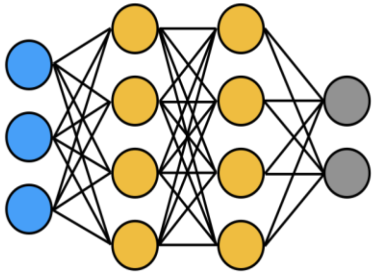
Lecture 18: Explainability

Trustworthy Machine Learning

Spring 2024

Beyond Accuracy



Malignant

Why did the model make this prediction?

"… the algorithm appeared more likely to label images with rulers as malignant … "

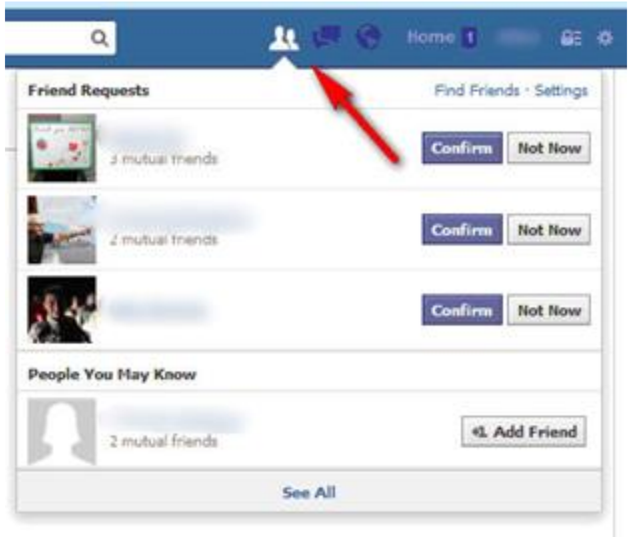
Goals of Explainable ML

- Explain why the model made a particular prediction on a specific input
- Explain how the model makes predictions across all inputs
- Explain how the training data affects model predictions
- Explain what changes to the input can cause the model make a different decision

Agenda

- Today:
  - Introduction
  - Feature attribution problem
  - LIME (Local Interpretable Model-agnostic Explanations) algorithm
- Resources:
  - Tutorial lectures on “Interpreting ML Models” by Hima Lakkaraju (Harvard)
  - “Why should I trust you?” Explaining the predictions of any classifier
    Ribeiro et al, KDD 2016 (LIME paper)

ML is everywhere, but is “explainable ML” needed everywhere?



When and Why "Explainable ML" ?

Explainability and Emerging AI Policy

EU General Data Protection Regulation (2018)

Right to explanation

…

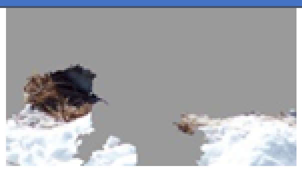
In any case, such processing should be subject to suitable safeguards, which should include specific information to the data subject and the right to obtain human intervention, to express his or her point of view, to obtain an explanation of the decision reached after such assessment and to challenge the decision.

Motivating Example

Input

Predictive Model

Explaining predictions facilitates debugging



Prediction = Siberian Husky

This model is

on incorrect

s to make

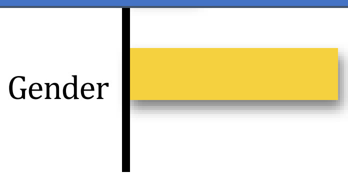
iction!! Let

he model

Motivating Example

Defendant De

Predictive
Model

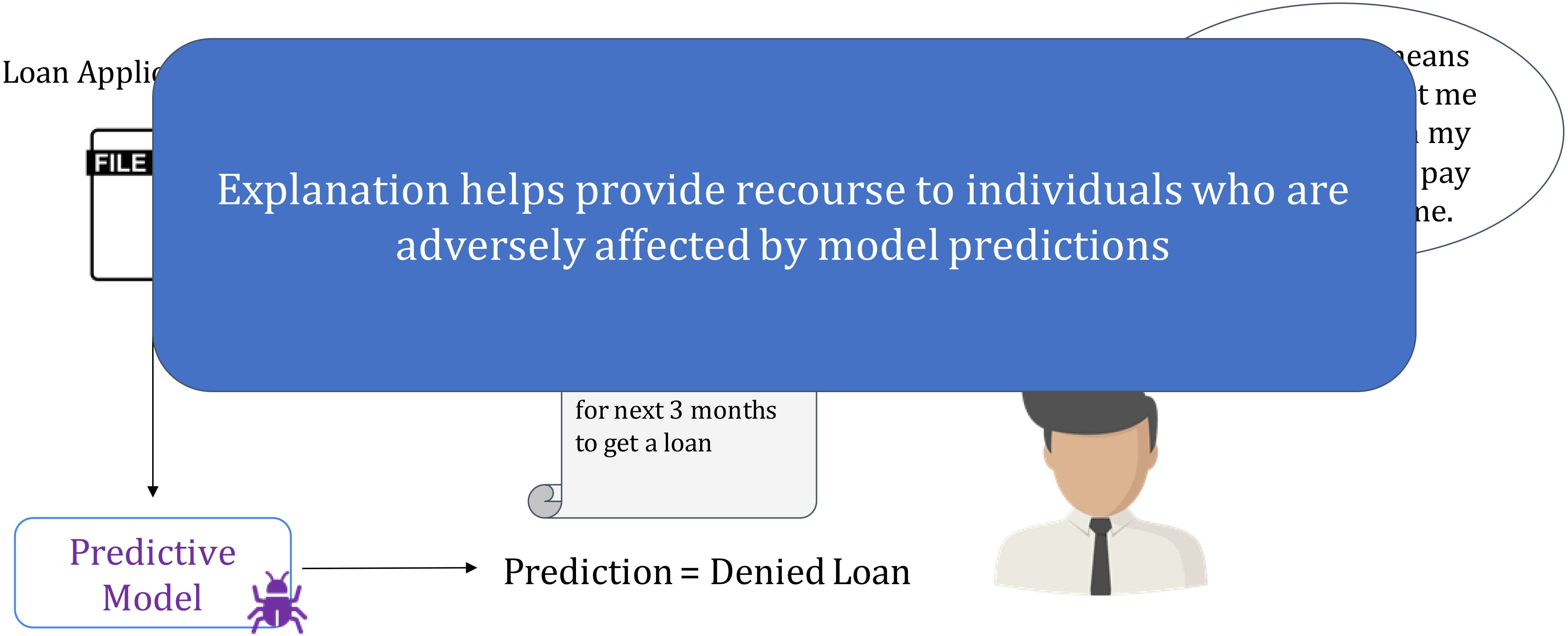


Prediction = Risky to Release

This prediction is
ce and
eing
e the
n!!

