Explicabilidade

SARAJANE MARQUES PERES

Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI

Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI

Alejandro Barredo Arrieta[a], Natalia Díaz-Rodríguez[b], Javier Del Ser[a,c,d], Adrien Bennetot[b,e,f], Siham Tabik[g], Alberto Barbado[h], Salvador Garcia[g], Sergio Gil-Lopez[a], Daniel Molina[g], Richard Benjamins[h], Raja Chatila[f], and Francisco Herrera[g]

[a]*TECNALIA, 48160 Derio, Spain*
[b]*ENSTA, Institute Polytechnique Paris and INRIA Flowers Team, Palaiseau, France*
[c]*University of the Basque Country (UPV/EHU), 48013 Bilbao, Spain*
[d]*Basque Center for Applied Mathematics (BCAM), 48009 Bilbao, Bizkaia, Spain*
[e]*Segula Technologies, Parc d'activité de Pissaloup, Trappes, France*
[f]*Institut des Systèmes Intelligents et de Robotique, Sorbonne Université, France*
[g]*DaSCI Andalusian Institute of Data Science and Computational Intelligence, University of Granada, 18071 Granada, Spain*
[h]*Telefonica, 28050 Madrid, Spain*

Abstract

In the last few years, Artificial Intelligence (AI) has achieved a notable momentum that, if harnessed appropriately, may deliver the best of expectations over many application sectors across the field. For this to occur shortly in Machine Learning, the entire community stands in front of the barrier of explainability, an inherent problem of the latest techniques brought by sub-symbolism (e.g. ensembles or Deep Neural Networks) that were not present in the last hype of AI (namely, expert systems and rule based models). Paradigms underlying this problem fall within the so-called *eXplainable* AI (XAI) field, which is widely acknowledged as a crucial feature for the practical deployment of AI models. The overview presented in this article examines the existing literature and contributions already done in the field of XAI, including a prospect toward what is yet to be reached. For this purpose we summarize previous efforts made to define explainability in Machine Learning, establishing a novel definition of explainable Machine Learning that covers such prior conceptual propositions with a major focus on the audience for which the explainability is sought. Departing from this definition, we propose and discuss about a taxonomy of recent contributions related to the explainability of different Machine Learning models, including those aimed at explaining Deep Learning methods for which a second dedicated taxonomy is built and examined in detail. This critical literature analysis serves as the motivating background for a series of challenges faced by XAI, such as the interesting crossroads of data fusion and explainability. Our prospects lead toward the concept of *Responsible Artificial Intelligence*, namely, a methodology for the large-scale implementation of AI methods in real organizations with fairness, model explainability and accountability at its core. Our ultimate goal is to provide newcomers to the field of XAI with a thorough taxonomy that can serve as reference material in order to stimulate future research advances, but also to encourage experts and professionals from other disciplines to embrace the benefits of AI in their activity sectors, without any prior bias for its lack of interpretability.

Keywords: Explainable Artificial Intelligence, Machine Learning, Deep Learning, Data Fusion, Interpretability, Comprehensibility, Transparency, Privacy, Fairness, Accountability, Responsible Artificial Intelligence.

*Corresponding author. TECNALIA. P. Tecnologico, Ed. 700. 48170 Derio (Bizkaia), Spain. E-mail: javier.delser@tecnalia.com

Taxonomia

Understandability/intelligibility: diz respeito à características de um modelo em fazer sua função ser entendida por um humano – **COMO O MODELO TRABALHA** – sem a necessidade de explicar sua estrutura interna ou os meios pelos quais o modelo processa dados internamente.

Comprehensibility: quando pensada para modelos de aprendizado de máquina, compreensibilidade se refere à **habilidade de um algoritmo de aprendizado de representar O SEU CONHECIMENTO APRENDIDO de forma compreensível para um humano**.

POSTULADO: os resultados da indução por computador deveriam ser descrições simbólicas de dadas entidades, semanticamente e estruturalmente similares ao que o especialista humano produz observando as mesmas entidades. Componentes destas descrições deveriam ser compreensíveis como pedaços únicos de informação, diretamente interpretáveis em linguagem natural, ou deveriam relacionar conceitos quantitativamente e qualitativamente de forma integrada.

Taxonomia

Interpretability: é definida como a habilidade de explicar ou fornecer um significado em termos compreensíveis para um humano.

Explainability: está associada com a noção de explicação **COMO UMA INTERFACE** entre humanos e um tomador de decisão. A explicação é, ao mesmo tempo, **um proxy preciso do tomador de decisão e compreensível para os humanos.**

Transparency: um modelo é considerado transparente se ele é **entendível por si próprio**. Como um modelo pode ter diferentes graus de *understandability*, os modelos transparentes podem ser divididos em: modelos simuláveis, modelos decomponíveis e modelos algoritmicamente transparentes.

Definições

IA explicável criará um conjunto de técnicas de aprendizado de máquina que capacita usuários humanos para **entender, confiar e gerenciar** a geração emergente de parceiros de inteligência artificial.

Dada uma certa audiência, explicabilidade se refere aos detalhes e razões que um modelo dá para tornar seu funcionamento claro e fácil de entender.

Dada uma certa audiência, uma inteligência artificial explicável é aquela que produz detalhes ou razões para tornar seu funcionamento claro e fácil de entender.

