CAI 4104/6108 – Machine Learning Engineering:

Explainable/Interpretable ML

Prof. Vincent Bindschaedler

Spring 2024

Administrivia

- Homework grading
  - Let us know if you have concerns (as soon as possible)
    - Be proactive. Don’t wait until the end of the semester
  - We can regrade to make sure you **do not** lose many points for a small mistake
    - Show us that submission runs fine otherwise
  - Contact the grader first, then me

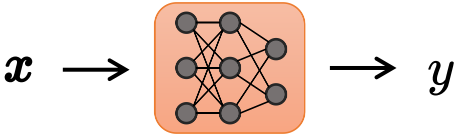
Administrivia

- Final Exam
  - When: May 2, 2024 — 7:30AM to 9:30AM
  - Where: Online (Canvas + **Honorlock**)
  - Note:
    - The CAI4104 and CAI6108 exams will be (slightly) different
  - Format:
    - Some Short answer questions (may include multiple choice)
    - Some multi-part problems

Administrivia

- Midterm Grade Adjustment
  - We have adjusted grades: `adj_score = 100 * (orig_score / new_max)`
    - For CAI4104: `new_max = 96`
    - For CAI6108: `new_max = 94`

Interpretable ML



Model

- Most ML models are black-box in terms of their decisions
  - You feed an input x, you get an output y
  - Why did we get output y (and not some y'≠y)?

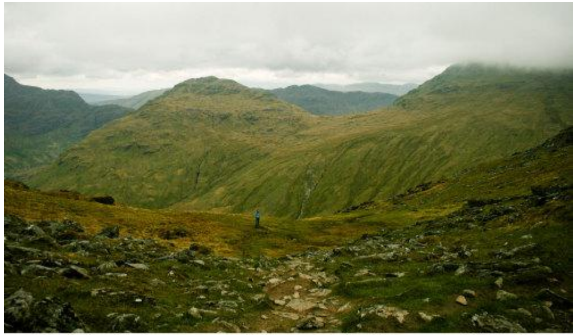
- We want human-understandable explanations
  - How?

Interpretable ML: Motivation

- Why do we need explanations?
  - For technology to be accepted and used, users must **trust** it (to some extent)
    - How much does the human understand the model's behavior?
  - Note: sometimes the stakes are high
    - E.g., some models are used for decision making (e.g., medical diagnosis, terrorism)
- Applications of interpretable ML
  - Validating models
    - E.g., fairness
  - Debugging models
    - E.g., adversarial examples
  - Human learning

Interpretable ML: Applications

- Models can be correct for the wrong reasons!
  - Example from Mudrakarta et al. "*Did the model understand the question?*." ACL, 2018.
  - Question to Neural Programmer: “which nation earned the most gold medals?”
    - Answer is the first row of the table (correct)
    - But the table is sorted by rank! (correct answer appears first!)
- High accuracy does not (always) imply reasonable predictions!
  - Lehman et al. "The surprising creativity of digital evolution: A collection of anecdotes from the evolutionary computation and artificial life research communities." arXiv, 2018.



A herd of sheep grazing on a lush green hillside
Tags: grazing, sheep, mountain, cattle, horse

source: aiweirdness.com

Interpretable ML: Terminology

- Explainability and interpretability are often used interchangeably

- LIME: Ribeiro et al. KDD, 2016
  - *Explaining* a prediction means presenting textual or visual **evidence** that gives a **qualitative understanding** of the relationship between the model's input and prediction
  - *Interpretability* means making the explanation **understandable by humans**

- Gilpin et al. 2018
  - *Explainability*: summarize reasons for behavior and provide insights about causes of decisions
  - *Interpretability*: describe internals of models in a way **understandable to humans**

Interpretable ML: Taxonomy of Techniques

- Techniques for explaining processing
  - Explain how the model processed the data
  - “How did the model produce this output from this input?”
    - E.g.: proxy models, saliency mapping, etc.

