram selecionados aleatoriamente 2 exemplos, cujas explicações são exibidas por meio de gráficos, com saídas tanto da abordagem proposta neste trabalho quanto do LIME para fins de comparação.

Nas figuras das Árovres de Decisão, o caminho percorrido pela instância na tarefa de predição é destacado pela cor **verde**. *Features* com importâncias iguais a 0 (zero) são desconsideradas e não exibidas nas figuras com as importâncias calculadas a partir das árvores, uma vez que não contribuem para a tarefa de interpretabilidade.

Diferente das importâncias calculadas pelo LIME, que são os coeficientes da regressão linear, a importância gerada pelas Árvores de Decisão terão sempre sinal positivo, uma vez que se baseiam em medidas de impureza e/ou erro médio absoluto, que por sua vez não admitem valores negativos.

4.3.1 Iris dataset

Instância 1

Na Tabela 4.4 são apresentados os valores das *features* referentes à primeira instância selecionada aleatoriamente:

sepal_length	5.1
sepal_width	3.5
petal_length	1.4
petal_width	0.2
target	Iris-setosa

Tabela 4.4: *Instância 1 selecionada aleatoriamente do dataset Iris*

A Figura 4.1 exibe a explicação gerada pelo LIME da predição que classificou a instância como "Iris-setosa". No eixo das ordenadas encontram-se as *features* e no eixo das abscissas encontram-se suas respectivas importâncias, dadas pelos coeficientes da regressão linear. Cada *feature* é apresentada com informações acerca do intervalo, delimitado por "<", "<=", ">" ou ">=" em que esta alcança a máxima importância. As barras verdes apontadas para a direita indicam que a *feature* contribui para que a instância seja classificada corretamente de acordo com o seu *target* "Iris-setosa".

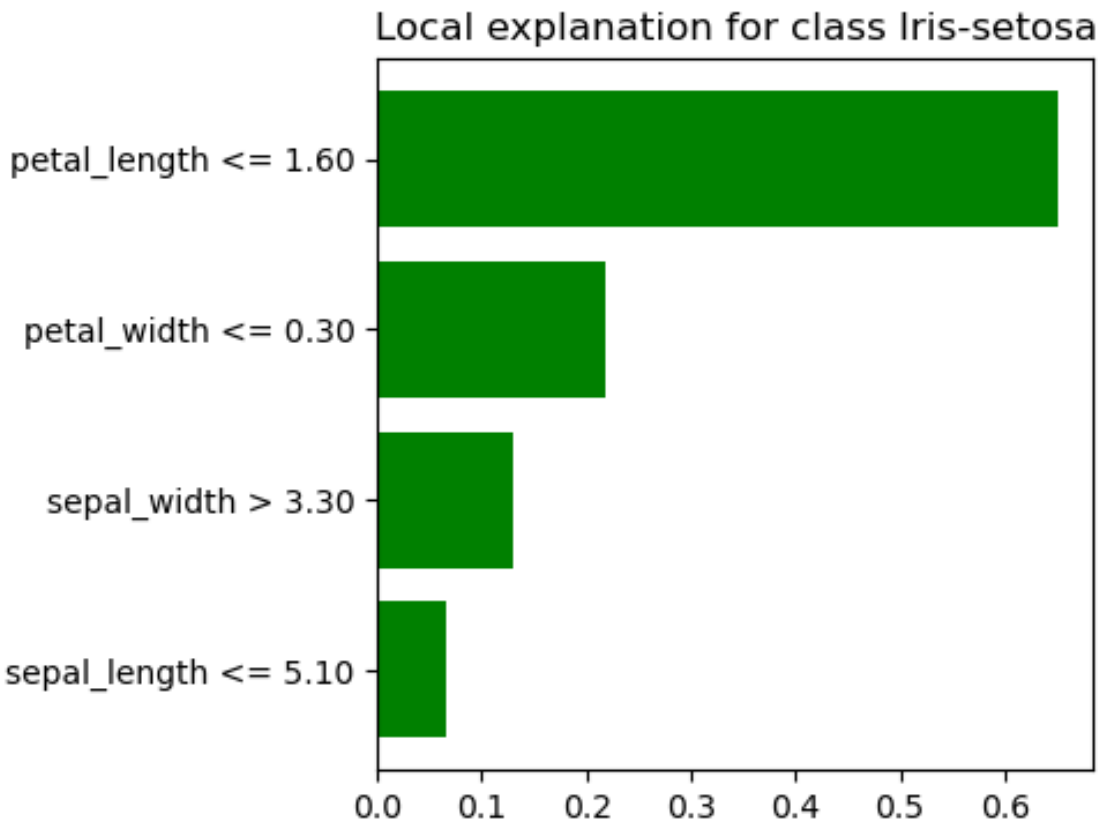

Figura 4.1: *Importância das features geradas pelo LIME.* ***Fonte****: O autor.*

A Figura 4.2 apresenta a importância das *features* de acordo com o método proposto, com as importâncias extraídas da Árvore de Decisão. Assim como no

resultado do LIME, no eixo das ordenadas encontram-se as *features* e no eixo das abscissas encontram-se suas respectivas importâncias, calculadas a partir do Índice *Gini* da árvore de decisão. Pelo método proposto, apenas as *features* que aparecem na árvore de decisão são apresentadas na Figura 4.2. Neste caso em particular, a feature "sepal_width" não aparece na árvore, assumindo-se que sua importância é zero, não sendo apresentada na figura.

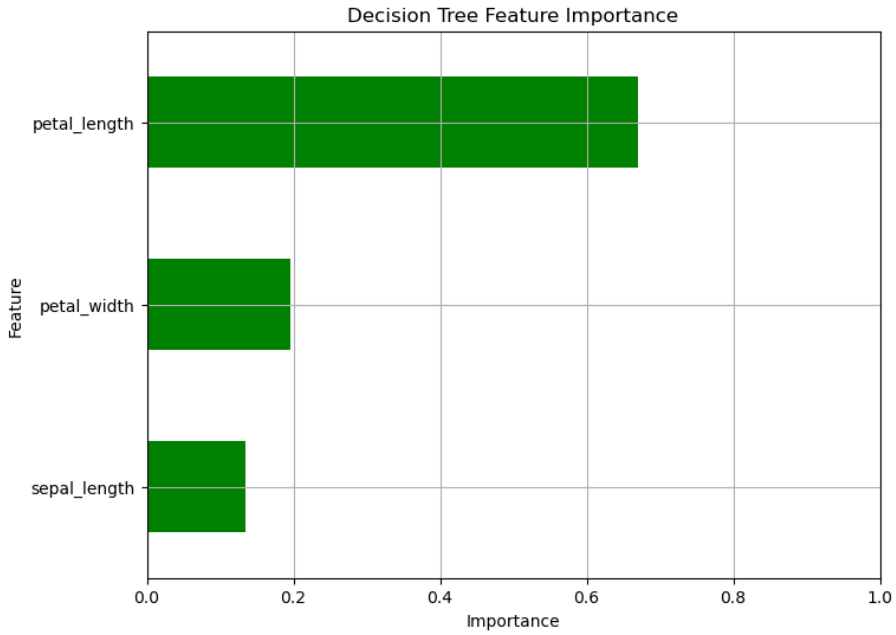

Figura 4.2: *Importância das features geradas pela Árvore de Decisão.* ***Fonte****: O autor.*

Por fim, a Figura 4.3 apresenta a Árvore de Decisão gerada a partir das perturbações da instância avaliada. O caminho destacado pela cor verde indica quais condições, expressas nos nós internos da árvore, e quais *features* foram mais importantes para predição. Ao todo 3 condições são executadas, avaliando-se os valores das *features* "petal_length" e "petal_width". O último nó do caminho destaca a classe atribuída à instância, "Iris-setosa".

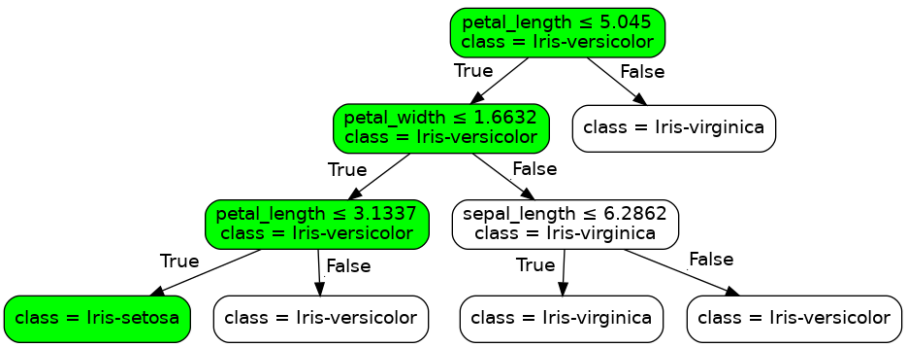

Figura 4.3: *Interpretação da predição com Árvore de Decisão.* ***Fonte****: O autor.*

Instância 2

Na Tabela 4.5 são apresentados os valores das *features* referentes à segunda instância selecionada aleatoriamente:

sepal_length	5.8
sepal_width	2.7
petal_length	4.1
petal_width	1.0
target	Iris-versicolor

Tabela 4.5: *Instância 2 selecionada aleatoriamente do dataset Iris*

As barras verdes apontadas para a direita indicam que a *feature* contribui para que a instância seja classificada corretamente de acordo com o seu *target* "Iris-setosa".

A Figura 4.4 exibe a explicação gerada pelo LIME da predição que classificou a instância como "Iris-versicolor". As *features* "petal_length", "petal_width" e "sepal_width" contribuem para que a instância seja classificada corretamente de acordo com o seu *target* "Iris-versicolor". A barra correspondente à *feature* "sepal_length", em vermelho e com importância negativa, contribui para que a instância seja classificada incorretamente para qualquer outra classe diferente do *target* "Iris-versicolor".

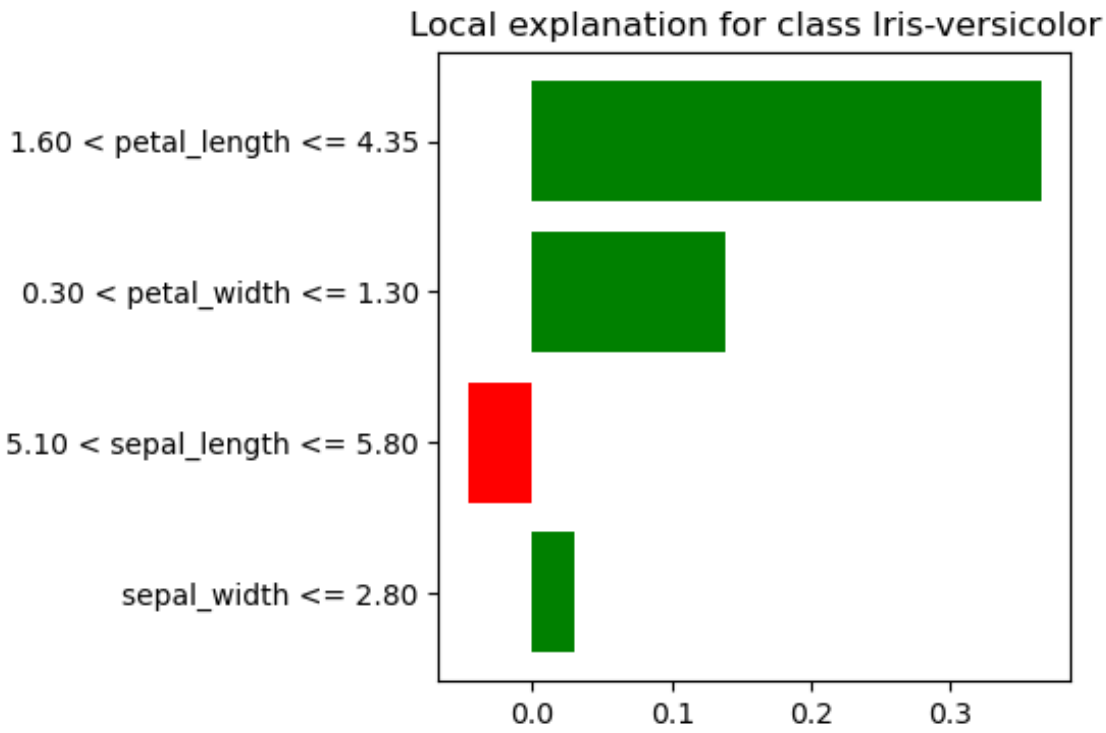

Figura 4.4: *Importância das features geradas pelo LIME.* ***Fonte****: O autor.*

A Figura 4.2 apresenta a importância das *features* de acordo com o método proposto, com as importâncias extraídas da Árvore de Decisão. Para a instância em específico, apenas as *features* "petal_length" e "petal_width" foram tidas como importantes pelo modelo interpretável.