Audiência

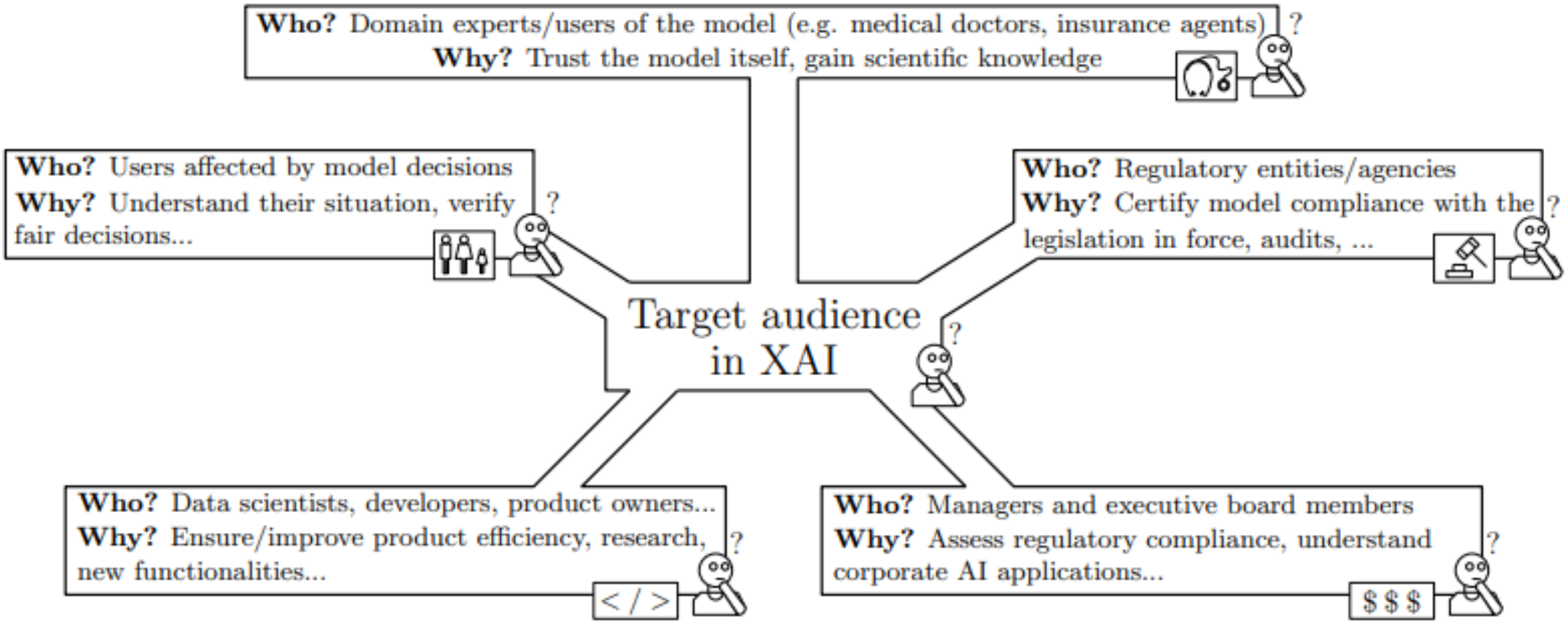


Figura 2

Explicabilidade

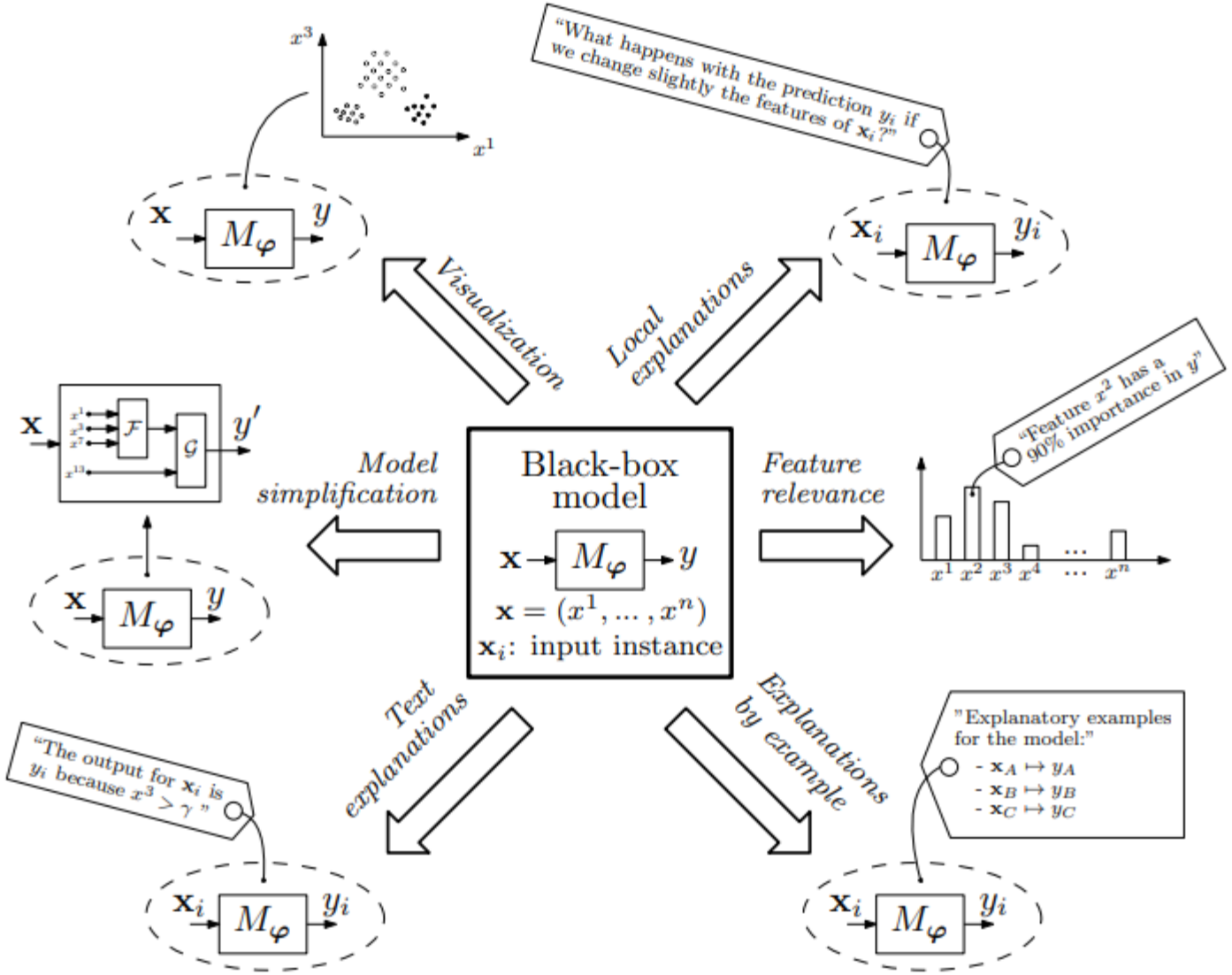


Figura 4

“Why Should I Trust You?” Explaining the Predictions of Any Classifier

Marco Tulio Ribeiro

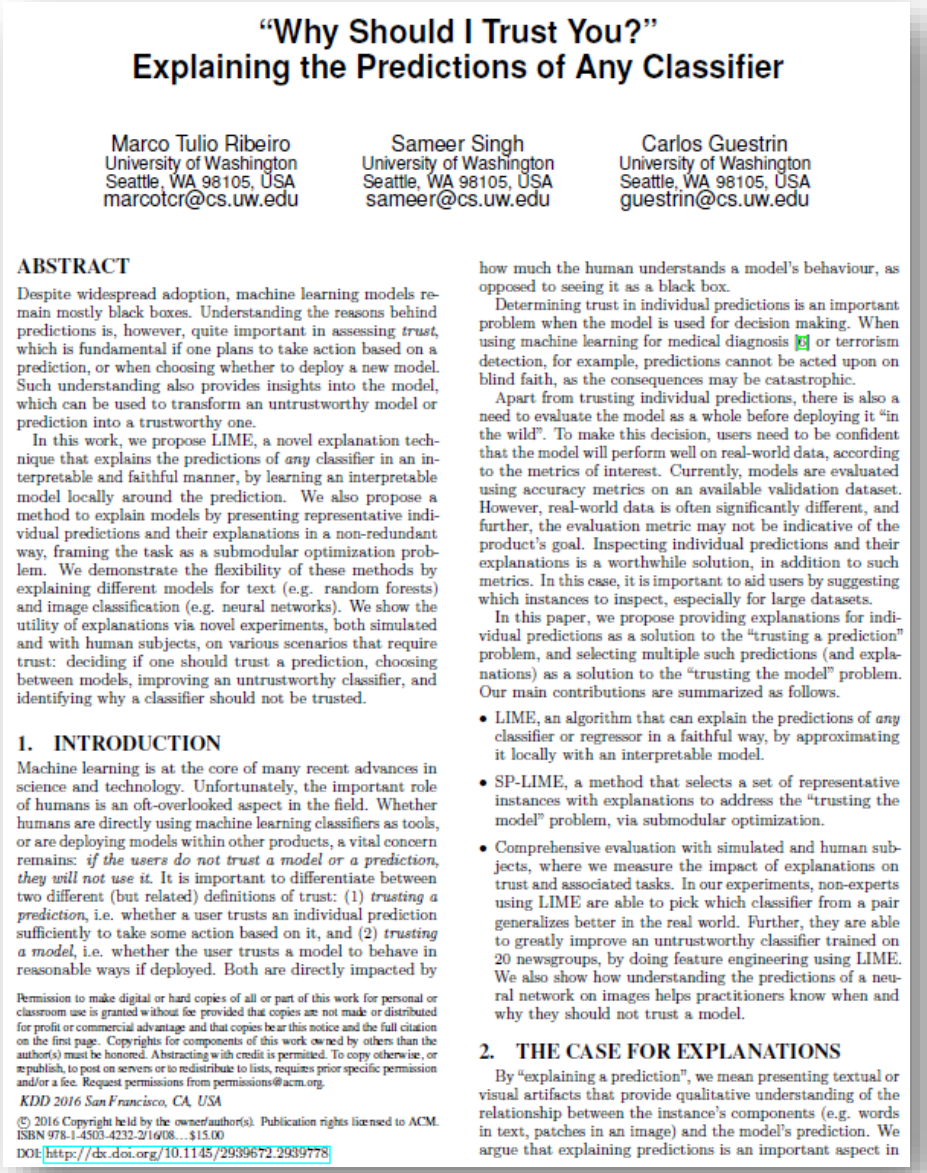
Sameer Singh

Carlos Guestrin

University of Washington

KDD 2016.

https://github.com/marcotcr/lime



“Why Should I Trust You?”
Explaining the Predictions of Any Classifier

Marco Tulio Ribeiro
University of Washington
Seattle, WA 98105, USA
marcotcr@cs.uw.edu

Sameer Singh
University of Washington
Seattle, WA 98105, USA
sameer@cs.uw.edu

Carlos Guestrin
University of Washington
Seattle, WA 98105, USA
guestrin@cs.uw.edu

ABSTRACT

Despite widespread adoption, machine learning models remain mostly black boxes. Understanding the reasons behind predictions is, however, quite important in assessing *trust*, which is fundamental if one plans to take action based on a prediction, or when choosing whether to deploy a new model. Such understanding also provides insights into the model, which can be used to transform an untrustworthy model or prediction into a trustworthy one.

In this work, we propose LIME, a novel explanation technique that explains the predictions of *any* classifier in an interpretable and faithful manner, by learning an interpretable model locally around the prediction. We also propose a method to explain models by presenting representative individual predictions and their explanations in a non-redundant way, framing the task as a submodular optimization problem. We demonstrate the flexibility of these methods by explaining different models for text (e.g. random forests) and image classification (e.g. neural networks). We show the utility of explanations via novel experiments, both simulated and with human subjects, on various scenarios that require trust: deciding if one should trust a prediction, choosing between models, improving an untrustworthy classifier, and identifying why a classifier should not be trusted.

1. INTRODUCTION

Machine learning is at the core of many recent advances in science and technology. Unfortunately, the important role of humans is an oft-overlooked aspect in the field. Whether humans are directly using machine learning classifiers as tools, or are deploying models within other products, a vital concern remains: *if the users do not trust a model or a prediction, they will not use it.* It is important to differentiate between two different (but related) definitions of trust: (1) *trusting a prediction*, i.e. whether a user trusts an individual prediction sufficiently to take some action based on it, and (2) *trusting a model*, i.e. whether the user trusts a model to behave in reasonable ways if deployed. Both are directly impacted by how much the human understands a model's behaviour, as opposed to seeing it as a black box.

Determining trust in individual predictions is an important problem when the model is used for decision making. When using machine learning for medical diagnosis [6] or terrorism detection, for example, predictions cannot be acted upon on blind faith, as the consequences may be catastrophic.

Apart from trusting individual predictions, there is also a need to evaluate the model as a whole before deploying it “in the wild”. To make this decision, users need to be confident that the model will perform well on real-world data, according to the metrics of interest. Currently, models are evaluated using accuracy metrics on an available validation dataset. However, real-world data is often significantly different, and further, the evaluation metric may not be indicative of the product's goal. Inspecting individual predictions and their explanations is a worthwhile solution, in addition to such metrics. In this case, it is important to aid users by suggesting which instances to inspect, especially for large datasets.

In this paper, we propose providing explanations for individual predictions as a solution to the “trusting a prediction” problem, and selecting multiple such predictions (and explanations) as a solution to the “trusting the model” problem. Our main contributions are summarized as follows.

- LIME, an algorithm that can explain the predictions of *any* classifier or regressor in a faithful way, by approximating it locally with an interpretable model.
- SP-LIME, a method that selects a set of representative instances with explanations to address the “trusting the model” problem, via submodular optimization.
- Comprehensive evaluation with simulated and human subjects, where we measure the impact of explanations on trust and associated tasks. In our experiments, non-experts using LIME are able to pick which classifier from a pair generalizes better in the real world. Further, they are able to greatly improve an untrustworthy classifier trained on 20 newsgroups, by doing feature engineering using LIME. We also show how understanding the predictions of a neural network on images helps practitioners know when and why they should not trust a model.

2. THE CASE FOR EXPLANATIONS

By “explaining a prediction”, we mean presenting textual or visual artifacts that provide qualitative understanding of the relationship between the instance's components (e.g. words in text, patches in an image) and the model's prediction. We argue that explaining predictions is an important aspect in