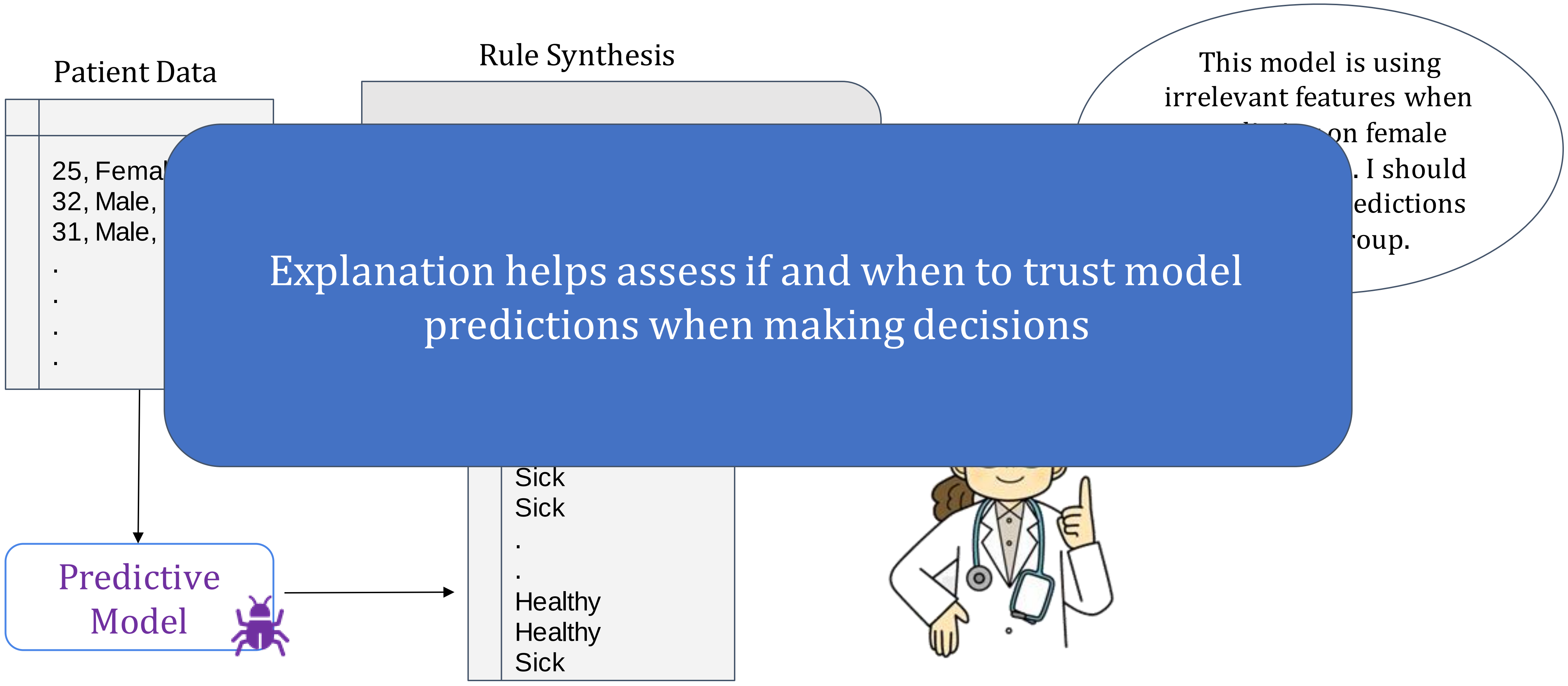
Explaining predictions facilitates bias detection

Motivating Example

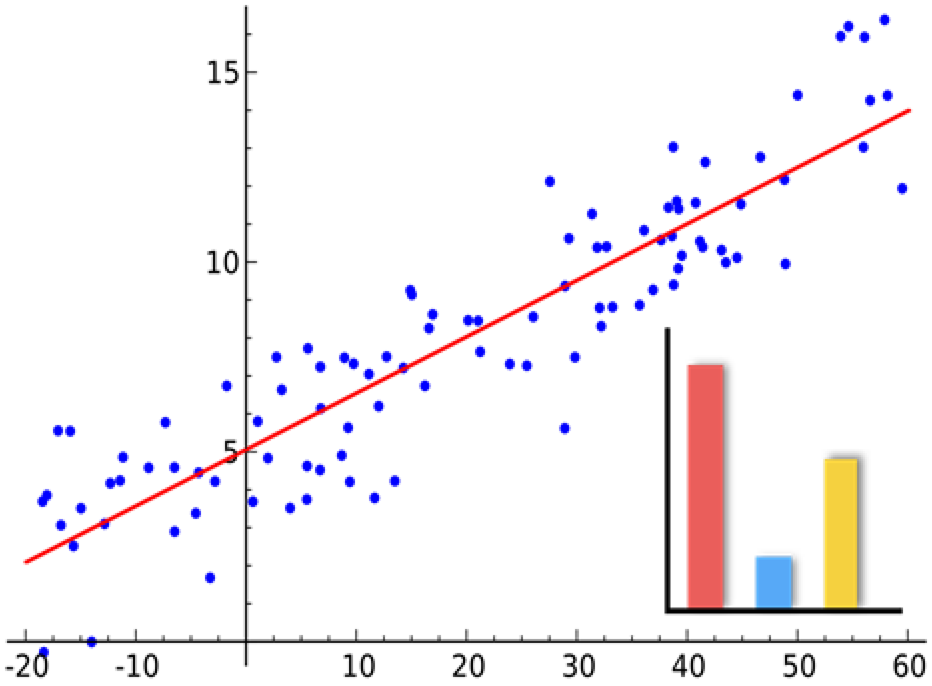


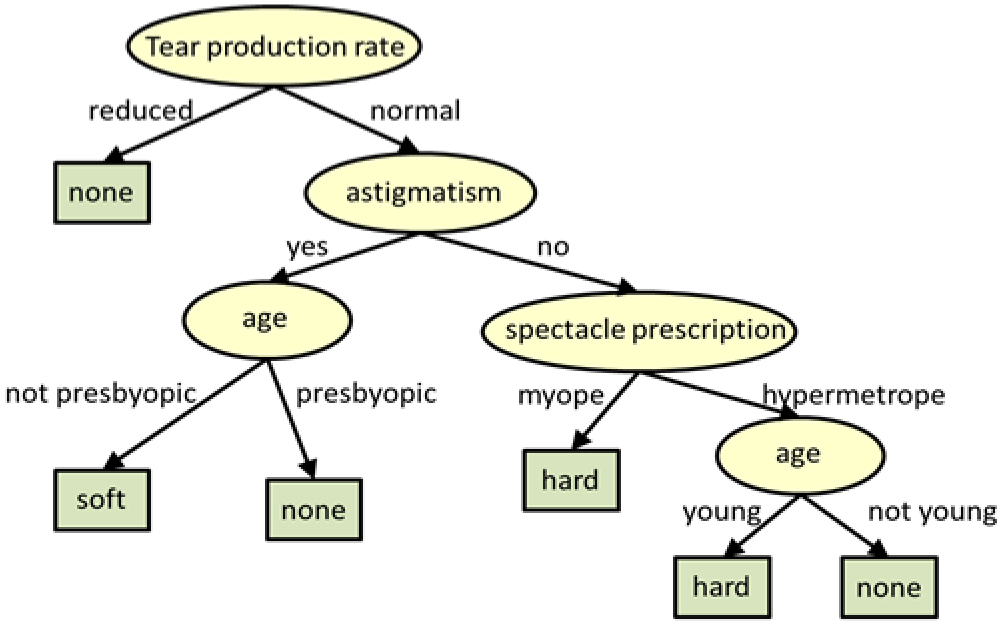
Explanation helps provide recourse to individuals who are adversely affected by model predictions