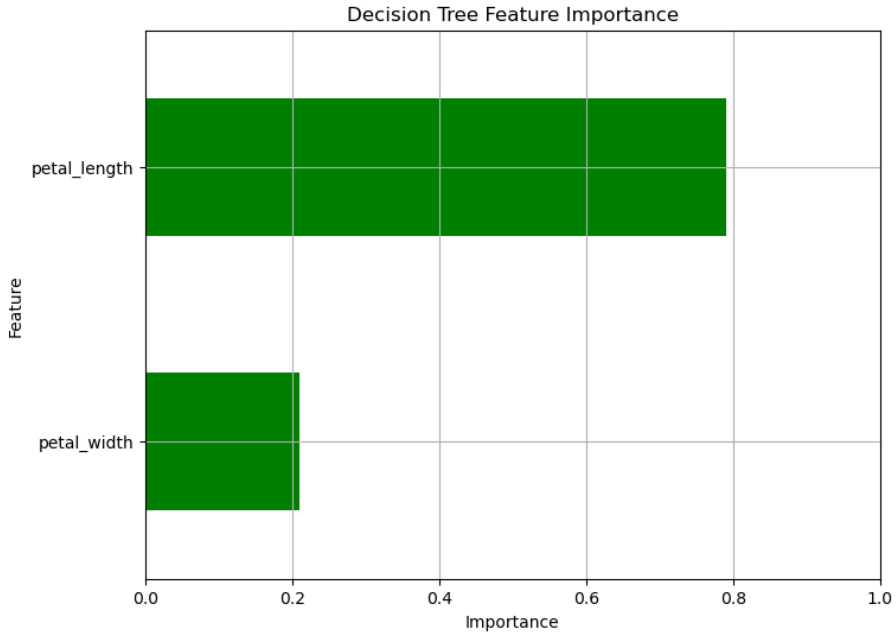

Figura 4.5: *Importância das features geradas pela Árvore de Decisão.* ***Fonte****: O autor.*

Por fim, a Figura 4.6 apresenta a Árvore de Decisão gerada a partir das perturbações da instância avaliada. Neste caso, as *features* "petal_length" e "petal_width" fazem parte das condições que levam o modelo a classificar a instância como "Iris-versicolor"

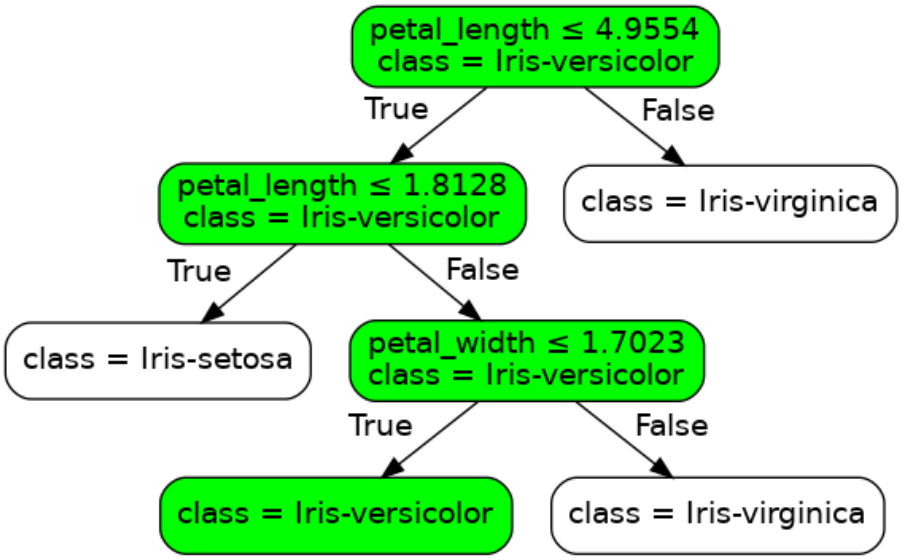

Figura 4.6: *Interpretação da predição com Árvore de Decisão.* ***Fonte****: O autor.*

4.3.2 Wine dataset

Instância 1

Na Tabela 4.6 são apresentados os valores das *features* referentes à instância selecionada aleatoriamente:

Alcohol	14.23
Malicacid	1.71
Ash	2.43
Alcalinity_of_ash	15.6
Magnesium	127
Total_phenols	2.8
Flavanoids	3.06
Nonflavanoid_phenols	0.28
Proanthocyanins	2.29
Color_intensity	5.64
Hue	1.04
0D280_0D315_of_diluted_wines	3.92
Proline	1065
target	1

Tabela 4.6: *Instância 1 selecionada aleatoriamente do dataset Wine*

A Figura 4.7 exibe a explicação gerada pelo LIME da predição que classificou a instância como sendo da classe "1". As *features* "Proline", "Color_intensity" e "Flavanoids" são as mais importantes, sendo "Proline" consideravelmente maior que as demais. Todas as *features* contribuem positivamente para a predição da classe "1".

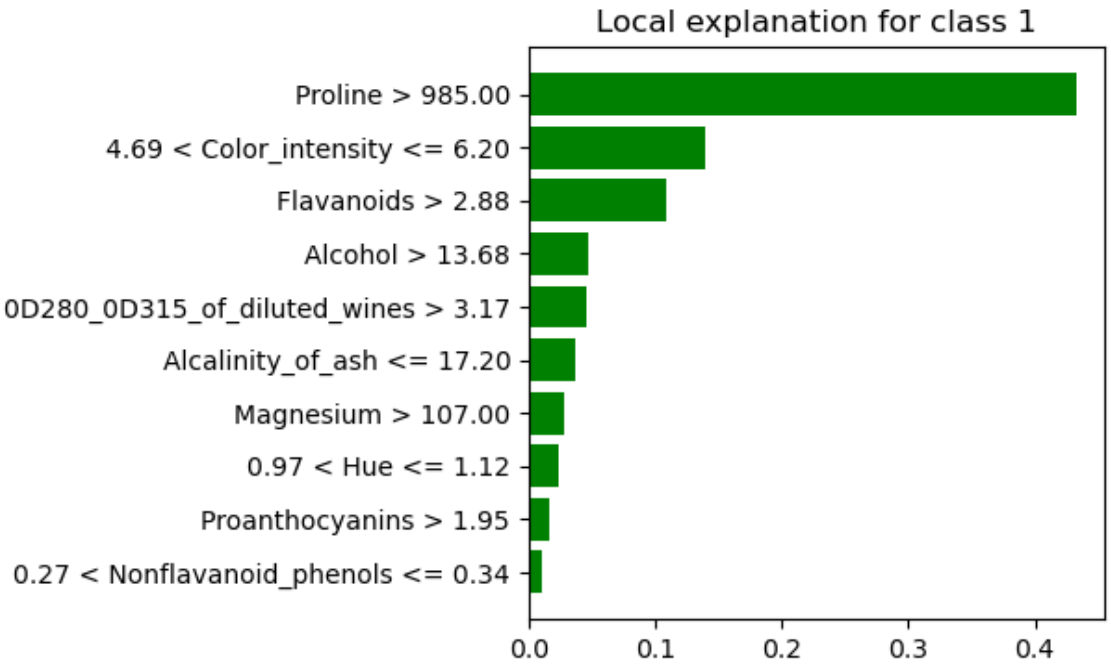

Figura 4.7: *Importância das features geradas pelo LIME.* ***Fonte****: O autor.*

A Figura 4.8 apresenta a importância das *features* de acordo com o método proposto, com as importâncias extraídas da Árvore de Decisão. Para a instância em específico, apenas as *features* "Proline", "Flavanoids" e "Color_intensity" foram tidas como importantes pelo modelo interpretável.

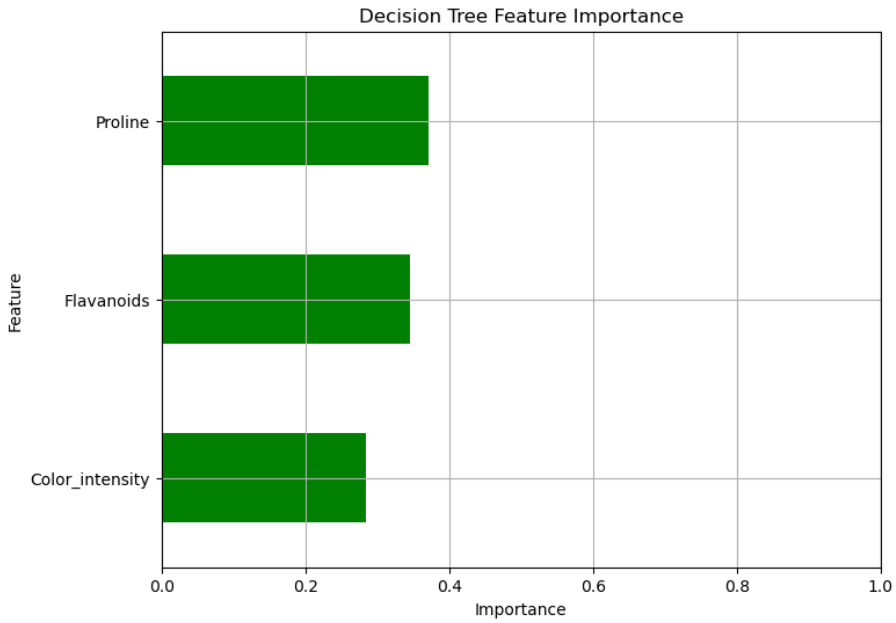

Figura 4.8: *Importância das features geradas pela Árvore de Decisão.* ***Fonte****: O autor.*

Por fim, a Figura 4.9 apresenta a Árvore de Decisão gerada a partir das perturbações da instância avaliada. Neste caso, as *features* "Color_intensity", "Proline" e "Flavanoids" fazem parte das condições que levam o modelo a classificar a instância como "1".

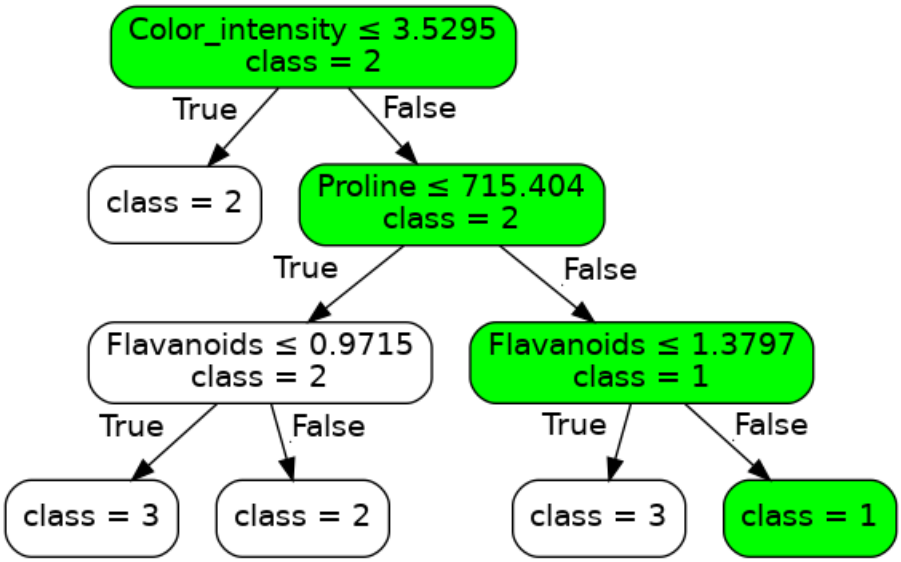

Figura 4.9: *Interpretação da predição com Árvore de Decisão.*
***Fonte**: O autor.*

Instância 2

Na Tabela 4.7 são apresentados os valores das *features* referentes à instância selecionada aleatoriamente:

Alcohol	12.87
Malicacid	4.61
Ash	2.48
Alcalinity_of_ash	21.5
Magnesium	86
Total_phenols	1.7
Flavanoids	0.65
Nonflavanoid_phenols	0.47
Proanthocyanins	0.86
Color_intensity	7.65
Hue	0.54
0D280_0D315_of_diluted_wines	1.86
Proline	625
target	3

Tabela 4.7: *Instância 2 selecionada aleatoriamente do dataset Wine*

A Figura 4.10 exibe a explicação gerada pelo LIME da predição que classificou a instância como "3". As *features* "Flavanoids", "Color_intensity" e "Hue", e são as mais importantes, sendo "Flavanoids" consideravelmente maior que as demais. Quase todas as *features* contribuem positivamente para a predição da classe "3", com exceção de "Magnesium" e "Malicacid", que contribuem negativamente, no entanto com importância baixa se comparada com as mais importantes.