- Techniques for explaining representations
  - Explain how the model represents information that influences the decision
  - “What information is represented by the model?”
    - E.g.: studying layers/neurons of a neural network

- Explanation-producing systems
  - We can design and use models themselves that produce simple interpretations of their behavior
    - E.g.: attention networks

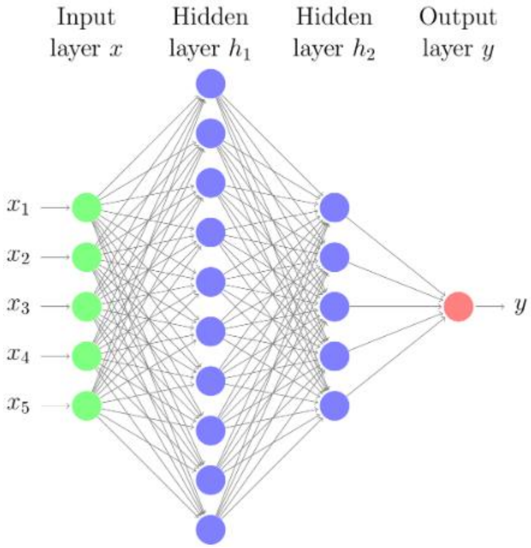
Interpretable ML: Explaining Processing

- Proxy models techniques
  - Idea: create (train) a *proxy model* that behaves similarly but is easier to explain
    - Linear proxy models (e.g., LIME)
    - Decision trees
    - Rule extraction (e.g., if-then rules, MofN)

- Saliency mapping
  - Identify and highlight the salient features
    - Shows a small portion that is the most relevant
  - Typically applied to the image domain, but can be used more generally

Interpretable ML: DeepRED

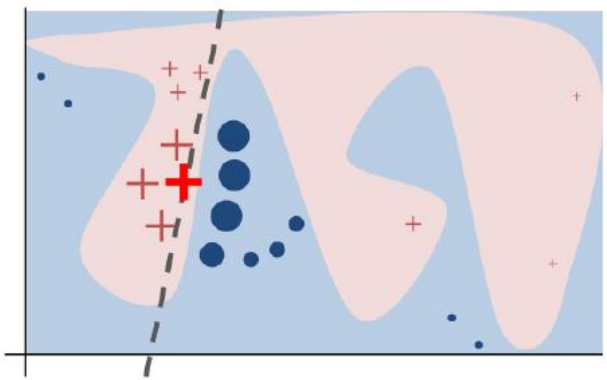
- Zilke, Jan Ruben, Eneldo Loza Mencía, and Frederik Janssen. "*DeepRED–Rule extraction from deep neural networks*." International Conference on Discovery Science, 2016.



```
IF X1<0.5 AND X2>0.75 THEN OUT=1
IF X1>0.9 THEN OUT=1
IF X1>0.5 AND X1<0.9 AND X3>0.2 THEN OUT=1
IF X2>0.2 AND X3<0.5 AND X5<0.5 THEN OUT=1
IF X2>0.4 AND X3<0.7 THEN OUT=1
IF X2<0.2 THEN OUT=1
IF X4>0.8 THEN OUT=1
IF X3<0.7 AND X3>0.2 AND X4<0.3 THEN OUT=1
```

Interpretable ML: LIME

- Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "*Why should I trust you?: Explaining the predictions of any classifier*." KDD, 2016
  - Given an instance x
  - Idea: approximate the behavior of the model in a neighborhood of x using a **proxy model**
  - Choice for the proxy model:
    - Linear model, decision tree, falling rule list



- Decision function: blue-pink background
- Instance being explained: bold red cross
- Learned explanation: dashed line

