Thirty-Fifth AAAI Conference on Artificial Intelligence

From Explainability to Model Quality and Back Again

Anupam Datta, Matt Fredrikson, Klas Leino, Kaiji Lu, Shayak Sen and Zifan Wang

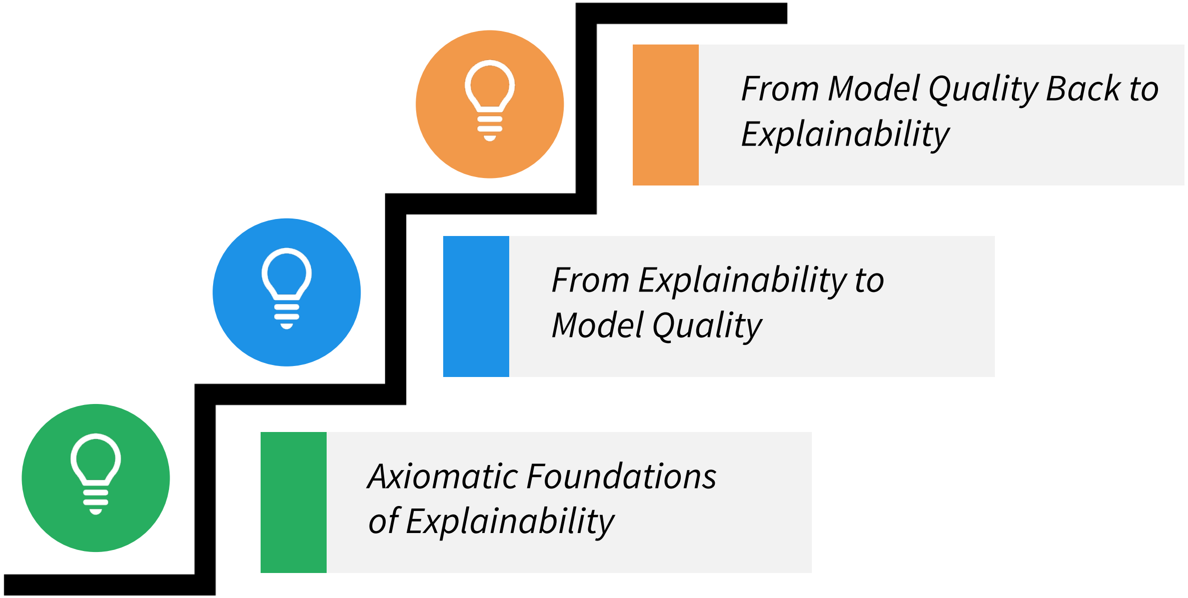
Anupam Datta

Matt Fredrikson

Klas Leino

Shayak Sen

From Explainability to Model Quality and Back Again



Machine Learning Systems are Ubiquitous

Google

April 3, 2013, Vol 309, No. 13 >

< Previous Article   Next Article >

Viewpoint | April 3, 2013

The Inevitable Application of Big Data to Health Care

Travis B. Murdoch, MD, MSc; Allan S. Detsky, MD, PhD

[+] Author Affiliations

Big Data in Government, Defense and Homeland Security 2015 - 2020

Share +1 in Pin it

NEW YORK, May 12, 2015 /PRNewsw

How Big Data Could Replace Your Credit Score

Credit scores are useful in determining who gets loans, but they're far from perfect. AvantCredit determines loan-worthiness based on all sorts of factors, including your use of social media and prepaid cell phones.

Big Data in Education

Learn how and when to use key methods for educational data mining and learning analytics on large-scale educational data.