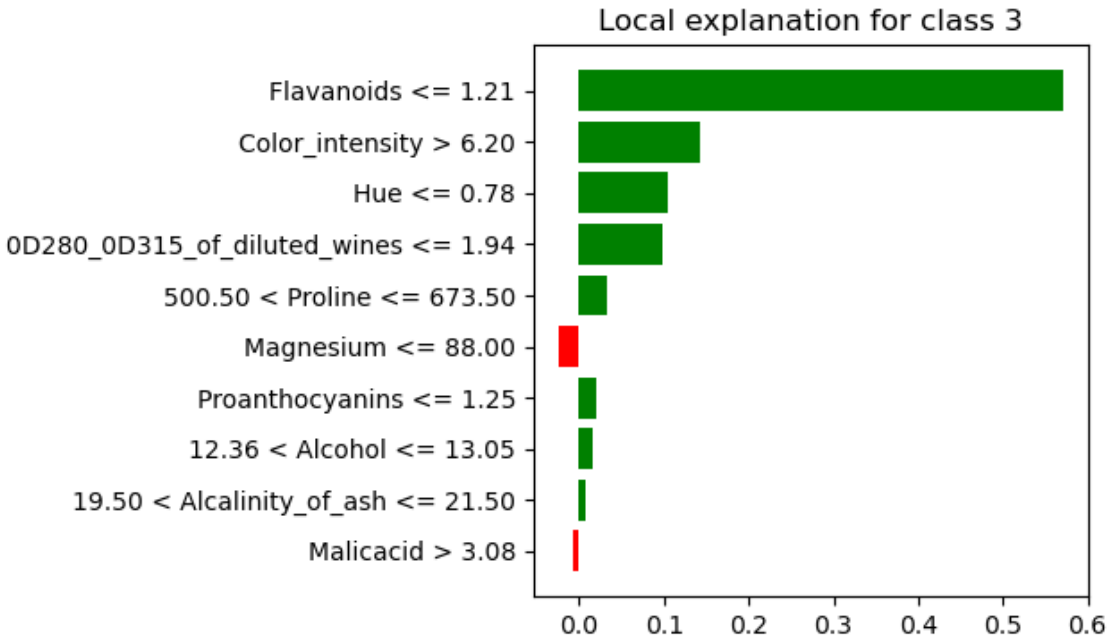

Figura 4.10: *Importância das features geradas pelo LIME.* ***Fonte****: O autor.*

A Figura 4.11 apresenta a importância das *features* de acordo com o método proposto, com as importâncias extraídas da Árvore de Decisão. Para a instância em específico, apenas as *features* "Flavanoids", "Proline" e "Color_intensity" foram tidas como importantes pelo modelo interpretável.

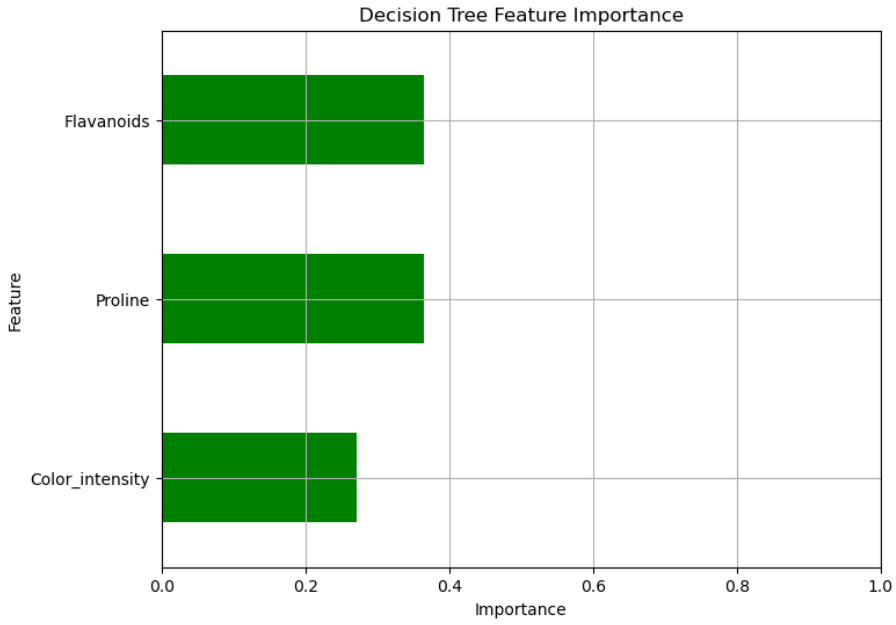

Figura 4.11: *Importância das features geradas pela Árvore de Decisão.* ***Fonte****: O autor.*

Por fim, a Figura 4.12 apresenta a Árvore de Decisão gerada a partir

das perturbações da instância avaliada. Neste caso, as *features* "Color_intensity", "Proline" e "Flavanoids" fazem parte das condições que levam o modelo a classificar a instância como "3".

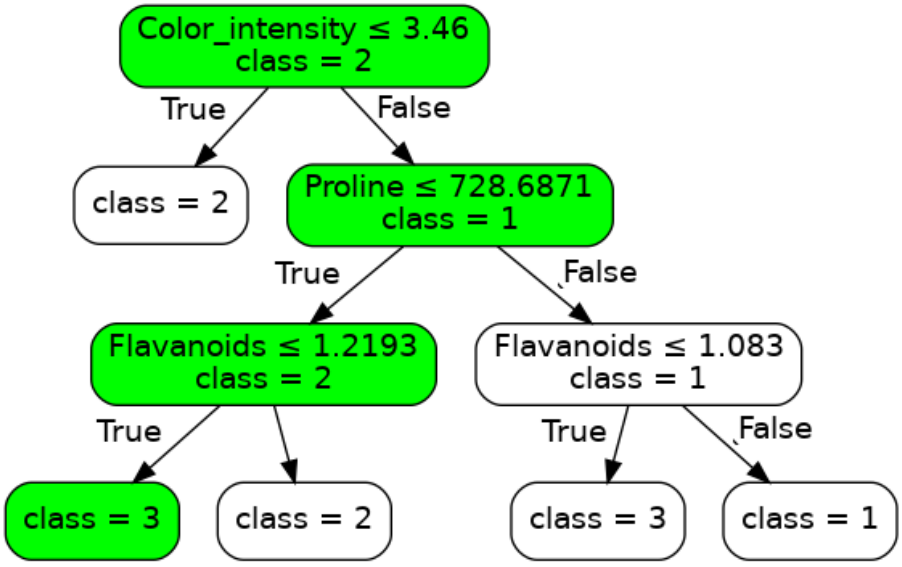

Figura 4.12: *Interpretação da predição com Árvore de Decisão.*
***Fonte**: O autor.*

4.3.3 Abalone dataset

Instância 1

Na Tabela 4.8 são apresentados os valores das *features* referentes à instância selecionada aleatoriamente:

Sex	0
Length	0.53
Diameter	0.42
Height	135
Whole_weight	677
Shucked_weight	0.2565
Viscera_weight	0.1415
Shell_weight	0.21
target	9

Tabela 4.8: *Instância 1 selecionada aleatoriamente do dataset Abalone*

A Figura 4.13 exibe a explicação gerada pelo LIME da predição para a instância selecionada. As *features* "Shucked_weight", "Viscera_weight", "Height",

"Sex", "Length" e "Diameter" contribuem positivamente para a predição do exemplo, enquanto que "Whole_weight" e "Shell_weight" contribuem negativamente para a predição. No caso dos problemas de regressão, como este, a contribuição positiva leva a predição a se aproximar do valor real, enquanto a contribuição negativa leva a predição a se distanciar do valor real.

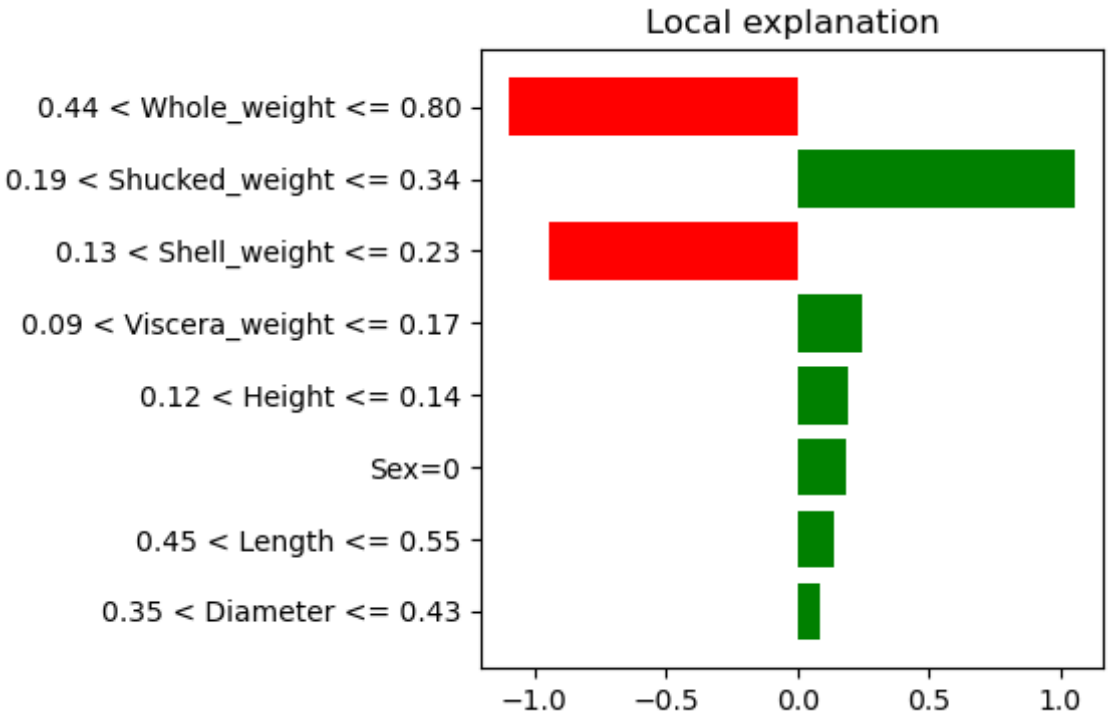

Figura 4.13: *Importância das features geradas pelo LIME.* ***Fonte****: O autor.*

A Figura 4.14 apresenta a importância das *features* de acordo com o método proposto, com as importâncias extraídas da Árvore de Decisão. Para a instância em específico, apenas as *features* "Shell_weight", "Shucked_weight" e "Whole_weight" foram tidas como importantes pelo modelo interpretável.

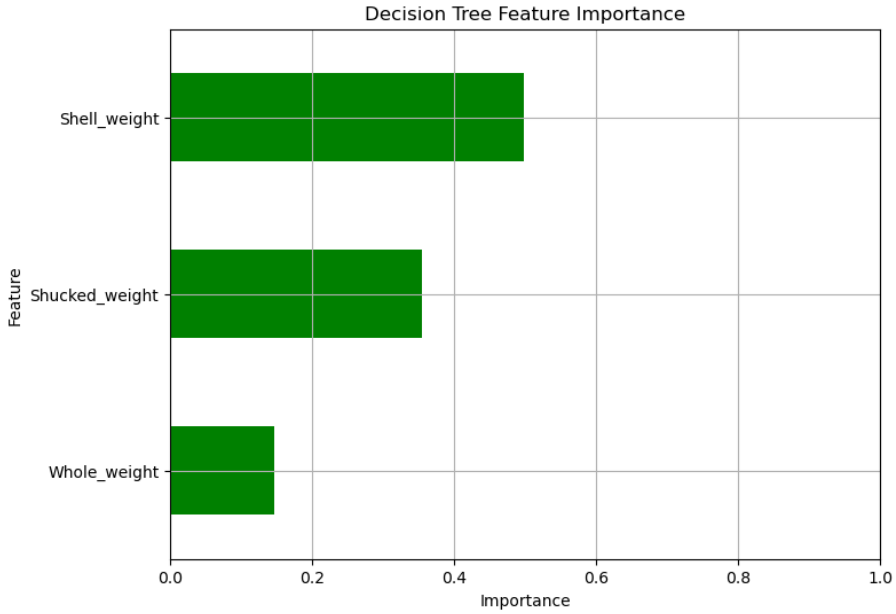

Figura 4.14: *Importância das features geradas pela Árvore de Decisão.* ***Fonte****: O autor.*

Por fim, as Figuras 4.15 e 4.16 apresentam a Árvore de Decisão gerada a partir das perturbações da instância avaliada. Neste caso, as *features* "Shell_weight", "Shucked_weight" e "Whole_weight" fazem parte das condições que levam o modelo a predizer o valor "10.7938" para a instância.