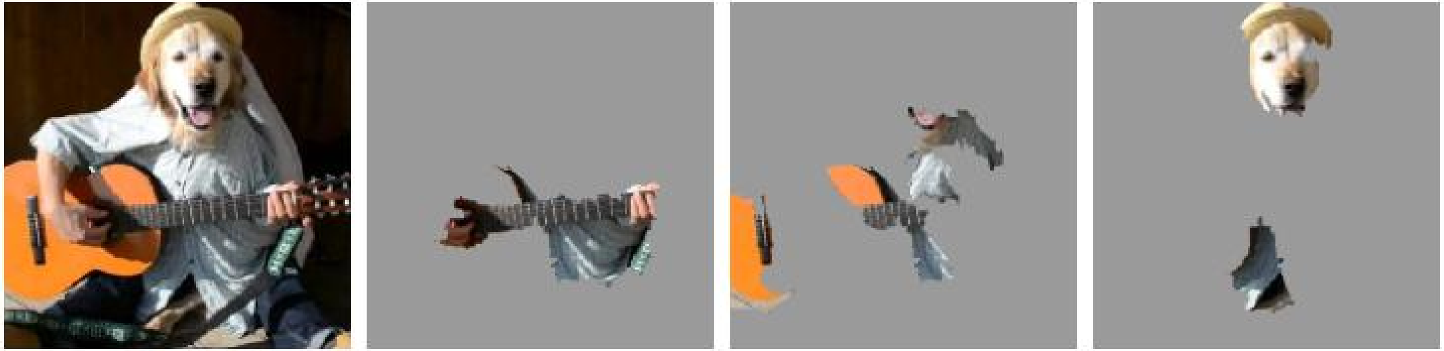
Interpretable ML: LIME

- Setup
  - f is the classifier
  - $\boldsymbol{x}$ is the input we want to explain ($f(\boldsymbol{x})=y$ for some y)
  - G is a class of interpretable models (e.g., linear models, decision trees, etc.)
    - $g \in G$ is an **explanation**
  - $\Omega(g)$ is a measure of complexity of explanation (e.g., depth of tree if g is a decision tree)
  - $\pi_{\boldsymbol{x}}(\boldsymbol{z})$ is a proximity measure of input $\boldsymbol{z}$ to $\boldsymbol{x}$
  - $L(f, g, \pi_x)$ is a measure of how unfaithful g is in approximating f in a neighborhood given by $\pi_{\boldsymbol{x}}$

- Explanation:
  - $g^* = \mathrm{argmin}_{g \in G} L(f, g, \pi_{\boldsymbol{x}}) + \Omega(g)$
  - How do we solve it? This is an optimization problem.
    - Use sampling (local exploration of points $\boldsymbol{z}$ in a neighborhood of $\boldsymbol{x}$, weighted by $\pi_{\boldsymbol{x}}$)

Interpretable ML: LIME Example

- Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "*Why should I trust you?: Explaining the predictions of any classifier*." KDD, 2016
  - Google's pre-trained Inception neural network



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

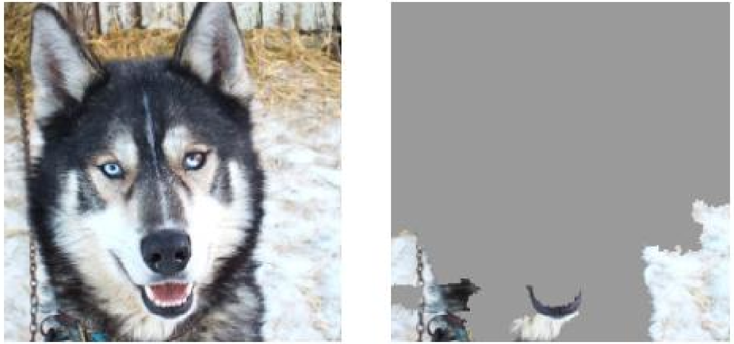
Figure 4: Explaining an image classification prediction made by Google's Inception neural network. The top 3 classes predicted are "Electric Guitar" ($p = 0.32$), "Acoustic guitar" ($p = 0.24$) and "Labrador" ($p = 0.21$)

Super-pixels explanations (with K = 10)

- Note: top predicted class is wrong

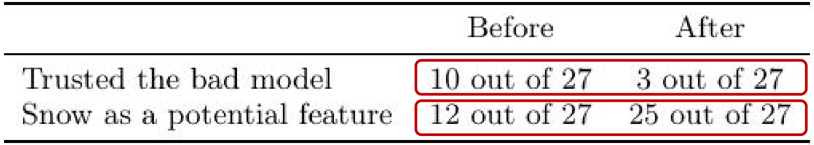
Interpretable ML: LIME Example

- Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "*Why should I trust you?: Explaining the predictions of any classifier*." KDD, 2016
  - Can explanations provide insights? (experiment with graduate students.)
    - Experiment: train a bad model to distinguish “husky” from “wolf”, then ask questions to subjects before and then again after they are show the explanation



(a) Husky classified as wolf (b) Explanation

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

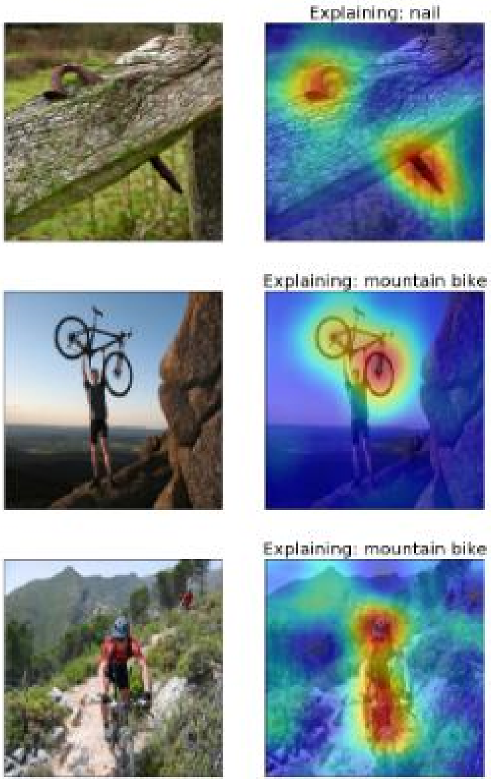
Table 2: “Husky vs Wolf” experiment results.