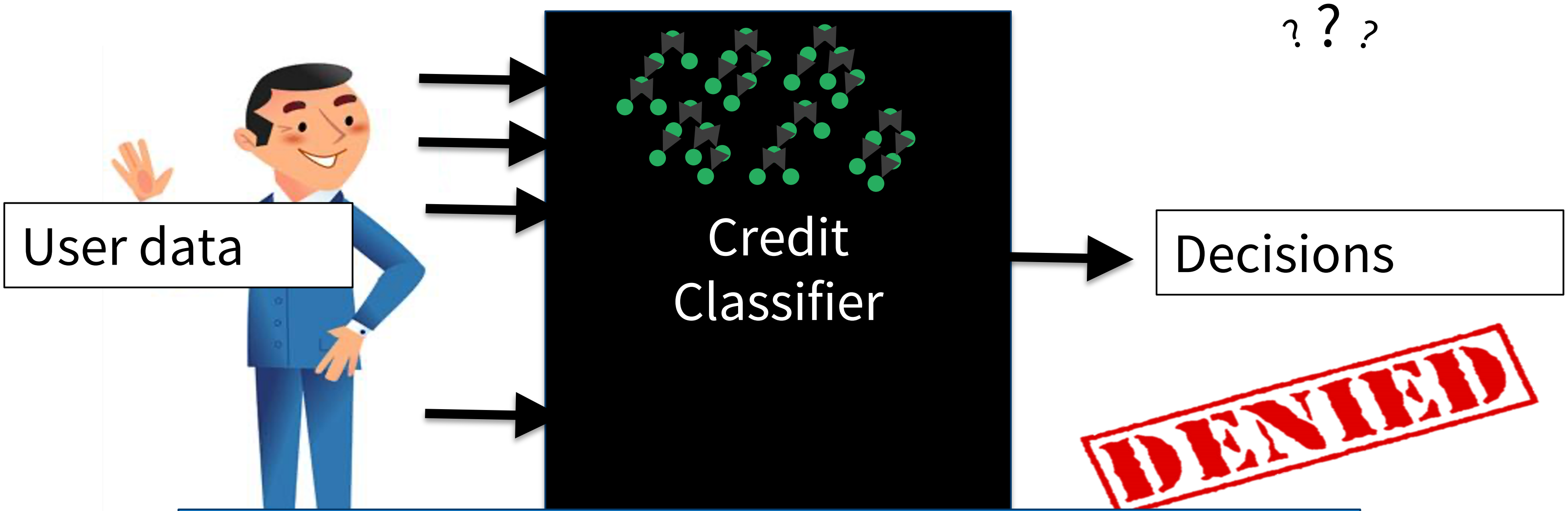
TEACHERS COLLEGE
COLUMBIA UNIVERSITY

amazon

facebook

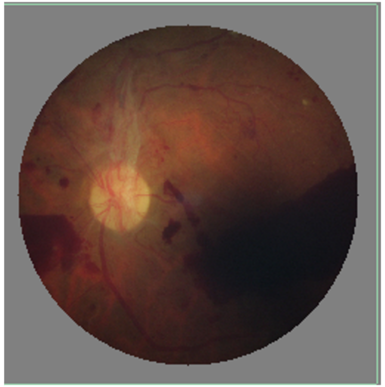
bing

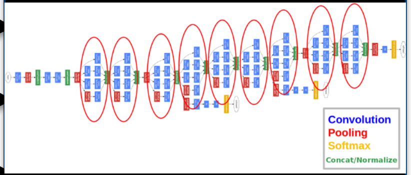
Machine Learning Systems are Opaque



Why was Joe denied credit by the tree ensemble model?

Machine Learning Systems are Opaque





DR Classifier

? ? ?

Diabetic retinopathy Stage 5

Why this diagnosis from the GoogleNet neural network?

Vision: Explanations ⟺ Machine Learning Model Quality

Explanations to enhance transparency, assess & improve model quality

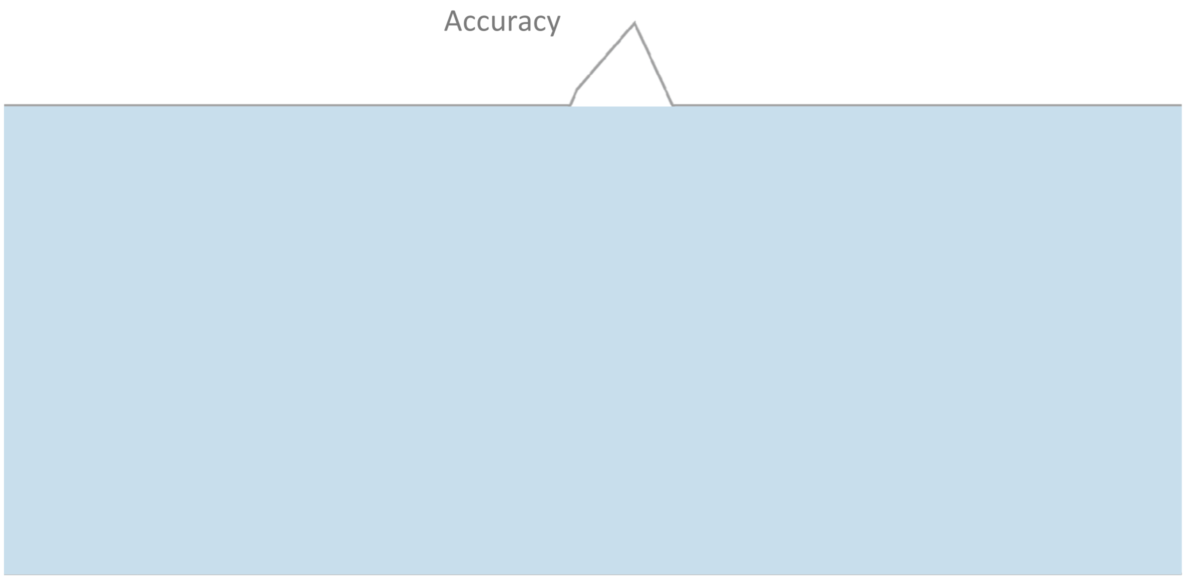
- What are requirements for “good” explanations?
- How can explanations enable model quality assessment & improvement?
  - Privacy, Fairness, Accuracy…

