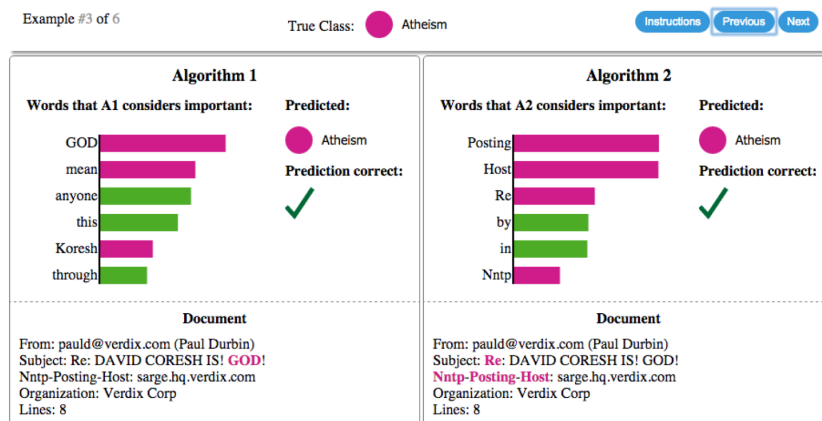


**What is “*interpretability*” in the context of machine learning?**

**What types of explanations are interpretable?**

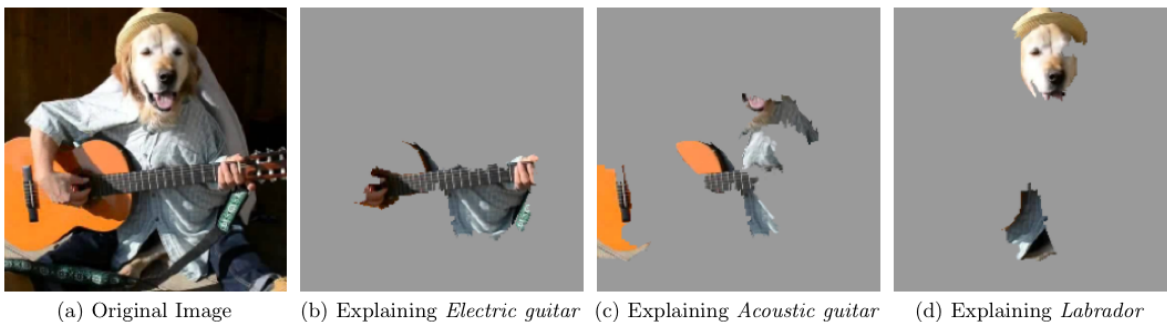
**Why is interpretability important/useful?**

## Example where interpretability can help find a better classifier



**Figure 2: Explaining individual predictions of competing classifiers trying to determine if a document is about “Christianity” or “Atheism”.** The bar chart represents the importance given to the most relevant words, also highlighted in the text. Color indicates which class the word contributes to (green for “Christianity”, magenta for “Atheism”).

**Does this explanation lead you to trust this model more even though it’s wrong?**



**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” ( $p = 0.21$ )

Do you trust this model?



What if you see the explanation of its prediction?

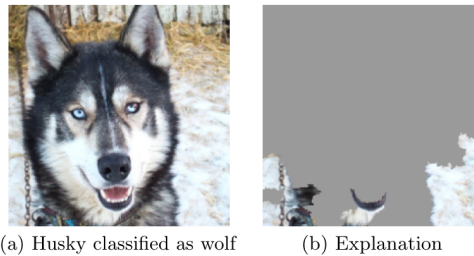


Figure 11: Raw data and explanation of a bad model’s prediction in the “Husky vs Wolf” task.

	Before	After
Trusted the bad model	10 out of 27	3 out of 27
Snow as a potential feature	12 out of 27	25 out of 27

Table 2: “Husky vs Wolf” experiment results.

How would you improve the classifier based on this explanation?

**Should you trade off accuracy for interpretability?**

## Types of interpretability:

Survey on Neural Network Interpretability, Zhang et al. 2020

Dimension 1 — Passive vs. Active Approaches	
{ Passive	Post-hoc explain trained neural networks
{ Active	Actively change the network architecture or training process for better interpretability





Dimension 2 — Type of Explanations (in the order of increasing explanatory power)	
To explain a prediction/class by	
↑ Examples	Provide example(s) which may be considered similar or as prototype(s)
↑ Attribution	Assign credit (or blame) to the input features (e.g. feature importance, saliency masks)
↑ Hidden semantics	Make sense of certain hidden neurons/layers
↓ Rules	Extract logic rules (e.g. decision trees, rule sets and other rule formats)