KDD 2016 San Francisco, CA, USA

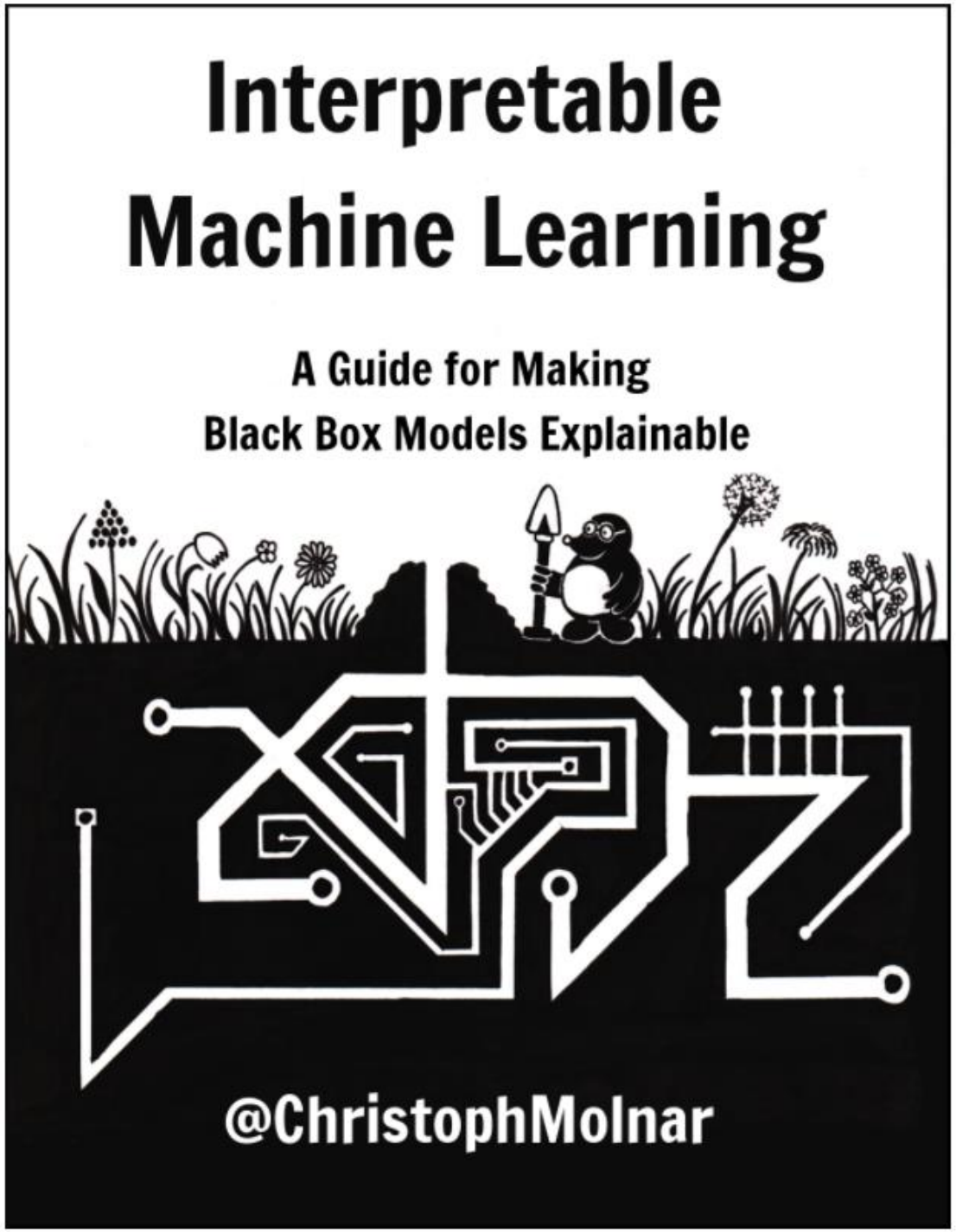
ISBN 978-1-4503-4232-2/16/08…$15.00
DOI: http://dx.doi.org/10.1145/2939672.2939778

Interpretable Machine Learning

A guide for making black box models explainable

Christoph Molnar

https://christophm.github.io/interpretable-ml-book/



Motivando …

Explicar uma decisão (predição) significa:

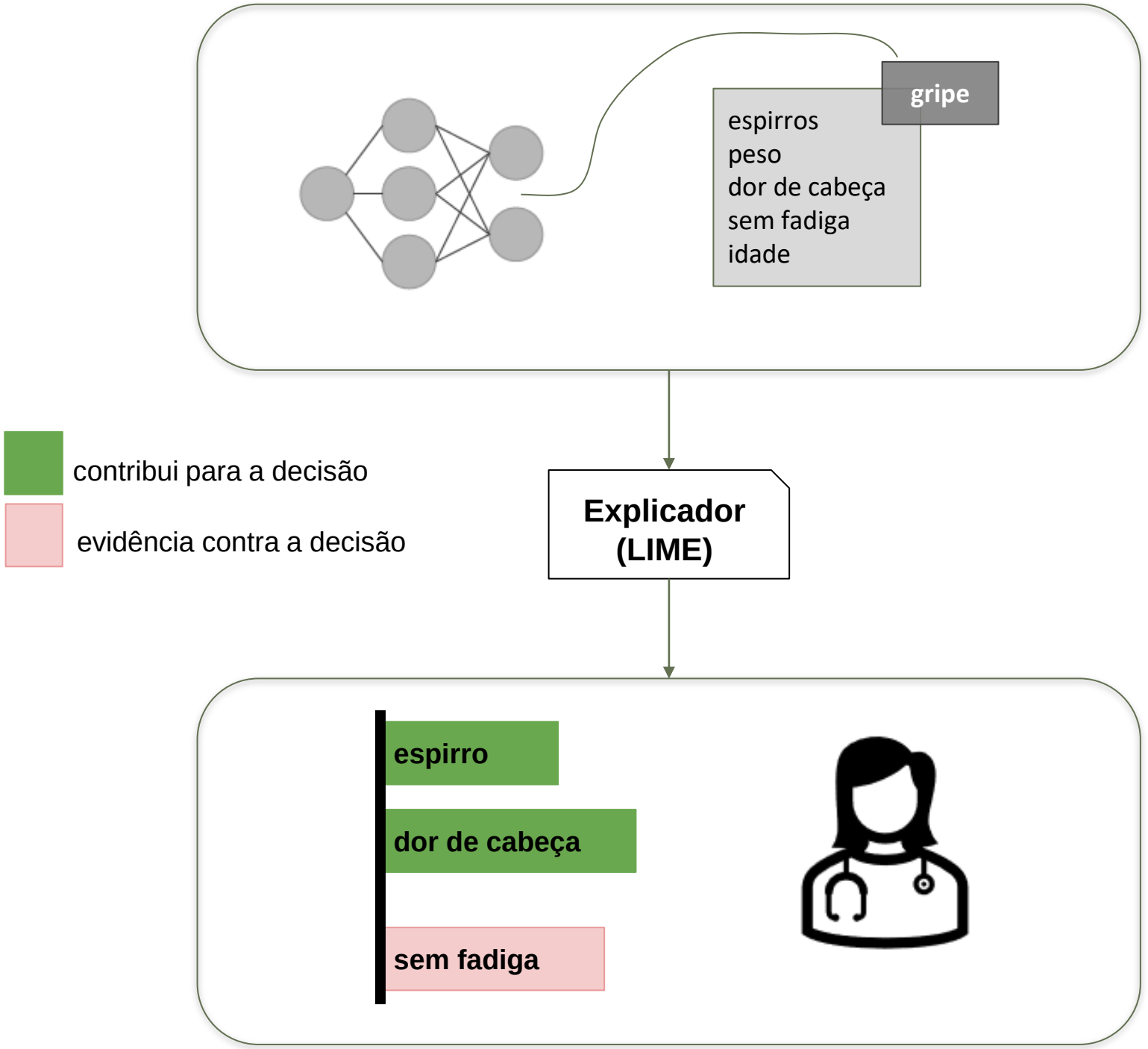
apresentar textos ou artefatos visuais que forneçam uma compreensão qualitativa da relação entre os componentes (atributos descritivos) da instância e a predição fornecida por um modelo para aquela instância.

Explicar é importante para fazer com que os humanos confiem e usem o aprendizado de máquina de maneira eficaz.

Isso ocorrerá se as explicações forem fiéis e compreensíveis.

Ribeiro, Singh and Guestrin, 2016

Motivando ...



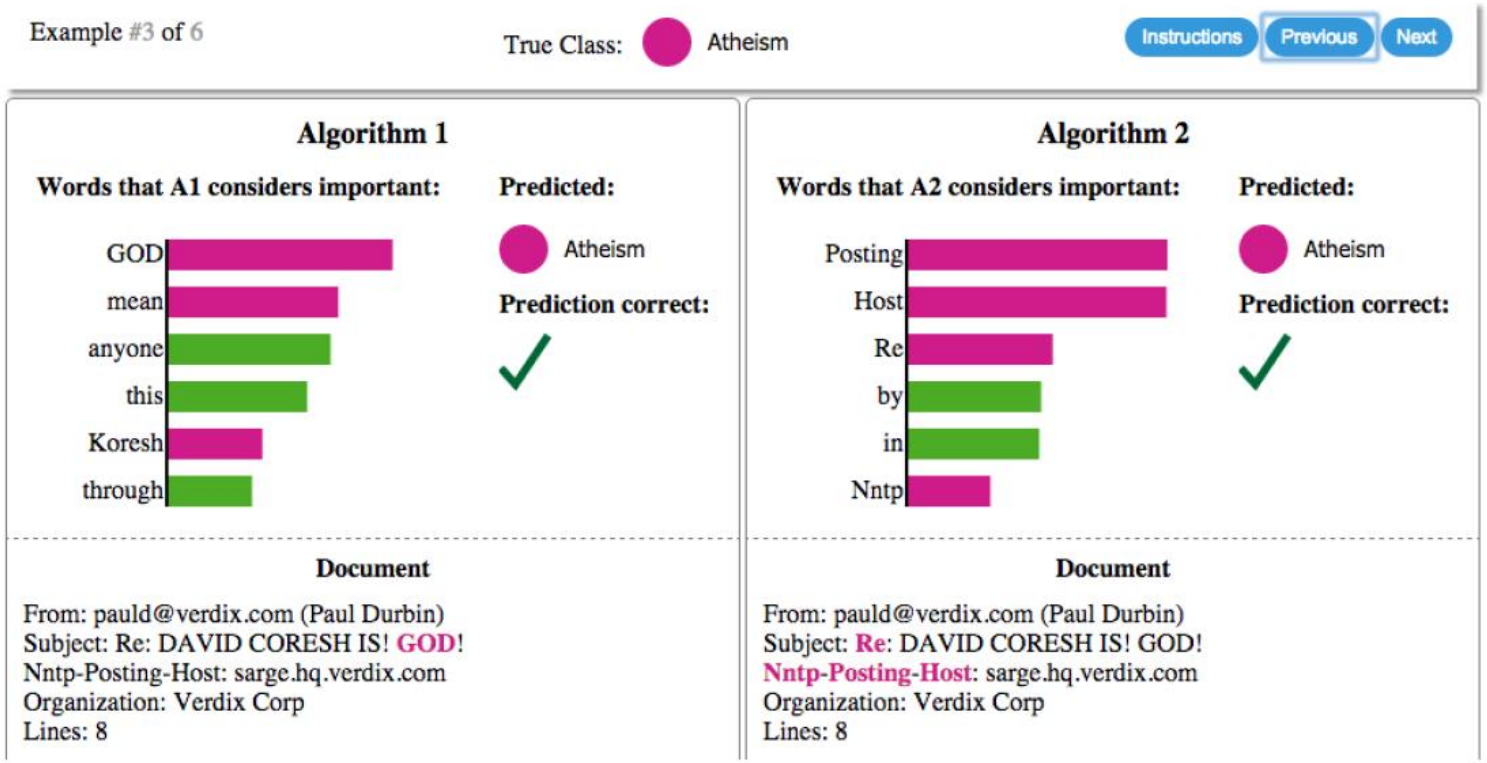
Ribeiro, Singh and Guestrin, 2016

Motivando ...

Problema: classificar um texto como referente a “ateísmo” ou “cristianismo”

Contribui para “ateísmo”

Contribui para “cristianismo”



Explicações podem ajudar a confiar mais em um algoritmo do que em outro.

Propiciam:

- Escolha entre algoritmos com acurácias competitivas.
- Escolha de um algoritmo menos acurado.

Ribeiro, Singh and Guestrin, 2016

Modelos substitutos locais usados para explicar decisões (predições) de modelos de aprendizado de máquina do tipo “caixa preta”.