Applications: Finance, healthcare, …

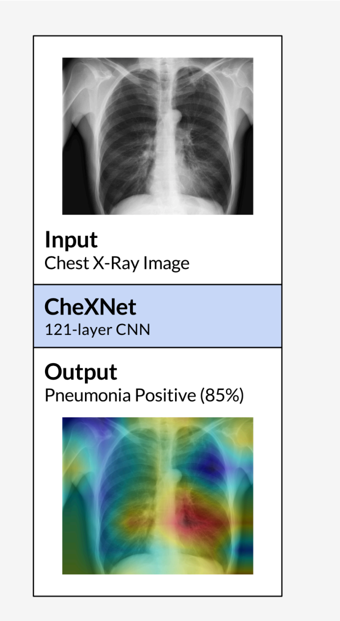
Vision 1 : Explanations & Machine Learning Model Quality

Model quality today: focused on model accuracy metrics

Emerging research: A lot more to model quality than accuracy



Vision 2: Explanations Enhances Trust and Transparency



[Andrew Y. Ng et. al. 2017]

EDITORS' PICK | Oct 16, 2019, 03:35pm EDT | 4,178 views

Explainable AI In Health Care: Gaining Context Behind A Diagnosis

THOUGHT LEADERS

Explainability: The Next Frontier for Artificial Intelligence in Insurance and Banking

Published 9 seconds ago on January 6, 2021
By Dr. Ori Katz

Artificial intelligence / Machine learning

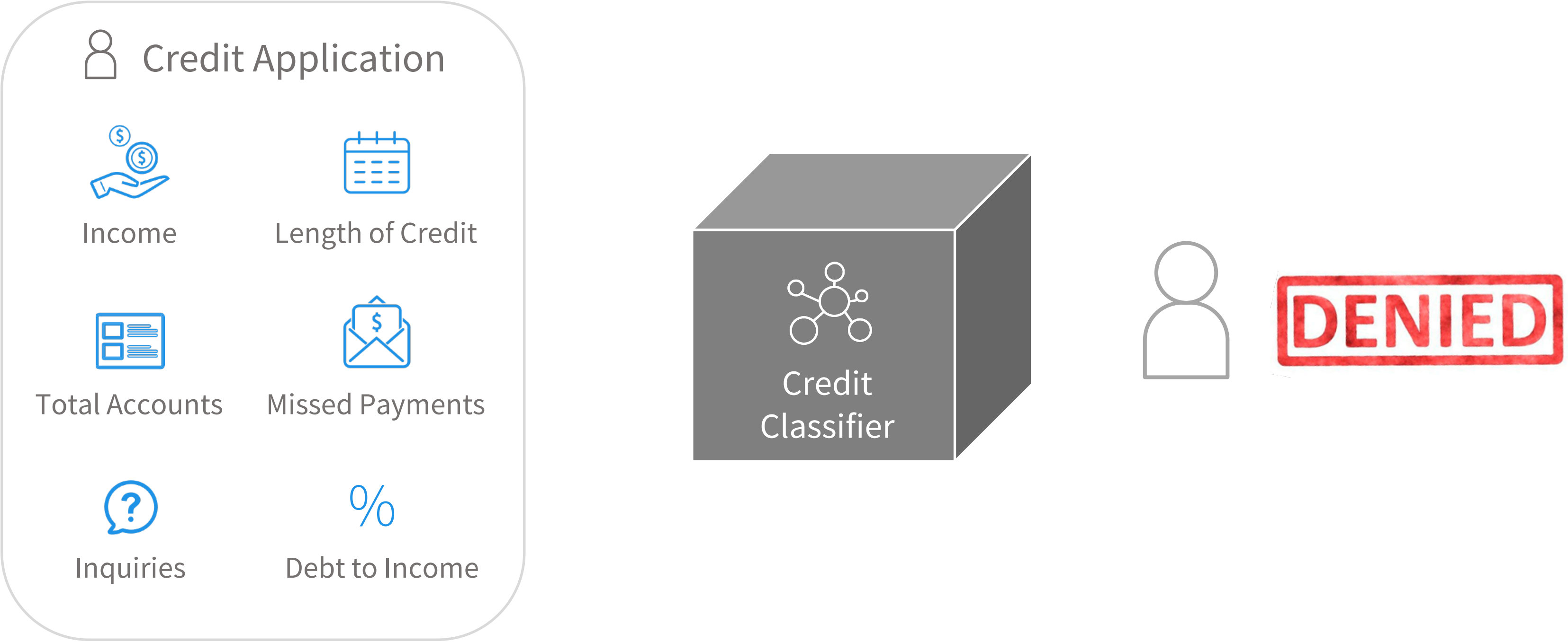
Nvidia Lets You Peer Inside the Black Box of Its Self-Driving AI