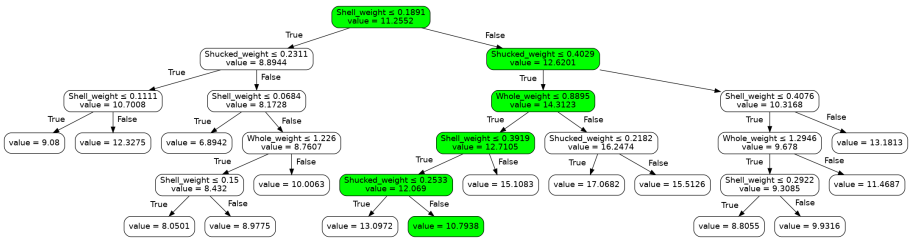

Figura 4.15: *Interpretação da predição com Árvore de Decisão.* ***Fonte****: O autor.*

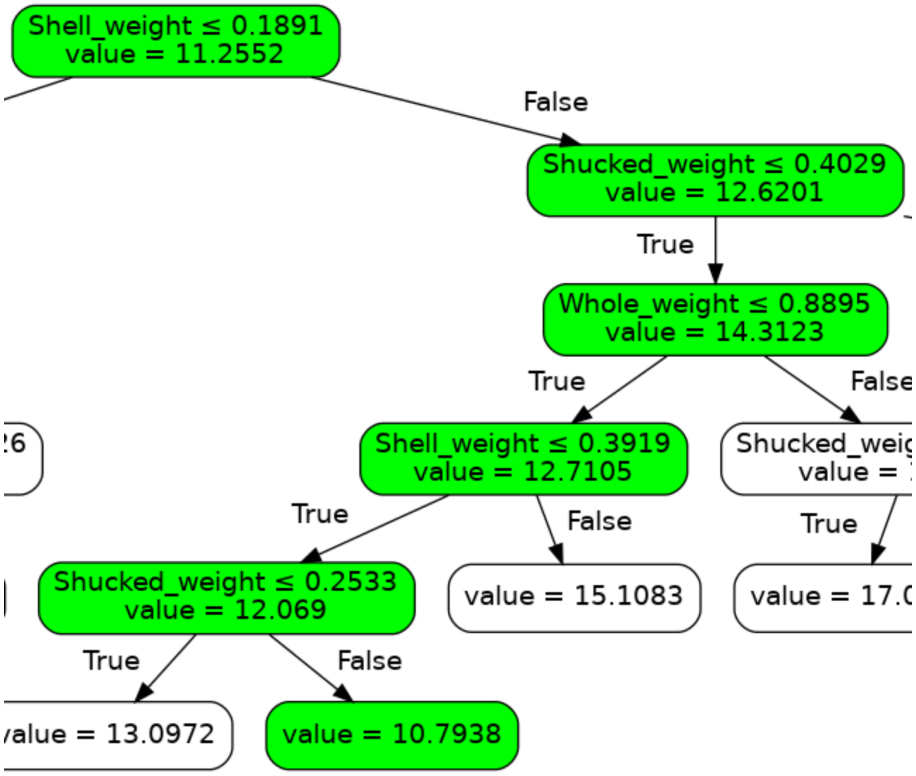

Figura 4.16: *Ampliação da Árvore de Decisão.* ***Fonte****: O autor.*

Instância 2

Na Tabela 4.9 são apresentados os valores das *features* referentes à instância selecionada aleatoriamente:

Sex	2
Length	0.49
Diameter	0.38
Height	135
Whole_weight	0.5415
Shucked_weight	0.2175
Viscera_weight	95
Shell_weight	0.19
target	11

Tabela 4.9: *Instância 2 selecionada aleatoriamente do dataset Abalone*

A Figura 4.17 exibe a explicação gerada pelo LIME da predição para a instância selecionada. As *features* "Shucked_weight", "Sex", "Length" e "Viscera_weight" contribuem positivamente para a predição do exemplo, enquanto

que “Whole_weight”, “Shell_weight”, “Diameter” e “Height” contribuem negativamente para a predição.

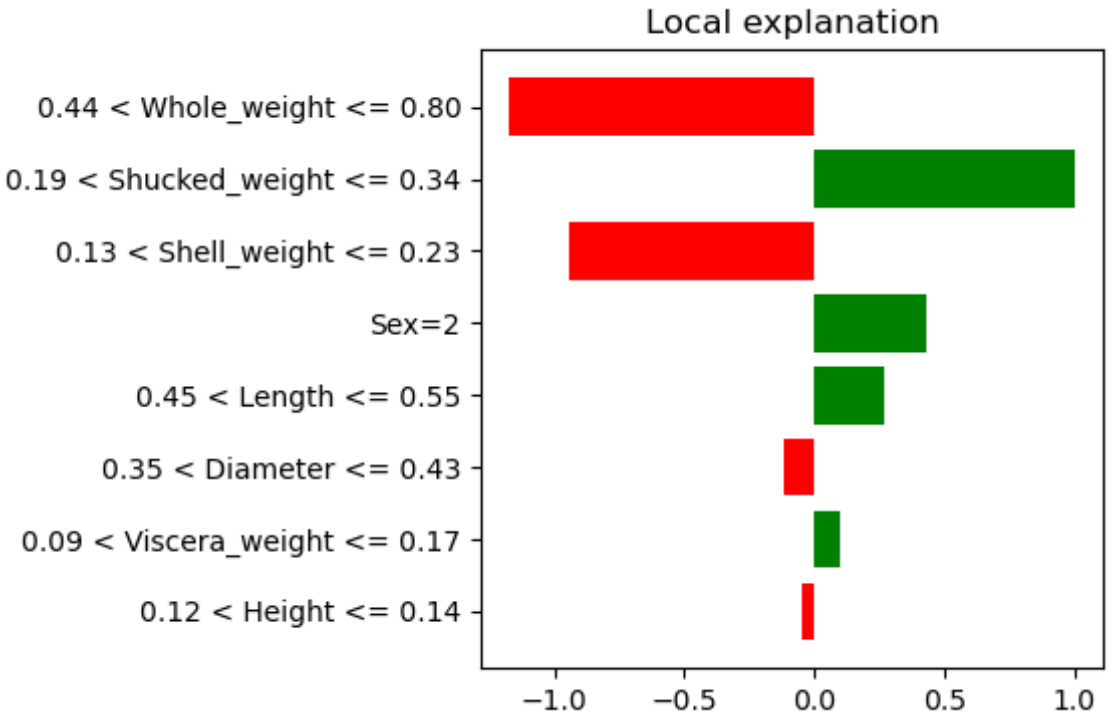

Figura 4.17: *Importância das features geradas pelo LIME.* ***Fonte:*** *O autor.*

A Figura 4.18 apresenta a importância das *features* de acordo com o método proposto, com as importâncias extraídas da Árvore de Decisão. Para a instância em específico, apenas as *features* “Shell_weight”, “Shucked_weight” e “Whole_weight” foram tidas como importantes pelo modelo interpretável.

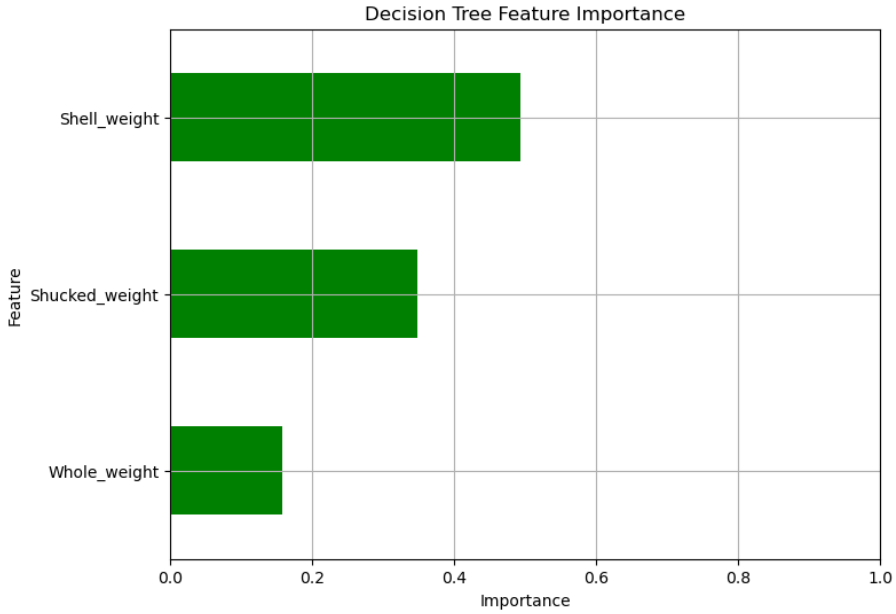

Figura 4.18: *Importância das features geradas pela Árvore de Decisão.* ***Fonte****: O autor.*

Por fim, nas Figuras 4.19 e 4.20 apresenta a Árvore de Decisão gerada a partir das perturbações da instância avaliada. Neste caso, as *features* "Shell_weight", "Shucked_weight" e "Whole_weight" fazem parte das condições que levam o modelo a predizer o valor "13.1329" para a instância.

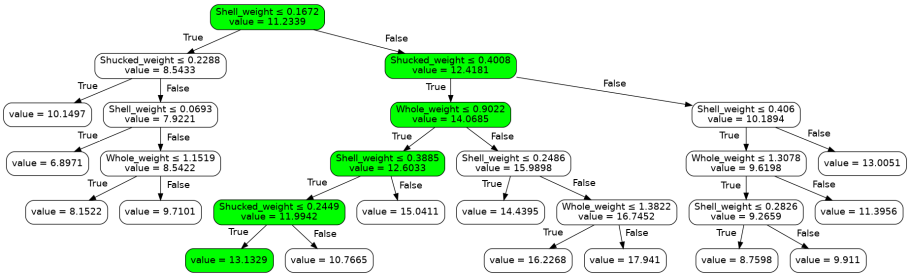

Figura 4.19: *Interpretação da predição com Árvore de Decisão.* ***Fonte****: O autor.*

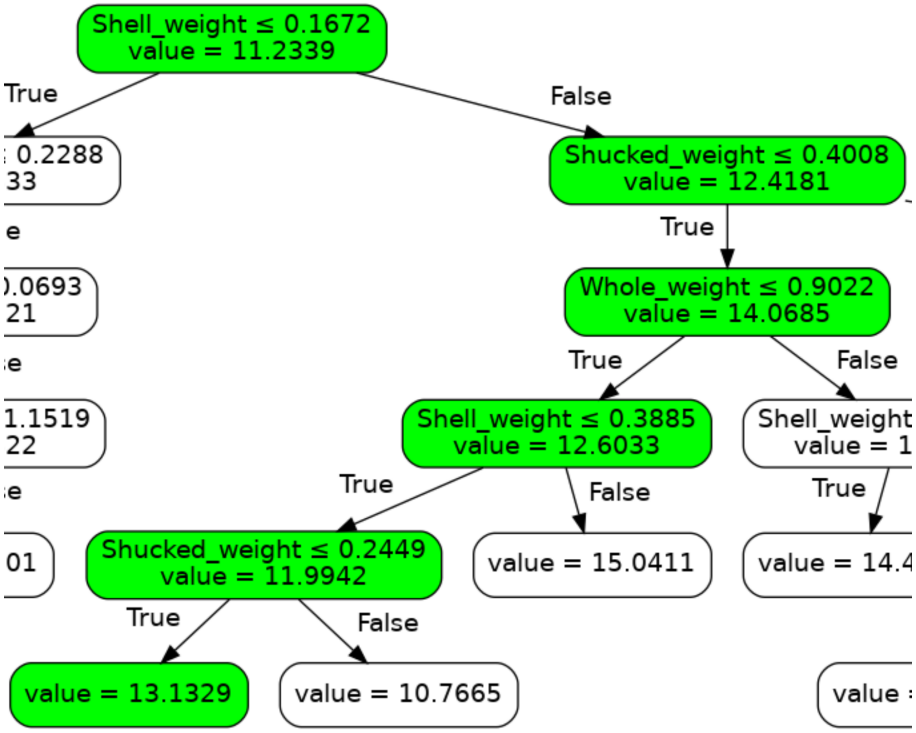

Figura 4.20: *Ampliação da Árvore de Decisão.* ***Fonte****: O autor.*

4.3.4 Parkinsons Telemonitoring dataset

Instância 1

Na Tabela 4.10 são apresentados os valores das *features* referentes à instância selecionada aleatoriamente:

age	72
sex	0
test_time	5.6431
Jitter	0.00662
Jitter Abs	3.38e-05
Jitter RAP	0.00401
Jitter PPQ5	0.00317
Jitter DDP	0.01204
Shimmer	0.02565
Shimmer dB	0.23
Shimmer APQ3	0.01438
Shimmer APQ5	0.01309
Shimmer APQ11	0.01662
Shimmer DDA	0.04314
NHR	0.01429
HNR	21.64
RPDE	0.41888
DFA	0.54842
PPE	0.16006
target	34.398

Tabela 4.10: *Instância 1 selecionada aleatoriamente do dataset Parkinsons Telemonitoring*

A Figura 4.21 exibe a explicação gerada pelo LIME da predição para a instância selecionada. As *features* "age", "DFA", "Jitter Abs", "sex", "Jitter", "Jitter RAP" e "Jitter PPQ5" contribuem positivamente para a predição do exemplo, enquanto que "test_time", "NHR", e "Shimmer APQ11" contribuem negativamente para a predição. É interessante notar a importância significativamente maior da *features* "age".