Os modelos substitutos são treinados para aproximar as decisões (predições) de um outro modelo (caixa preta) subjacente.

> **LIME (***Local interpretable model-agnostic explanations***)** se concentra no treinamento de modelos substitutos locais para explicar as previsões individuais.

Caixa preta: **tratamento** dado a um modelo. Olhamos o modelo como um objeto que recebe uma entrada e produz uma saída, sem que o processamento feito para consumo da entrada e produção da saída seja de nosso interesse.

https://christophm.github.io/interpretable-ml-book/

Surrogate local models

Características desejáveis para um explicador.

LIME - *Local interpretable model-agnostic explanations*

Explicações devem ser interpretáveis

Devem fornecer entendimento qualitativo entre as variáveis de entrada e a resposta associada a elas.

Isso implica levar em conta as limitações do usuário (quem é esse usuário?)

Apresentar uma centena de características importantes para a predição não vai contribuir com a compreensão do porquê a predição foi como foi.

Agnóstico ao modelo

Agnóstico ao modelo significa TRATAR o modelo original como um modelo caixa preta.

Traz flexibilidade para ser usado inclusive com novos (futuros) classificadores.

Fidelidade local

A explicação DEVE ser localmente fiel. Ela deve corresponder a como o modelo se comporta na vizinhança da instância de interesse.

Características que são globalmente importantes podem não ser importantes em um contexto local e vice-versa.

Explicações com **fidelidade** global ainda são um desafio.

Perspectiva global

Além de explicar as previsões, fornecer uma visão em **perspectiva** global é importante para verificar a confiança no modelo.

Com base nas explicações para cada previsão, devemos selecionar algumas explicações (aquelas mais representativas) para apresentar ao usuário.

Ribeiro, Singh and Guestrin, 2016

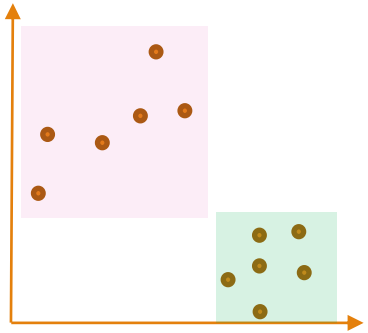
LIME - *Local interpretable model-agnostic explanations*

- Explicações interpretáveis localmente agnósticas ao modelo.

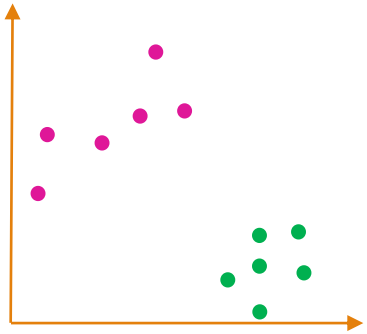
- Ideia geral:
  - **Selecione uma instância (dado)** para a qual você deseja obter uma explicação referente à predição oferecida pelo modelo *caixa preta*.
  - **Perturbe o conjunto de dados** e **obtenha predições** do modelo caixa preta para essas novas instâncias (o conjunto de dados perturbado).
  - **Pondere as novas instâncias** de acordo com a **similaridade** delas à instância de interesse.
  - Treine um **modelo interpretável** ponderado sobre o conjunto de dados com as variações.
  - **Explique** a predição **interpretando** esse modelo local.

LIME

Classificador

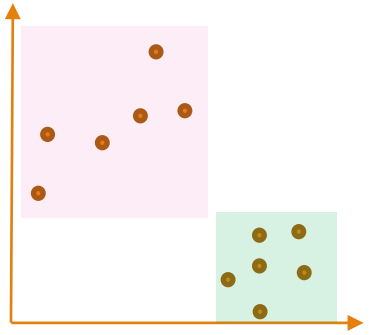


Modelo original – **tratado** como caixa preta

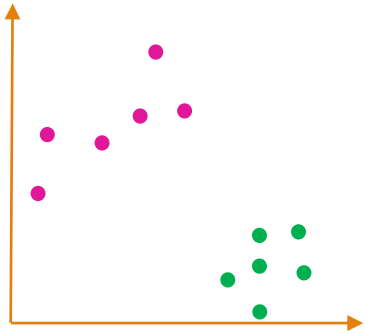


LIME

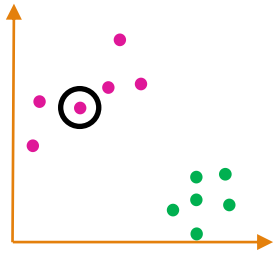
Classificador



Modelo original – **tratado** como caixa preta

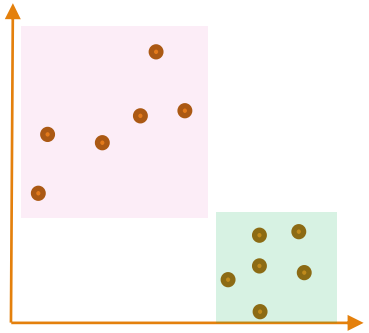


1 **Seleção de instância**

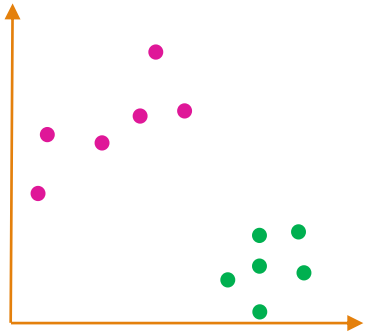


LIME

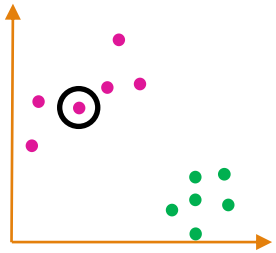
Classificador



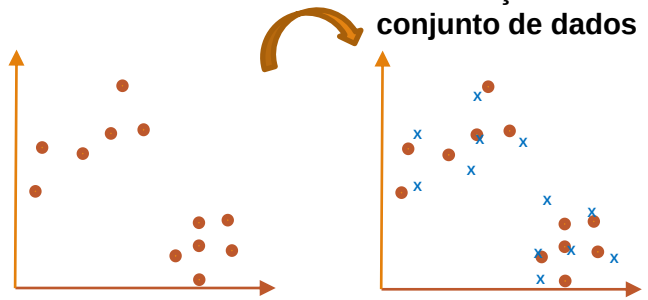
Modelo original – **tratado** como caixa preta



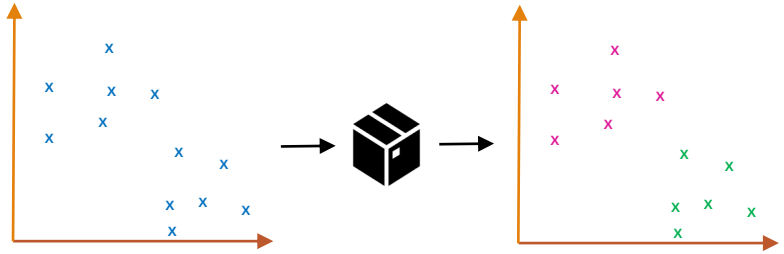
1 **Seleção de instância**



2 **Perturbação do conjunto de dados**



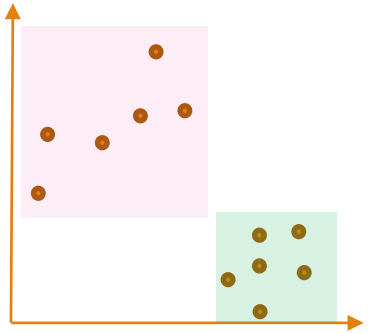
Predição (caixa preta) para novas instâncias



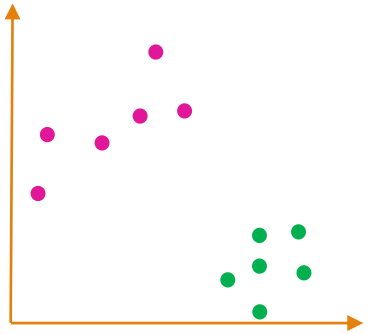
https://christophm.github.io/interpretable-ml-book/

LIME

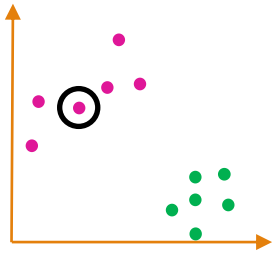
Classificador



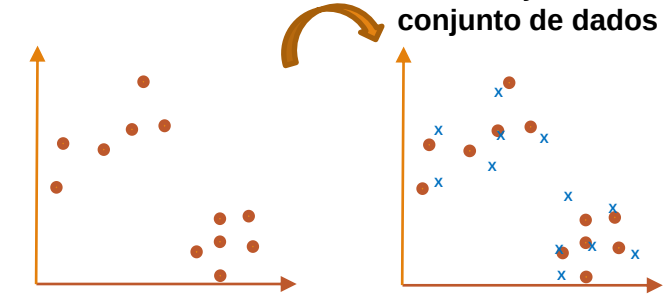
Modelo original – **tratado** como caixa preta



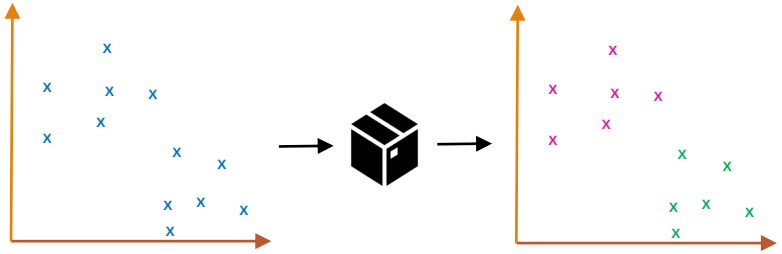
1 **Seleção de instância**



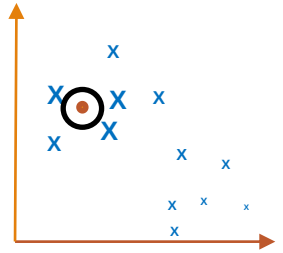
2 **Perturbação do conjunto de dados**



Predição (caixa preta) para novas instâncias



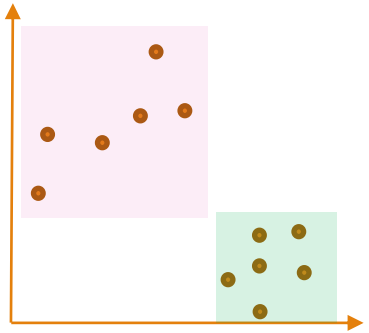
3 **Ponderação das novas instâncias**



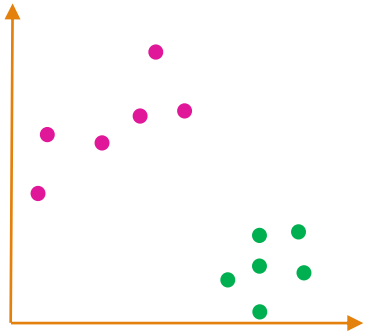
https://christophm.github.io/interpretable-ml-book/