In a step toward making AI more accountable, Nvidia has developed a neural network for autonomous driving that highlights what it's focusing on.

Section I

Foundations of XAI

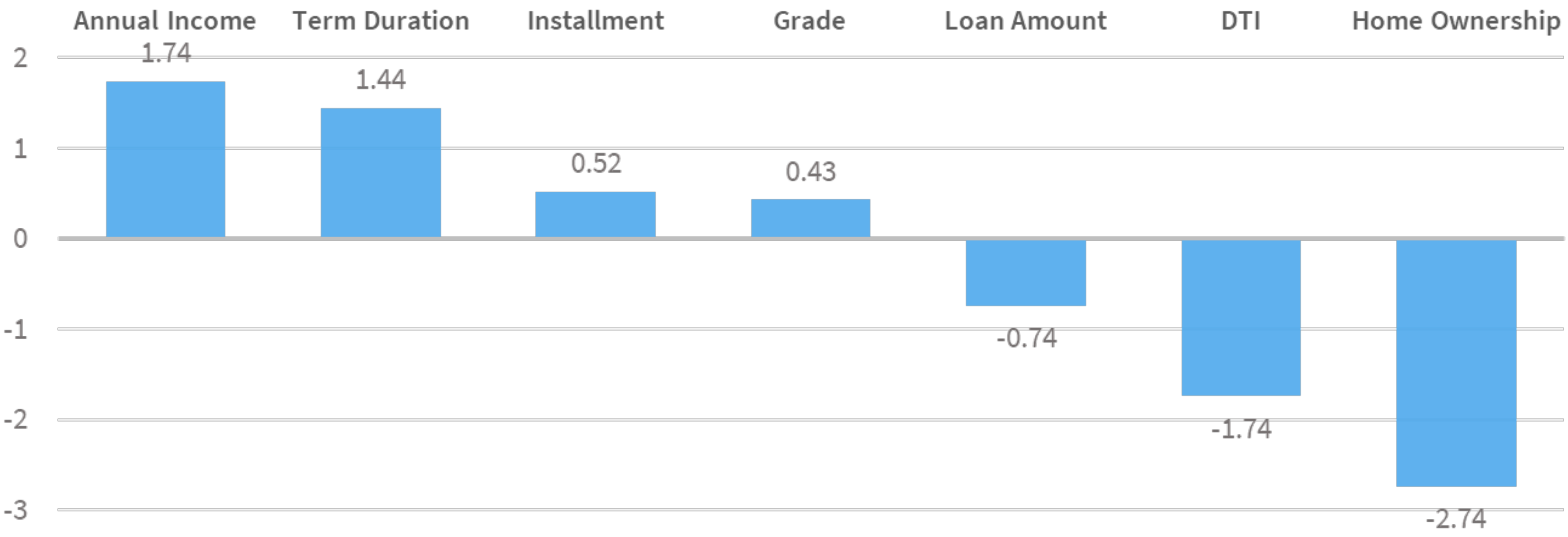
Explanations are Necessary



Requirements for "Good" Explanations

- Answer rich set of queries
- Capture causal influence
- Reflect "power" of a feature
- Be accurate

Input Feature Importance



Methods for Computing Input Feature Importance

2001

Permutation Importance (PI)
Breiman

2016

Quantitative Input Influence (QII)
Datta, Sen, Zick

Local Interpretable Model-Agnostic Explanations (LIME)
Ribeiro et al.