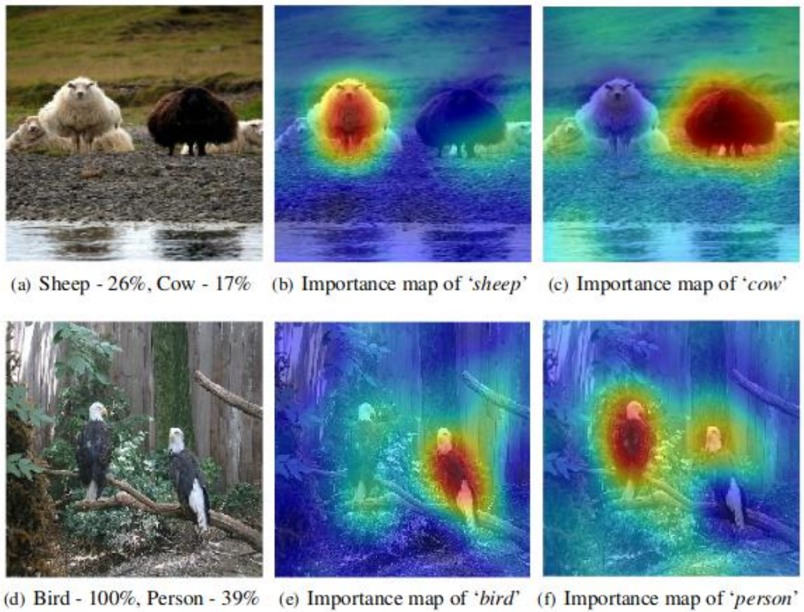
Interpretable ML: Saliency Maps

- Instead of training a proxy model, can we directly highlight the salient features?
  - Similar to what LIME is doing with super-pixels
  - We want to see the small portion of the input that is the most relevant
    - Note: sometimes called “attribution”

- How?
  - Simple solution:
    - Repeatedly occlude portions of the input
    - Create a map showing which part of the input actually influences the output
  - More efficient (& better):
    - Compute the saliency map directly by computing the input gradient (or other related quantities)
    - Example of techniques: CAM, Grad-CAM, Integrated gradients, etc.

Interpretable ML: Saliency Maps




(a) Sheep - 26%, Cow - 17% (b) Importance map of '*sheep*' (c) Importance map of '*cow*'

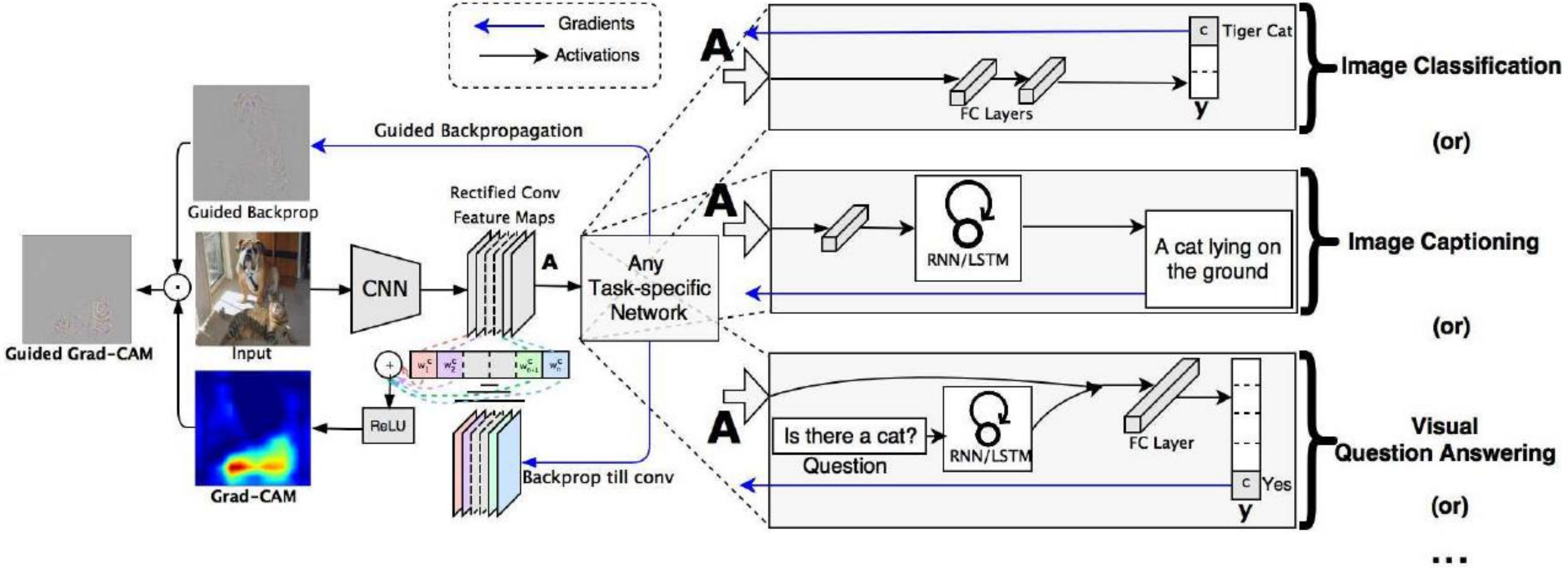
(d) Bird - 100%, Person - 39% (e) Importance map of '*bird*' (f) Importance map of '*person*'