LIME

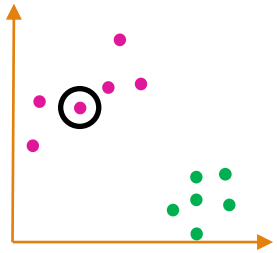
Classificador



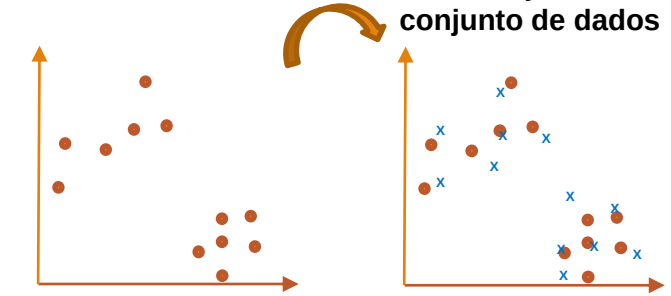
Modelo original – **tratado** como caixa preta



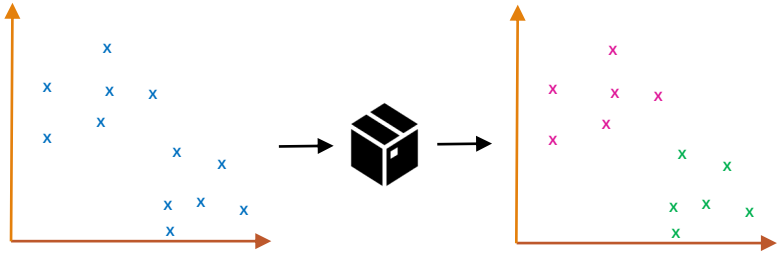
1 **Seleção de instância**



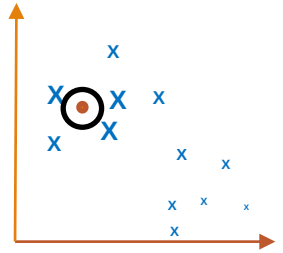
2 **Perturbação do conjunto de dados**



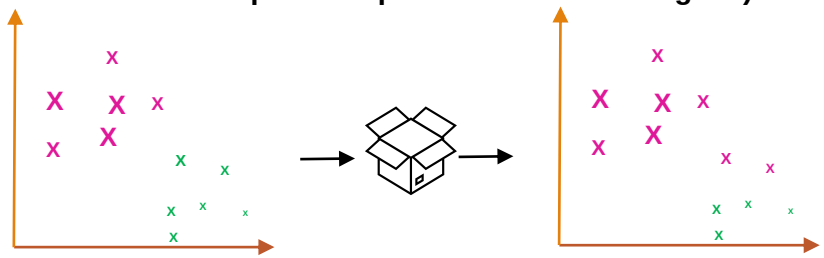
Predição (caixa preta) para novas instâncias



3 **Ponderação das novas instâncias**

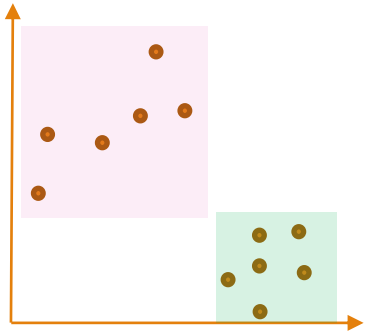


4 **Treinamento do modelo interpretável (com labels definidos pelas respostas do modelo original)**

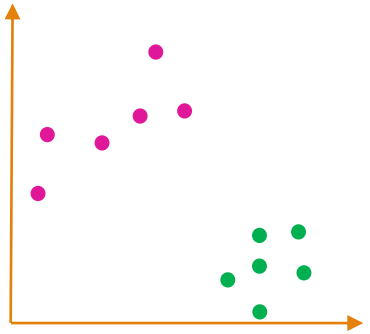


LIME

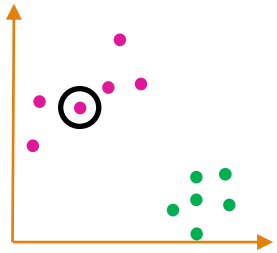
Classificador



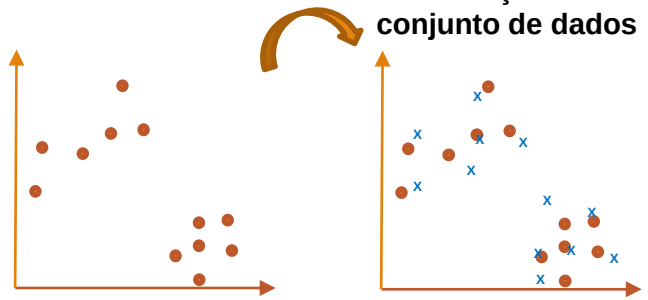
Modelo original – **tratado** como caixa preta



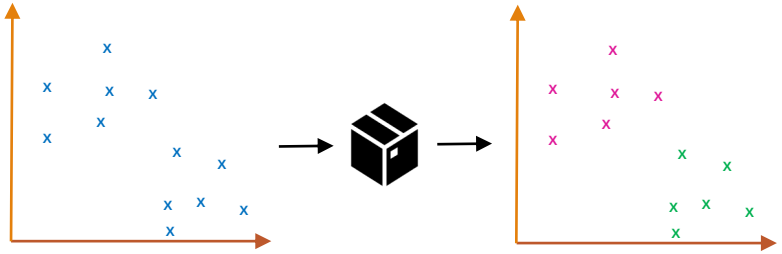
1 **Seleção de instância**



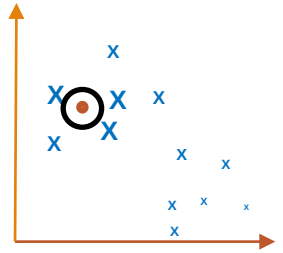
2 **Perturbação do conjunto de dados**



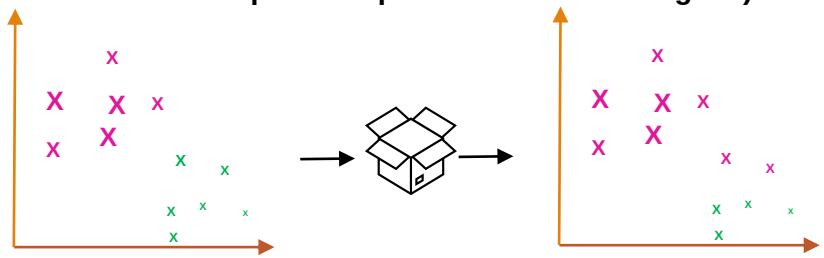
Predição (caixa preta) para novas instâncias



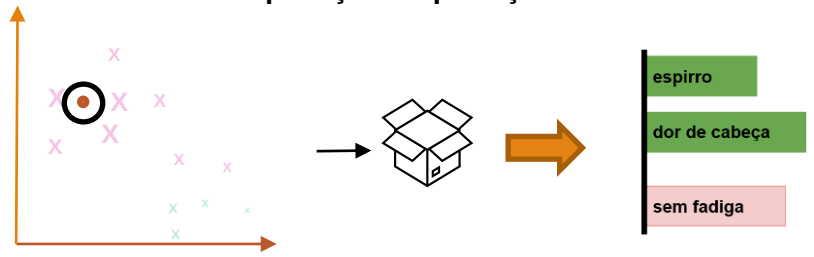
3 **Ponderação das novas instâncias**



4 **Treinamento do modelo interpretável (com labels definidos pelas respostas do modelo original)**



5 **Explicação da predição**



LIME - preliminares

Conceito de representação interpretável

Tensores ou *embeddings* não são interpretáveis.

Vetores binários com ausência ou presença de palavras ou super-pixels são interpretáveis.

- Representação original: conjunto de características “incompreensíveis”.

$$x \in \mathbb{R}^d$$

- Representação interpretável: um vetor binário (a representação interpretável).

$$x' \in \{0,1\}^{d'}$$

Ribeiro, Singh and Guestrin, 2016

LIME - preliminares

- Uma explicação é um modelo pertencente a uma classe de modelos potencialmente interpretáveis (por exemplo, uma árvore de decisão).

$$g \in G$$

- Idealmente, um modelo interpretável pode ser prontamente apresentado ao usuário por meio de artefatos visuais ou textuais.

O domínio de *g* é

$$\{0,1\}^{d'}$$

ou seja, *g* atua sobre a ausência ou presença de componentes interpretáveis.

Ribeiro, Singh and Guestrin, 2016

LIME - preliminares

- Nem todo modelo *g* será simples o suficiente para ser interpretável. Assim, uma medida de **complexidade** da explicação é definida.

$$\Omega(g)$$

- Por exemplo, para árvores de decisão, essa medida de complexidade da explicação pode ser a profundidade da árvore.

- O modelo que será explicado é denotado por

$$f : \mathbb{R}^d \rightarrow \mathbb{R}$$

- Em um classificador, *f(x)* é a probabilidade (ou a indicação binária) com que *x* pertence a uma certa classe.

Se o modelo é um **classificador**, ele mapeia o espaço de representação em um espaço de classes (discretas).

Ribeiro, Singh and Guestrin, 2016

LIME - preliminares

- A medida de proximidade entre duas instâncias, *z* e *x*, é

$$\pi_x(z)$$

e define a localidade ao redor de *x* (vizinhança usada para ponderação).

- Finalmente,

$$\mathcal{L}(f, g, \pi_x)$$

é a medida do quão **infiel** *g* é em aproximar *f* na localidade definida p π_x .

Ribeiro, Singh and Guestrin, 2016

LIME - preliminares

- Para assegurar tanto interpretabilidade quanto fidelidade local, nós temos que minimizar

$$\mathcal{L}(f, g, \pi_x)$$

enquanto temos

$$\Omega(g)$$

baixa o suficiente para que possa ser interpretada por humanos. Assim, a explicação produzida pelo LIME é obtida por

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

A busca para minimização poder ser feita por perturbações (criando a vizinhança).

Ribeiro, Singh and Guestrin, 2016

LIME - preliminares

Nós queremos um explicador agnóstico, ou seja, nós não temos que fazer nenhuma suposição sobre o modelo *f*. Então, para aprender um comportamento local de *f*, nós usamos uma aproximação.