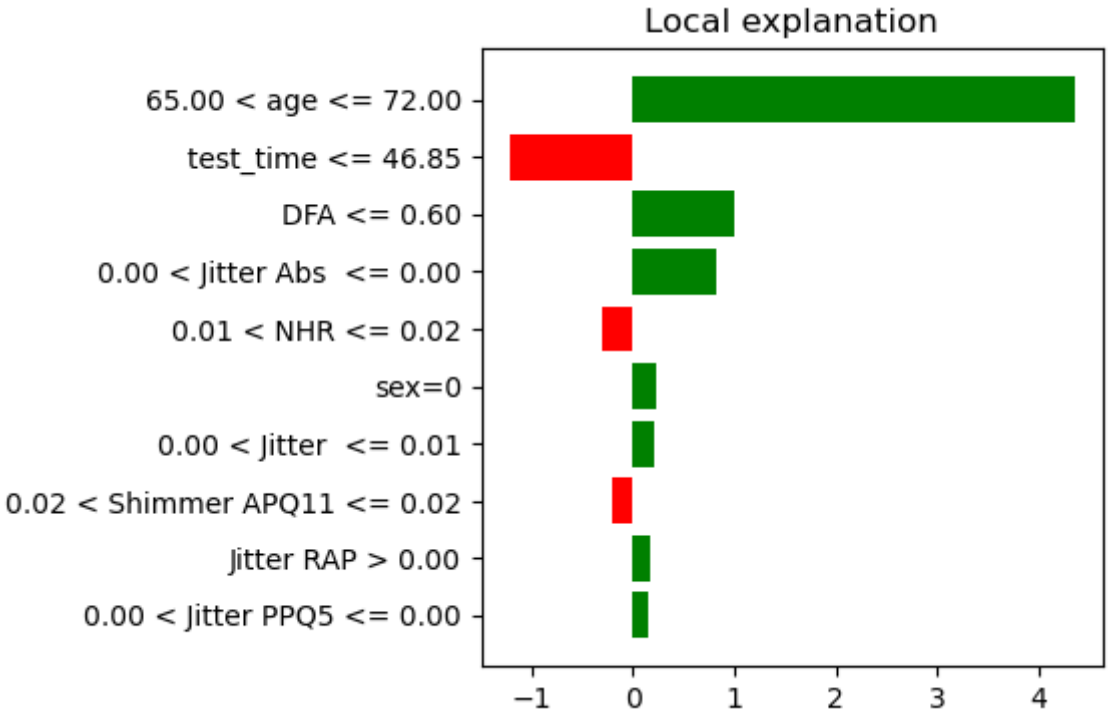

Figura 4.21: *Importância das features geradas pelo LIME.* ***Fonte:*** *O autor.*

A Figura 4.22 apresenta a importância das *features* de acordo com o método proposto, com as importâncias extraídas da Árvore de Decisão. Para a instância em específico, apenas as *features* "age" e "sex" foram tidas como importantes pelo modelo interpretável.

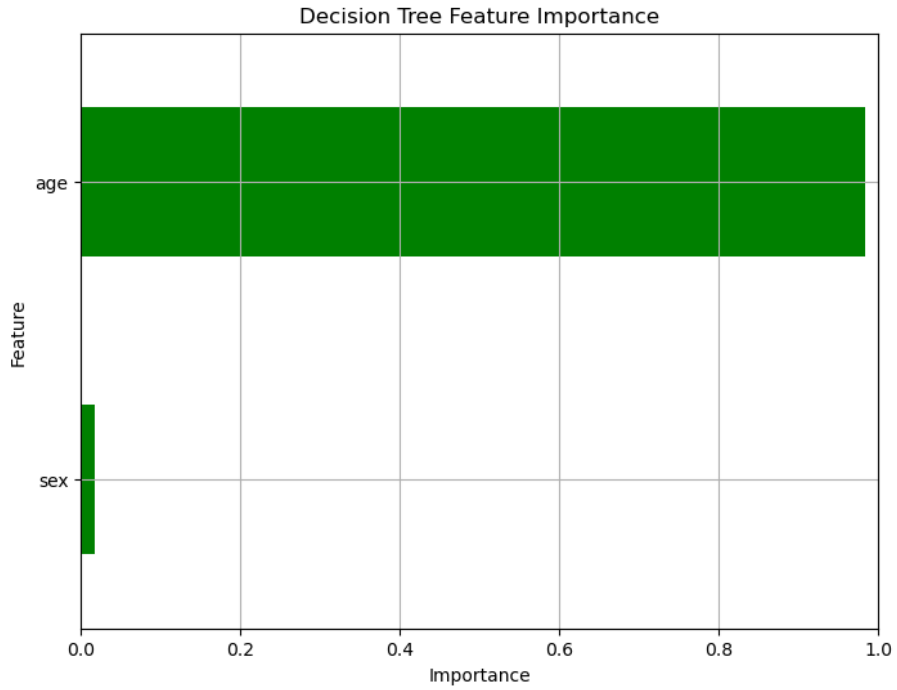

Figura 4.22: *Importância das features geradas pela Árvore de Decisão.* ***Fonte****: O autor.*

Por fim, nas Figuras 4.23 e 4.24 apresenta a Árvore de Decisão gerada a partir das perturbações da instância avaliada. Neste caso, apenas a *feature* "age" tem impacto sobre a predição, que resulta no valor "32.1805".

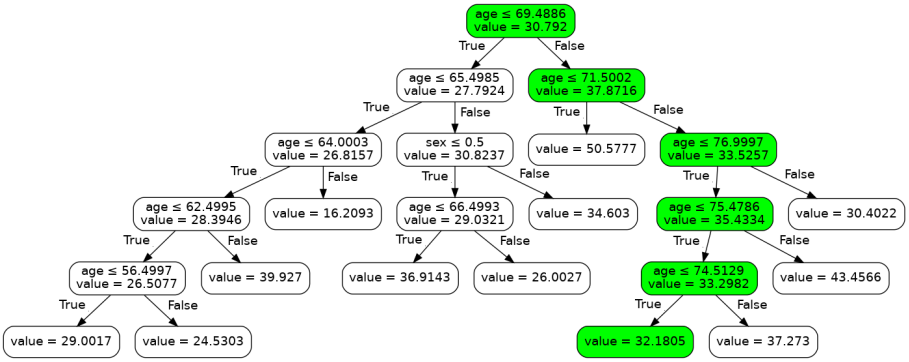

Figura 4.23: *Interpretação da predição com Árvore de Decisão.* ***Fonte****: O autor.*

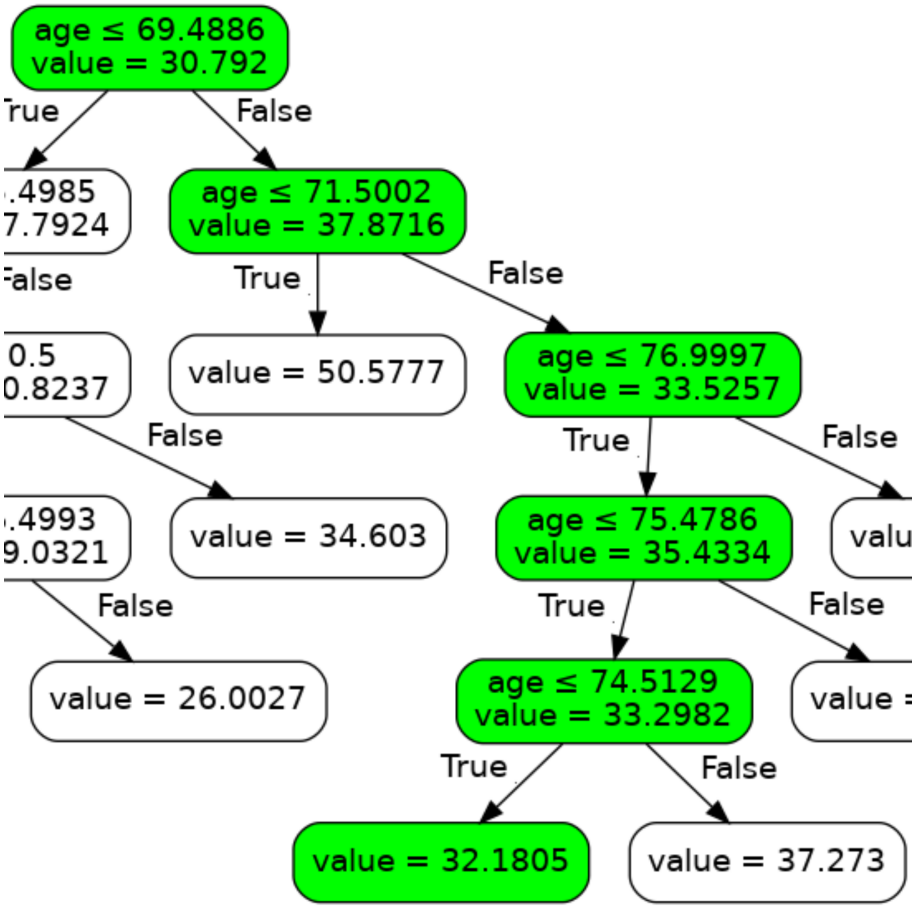

Figura 4.24: *Ampliação da Árvore de Decisão.* ***Fonte****: O autor.*

Instância 2

Na Tabela 4.11 são apresentados os valores das *features* referentes à instância selecionada aleatoriamente:

age	72
sex	0
test_time	174.66
Jitter	0.00526
Jitter Abs	0,02573
Jitter RAP	0.00225
Jitter PPQ5	0.00177
Jitter DDP	0.00675
Shimmer	0.03183
Shimmer dB	304
Shimmer APQ3	0.01906
Shimmer APQ5	0.01497
Shimmer APQ11	0.02093
Shimmer DDA	0.05718
NHR	0.033822
HNR	25852
RPDE	0.45109
DFA	0.54152
PPE	0.23613
target	47.97

Tabela 4.11: *Instância 2 selecionada aleatoriamente do dataset Parkinsons Telemonitoring*

A Figura 4.25 exibe a explicação gerada pelo LIME da predição para a instância selecionada. As *features* "age", "Shimmer", "DFA", "Jitter Abs", "Jitter PPQ5", "Shimmer APQ5" e "Jitter DDP" contribuem positivamente para a predição do exemplo, enquanto que "Shimmer dB", "HNR", e "RPDE" contribuem negativamente para a predição.