Figure 1: Our proposed RISE approach can explain why a black-box model (here, ResNet50) makes classification decisions by generating a pixel importance map for each decision (redder is more important). For the top image, it reveals that the model only recognizes the white sheep and confuses the black one with a cow; for the bottom image it confuses parts of birds with a person. (Images taken from the PASCAL VOC dataset.)

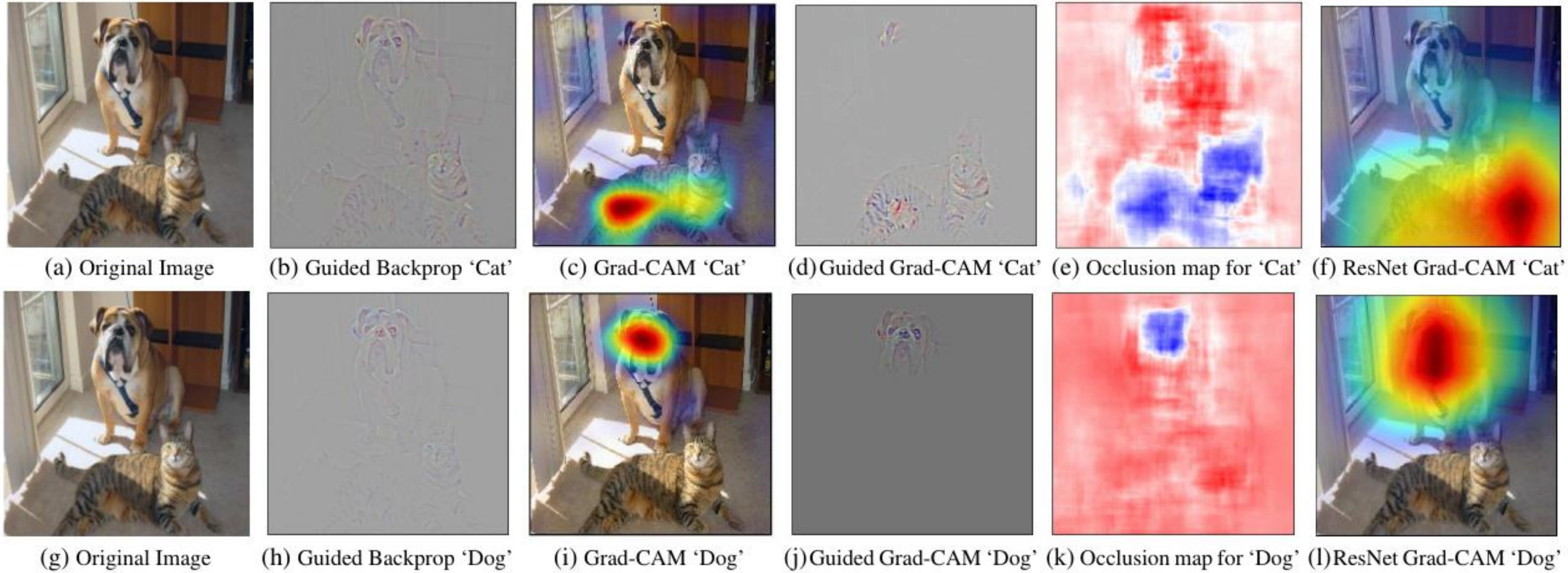
Source: Petsiuk, Das, and Saenko. "RISE: Randomized input sampling for explanation of black-box models." BMVC, 2018.