2017

Shapley Additive Explanations (SHAP)
Lundberg & Lee

Integrated Gradients
Sundararajan, Taly, Yan

2018

Influence-Directed Explanations
Leino, Sen, Datta, Fredrikson, Li

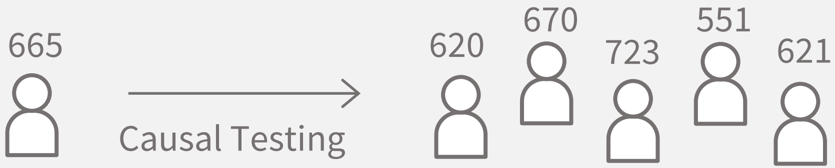
Similarities Across Methods

1 QUERY DEFINITION

Why does the model:

- have a score of 665 for Jane
- have disparate impact
- deny Jane

2 OUTPUT COMPARISON



3 SUMMARIZATION

Of 665, 133 is accounted for by DTI, -45 by income, etc.

(Aumann) Shapley Values

Power of a State (Feature)

Which states contribute
the most electoral votes?

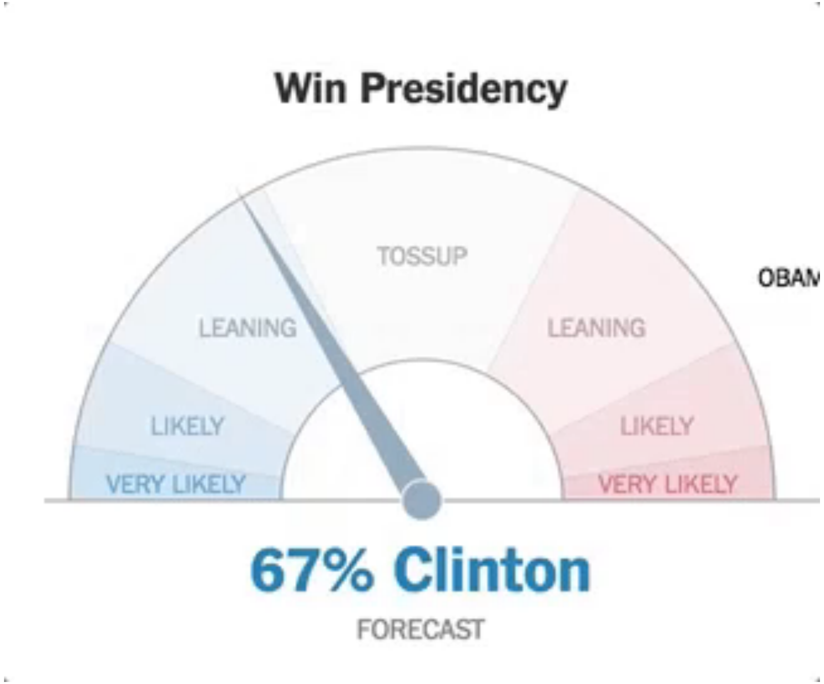
Power of a State (Feature)

Which states decide the winner?

Causal Influence of Pennsylvania is high

Power Depends on Marginal Influence

What is the effect of PA after results from IN, GA, MD are in?