Motivating Example



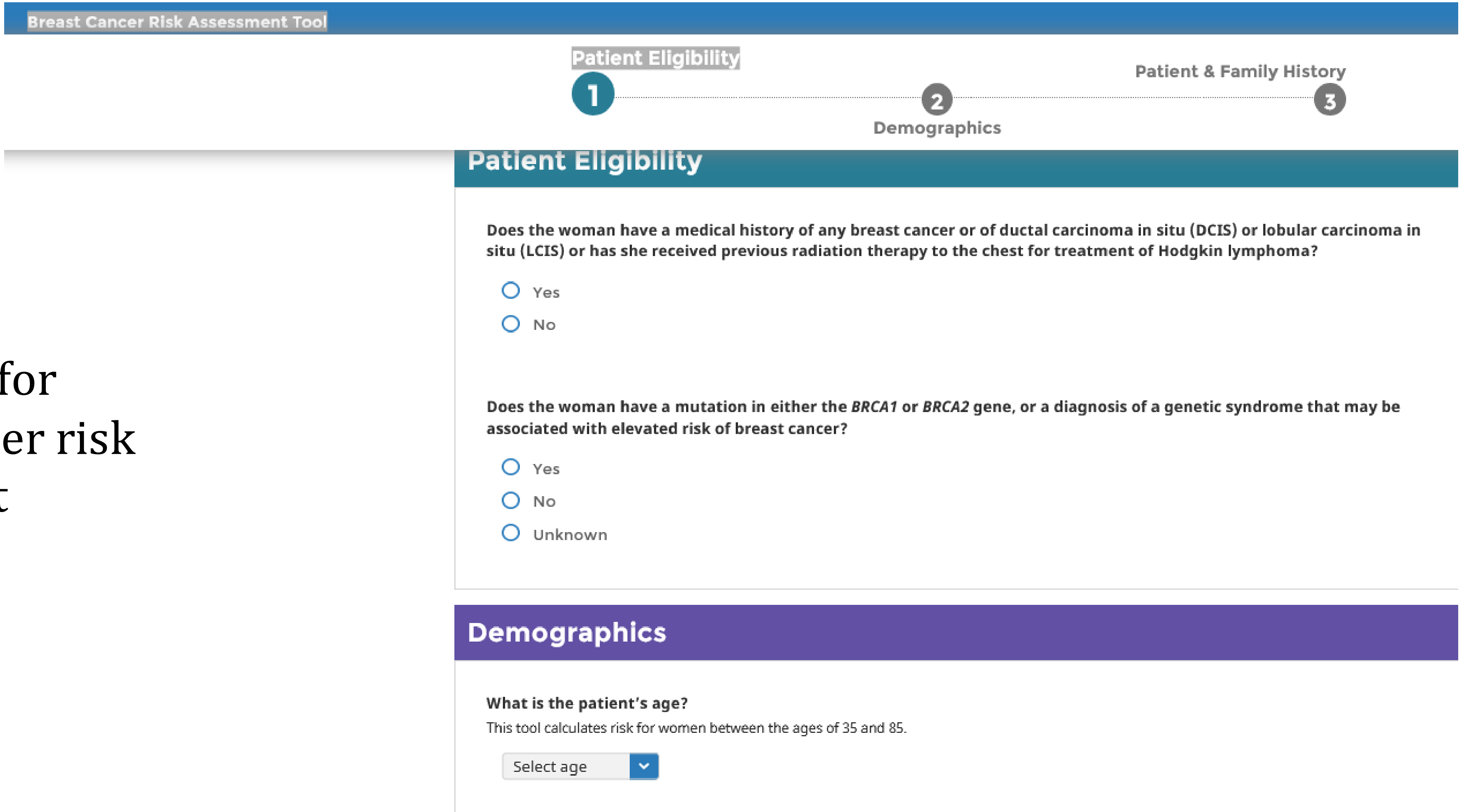
Achieving Explainability: Inherently Interpretable Models





if $(age = 18-20)$ **and** $(sex = male)$ **then predict** yes
else if $(age = 21-23)$ **and** $(priors = 2-3)$ **then predict** yes
else if $(priors > 3)$ **then predict** yes
else predict no

Interpretable Models are Trustworthy and Widely Deployed!



Breast Cancer Risk Assessment Tool

Patient Eligibility

1

2

Demographics

Patient & Family History

3

Patient Eligibility

Does the woman have a medical history of any breast cancer or of ductal carcinoma in situ (DCIS) or lobular carcinoma in situ (LCIS) or has she received previous radiation therapy to the chest for treatment of Hodgkin lymphoma?

- Yes
- No

Does the woman have a mutation in either the *BRCA1* or *BRCA2* gene, or a diagnosis of a genetic syndrome that may be associated with elevated risk of breast cancer?

- Yes
- No
- Unknown

Demographics

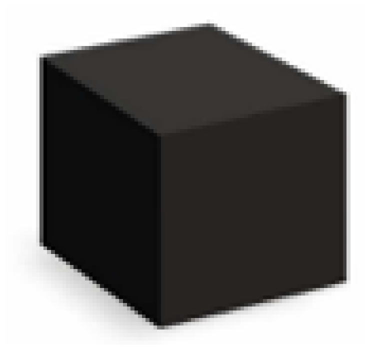
What is the patient's age?

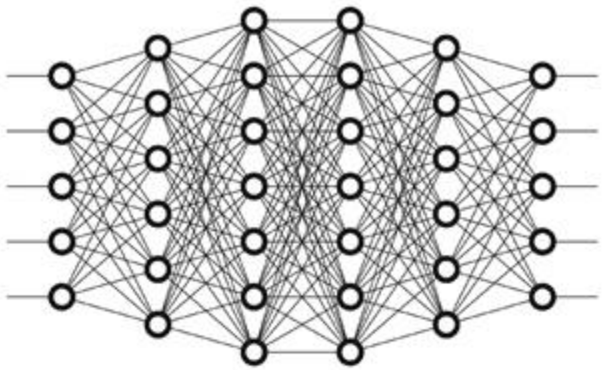
This tool calculates risk for women between the ages of 35 and 85.

Select age

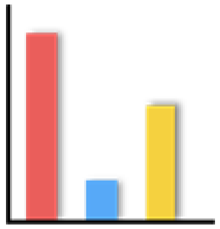
Gail model for breast cancer risk assessment

Achieving Explainability: Post-hoc Explanations

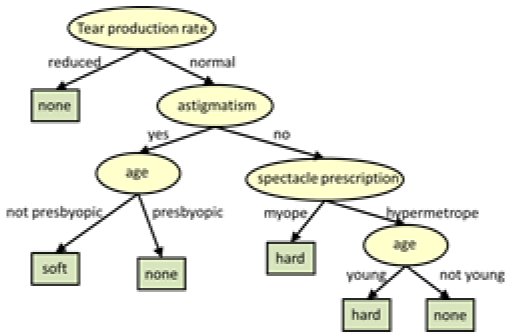




Explainer



if $(age = 18 - 20)$ **and** $(sex = male)$ **then predict** yes
else if $(age = 21 - 23)$ **and** $(priors = 2 - 3)$ **then predict** yes
else if $(priors > 3)$ **then predict** yes
else predict no



Interpretability vs Accuracy Trade-offs

If you *can build* an ***interpretable model*** which is also adequately accurate for your setting, DO IT!

Otherwise, ***post hoc explanations*** come to the rescue!

Interp
abil

Accuracy

What is an Explanation ?

Ideally, interpretable description of the model behavior

Classifier

Faithful

Explanation

Understandable

User

What is an Explanation ?

Ideally, interpretable description of the model behavior

Classifier

Send all the model parameters θ?

Send many example predictions?

Summarize with a program/rule/tree

Select most important features/points

Describe how to *flip* the model prediction

...