Dimension 3 — Local vs. Global Interpretability (in terms of the input space)	
↑ Local	Explain network's <i>predictions on individual samples</i> (e.g. a saliency mask for a input image)
↑ Semi-local	In between, for example, explain a group of similar inputs together
↓ Global	Explain the network <i>as a whole</i> (e.g. a set of rules/a decision tree)

## Local versus global interpretability

## Local methods with different explanation types:

<p style="text-align: center;"><b>Attribution as explanation</b></p>	<p>For <math>\mathbf{x}^{(i)}</math>:  → neural net → <math>\hat{y}^{(i)}</math>: junco bird</p> <p>The “contribution”<sup>1</sup> of each pixel:  [45]</p> <p>a.k.a. saliency map, which can be computed by different methods like gradients [40], sensitivity analysis<sup>2</sup> [41] etc.</p>
<p style="text-align: center;"><b>Explanation by showing examples</b></p>	<p>For <math>\mathbf{x}^{(i)}</math>:  → neural net → <math>\hat{y}^{(i)}</math>: fish</p> <p>By asking how much the network will change <math>\hat{y}^{(i)}</math> if removing a certain training image, we can find:</p> <p>most helpful<sup>2</sup> training images:  [43]</p>

### 3. LOCAL INTERPRETABLE MODEL-AGNOSTIC EXPLANATIONS

We now present Local Interpretable Model-agnostic Explanations (**LIME**). The overall goal of LIME is to identify an **interpretable** model over the *interpretable representation* that is **locally faithful** to the classifier.

$$\xi(x) = \underset{g \in G}{\operatorname{argmin}} \mathcal{L}(f, g, \pi_x) + \Omega(g) \quad (1)$$

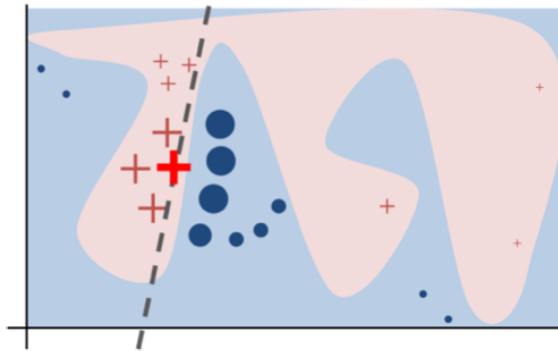


Figure 3: Toy example to present intuition for LIME. The black-box model's complex decision function  $f$  (unknown to LIME) is represented by the blue/pink background, which cannot be approximated well by a linear model. The bold red cross is the instance being explained. LIME samples instances, gets predictions using  $f$ , and weighs them by the proximity to the instance being explained (represented here by size). The dashed line is the learned explanation that is locally (but not globally) faithful.

Sparse

Explanations

---

**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  $\pi_x$ , Length of explanation  $K$

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

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

$z'_i \leftarrow \text{sample\_around}(x')$

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

**end for**

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

**return**  $w$

---

For text classification, we ensure that the explanation is **interpretable** by letting the *interpretable representation* be a bag of words, and by setting a limit  $K$  on the number of words, i.e.  $\Omega(g) = \infty \mathbb{1}[\|w_g\|_0 > K]$ . Potentially,  $K$  can be adapted to be as big as the user can handle, or we could have different values of  $K$  for different instances. In this paper we use a constant value for  $K$ , leaving the exploration of different values to future work. We use the same  $\Omega$  for image classification, using “super-pixels” (computed using any standard algorithm) instead of words, such that the interpretable representation of an image is a binary vector where 1 indicates the original super-pixel and 0 indicates a grayed out super-pixel. This particular choice of  $\Omega$  makes directly solving Eq. (1) intractable, but we approximate it by first selecting  $K$  features with Lasso (using the regularization path [9]) and then learning the weights via least squares (a procedure we call K-LASSO in Algorithm 1). Since Algorithm 1 produces an explanation for an individual prediction, its complexity does not depend on the size of the dataset, but instead on time to compute  $f(x)$  and on the number of samples  $N$ . In practice, explaining random forests with 1000 trees using scikit-learn (<http://scikit-learn.org>) on a laptop with  $N = 5000$  takes under 3 seconds without any optimizations such as using gpus or parallelization. Explaining each prediction of the Inception network [25] for image classification takes around 10 minutes.



**What is a regularization path? How is it used here in K-Lasso?**

**Superpixels**

Explain the  
hted line



highlig

Any choice of interpretable representations and  $G$  will have some inherent drawbacks. First, while the underlying model can be treated as a black-box, certain interpretable representations will not be powerful enough to explain certain behaviors. For example, a model that predicts sepia-toned images to be *retro* cannot be explained by presence of absence of super pixels. Second, our choice of  $G$  (sparse linear models) means that if the underlying model is highly non-linear even in the locality of the prediction, there may not be a faithful explanation. However, we can estimate the faithfulness of