Shapley Value Averages Marginal Influence

$$\phi_i(N, v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!\,(n - |S| - 1)!}{n!}\, m_i(S)$$

Symmetry	Dummy	Monotonicity
• Equal marginal contribution implies equal influence • Example: cloned features	• Zero marginal contribution implies zero influence • Example: features never touched by ML model	• Consistently higher marginal contribution yields higher influence • Necessary to compare feature influence scores of individuals

Reflect “power” of a feature

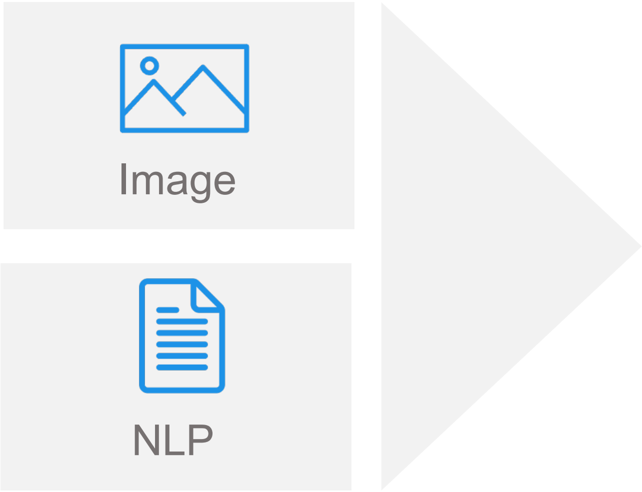
Efficient Shapley Value Estimation

- Exact computation is exponential in the number of features
- Efficient estimation
  - Sampling
  - Leveraging structure of tree models
- PAC-style bounds on accuracy of estimation
- High empirical accuracy

Takeaways

- Shapley Value based methods can be the basis for meaningful reason codes
  - Captures “power” of a feature while accounting for feature interactions
- Reason codes vary significantly based on which comparison group is chosen
  - Approved applicants vs All applicants
- Explanations vary based on model output type
  - Log-odds vs probabilities vs classification outcomes
- Explanation accuracy is critical
  - Methods like TreeSHAP are accurate for risk scores but can be very inaccurate for classification outcomes
  - QII method is accurate for risk scores, probabilities, classification outcomes

Explaining Deep Neural Networks



1. Input Feature Importance
2. Internal Explanations

Integrated Gradient

Shapley Value → Aumann-Shapley → Integrated Gradient

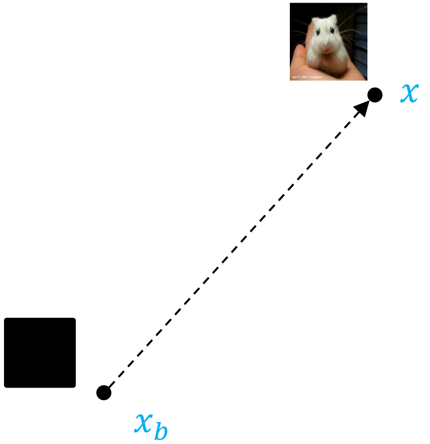
continuous features

neural network

[Sundararajan et al. ICML 2017]

$$IG(x;\ x_b, F) = (x - x_b)\int_0^1 \frac{\partial F(\gamma(\alpha; x, x_b))}{\partial \gamma} d\alpha$$

$$\text{where } \gamma(\alpha; x, x_b) = x_b + \alpha(x - x_b)$$



Aggregating the gradient of all points on a linear path from a user-selected baseline to the target input

Integrated Gradient

Shapley Value → Aumann Shapley → Integrated Gradient

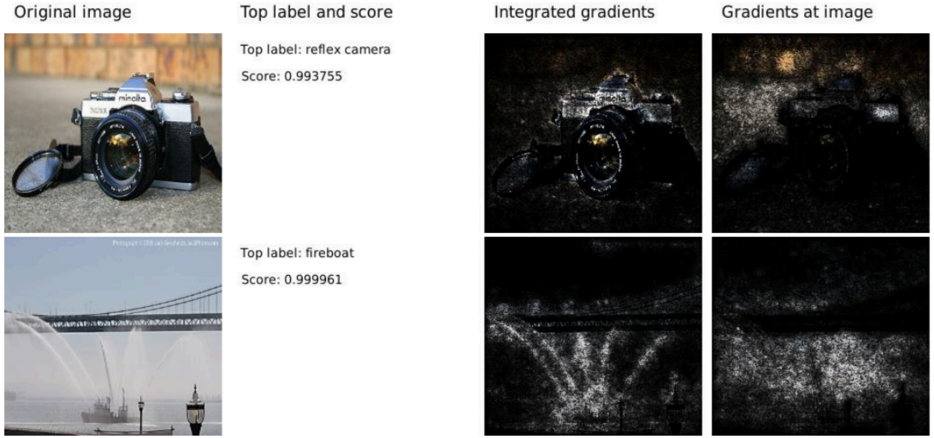
continues features

differentiable output

[Sundararajan et al. ICML 2017]

Integrated Gradient is the **only** path method that satisfies

- Symmetry
- Dummy
- Efficiency(Completeness)
- Additivity



Now It's Time to Dive Deeper…

Input Attributions → **Internal** Attributions

Why we are interested in internal representations?