User

Local vs. Global Explanations

Local	Global
Explain individual predictions	Explain complete behavior of the model
Help unearth biases in the *local neighborhood* of a given instance	Help shed light on *big picture biases* affecting larger subgroups
Help vet if individual predictions are being made for the right reasons	Help vet if the model, at a high level, is suitable for deployment

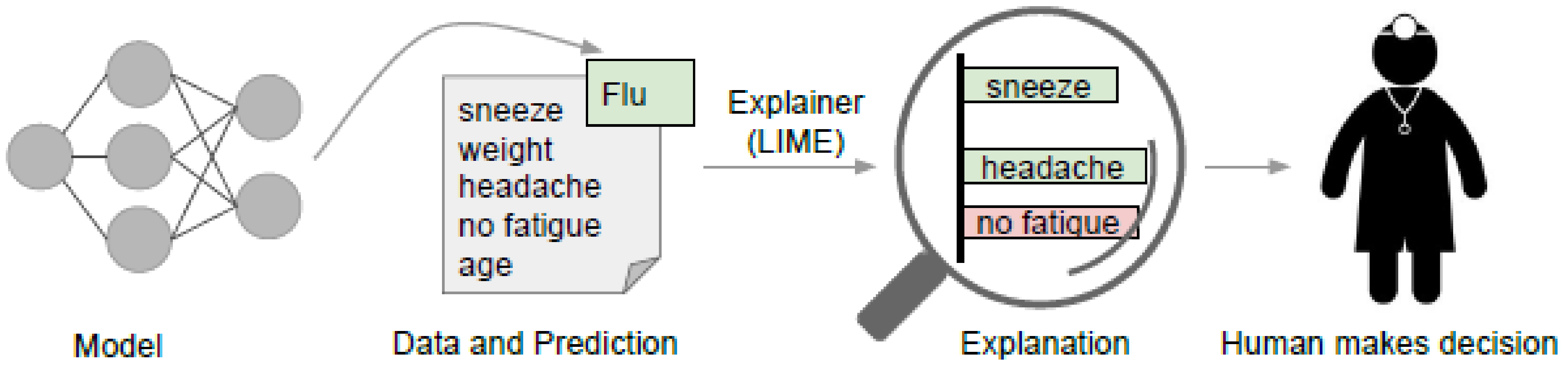
Feature Attribution Methods

- Goal: Explain why the model made a particular prediction on a specific input

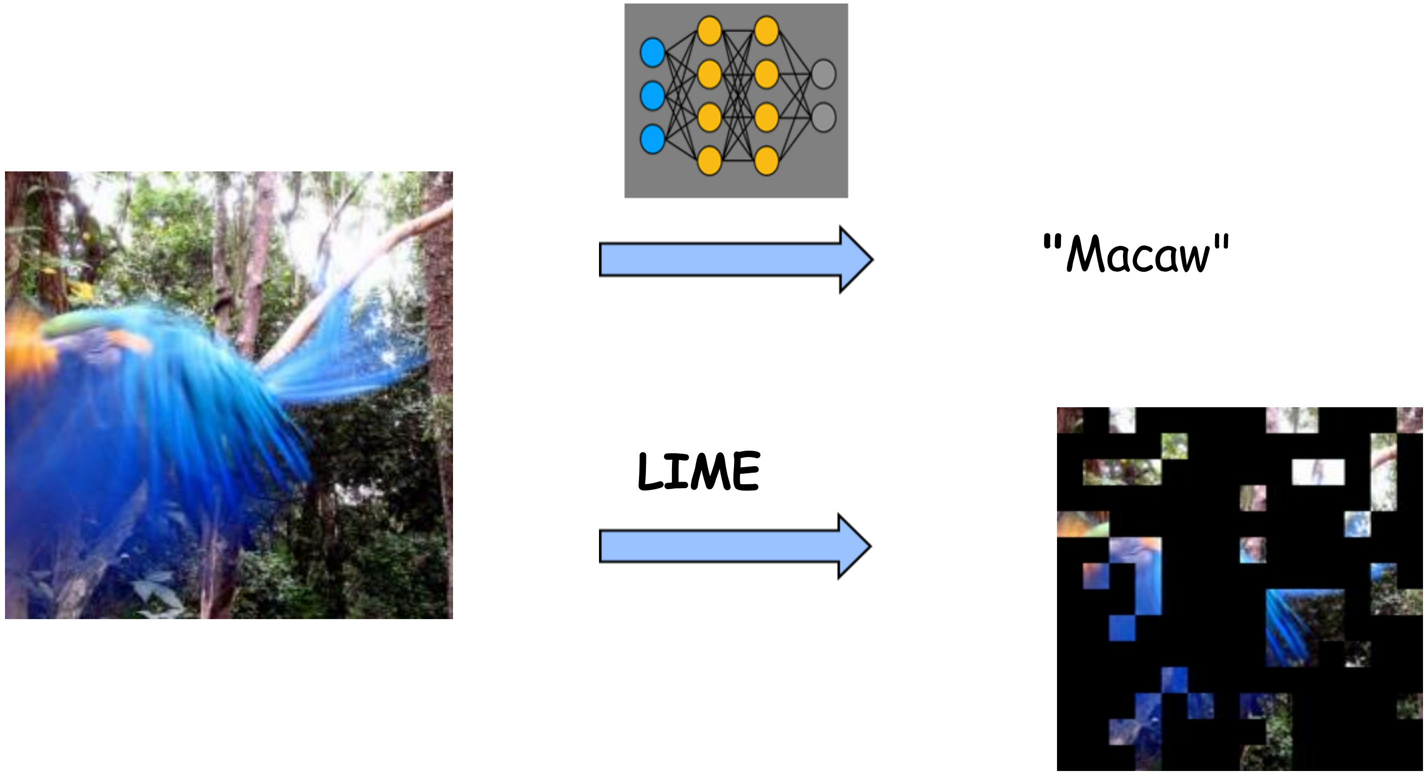
- Solution: Select/rank a subset of input features that contributed the most to model prediction

- Today:
  - How to formalize the problem
  - LIME algorithm: “Why should I trust you?” Explaining the predictions of any classifier (Ribeiro et al, KDD 2016)

LIME use-case illustrated



LIME Explanation for Image Classification



Classifier: Vision Transformer (Dosovitskiy, 2020)
Dataset: ImageNet

Formalizing Feature Attribution Problem

- Given an input x and model f, select a subset (of specified size) of features of x that contribute the most to the prediction f(x)

- First attempt: if $x \in \mathbb{R}^d$ output should be d-dim vector over {0,1} specifying for each feature whether it is selected or not x

- Problem: Features in representation of x may not be interpretable to humans

  e.g. an input image is a tensor with three color channels per pixel

Formalizing Feature Attribution Problem

- If input x is d-dimensional, first define a simpler d'-dimensional "interpretable" representation

- Output of explanation method g, for given input x and model f, is d'-dim vector over {0,1} that selects a subset of interpretable features (possibly with weights)

- Desired properties
    - Model agnostic: Method works for any model f
    - Interpretability: Minimize complexity of g (e.g. select at most 25% features)
    - Local fidelity: g approximates f well in the vicinity of x

  Note: to formalize local fidelity, need a way to map x and g to d-dim vectors

Interpretable Features: From Pixels to Superpixels

- Superpixel is a technique to segment an image into regions by considering similarity according to perceptual features

- Segmentation is dependent on specific input

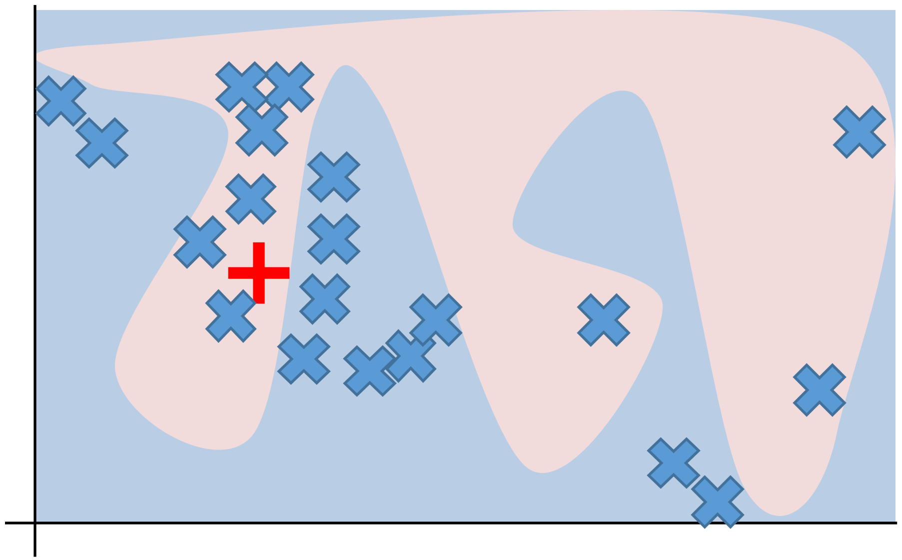
- Well-known algorithms based on graph partitioning and clustering