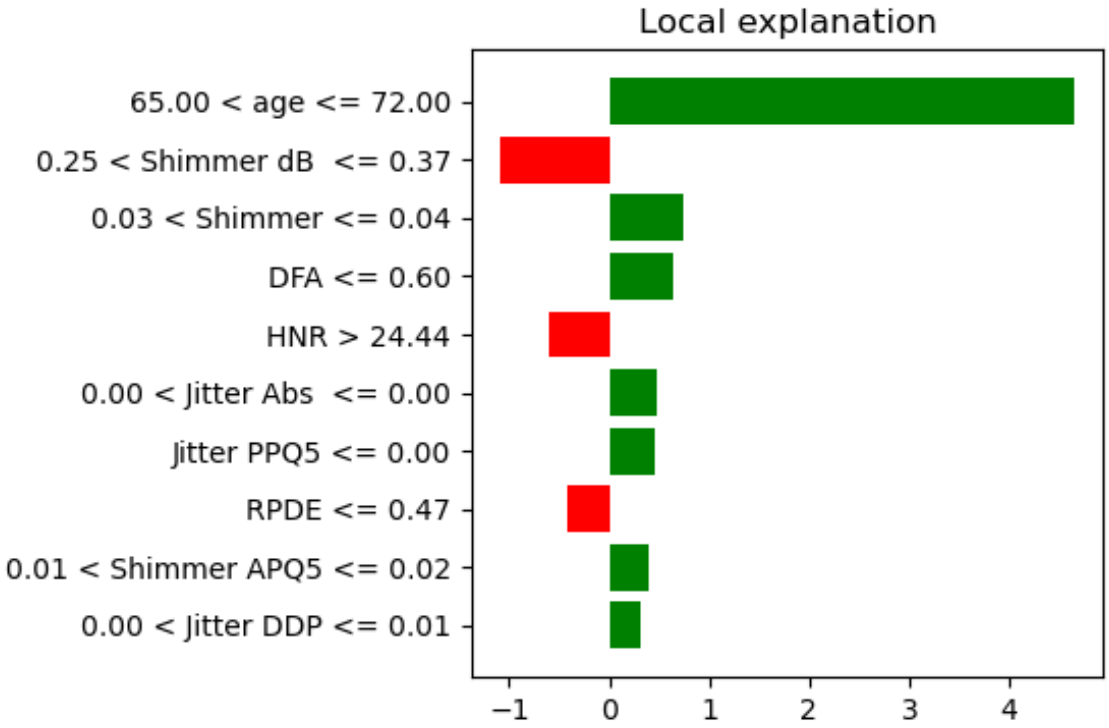

Figura 4.25: *Importância das features geradas pelo LIME.* ***Fonte****: O autor.*

A Figura 4.26 apresenta a importância das *features* de acordo com o método proposto, com as importâncias extraídas da Árvore de Decisão. Para a instância em específico, apenas as *features* "age" e "sex" foram tidas como importantes pelo modelo interpretável.

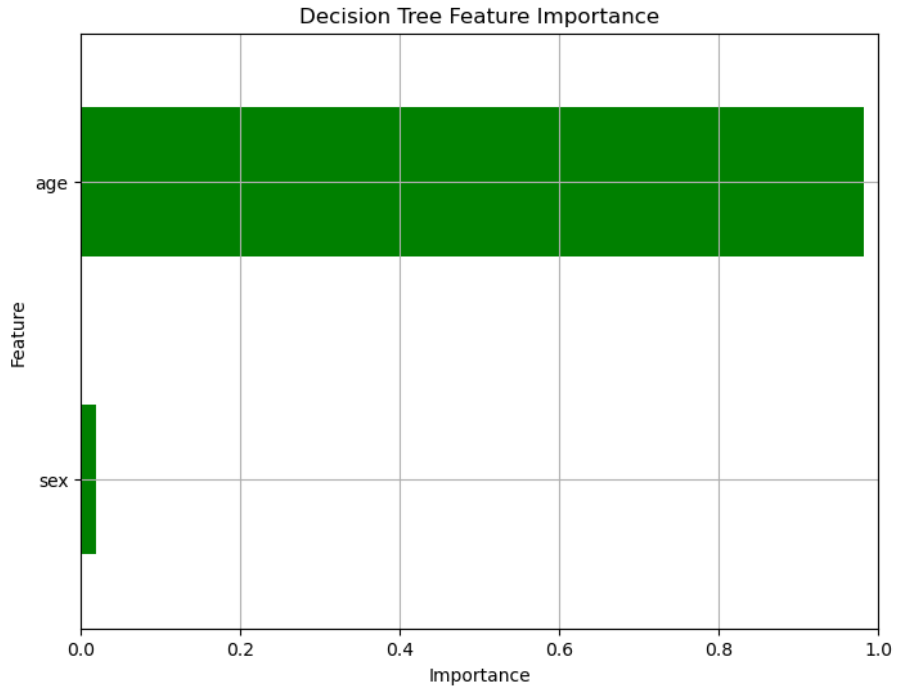

Figura 4.26: *Importância das features geradas pela Árvore de Decisão.* ***Fonte****: O autor.*

Por fim, nas Figuras 4.27 e 4.28 apresenta a Árvore de Decisão gerada a partir das perturbações da instância avaliada. Neste caso, apenas a *feature* "age" tem impacto sobre a predição, que resulta no valor "32.2171".

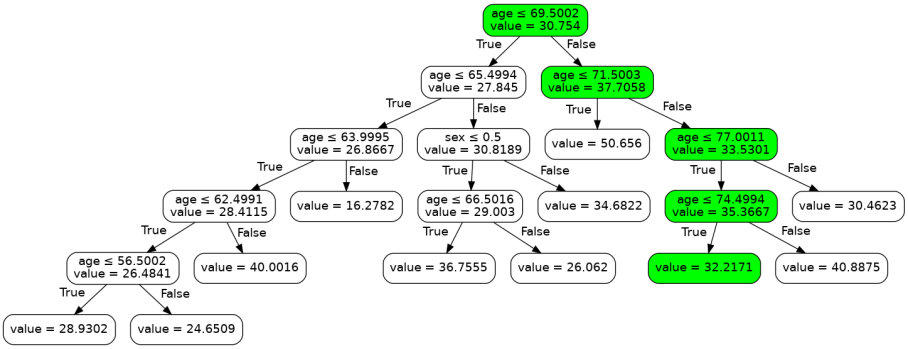

Figura 4.27: *Interpretação da predição com Árvore de Decisão.* ***Fonte****: O autor.*

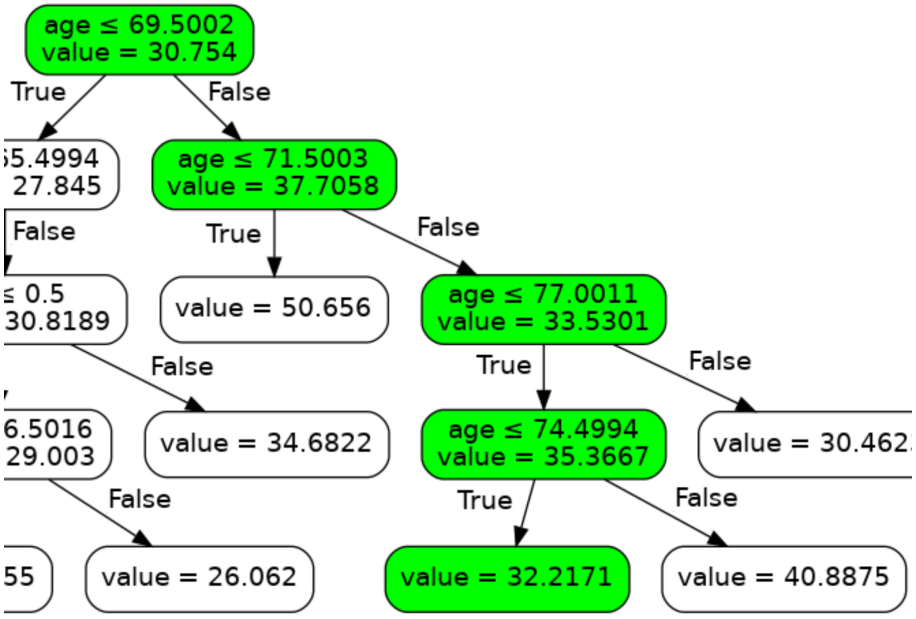

Figura 4.28: *Ampliação da Árvore de Decisão.* ***Fonte****: O autor.*

4.4 Fidelidade dos modelos interpretáveis

Nesta seção, os modelos interpretáveis do método proposto (árvore de decisão) e do LIME (regressão linear) são comparados em relação a qual modelo tem suas previsões mais fieis ao modelo preditor. Para isto, foram utilizadas as métricas de fidelidade como propostas por Li [18] e Archer [3]. Tais métricas foram computadas para cada exemplo, uma vez a interpretabilidade leva em consideração um modelo interpretável para cada instância do *dataset*. Os resultados a seguir referem-se à média, juntamente com o desvio padrão, entre as fidelidades obtidas para cada exemplo.

A Tabela 4.12 apresenta as métricas de fidelidade obtidas nos problemas de classificação. Para esse tipo de problema, a fidelidade diz respeito à porcentagem em que o modelo interpretável concorda com o modelo de predição. Dessa forma, maiores valores indicam maior concordância e assim, são ditos melhores.

Como é possível observar, os modelos interpretáveis baseados em Árvore de Decisão alcançaram melhores resultados, comprovando a vantagem destes modelos se comparados com a regressão linear utilizada pelo LIME.

Dataset/modelo	Regressão linear (LIME)	Desvio padrão	Árvore de decisão	Desvio padrão
(Iris)	0.692	0.005	**0.957**	0.045
(Wine)	0.766	0.005	**0.951**	0.018

Tabela 4.12: *Fidelidade média calculada através da Equação 1-1 para problemas de classificação.*

A Tabela 4.13 apresenta as métricas de fidelidade obtidas nos problemas de regressão. Para esse tipo de problema, a fidelidade diz respeito ao erro absoluto médio (MAE) entre as predições do modelo interpretável e o modelo de predição. Por ser uma métrica que calcula o erro sua interpretação é inversa à usada para classificação, ou seja, quanto menor o erro maior será a fidelidade.

É possível verificar que para os experimentos de regressão linear o LIME se saiu melhor para o *dataset Abalone*, alcançando menor erro médio absoluto. No o *dataset Parkinsons Telemonitoing* os modelos baseados em Árvore de Decisão novamente mostraram sua eficácia, obtendo o erro médio absoluto mais baixo.

Dataset/modelo	Regressão linear (LIME)	Desvio padrão	Árvore de decisão	Desvio padrão
(Abalone)	**1.176**	0.013	1.807	0.059
Parkinsons Telemonitoing	6.858	0.075	**5.638**	0.099

Tabela 4.13: *Fidelidade média calculada através da Equação 1-2 para problemas de regressão.*

O resultados obtidos e avaliados através das métricas de fidelidade demonstram a vantagem das Árvores de Decisão se comparadas com a regressão linear, o que destaca o ganho da abordagem apresentada neste trabalho no que diz respeito à fidelidade dos modelos interpretáveis ao simular o comportamento do modelo de predição.

4.5 *Features* relevantes dos modelos interpretáveis nas predições

A importância das *features* geradas pelos métodos e técnicas de interpretabilidade de modelos de Aprendizado de Máquina são cruciais para o entendimento dos resultados de tais modelos. Dessa forma, surge a necessidade de se

comparar novas abordagens, como a apresentada neste trabalho, com técnicas já existentes, como LIME.

Como é possível verificar nos experimentos apresentados na Seção 4.3, as *features* selecionadas através das Árvores de Decisão se aproximam das selecionadas pelo LIME no que diz respeito ao conjunto das mais importantes. Apesar de posição relativa nos gráficos de importâncias das Árvores de Decisão e do LIME não coincidirem em todos os casos, podemos observar que as *features* selecionadas a partir das árvores ainda assim apontam em posições elevadas (maior importância) na sequência gerada pelo LIME.

Tomando como exemplo as importâncias exibidas nas Figuras 4.1 e 4.2 notamos que a primeira e segunda posições em ambas as abordagens coincidem, com as *features* "petal_length" e "petal_width", respectivamente. No entanto, "petal_length" obteve uma importância consideravelmente maior se comparada às demais, o que pode indicar, em casos como este, que o modelo preditor considera uma única *feature* (ou um pequeno conjunto) para tomar sua decisão.