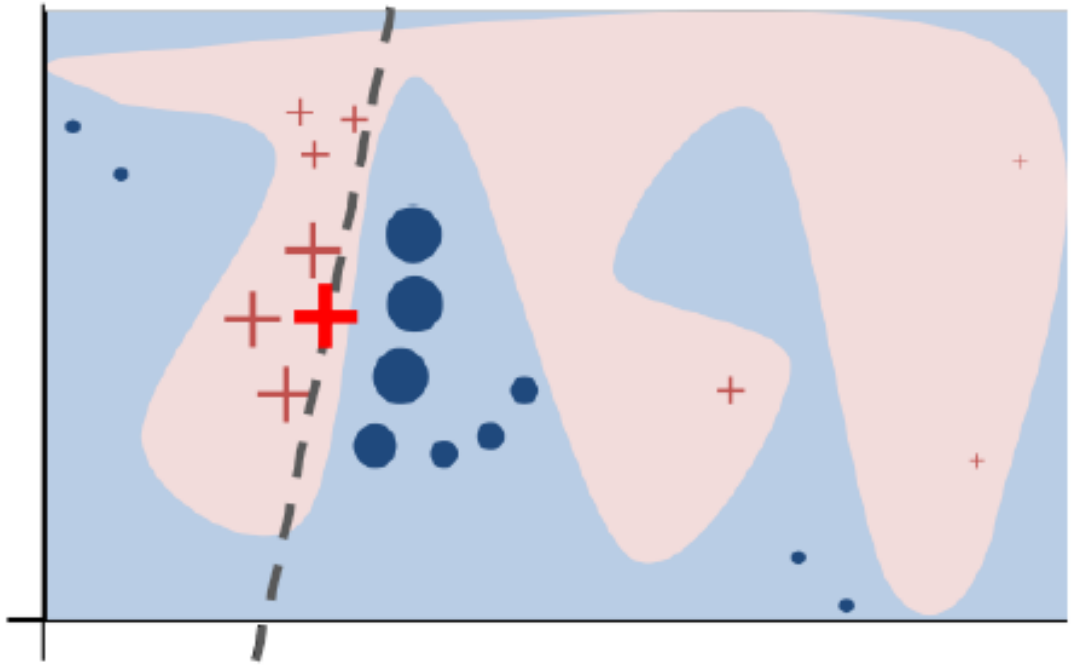
Instâncias (z') são amostradas uniformemente por aleatoridade e ponderadas de acordo com a vizinhança de *x'*. As instâncias são perturbadas de acordo com o espaço de representação explicável.

Para cada instância *z'* amostrada, nós recuperamos a amostra no espaço de representação original, obtendo *z*, e executamos *f(z)*. Assim saberemos o label que o modelo original dá a ela.

Dado o conjunto *Z* de amostras perturbadas e rotuladas, nós otimizamos o problema enunciado e obtemos as explicações $\xi(x)$.

Ribeiro, Singh and Guestrin, 2016

Intuição



- *f* (desconhecida para o LIME) é representada pelo fundo rosa/azul - uma superfície de decisão que não pode ser aproximada adequadamente por um modelo linear.
- A cruz vermelha em negrito é a instância que deve ser explicada
- Outras cruzes e círculos são instâncias amostradas, predições obtidas usando *f,* e seus respectivos pesos por proximidade da instância sob explicação (representados pelo tamanho)
- A linha tracejada é a explicação que foi aprendida localmente.

Ribeiro, Singh and Guestrin, 2016

LIME

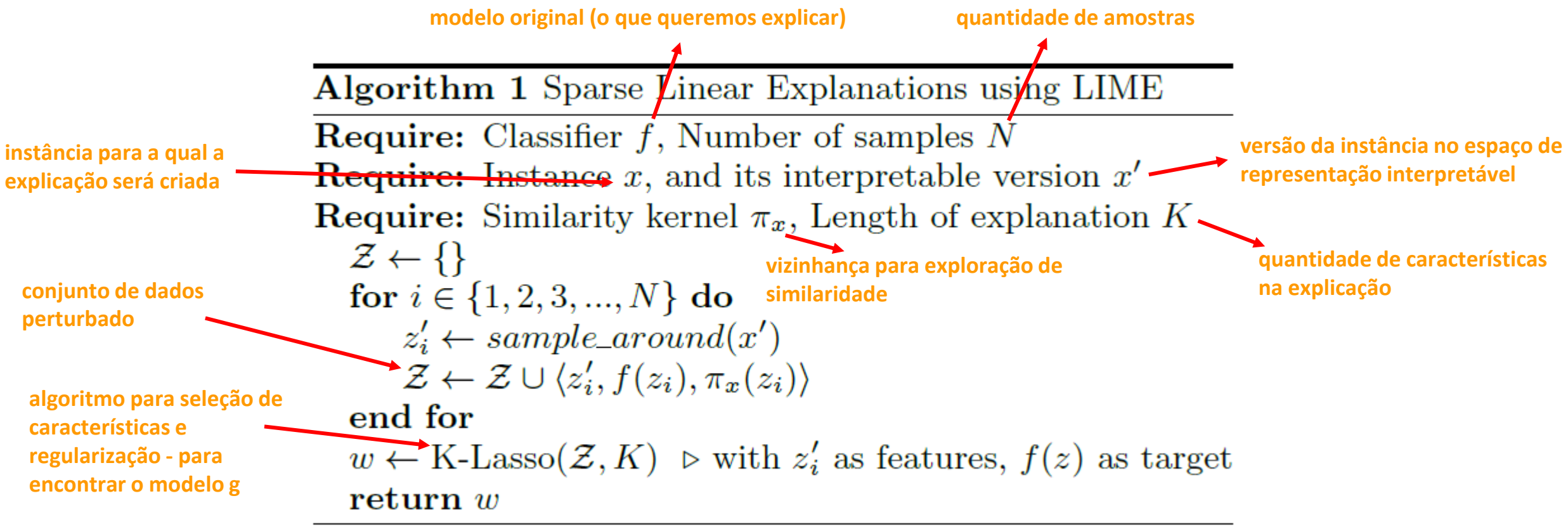
Algorithm 1 Sparse Linear Explanations using LIME

Require: Classifier f, Number of samples N
Require: Instance x, and its interpretable version x'
Require: Similarity kernel π_x, Length of explanation K

$\mathcal{Z} \leftarrow \{\}$
for $i \in \{1, 2, 3, ..., N\}$ **do**
  $z'_i \leftarrow sample\_around(x')$
  $\mathcal{Z} \leftarrow \mathcal{Z} \cup \langle z'_i, f(z_i), \pi_x(z_i) \rangle$
end for
$w \leftarrow$ K-Lasso$(\mathcal{Z}, K)$ ▷ with z'_i as features, $f(z)$ as target
return w

Ribeiro, Singh and Guestrin, 2016

LIME



Algorithm 1 Sparse Linear Explanations using LIME

Require: Classifier f, Number of samples N

Require: ~~Instance~~ x, and its interpretable version x'

Require: Similarity kernel π_x, Length of explanation K

$\mathcal{Z} \leftarrow \{\}$

for $i \in \{1, 2, 3, ..., N\}$ **do**

$z_i' \leftarrow sample\_around(x')$

$\mathcal{Z} \leftarrow \mathcal{Z} \cup \langle z_i', f(z_i), \pi_x(z_i) \rangle$

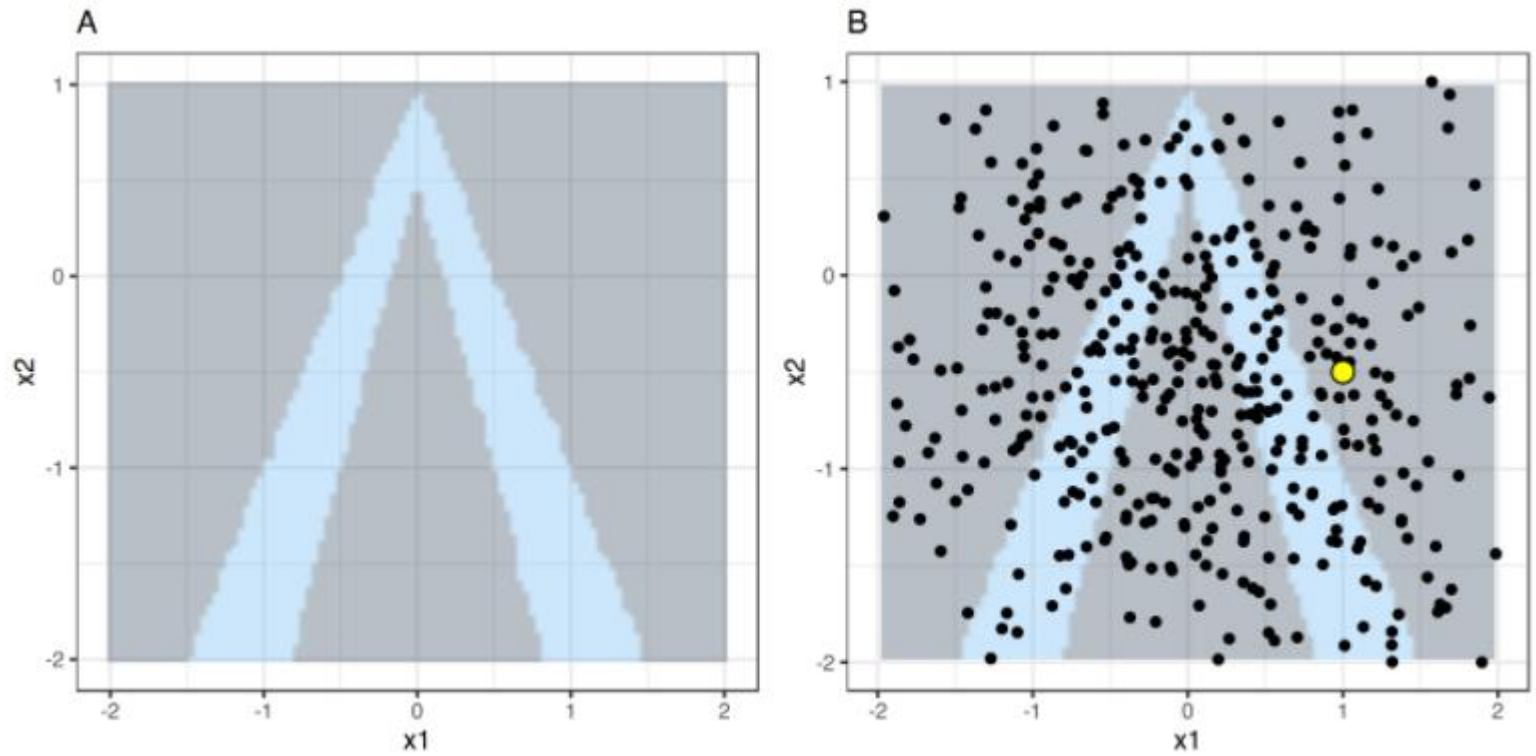
end for

$w \leftarrow$ K-Lasso$(\mathcal{Z}, K)$ ▷ with z_i' as features, $f(z)$ as target