Interpretable Features: From Pixels to Superpixels

Original image

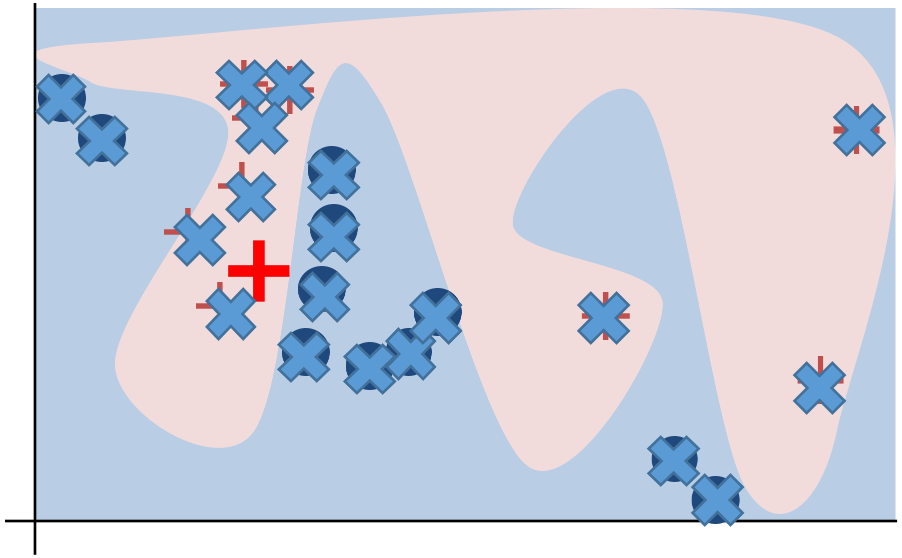
LIME: Local Interpretable Model-agnostic Explanations

1. Sample points around x



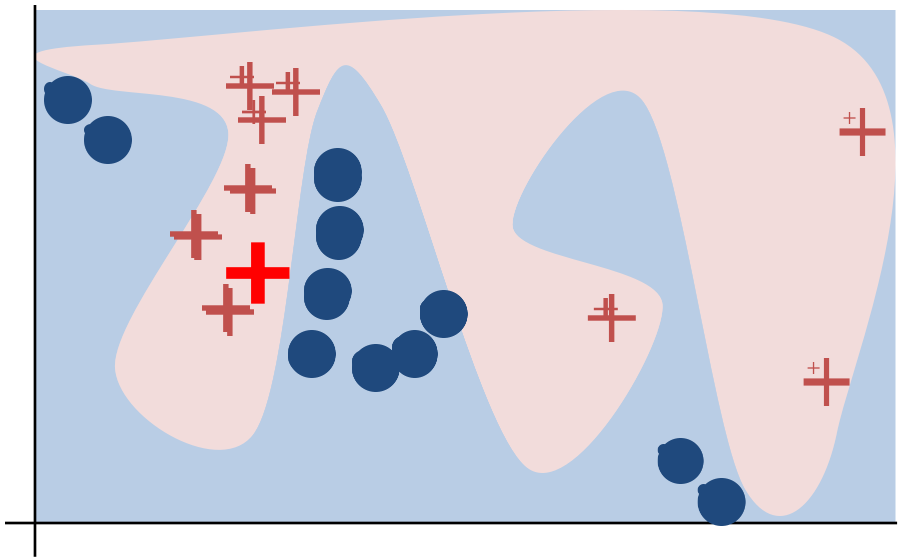
LIME Algorithm

1. Sample points around x
2. Use model to predict labels for each sample



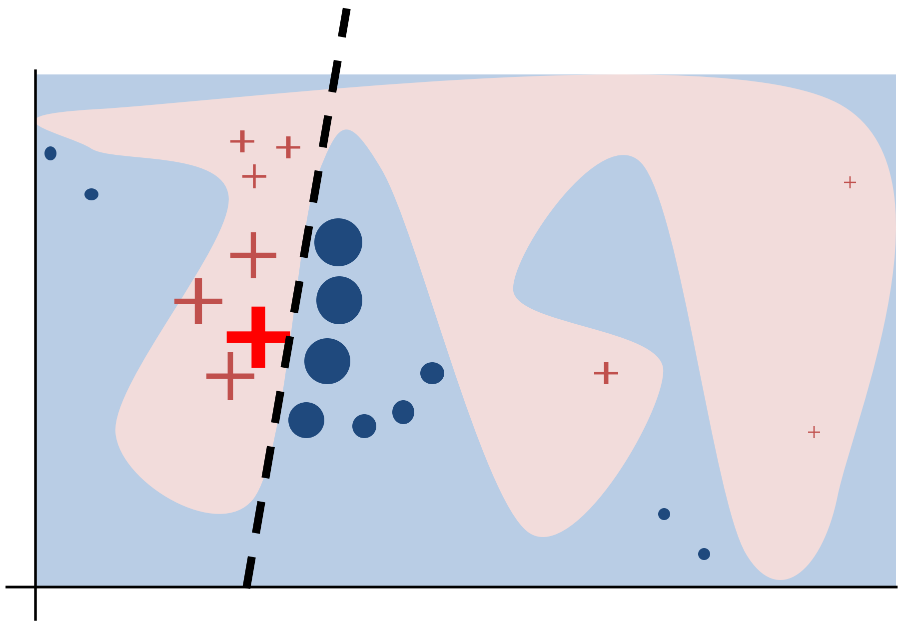
LIME Algorithm

1. Sample points around x
2. Use model to predict labels for each sample
3. Weigh samples according to distance to x



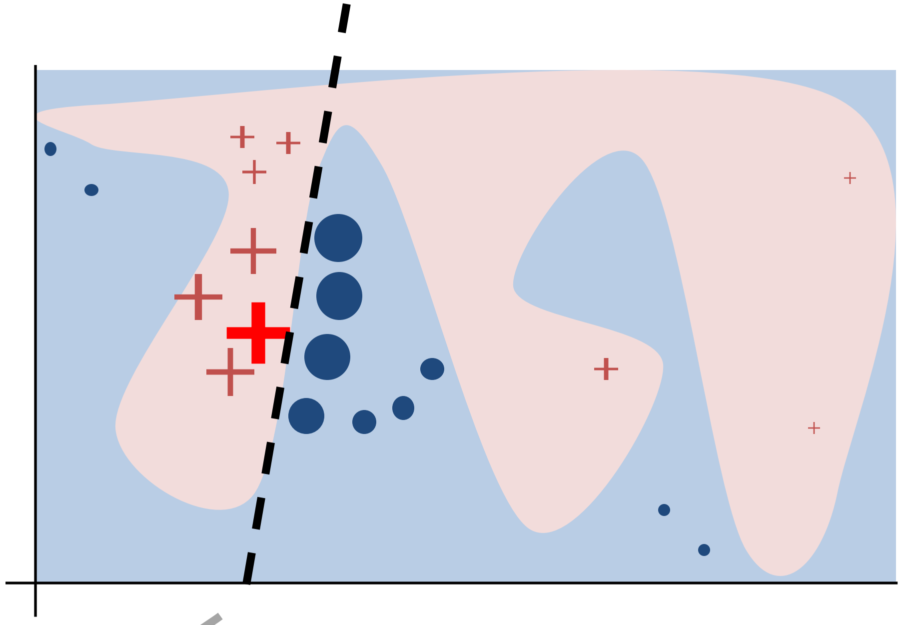
LIME Algorithm

1. Sample points around x
2. Use model to predict labels for each sample
3. Weigh samples according to distance to x
4. Learn simple linear model on weighted samples