Interpretable ML: Grad-CAM

- Selvaraju, Ramprasaath R., Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. "*Grad-CAM: Visual explanations from deep networks via gradient-based localization*." ICCV, 2017.
  - Gradient-weighted Class Activation Mapping (Grad-CAM)



Interpretable ML: Grad-CAM

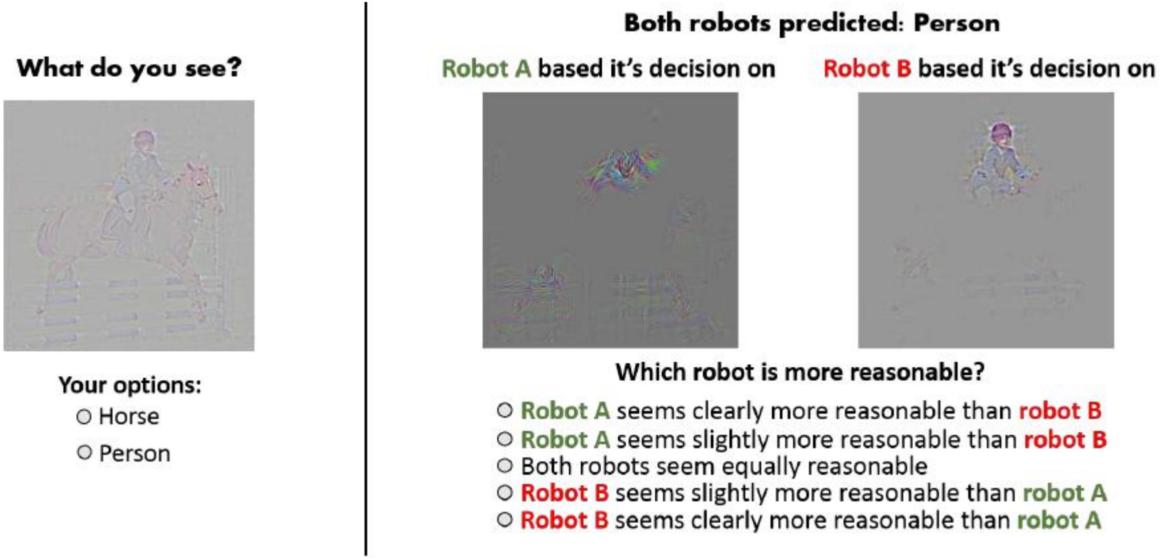


(a) Original Image (b) Guided Backprop ‘Cat’ (c) Grad-CAM ‘Cat’ (d) Guided Grad-CAM ‘Cat’ (e) Occlusion map for ‘Cat’ (f) ResNet Grad-CAM ‘Cat’

(g) Original Image (h) Guided Backprop ‘Dog’ (i) Grad-CAM ‘Dog’ (j) Guided Grad-CAM ‘Dog’ (k) Occlusion map for ‘Dog’ (l) ResNet Grad-CAM ‘Dog’

Figure 1: (a) Original image with a cat and a dog. (b-f) Support for the cat category according to various visualizations for VGG-16 and ResNet. (b) Guided Backpropagation [42]: highlights all contributing features. (c, f) Grad-CAM (Ours): localizes class-discriminative regions, (d) Combining (b) and (c) gives Guided Grad-CAM, which gives high-resolution class-discriminative visualizations. Interestingly, the localizations achieved by our Grad-CAM technique, (c) are very similar to results from occlusion sensitivity (e), while being orders of magnitude cheaper to compute. (f, l) are Grad-CAM visualizations for ResNet-18 layer. Note that in (c, f, i, l), red regions corresponds to high score for class, while in (e, k), blue corresponds to evidence for the class. Figure best viewed in color.

Interpretable ML: Grad-CAM



What do you see?

Your options:

- ○ Horse
- ○ Person

Both robots predicted: Person

Robot A based it's decision on

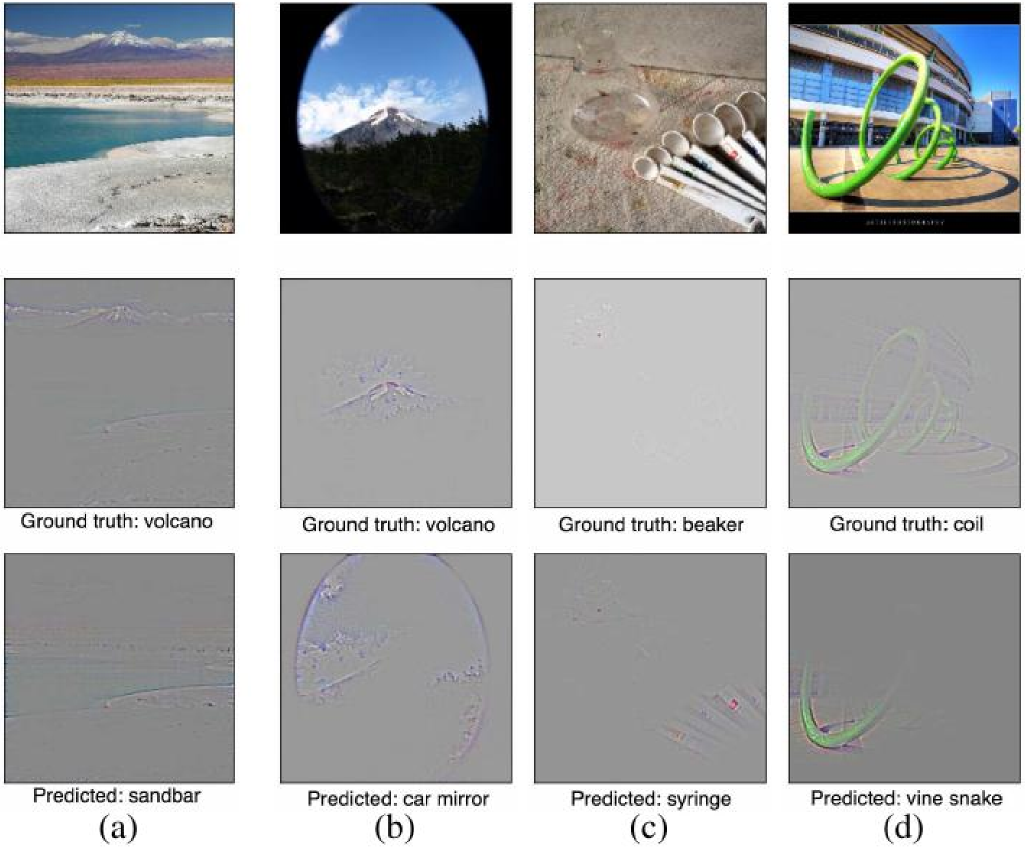
Robot B based it's decision on

Which robot is more reasonable?

- ○ Robot A seems clearly more reasonable than robot B
- ○ Robot A seems slightly more reasonable than robot B
- ○ Both robots seem equally reasonable
- ○ Robot B seems slightly more reasonable than robot A
- ○ Robot B seems clearly more reasonable than robot A

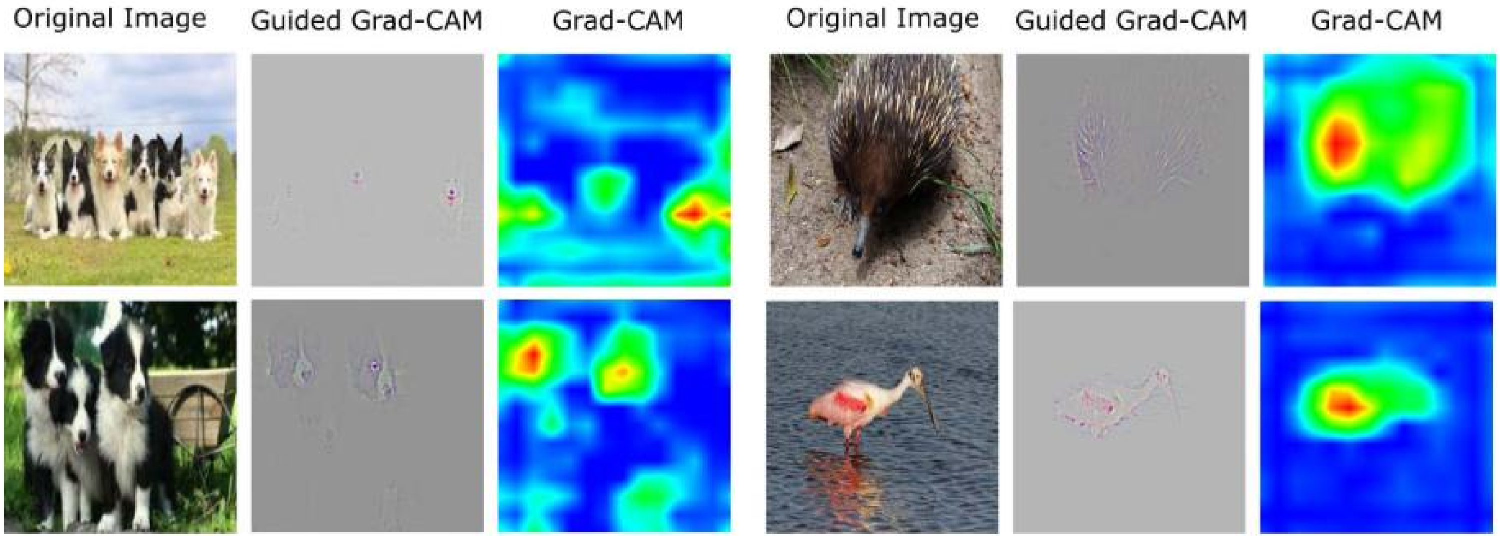
Figure 3: AMT interfaces for evaluating different visualizations for class discrimination (left) and trustworthiness (right). Guided Grad-CAM outperforms baseline approaches (Guided-backprop and Deconvolution) showing that our visualizations are more class-discriminative and help humans place trust in a more accurate classifier.