return w

Ribeiro, Singh and Guestrin, 2016

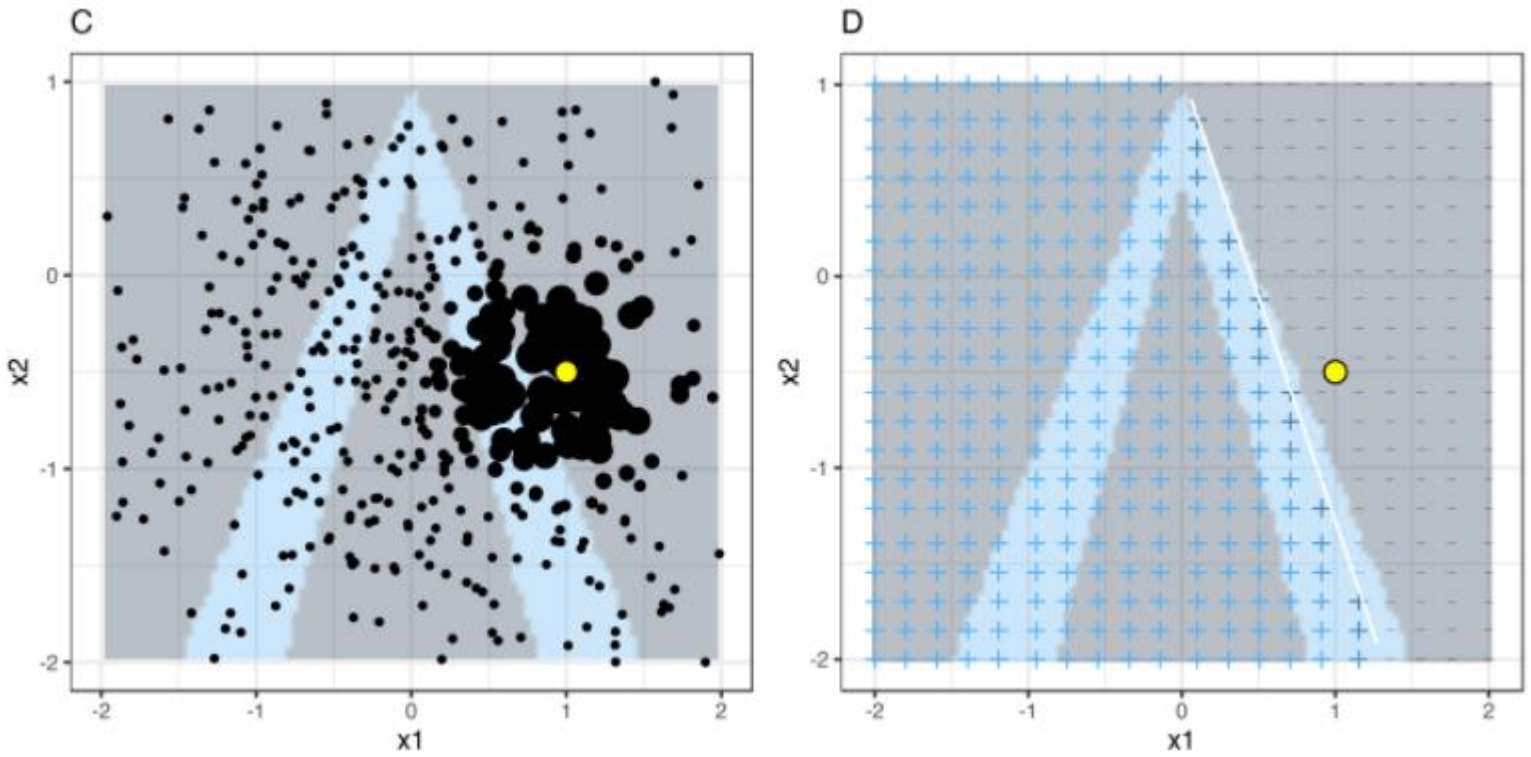
LIME para dados tabulares



- As amostras não são selecionadas apenas ao redor da instância de interesse, mas a partir do conjunto de dados todo. Esse procedimento aumenta a probabilidade de alguns resultados de predição diferirem do resultado de predição do ponto de interesse.
- Problema: predições de uma floresta aleatória dadas as características x1 e x2. Classes da predição: 1 (cinza) ou 0 (azul).
- A instância de interesse é o círculo amarelo. As amostras vêm de uma distribuição normal (pontos pretos).

https://christophm.github.io/interpretable-ml-book/

LIME para dados tabulares

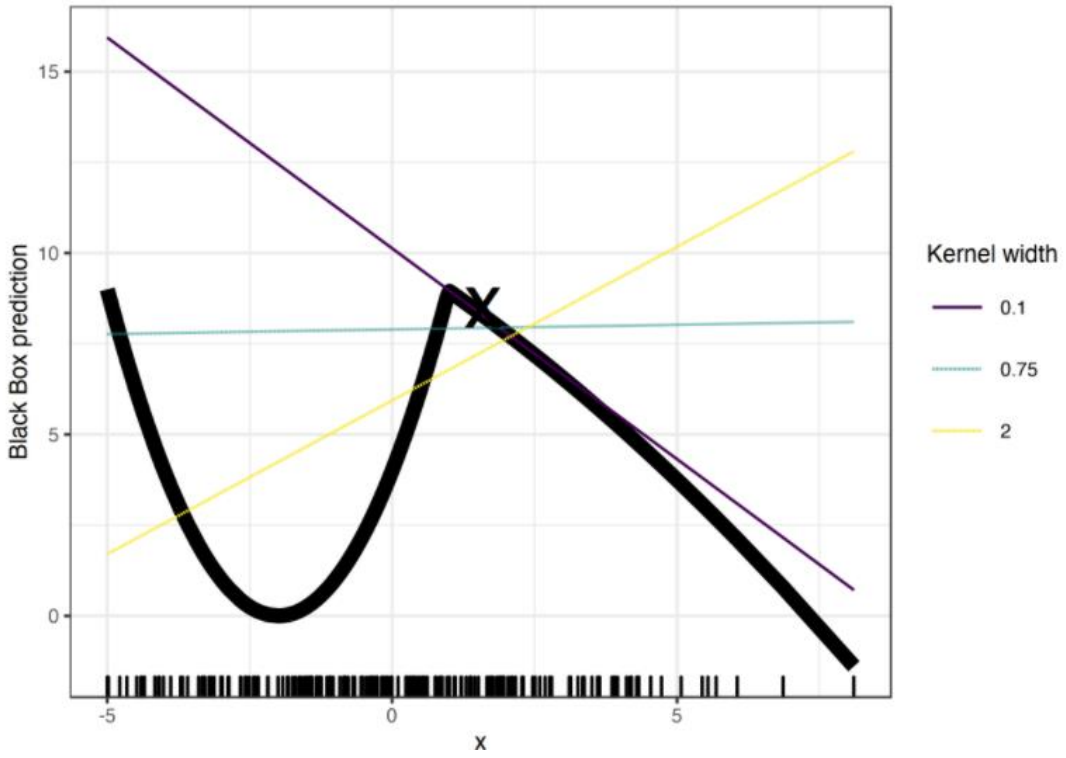


- Pesos altos são associados para as amostras na vizinhança da instância de interesse.
- A grade de sinais mostra as classificações obtidas a partir do modelo local aprendido a partir das amostras ponderadas.
- A linha branca marca a superfícies de decisão.

- A grande dificuldade do método LIME está na definição da função de vizinhança. Implementações sugerem estratégias nem sempre justificadas.

https://christophm.github.io/interpretable-ml-book/

Lime para dados tabulares



Explicação da predição de uma instância: x = 1.6

- A predição do modelo “caixa preta”, dependendo de uma única característica (x) é mostrada como uma linha grossa.
- A distribuição dos dados é mostrada na parte inferior da figura, sobre o eixo x.

Três modelos substitutos locais com **largura de kernels** (função para cálculo da vizinhança / da similaridade entre uma amostra e a instância de interesse) **diferentes**.

A similaridade/distância computada em cada característica pode ter características diferentes e podem não ser comparáveis entre si: kernels diferentes podem ter que ser usados.

https://christophm.github.io/interpretable-ml-book/

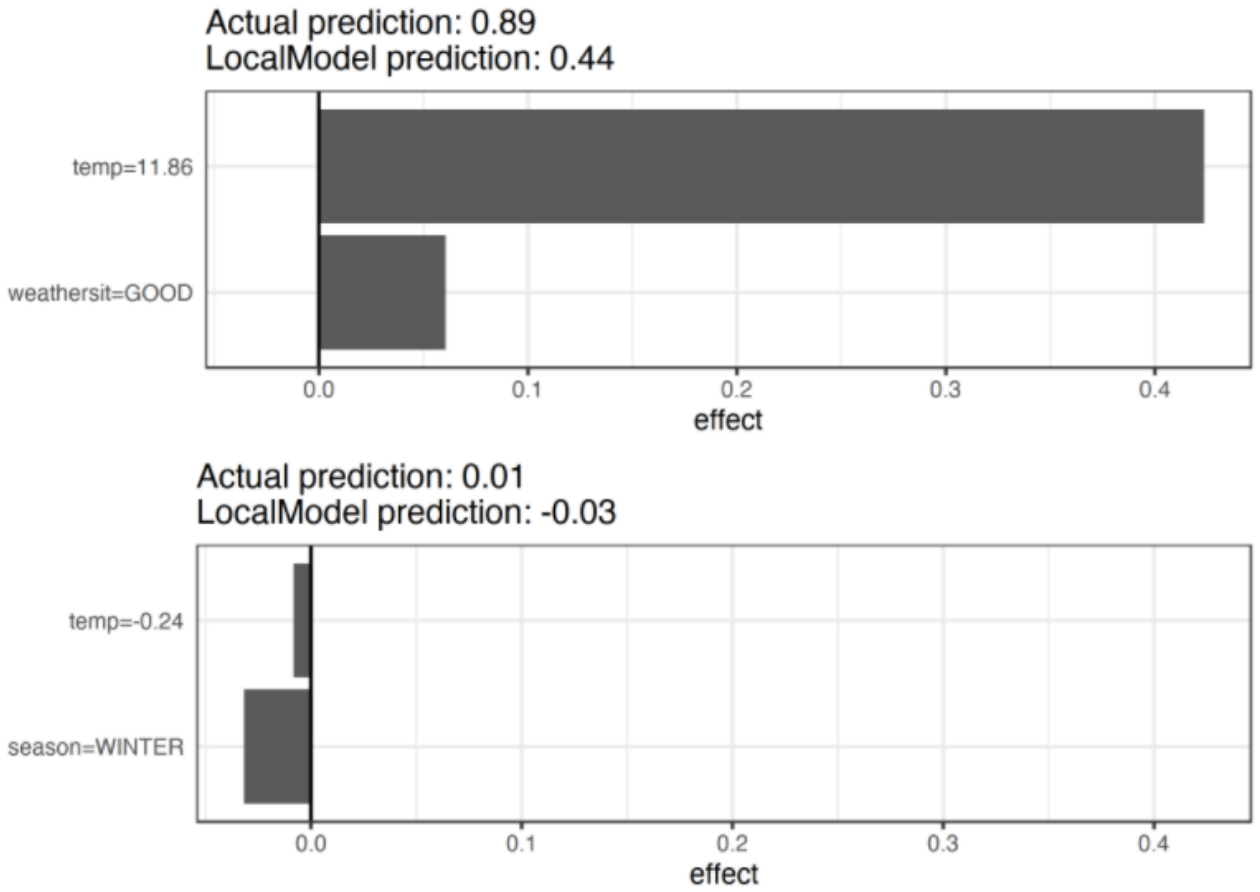
Exemplo: *Bike rental data*

Alugar bicicletas tem se tornado uma prática cada vez mais popular. Nós queremos saber se em um certo dia a quantidade de bicicletas alugadas estará acima ou abaixo de uma linha de tendência.