Outro ponto a se destacar é que o número reduzido de *features* geradas pela nova abordagem, como podemos observar nos resultados apresentados, torna a interpretabilidade mais concisa, uma vez que mostram apenas *features* verdadeiramente importantes. Além disso, tanto a análise dos resultados quanto a tomada de decisões se beneficiam de tal fato. *Features* com importâncias baixas ou iguais a 0 (zero) podem vir a ser desconsideradas na tomada de decisão ou mesmo na modelagem de modelos posteriores.

Além do número reduzido de *features*, há casos, como podemos observar nas Figuras 4.8 e 4.11 em que as importâncias aparecem de forma mais equilibrada, ou seja, com valores que não se distanciam de forma anormal.

O uso do Índice *Gini* e da MSE para o cálculo das importâncias agregam consistência à nova abordagem. Diferente dos coeficientes da regressão linear utilizadas no Lime, tais métricas são obtidas de forma mais natural, calculando-se através da árvore, e confiáveis, uma vez que são apoiadas pela literatura para essa finalidade.

4.6 Explicações contrafactuais das árvores de decisão

Além da possibilidade de se calcular a importância das *features*, as Árvores de Decisão também possibilitam a interpretação através da sua própria estrutura. As condições que regem a predição de uma instância qualquer podem ser encontradas nos nós internos da árvore.

A partir das árvores apresentas na Seção 4.3 podemos obter as explicações contrafactuais analisando as *features* que são consideradas na tarefa de predição. Como citado na Seção 2.5, a explicação contrafactual busca a menor alteração possível na instância de modo que o resultado da predição seja diferente.

A busca pela menor alteração possível resume-se em encontrar a *feature* e uma condição relacionada a essa *feature* de forma que o resultado da predição seja diferente. Tal busca é então direta e trivial a partir da Árvore de Decisão, uma vez que esta já conta com essas informações em sua estrutura.

Dessa forma, nas Árvores de Decisão apresentas na Seção 4.3, as explicações contrafactuais serão dadas pelas condições presentes no último nó antes do nó folha. A inversão da condição desse nó resulta na obtenção de um novo valor para a predição e podemos obter o conjunto (*feature*, <novo valor>) que caracteriza a explicação contrafactual.

A hipótese da explicação contrafactual pode então ser comprovada a partir da busca de exemplos que comprovem tal fato. A seguir apresentamos exemplos que comprovam a explicação contrafactual para os experimentos apresentados na Seção 4.3.

4.6.1 Iris dataset

Instância 1

Consideremos a instância 1 do *dataset Iris* apresentada na Seção 4.3.1, e sua respectiva Árvore de Decisão do modelo interpretável (Figura 4.3). A explicação contrafactual é obtida alterando-se a decisão do nó "$petal\_length \leq$ 3.1337" para o valor "falso" e pode ser expressa pela proposição:

"petal_length" deve ser MAIOR que 3.1337 *para que o resultado da predição seja "Iris-versicolor", de forma que todas as outras condições sejam respeitadas*

De fato, a Tabela 4.14 exibe um exemplo presente no *dataset* que demonstra a validade da explicação contrafactual obtida através da interpretação da Árvore de Decisão. Trata-se de um exemplo/instância real, que respeita as condições da árvore (vide coluna "Condições respeitadas"), com exceção do último nó antes da folha na Figura 4.3.

feature	valor	Condições respeitadas
sepal_length	7.0	
sepal_width	3.2	
petal_length	4.7	<= 5.045
petal_width	1.4	<= 1.6632
target	**Iris-versicolor**	

Tabela 4.14: *Exemplo que comprova a explicação contrafactual do experimento 1 do dataset (Iris)*

Instância 2

Consideremos a instância 2 do *dataset Iris* apresentada na Seção 4.3.1, e sua respectiva Árvore de Decisão do modelo interpretável (Figura 4.6). A explicação contrafactual é obtida alterando-se a decisão do nó "*petal_width* $\leq$ 1.7023" para o valor "falso" e pode ser expressa pela proposição:

"petal_width" deve ser MAIOR que 1.7023 *para que o resultado da predição seja "Iris-virginica", de forma que todas as outras condições sejam respeitadas*

De fato, a Tabela 4.15 exibe um exemplo presente no *dataset* que demonstra a validade da explicação contrafactual obtida através da interpretação da Árvore de Decisão. Trata-se de um exemplo/instância real, que respeita as condições da árvore (vide coluna "Condições respeitadas"), com exceção do último nó antes da folha na Figura 4.6.

feature	valor	Condições respeitadas
sepal_length	5.6	
sepal_width	2.8	
petal_length	4.9	<= 4.9554 >1.8128
petal_width	2.0	
target	**Iris-virginica**	

Tabela 4.15: *Exemplo que comprova a explicação contrafactual do experimento 2 do dataset (Iris)*

4.6.2 Wine dataset

Instância 1

Consideremos a instância 1 do *dataset Wine* apresentada na Seção 4.3.2, e sua respectiva Árvore de Decisão do modelo interpretável (Figura 4.9). A

explicação contrafactual é obtida alterando-se a decisão do nó "*Flavanoids* $\leq$ 1.3797" para o valor "verdadeiro" e pode ser expressa pela proposição:

"Flavanoids" deve ser MENOR que 1.3797 *para que o resultado da predição seja* 3*, de forma que todas as outras condições sejam respeitadas*

De fato, a Tabela 4.16 exibe um exemplo presente no *dataset* que demonstra a validade da explicação contrafactual obtida através da interpretação da Árvore de Decisão. Trata-se de um exemplo/instância real, que respeita as condições da árvore (vide coluna "Condições respeitadas"), com exceção do último nó antes da folha na Figura 4.9.

feature	**valor**	**Condições respeitadas**
Alcohol	12.25	
Malicacid	4.72	
Ash	2.54	
Alcalinity_of_ash	21.0	
Magnesium	89	
Total_phenols	1.38	
Flavanoids	0.47	
Nonflavanoid_phenols	0.53	
Proanthocyanins	0.8	
Color_intensity	3.85	>3.5295
Hue	0.75	
0D280_0D315_of_diluted_wines	1.27	
Proline	720	>715.404
target	**3**	

Tabela 4.16: *Exemplo que comprova a explicação contrafactual do experimento 1 do dataset (Wine)*

Instância 2

Consideremos a instância 2 do *dataset Wine* apresentada na Seção 4.3.2, e sua respectiva Árvore de Decisão do modelo interpretável (Figura 4.12). A explicação contrafactual é obtida alterando-se a decisão do nó "*Flavanoids* $\leq$ 1.2193" para o valor "falso" e pode ser expressa pela proposição:

"Flavanoids" deve ser MAIOR que 1.2193 *para que o resultado da predição seja* 2*, de forma que todas as outras condições sejam respeitadas*

De fato, a Tabela 4.17 exibe um exemplo presente no *dataset* que demonstra a validade da explicação contrafactual obtida através da interpretação da Árvore de Decisão. Trata-se de um exemplo/instância real, que respeita as condições da árvore (vide coluna "Condições respeitadas"), com exceção do último nó antes da folha na Figura 4.12.

feature	**valor**	**Condições respeitadas**
Alcohol	11.76	
Malicacid	2.68	
Ash	2.92	
Alcalinity_of_ash	20.0	
Magnesium	103	
Total_phenols	1.75	
Flavanoids	2.03	
Nonflavanoid_phenols	0.6	
Proanthocyanins	1.05	
Color_intensity	3.8	>3.46
Hue	1.23	
0D280_0D315_of_diluted_wines	2.5	
Proline	607	<728.6871
target	**2**	

Tabela 4.17: *Exemplo que comprova a explicação contrafactual do experimento 2 do dataset (Wine)*

4.6.3 Abalone dataset

Instância 1

Consideremos a instância 1 do *dataset Abalone* apresentada na Seção 4.3.3, e sua respectiva Árvore de Decisão do modelo interpretável (Figura 4.16). A explicação contrafactual é obtida alterando-se a decisão do nó "*Shucked_weight* $\leq$ 0.2533" para o valor "verdadeiro" e pode ser expressa pela proposição:

"Shucked_weight" deve ser MENOR que 0.2533 *para que o resultado da predição seja aproximadamente* 13.0972*, de forma que todas as outras condições sejam respeitadas*

De fato, a Tabela 4.18 exibe um exemplo presente no *dataset* que demonstra a validade da explicação contrafactual obtida através da interpretação da Árvore de Decisão. Trata-se de um exemplo/instância real, que respeita as condições

da árvore (vide coluna "Condições respeitadas"), com exceção do último nó antes da folha na Figura 4.16.

feature	valor	Condições respeitadas
Sex	0	
Length	0.49	
Diameter	0.39	
Height	135	
Whole_weight	0.5785	<0.8895
Shucked_weight	0.2465	<0.4029
Viscera_weight	123	
Shell_weight	0.2	>0.1891 <0.2919
target	**13**	

Tabela 4.18: *Exemplo que comprova a explicação contrafactual do experimento 1 do dataset (Abalone)*

Instância 2

Consideremos a instância 1 do *dataset Abalone* apresentada na Seção 4.3.3, e sua respectiva Árvore de Decisão do modelo interpretável (Figura 4.20). A explicação contrafactual é obtida alterando-se a decisão do nó "*Shucked_weight* $\leq$ 0.2449" para o valor "falso" e pode ser expressa pela proposição:

"Shucked_weight" deve ser MAIOR que 0.2449 *para que o resultado da predição seja aproximadamente* 10.7665, *de forma que todas as outras condições sejam respeitadas*

De fato, a Tabela 4.19 exibe um exemplo presente no *dataset* que demonstra a validade da explicação contrafactual obtida através da interpretação da Árvore de Decisão. Trata-se de um exemplo/instância real, que respeita as condições da árvore (vide coluna "Condições respeitadas"), com exceção do último nó antes da folha na Figura 4.20.

feature	valor	Condições respeitadas
Sex	2	
Length	0.56	
Diameter	0.44	
Height	0.16	
Whole_weight	0.8645	<0.9022
Shucked_weight	0.3305	<0.4008
Viscera_weight	0.2075	
Shell_weight	0.26	>0.1672 <0.3885
target	**10**	

Tabela 4.19: *Exemplo que comprova a explicação contrafactual do experimento 2 do dataset (Abalone)*

4.6.4 Parkinsons Telemonitoring dataset

Instância 1

Consideremos a instância 1 do *dataset Parkinsons Telemonitoring dataset* apresentada na Seção 4.3.4, e sua respectiva Árvore de Decisão do modelo interpretável (Figura 4.24). A explicação contrafactual é obtida alterando-se a decisão do nó "$age \leq 74.5129$" para o valor "falso" e pode ser expressa pela proposição:

"age" deve ser MAIOR que 74.5129 *para que o resultado da predição seja aproximadamente* 37.273*, de forma que todas as outras condições sejam respeitadas*

De fato, a Tabela 4.20 exibe um exemplo presente no *dataset* que demonstra a validade da explicação contrafactual obtida através da interpretação da Árvore de Decisão. Trata-se de um exemplo/instância real, que respeita as condições da árvore (vide coluna "Condições respeitadas"), com exceção do último nó antes da folha na Figura 4.24.

feature	valor	Condições respeitadas
age	75	>69.4886 >71.5002 <76.9997 <75.4786
sex	0	
test_time	7.3625	
Jitter	0.00397	
Jitter Abs	0,03246	
Jitter RAP	0.00176	
Jitter PPQ5	0.00194	
Jitter DDP	0.00528	
Shimmer	0.02709	
Shimmer dB	246	
Shimmer APQ3	0.01372	
Shimmer APQ5	0.01549	
Shimmer APQ11	0.02316	
Shimmer DDA	0.04115	
NHR	0.015902	
HNR	21993	
RPDE	0.61539	
DFA	0.63644	
PPE	0.18907	
target	**39.24**	

Tabela 4.20: *Exemplo que comprova a explicação contrafactual do experimento 1 do dataset Parkinsons Telemonitoring*

Instância 2

Consideremos a instância 1 do *dataset Parkinsons Telemonitoring dataset* apresentada na Seção 4.3.4, e sua respectiva Árvore de Decisão do modelo interpretável (Figura 4.28). A explicação contrafactual é obtida alterando-se a decisão do nó "$age \leq 74.4994$" para o valor "falso" e pode ser expressa pela proposição:

"age" deve ser MAIOR que 74.4994 *para que o resultado da predição seja aproximadamente* 40.8875*, de forma que todas as outras condições sejam respeitadas*

De fato, a Tabela 4.21 exibe um exemplo presente no *dataset* que demonstra a validade da explicação contrafactual obtida através da interpretação da Árvore de Decisão. Trata-se de um exemplo/instância real, que respeita as condições

da árvore (vide coluna "Condições respeitadas"), com exceção do último nó antes da folha na Figura 4.28.

feature	valor	Condições respeitadas
age	75	>69.5002 >71.5003 <77.0011 <75.4786
sex	0	
test_time	35416	
Jitter	0.00426	
Jitter Abs	0,03755	
Jitter RAP	0.00184	
Jitter PPQ5	0.00224	
Jitter DDP	0.00552	
Shimmer	0.02913	
Shimmer dB	244	
Shimmer APQ3	0.01649	
Shimmer APQ5	0.01757	
Shimmer APQ11	0.02006	
Shimmer DDA	0.04947	
NHR	0.015422	
HNR	20893	
RPDE	0.58411	
DFA	0.66542	
PPE	0.19139	
target	**40.155**	

Tabela 4.21: *Exemplo que comprova a explicação contrafactual do experimento 2 do dataset Parkinsons Telemonitoring*

CAPÍTULO 5

Conclusão

No presente trabalho, exploramos questões de pesquisa fundamentais relacionadas ao uso de árvores de decisão como modelos interpretáveis em comparação com a regressão linear usada no método do Lime, estado da arte na interpretação agnóstica de modelos. As análises e argumentos respaldam várias conclusões importantes.