Deep Neural Network

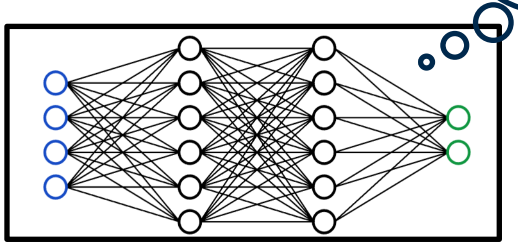
"Sports Car"

Now It's Time to Dive Deeper…

Input Attributions → **Internal** Attributions

Why we are interested in internal representations?

What does each neuron learn?



"Sports Car"

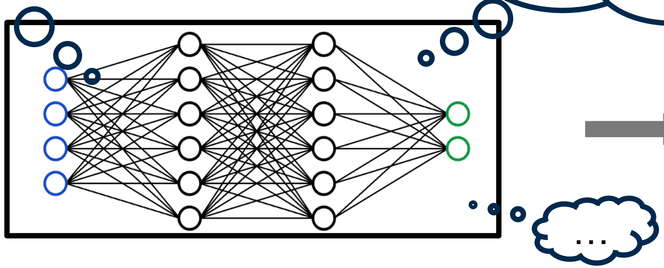
Now It’s Time to Dive Deeper…

Input Attributions

Internal Attributions

What is the key concept related to the label over a distribution of input?

What concept does each neuron learn?



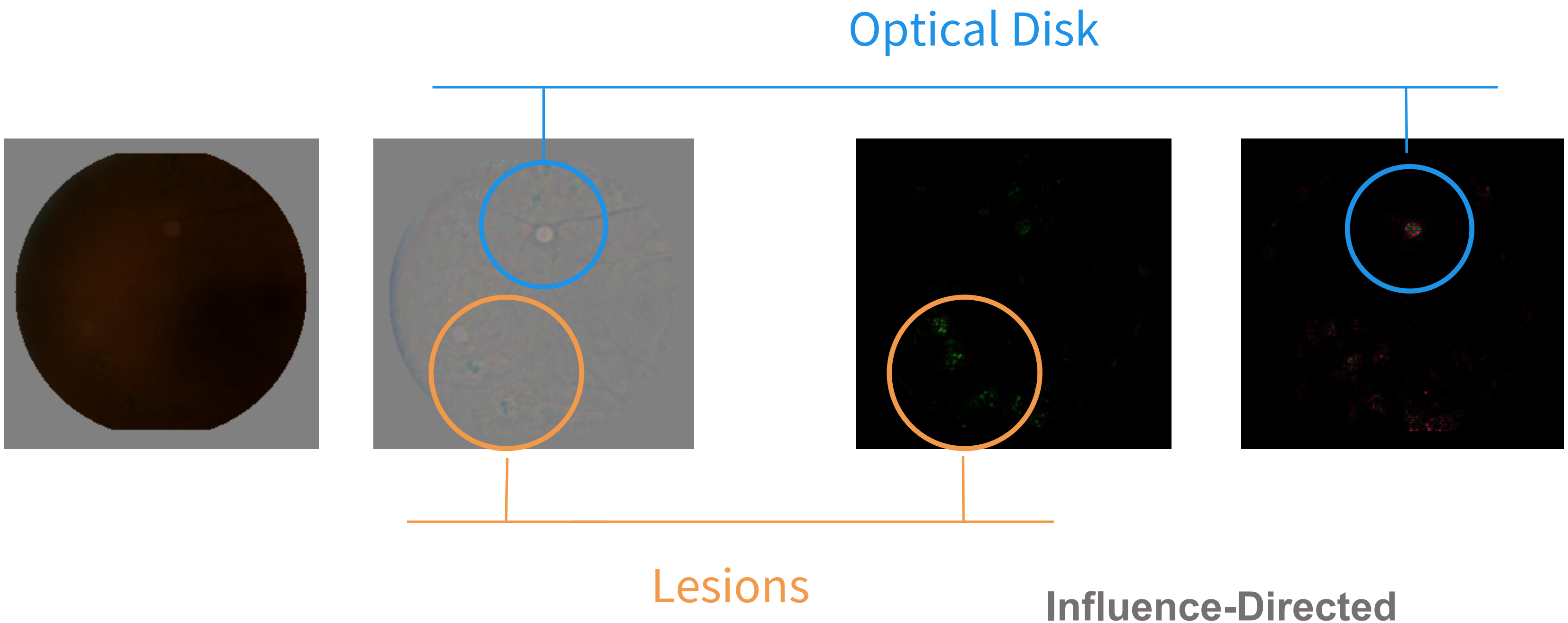
“Sports Car”

What Makes Orlando Bloom Orlando Bloom?