LIME Algorithm

1. Sample points around x
2. Use model to predict labels for each sample
3. Weigh samples according to distance to x
4. Learn simple linear model on weighted samples
5. Use simple linear model to explain



LIME in more detail

- Consider a (black-box) classifier that labels an input sentence as good (label 1) or bad (label 0)
- Suppose it labels “You are a very nice person” as 1
- We want as an explanation which 3 words contributed the most to this prediction

Interpretable Data Representation

X

X'

Raw Input:

"You are a very nice person" (Label: 1)

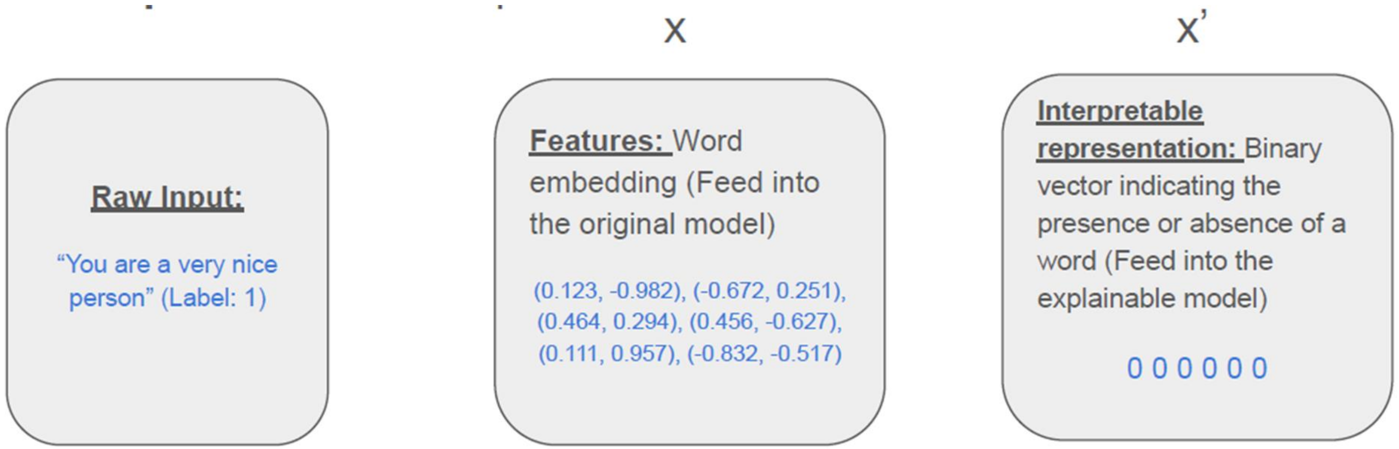
Features: Word embedding (Feed into the original model)

(0.123, -0.982), (-0.672, 0.251), (0.464, 0.294), (0.456, -0.627), (0.111, 0.957), (-0.832, -0.517)

Interpretable representation: Binary vector indicating the presence or absence of a word (Feed into the explainable model)

0 0 0 0 0 0

Sampling



- N: number of samples, say, 5
- K: length of explanation, say, 3
- Sample instances around x’ by drawing nonzero elements uniformly at random, say, ~U(2,4)

Sampling

Raw Input:

"You are a very nice person"
(Label: 1)

Interpretable representation:
Binary vector indicating the presence or absence of a word

0 0 0 0 0 0

Sampling →

Perturbed sample:
Interpretable binary vector indicating the presence or absence of a word (Feed into the explainable model)

z1': 0 1 1 0 0 0

z2': 0 0 1 1 1 0

z3': 1 0 0 1 0 0

z4': 1 1 1 1 0 0

z5': 0 1 0 1 1 0

- N: number of samples, 5
- K: length of explanation, 3
- Sample instances around x' uniformly at random ~ U(2,4)

Analyzing Samples

Raw Input:

"You are a very nice person"
(Label: 1)

Interpretable representation:
Binary vector indicating the presence or absence of a word

0 0 0 0 0 0

Sampling

Perturbed sample:
Interpretable binary vector indicating the presence or absence of a word (Feed into the explainable model)

z1': 0 1 1 0 0 0

z2': 0 0 1 1 1 0

z3': 1 0 0 1 0 0

z4': 1 1 1 1 0 0

z5': 0 1 0 1 1 0

- For each z', z is the corresponding input

  z'= 111100 maps to word vector of "You are a very"
- For each z, compute f(z)
- $\pi_x(z)$: Proximity measure between an instance z to x

Analyzing Samples

Raw Input:

"You are a very nice person"
(Label: 1)

Interpretable representation:
Binary vector indicating the presence or absence of a word

0 0 0 0 0 0

Sampling

Perturbed sample:
Interpretable binary vector indicating the presence or absence of a word (Feed into the explainable model)

z1': 0 1 1 0 0 0

z2': 0 0 1 1 1 0

z3': 1 0 0 1 0 0

z4': 1 1 1 1 0 0

z5': 0 1 0 1 1 0

Z ← {}

Z ← Z U (z1', f(z1), πx(z1)).

Z ← Z U (z2', f(z2), πx(z2)).

Z ← Z U (z3', f(z3), πx(z3)).

Z ← Z U (z4', f(z4), πx(z4)).

Z ← Z U (z5', f(z5), πx(z5)).

- For each z', z is the corresponding input
- For each z, compute f(z)
- $\pi_x(z)$: Proximity of z to x

Finding Explainable Model

Explainable model g:

Choose linear models here

Dataset:

Z (contains data & label & additional distance metric)

Objective function:

Local fidelity Interpretability

$$\min \quad \underbrace{\mathcal{L}(f, g, \pi_x)} + \underbrace{\Omega(g)}$$

Ω(g): A measure of complexity (as opposed to interpretability)

$$\mathcal{L}(f, g, \pi_x) = \sum_{z, z' \in \mathcal{Z}} \pi_x(z) \left(f(z) - g(z')\right)^2$$

Explanation:

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

(Corresponding weight for each feature)

Raw Input:

"You are a very nice person"
(Label: 1)

Perturbed sample:

Interpretable binary vector indicating the presence or absence of a word (Feed into the explainable model)

	Z ← {}
z1': 0 1 1 0 0 0	Z ← Z U (z1', f(z1), πx(z1)).
z2': 0 0 1 1 1 0	Z ← Z U (z2', f(z2), πx(z2)).
z3: 1 0 0 1 0 0	Z ← Z U (z3', f(z3), πx(z3)).
z4': 1 1 1 1 0 0	Z ← Z U (z4', f(z4), πx(z4)).
z5': 0 1 0 1 1 0	Z ← Z U (z5', f(z5), πx(z5)).