Primeiramente, mostramos que as árvores de decisão têm o potencial de oferecer uma maior precisão de predição do que a regressão linear. Isso sugere que as árvores de decisão são uma opção valiosa em problemas do mundo real que desafiam as simplificações da regressão linear.

A fidelidade alta por parte dos modelos baseados em Árvores de Decisão apoiam a hipótese que tais modelos simulam, de fato, o modelo preditor e, dessa forma, podemos concluir que as condições da árvore estão, de forma implícita, presentes nas "engrenagens"do modelo preditor.

A abordagem apresentada neste trabalho vai além da apresentação das importâncias das *features* de uma predição, mas também explora a possibilidade das Árvores de Decisão para geração de explicações contrafactuais.

Além disso, nossas investigações revelaram que as árvores de decisão apresentam os atributos mais importantes de forma mais concisa do que a regressão linear. A capacidade das árvores de decisão de classificar diretamente a importância dos atributos com base na estrutura da árvore facilita a interpretação e a tomada de decisões informadas.

Além das vantagens levantadas da nova abordagem com relação ao LIME, destacam-se também as vantagens com relação ao Tree-Lime, cujo leque de possibilidades de aplicação se restringe aos problemas de regressão, enquanto a abordagem apresentada explora tanto problemas de regressão quanto de classificação.

Finalmente, exploramos como as árvores de decisão podem ser usadas para gerar resultados alternativos, conhecidos como explicações contrafactuais. Essa abordagem permite não apenas entender o "porquê"das decisões do modelo,

mas também visualizar como diferentes valores de atributos poderiam ter levado a resultados alternativos. Isso tem implicações significativas para a interpretabilidade e a capacidade de justificar as decisões do modelo em uma variedade de cenários.

Em resumo, este trabalho oferece uma análise abrangente das vantagens das árvores de decisão como modelos interpretáveis em comparação com a regressão linear. Esses resultados indicam que as árvores de decisão podem ser uma escolha superior em muitos casos, especialmente quando se busca uma interpretação mais precisa, concisa e a capacidade de gerar explicações contrafactuais. No entanto, é importante destacar que a seleção entre esses modelos deve ser orientada pelo contexto e pelos requisitos específicos do problema em questão. A pesquisa aqui apresentada contribui significativamente para a compreensão dos benefícios das árvores de decisão como ferramentas valiosas na análise e interpretação de modelos de Aprendizado de Máquina.

5.1 Limitações

- **Uso apenas do CART para geração de árvore**: A exploração de outros algoritmos e técnicas para construção de árvores é válida, visto que diferentes algoritmos detém diferentes vantagens sobre determinados problemas;
- **Possibilidade de execução por instância, de forma individual**: A interpretabilidade global, através de amostras de instâncias, é possível a partir da interpretabilidade local, de uma única instância. No entanto, este trabalho limita-se apenas à segunda opção.
- **Restrição do domínio dos dados**: Técnicas de interpretabilidade como LIME e SHAP têm a capacidade lidar com dados de diferentes domínios, como texto e imagens. No entanto, a abordagem apresentada neste trabalho se limita aos dados tabulares.

5.2 Sugestão para trabalhos futuros

Como trabalhos futuros, sugerimos o teste de outros algoritmos de construção de Árvores de Decisão, a fim de avaliar seus pontos positivos e negativos, assim como a possibilidade de aplicação em problemas práticos do mundo real que possam ser avaliados por especialistas do domínio.

A fim de alcançar interpretabilidade com foco no modelo caixa-preta, para qualquer instância, a interpretabilidade através de amostras, como já realizada pelo LIME, há também de ser considerada como trabalhos futuros.

O aperfeiçoamento da técnica para expansão das possibilidades de aplicação em dados de domínios diferentes, como textos, também é algo a ser considerado como trabalho futuro.

Referências Bibliográficas

[1] AEBERHARD, S.; FORINA, M. **Wine**. UCI Machine Learning Repository, 1991. DOI: https://doi.org/10.24432/C5PC7J.

[2] ANYSZ, H.; BRZOZOWSKI, Ł.; KRETOWICZ, W.; NARLOCH, P. **Feature importance of stabilised rammed earth components affecting the compressive strength calculated with explainable artificial intelligence tools**. *Materials*, 13(10):2317, 2020.

[3] ARCHER, K. J.; KIMES, R. V. **Empirical characterization of random forest variable importance measures**. *Computational statistics & data analysis*, 52(4):2249–2260, 2008.

[4] BAEHRENS, D.; SCHROETER, T.; HARMELING, S.; KAWANABE, M.; HANSEN, K.; MÜLLER, K.-R. **How to explain individual classification decisions**. *The Journal of Machine Learning Research*, 11:1803–1831, 2010.

[5] BRAMER, M. **Principles of Data Mining**. Springer Publishing Company, Incorporated, 3rd edition, 2016.

[6] BREIMAN, L.; FRIEDMAN, J.; STONE, C.; OLSHEN, R. **Classification and Regression Trees**. Taylor & Francis, 1984.

[7] BUGAJ, M.; WROBEL, K.; IWANIEC, J. **Model explainability using shap values for lightgbm predictions**. In: *2021 IEEE XVIIth International Conference on the Perspective Technologies and Methods in MEMS Design (MEMSTECH)*, p. 102–106. IEEE, 2021.

[8] CRAVEN, M.; SHAVLIK, J. **Extracting tree-structured representations of trained networks**. *Advances in neural information processing systems*, 8, 1995.

[9] ERTEL, W. **Introduction to artificial intelligence**. Springer, 2018.

[10] FISHER, R. A. **Iris**. UCI Machine Learning Repository, 1988. DOI: https://doi.org/10.24432/C56C76.

[11] GUIDOTTI, R. **Counterfactual explanations and how to find them: literature review and benchmarking**. *Data Mining and Knowledge Discovery*, p. 1–55, 2022.

[12] HARRIS, C. R.; MILLMAN, K. J.; VAN DER WALT, S. J.; GOMMERS, R.; VIRTANEN, P.; COURNAPEAU, D.; WIESER, E.; TAYLOR, J.; BERG, S.; SMITH, N. J.; KERN, R.; PICUS, M.; HOYER, S.; VAN KERKWIJK, M. H.; BRETT, M.; HALDANE, A.; DEL RÍO, J. F.; WIEBE, M.; PETERSON, P.; GÉRARD-MARCHANT, P.; SHEPPARD, K.; REDDY, T.; WECKESSER, W.; ABBASI, H.; GOHLKE, C.; OLIPHANT, T. E. **Array programming with NumPy**. *Nature*, 585(7825):357–362, Sept. 2020.

[13] HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. H.; FRIEDMAN, J. H. **The elements of statistical learning: data mining, inference, and prediction**, volume 2. Springer, 2009.

[14] HUNTER, J. D. **Matplotlib: A 2d graphics environment**. *Computing in Science & Engineering*, 9(3):90–95, 2007.

[15] KE, G.; MENG, Q.; FINLEY, T.; WANG, T.; CHEN, W.; MA, W.; YE, Q.; LIU, T.-Y. **Lightgbm: A highly efficient gradient boosting decision tree**. *Advances in neural information processing systems*, 30:3146–3154, 2017.

[16] KUMAR, P.; SHARMA, M. **Predicting academic performance of international students using machine learning techniques and human interpretable explanations using lime—case study of an indian university**. In: *International Conference on Innovative Computing and Communications: Proceedings of ICICC 2019, Volume 1*, p. 289–303. Springer, 2020.

[17] KUMARAKULASINGHE, N. B.; BLOMBERG, T.; LIU, J.; LEAO, A. S.; PAPAPETROU, P. **Evaluating local interpretable model-agnostic explanations on clinical machine learning classification models**. In: *2020 IEEE 33rd International Symposium on Computer-Based Medical Systems (CBMS)*, p. 7–12. IEEE, 2020.

[18] LI, H.; FAN, W.; SHI, S.; CHOU, Q. **A modified lime and its application to explain service supply chain forecasting**. In: *CCF International Conference on Natural Language Processing and Chinese Computing*, p. 637–644. Springer, 2019.

[19] LIPTON, Z. C. **The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery.** *Queue*, 16(3):31–57, 2018.

[20] LUNDBERG, S. M.; LEE, S.-I. **A unified approach to interpreting model predictions**. In: Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan,

S.; Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, p. 4765–4774. Curran Associates, Inc., 2017.

[21] MAGESH, P. R.; MYLOTH, R. D.; TOM, R. J. **An explainable machine learning model for early detection of parkinson's disease using lime on datscan imagery**. *Computers in Biology and Medicine*, 126:104041, 2020.

[22] MANEK, A. S.; SHENOY, P. D.; MOHAN, M. C. **Aspect term extraction for sentiment analysis in large movie reviews using gini index feature selection method and svm classifier**. *World wide web*, 20:135–154, 2017.

[23] MILLER, T. **Explanation in artificial intelligence: Insights from the social sciences**. *Artificial intelligence*, 267:1–38, 2019.

[24] MOLNAR, C. **Interpretable Machine Learning**. 2 edition, 2022.

[25] NASH, WARWICK, S. T. T. S. C. A.; FORD, W. **Abalone**. UCI Machine Learning Repository, 1995. DOI: https://doi.org/10.24432/C55C7W.

[26] PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V.; VANDERPLAS, J.; PASSOS, A.; COURNAPEAU, D.; BRUCHER, M.; PERROT, M.; DUCHESNAY, E. **Scikit-learn: Machine learning in Python**. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[27] QIN, Q.; ZHOU, X.; JIANG, Y. **Prognosis prediction of stroke based on machine learning and explanation model**. *INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL*, 16(2), 2021.

[28] QUINLAN, J. R. **C4. 5: programs for machine learning**. Elsevier, 2014.

[29] RANJBAR, N.; SAFABAKHSH, R. **Using decision tree as local interpretable model in autoencoder-based lime**. In: *2022 27th International Computer Conference, Computer Society of Iran (CSICC)*, p. 1–7. IEEE, 2022.

[30] RIBEIRO, M. T.; SINGH, S.; GUESTRIN, C. **"why should i trust you?"explaining the predictions of any classifier**. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, p. 1135–1144, 2016.

[31] RIBEIRO, M. T.; SINGH, S.; GUESTRIN, C. **Model-agnostic interpretability of machine learning**. *arXiv preprint arXiv:1606.05386*, 2016.

[32] TSANAS, A.; LITTLE, M. **Parkinsons Telemonitoring**. UCI Machine Learning Repository, 2009. DOI: https://doi.org/10.24432/C5ZS3N.

[33] WES MCKINNEY. **Data Structures for Statistical Computing in Python**. In: Stéfan van der Walt.; Jarrod Millman., editors, *Proceedings of the 9th Python in Science Conference*, p. 56 – 61, 2010.

[34] ZAFAR, M. R.; KHAN, N. **Deterministic local interpretable model-agnostic explanations for stable explainability**. *Machine Learning and Knowledge Extraction*, 3(3):525–541, 2021.

[35] ZHOU, Z.-H. **Machine learning**. Springer Nature, 2021.

[36] ZHOU, Z.-H. **Open-environment machine learning**. *National Science Review*, 9(8):nwac123, 2022.