Interpretable ML: Grad-CAM

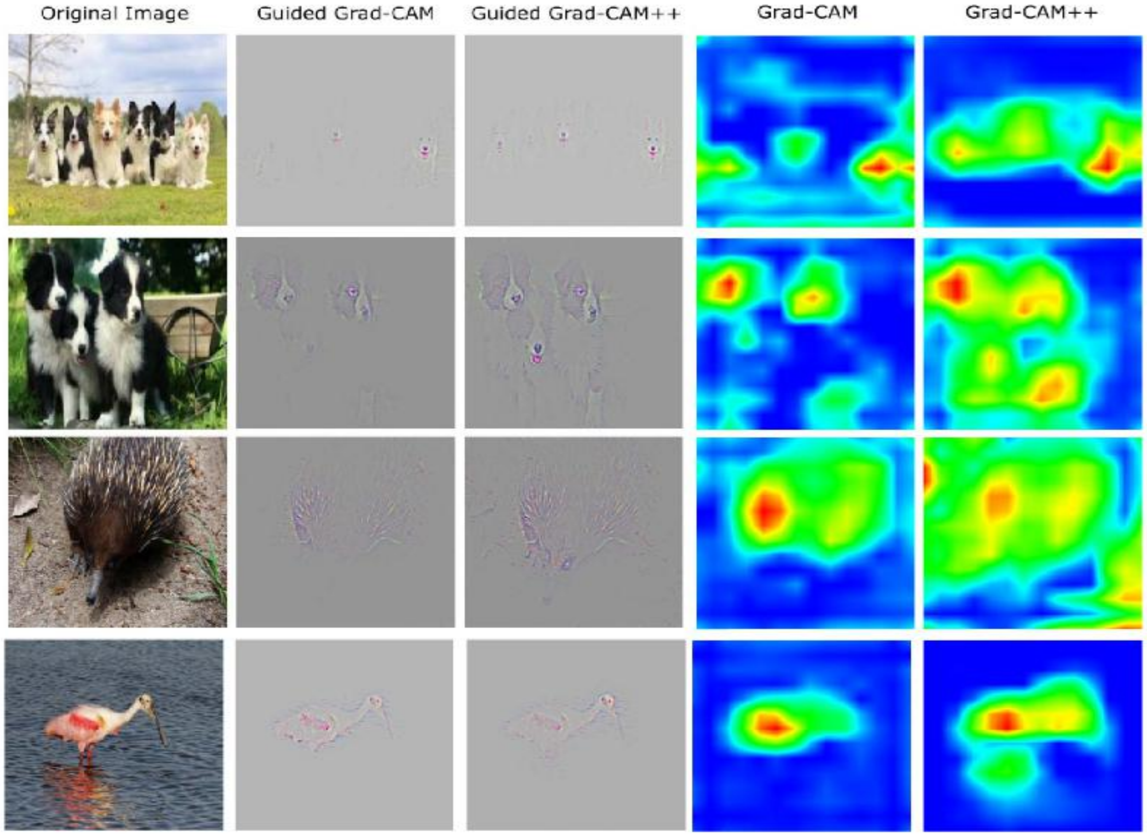


Interpretable ML: Grad-CAM++

- Chattopadhay et al. "Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks." In WACV, 2018.
  - Tries to improve on Grad-CAM



Interpretable ML: Grad-CAM++



Next Time

- Friday (4/19): Lecture
  - Topic: Fairness

- Upcoming:
  - Project due 4/24