Finding Explainable Model

Explainable model g:
Choose linear models here
Dataset:
Z (contains data & label & additional distance metric)
Objective function:

$$\min \underbrace{\mathcal{L}(f, g, \pi_x)}_{\text{Local fidelity}} + \underbrace{\Omega(g)}_{\text{Interpretability}}$$

Ω(g): A measure of complexity (as opposed to interpretability)

$$\mathcal{L}(f, g, \pi_x) = \sum_{z, z' \in \mathcal{Z}} \pi_x(z) \left(f(z) - g(z')\right)^2$$

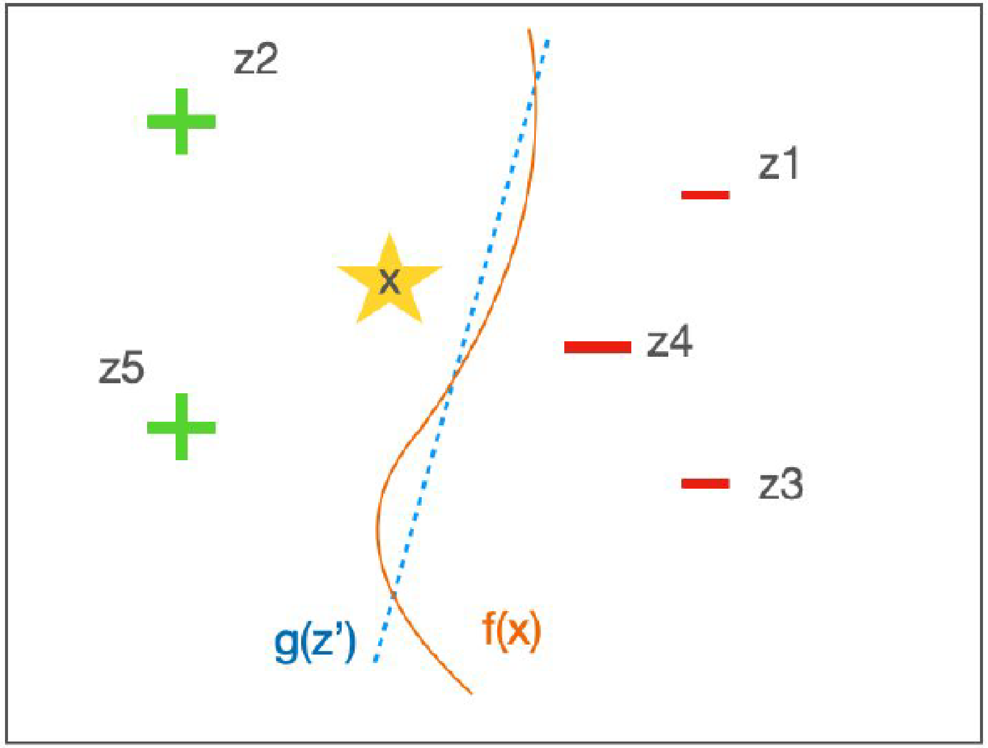
Explanation:

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

(Corresponding weight for each feature)

For the toy example:
Choose K-Lasso to limit # of explanations (K=3), i.e. we can only choose up to 3 words here for explanation

Explainable model g vs original model f



Explanation:

Nice 0.96
Very 0.56
You 0.43

LIME Algorithm

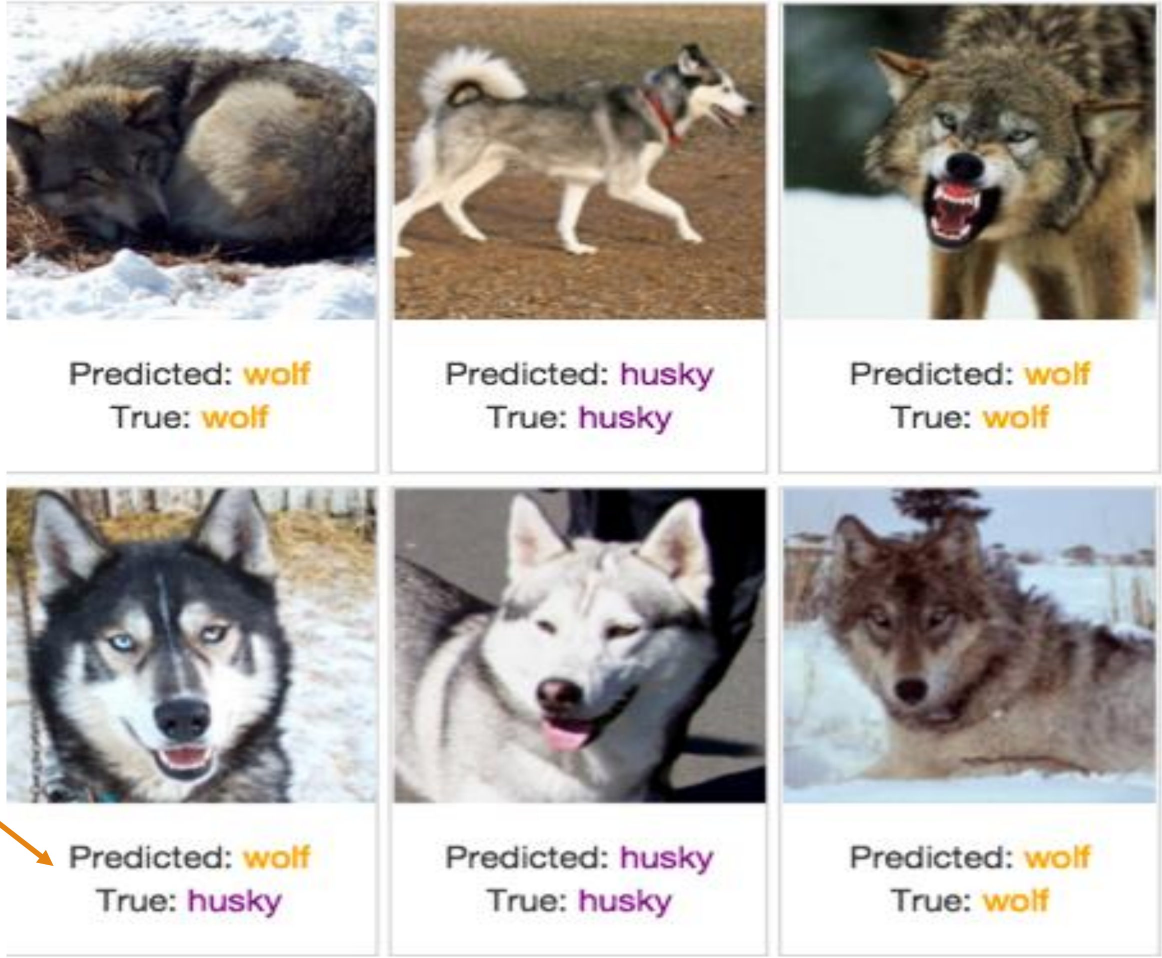
Algorithm 1 Sparse Linear Explanations using LIME