- Uma floresta aleatória com 100 árvores é treinada, dada uma tarefa de classificação.

- Há dias em que a quantidade de bicicletas alugadas estará acima de uma tendência. **Considerando informações de data e condições climáticas, o que explica o comportamento da predição?**

Exemplo: *Bike rental data*



https://christophm.github.io/interpretable-ml-book/

LIME para textos

- Amostragem: novos textos são criados a partir do texto original por meio de um procedimento de remoção aleatória de palavras.
- No exemplo, dois comentários (textos) são apresentados. O primeiro não foi classificado como spam. O segundo foi classificado como spam.

	CONTENT	CLASS
267	PSY is a good guy	0
173	For Christmas Song visit my channel! ;)	1

https://christophm.github.io/interpretable-ml-book/

LIME para textos

- Variações considerando o comentário 173 - classificado como spam pelo modelo original
- Algumas palavras foram removidas dando origem a novos textos na “vizinhança” da instância de interesse.
- **prob:** é a probabilidade de ser spam predita pelo modelo original
- **weight:** é a proximidade da sentença “perturbada” para a sentença original, calculada como 1 menos a proporção de palavras que foram removidas (se 1 de 7 palavras são removidas, a proximidade é 1 - 1/7 = 0.86).

For	Christmas	Song	visit	my	channel!	;)	prob	weight
1	0	1	1	0	0	1	0.17	0.57
0	1	1	1	1	0	1	0.17	0.71
1	0	0	1	1	1	1	0.99	0.71
1	0	1	1	1	1	1	0.99	0.86
0	1	1	1	0	0	1	0.17	0.57

https://christophm.github.io/interpretable-ml-book/

LIME para textos

Saída do algoritmo LIME

case	label_prob	feature	feature_weight
1	0.1701170	good	0.000000
1	0.1701170	a	0.000000
1	0.1701170	is	0.000000
2	0.9939024	channel!	6.180747
2	0.9939024	For	0.000000
2	0.9939024	;)	0.000000

not span

span

- A palavra “channel” indica a alta probabilidade de spam.
- Para o comentário que não é spam nenhum peso diferente de zero foi estimado. Isso significa que não importa qual palavra seja removida, a classe predita continua a mesma.

LIME para imagens

(a) Original Image

Intuitivamente, não faria sentido perturbar pixels individuais, porque muitos mais do que um pixel contribuem para a predição em uma classe.

Amostragem: As variações das imagens são criadas por um procedimento de segmentação da imagem em “super pixels” e conversão desse super píxel em “on” ou “off”.

“Super pixels” são pixels interconectados com cores similares e podem ser “desligados” por meio da substituição de cada pixels por uma cor especificada pelo usuário.

(c) Explaining *Acoustic guitar*

(b) Explaining *Electric guitar*

(d) Explaining *Labrador*

Ribeiro, Singh and Guestrin, 2016

Escolha submodular para explicar modelos

SP - LIME

Algorithm 2 Submodular pick (SP) algorithm

Require: Instances X, Budget B
  for all $x_i \in X$ **do**
    $\mathcal{W}_i \leftarrow$ **explain**(x_i, x_i') ▷ Using Algorithm 1
  end for
  for $j \in \{1 \dots d'\}$ **do**
    $I_j \leftarrow \sqrt{\sum_{i=1}^{n} |\mathcal{W}_{ij}|}$ ▷ Compute feature importances
  end for
  $V \leftarrow \{\}$
  while $|V| < B$ **do** ▷ Greedy optimization of Eq (4)
    $V \leftarrow V \cup \text{argmax}_i \, c(V \cup \{i\}, \mathcal{W}, I)$
  end while
  return V

- Explicações locais não são tão **confiáveis**.
- Explicações (totalmente) globais não são **viáveis**.

Dado um conjunto de dado *X*, selecionar *B* instâncias a serem inspecionadas (explicadas).

Para alcançar um bom resultado, o passo de escolha deveria levar em conta as explicações obtidas para cada predição.

É necessário escolher um conjunto representativo e diverso de explicações para mostrar ao usuário → explicações não redundantes que representem como o modelo se comporta globalmente.

Escolha submodular para explicar modelos

SP - LIME

Algorithm 2 Submodular pick (SP) algorithm

Require: Instances X, Budget B

for all $x_i \in X$ **do**
  $\mathcal{W}_i \leftarrow \textbf{explain}(x_i, x'_i)$ ▷ Using Algorithm 1
end for
for $j \in \{1 \dots d'\}$ **do**
  $I_j \leftarrow \sqrt{\sum_{i=1}^{n} |\mathcal{W}_{ij}|}$ ▷ Compute feature importances
end for
$V \leftarrow \{\}$
while $|V| < B$ **do** ▷ Greedy optimization of Eq (4)
  $V \leftarrow V \cup \text{argmax}_i \, c(V \cup \{i\}, \mathcal{W}, I)$
end while
return V

Dadas as explicações para um conjunto de instâncias *n* de *X*, nós construímos uma matriz de explicações *W*, de dimensões *n x d'*.

A matriz *W* representa a importância local dos componentes interpretáveis para cada instância.

A matriz *W* é construída aplicando o algoritmo LIME para cada uma das instâncias, e obtendo a "importância" local de cada características para cada instância - ou seja, a explicação para cada instância é uma linha da matriz *W*.

Escolha submodular para explicar modelos

SP - LIME

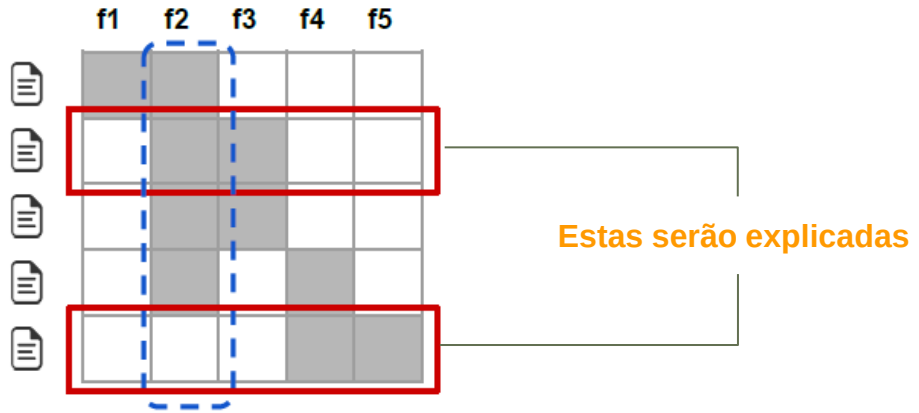
Algorithm 2 Submodular pick (SP) algorithm

Require: Instances X, Budget B
  for all $x_i \in X$ **do**
    $\mathcal{W}_i \leftarrow$ **explain**(x_i, x'_i) ▷ Using Algorithm 1
  end for
  for $j \in \{1 \dots d'\}$ **do**
    $I_j \leftarrow \sqrt{\sum_{i=1}^{n} |\mathcal{W}_{ij}|}$ ▷ Compute feature importances
  end for
  $V \leftarrow \{\}$
  while $|V| < B$ **do** ▷ Greedy optimization of Eq (4)
    $V \leftarrow V \cup \text{argmax}_i\, c(V \cup \{i\}, \mathcal{W}, I)$
  end while
  return V

Então, para cada uma das características (colunas *j* em *W*) é preciso calcular a importância global *Ij*.

Intuitivamente, *I* é tal que características que explicam várias instâncias receberão pontuações (importância) maiores.

f2 é a característica mais importante



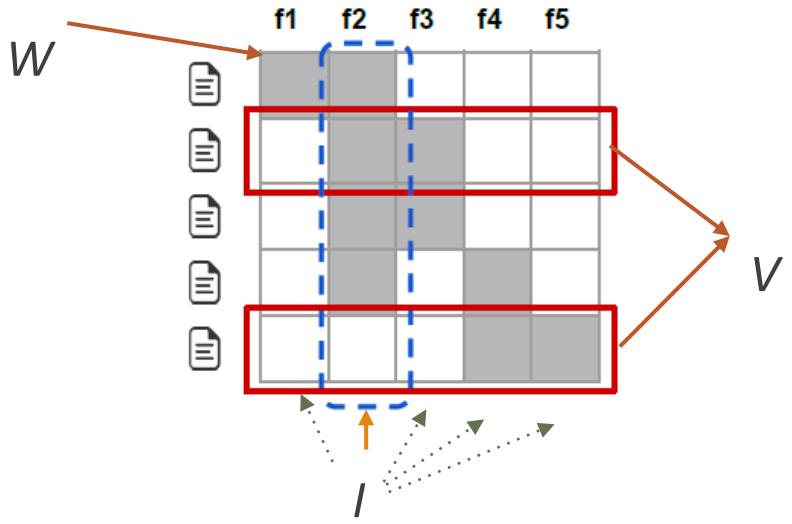
Escolha submodular para explicar modelos

SP - LIME

Algorithm 2 Submodular pick (SP) algorithm

Require: Instances X, Budget B
  for all $x_i \in X$ **do**
    $\mathcal{W}_i \leftarrow$ **explain**(x_i, x_i') ▷ Using Algorithm 1
  end for
  for $j \in \{1 \dots d'\}$ **do**
    $I_j \leftarrow \sqrt{\sum_{i=1}^{n} |\mathcal{W}_{ij}|}$ ▷ Compute feature importances
  end for
  $V \leftarrow \{\}$
  while $|V| < B$ **do** ▷ Greedy optimization of Eq (4)
    $V \leftarrow V \cup \text{argmax}_i\, c(V \cup \{i\}, \mathcal{W}, I)$
  end while
  return V

Construir um conjunto *V* com B instâncias que maximiza a importância total de características que aparecem em pelo menos uma instância de *V*. Isso é construído por uma função que recebe como entrada *W* e *I*.



Explicabilidade

SARAJANE MARQUES PERES