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

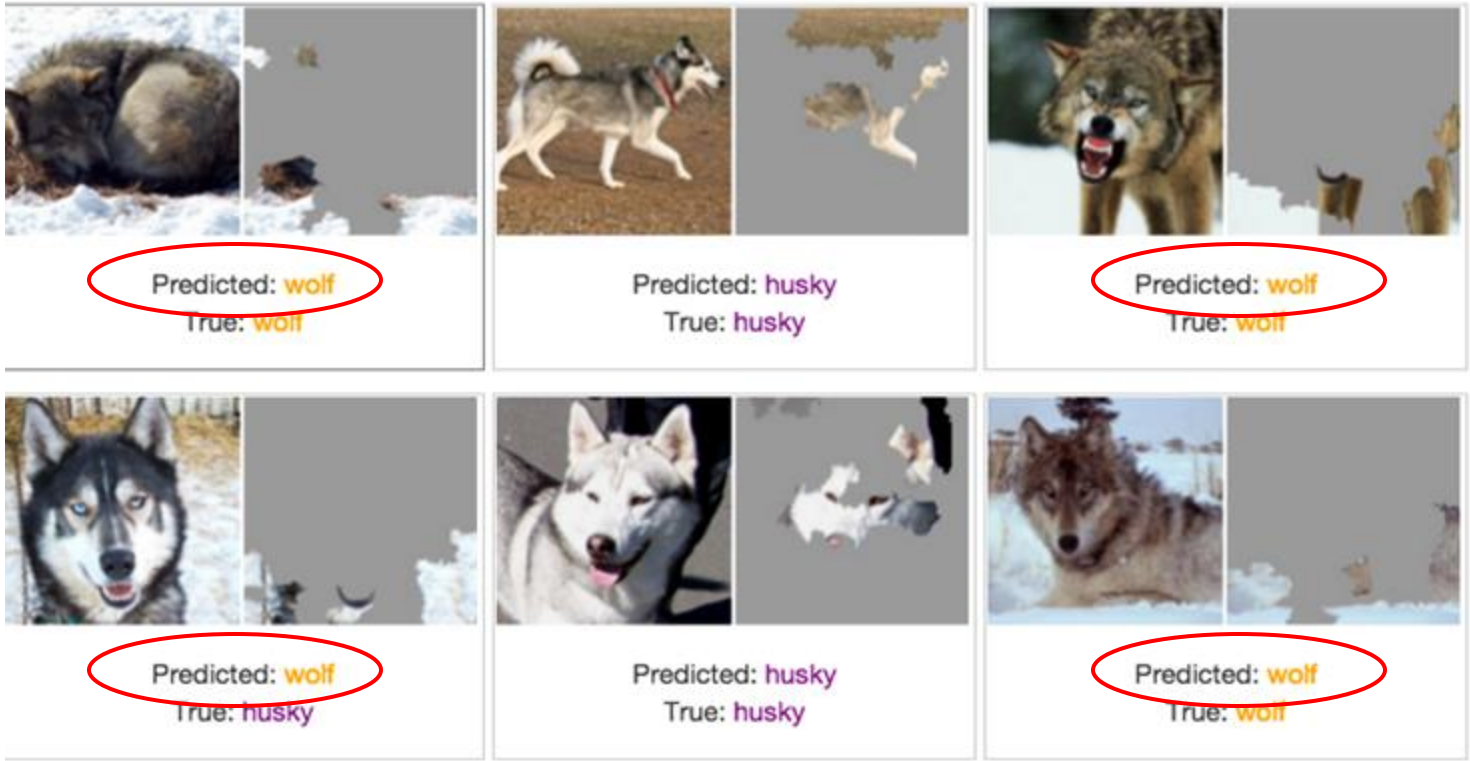
$\mathcal{Z} \leftarrow \{\}$
for $i \in \{1, 2, 3, ..., N\}$ **do**
  $z'_i \leftarrow sample\_around(x')$
  $\mathcal{Z} \leftarrow \mathcal{Z} \cup \langle z'_i, f(z_i), \pi_x(z_i) \rangle$
end for
$w \leftarrow$ K-Lasso$(\mathcal{Z}, K)$ ▷ with z'_i as features, $f(z)$ as target
return w

Image Classifier: Wolf vs Husky

Only 1 mistake!



Check Explanations with LIME



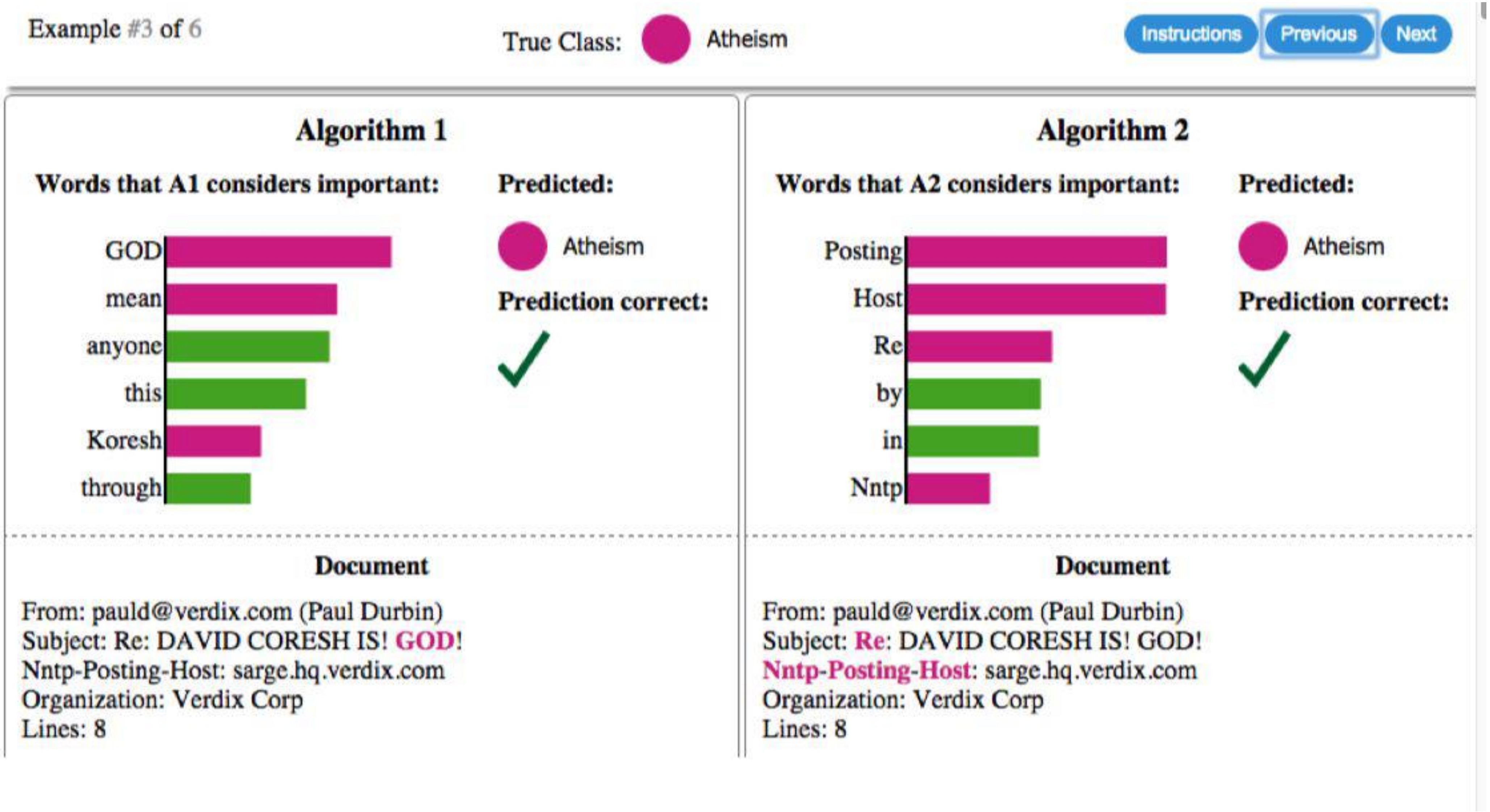
We've built a great snow detector...

Explanations with LIME

(a) Original Image (b) Explaining *Electric guitar* (c) Explaining *Acoustic guitar* (d) Explaining *Labrador*

Figure 4: Explaining an image classification prediction made by Google's Inception neural network. The top 3 classes predicted are "Electric Guitar" ($p = 0.32$), "Acoustic guitar" ($p = 0.24$) and "Labrador" ($p = 0.21$)

LIME Explanations can help choose between models



Agenda

- Today's recap:
  - Introduction
  - Feature attribution problem
  - LIME (Local Interpretable Model-agnostic Explanations) algorithm

- Coming up:
  - Next lecture: SHAP methods based on cooperative game theory
  - Review of other feature attribution methods (Saliency Maps)
  - Formal guarantees for feature attribution methods
  - Counterfactuals
  - Rule synthesis
  - Data attribution methods: Influence functions, Datamodels