Internal explanation for a deep network

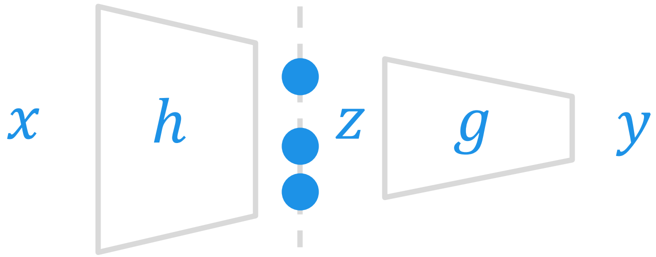
Influence-Directed Explanations
Leino, Sen, Fredrikson, Datta, Li, ITC '18

Detecting Diabetic Retinopathy Stage 5



Influence-Directed Explanations
Leino, Sen, Fredrikson, Datta, Li 2018

Requirements for “Good” Explanations



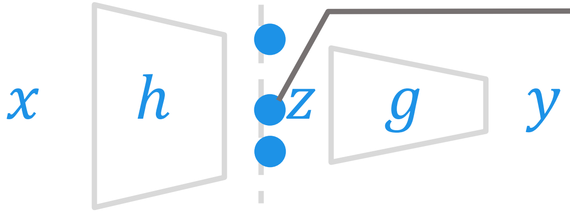
Causal	Succinct	Distributional Faithfulness
Identify features that are causing model predictions	A “few” features explain model predictions	Model is fed “familiar” inputs

Influence-Directed Explanations

Leino, Sen, Fredrikson, Datta, Li, ITC ‘18

Distributional Influence

Influence = average gradient over distribution of interest



$$y = F(x) = g(z), z = h(x)$$

$$\mathrm{I}_j^s(F, P) := \int_{x \in \mathcal{X}} \frac{\partial g(z)}{\partial z_j} P(x) dx$$

Slice with *neuron z_j*

Gradient of **Quantity of Interest** *g(.)*

For input x [note z = h(x)]

Weighted by probability of input x from **Distribution of Interest** P

Influence-Directed Explanations

Leino, Sen, Fredrikson, Datta, Li, ITC '18

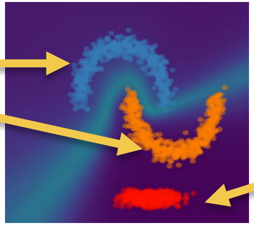
Axiomatic Foundation for Distributional Influence

$$\mathrm{I}_j^s(F, P) := \int_{x \in \mathcal{X}} \frac{\partial g(z)}{\partial z_j} P(x) dx$$

When s is the input slice($h(x) = x$), Distributional Influence satisfies:

- **Axiom (1), Linear Agreement**: If F behaves linearly over the distribution of interest, then $\mathrm{I}_j^s(F, P)$ returns the weight of the j-th feature .
- **Axiom (2), Distributional Marginality**: If the partial derivatives w.r.t. an input feature are identical for F_1, F_2 over the distribution of interest, then $\mathrm{I}_j^s(F_1, P) = \mathrm{I}_j^s(F_2, P)$
- …

We are interested in



We are **not** interested in

Influence-Directed Explanations

Leino, Sen, Fredrikson, Datta, Li ITC '18

Images Source: Simple and Principled Uncertainty Estimation with Deterministic Deep Learning via Distance Awareness [Liu et al. 2020]

Distributional Influence Generalizes Existing Methods

$$\mathrm{I}_j^s(F,P) := \int_{x\in\mathcal{X}} \frac{\partial g(z)}{\partial z_j} P(x)dx$$

When s is the input slice($h(x) = x$)

- and $\mathcal{X}$ is a set of points (uniformly) distributed on a linear path from a baseline input to the target input

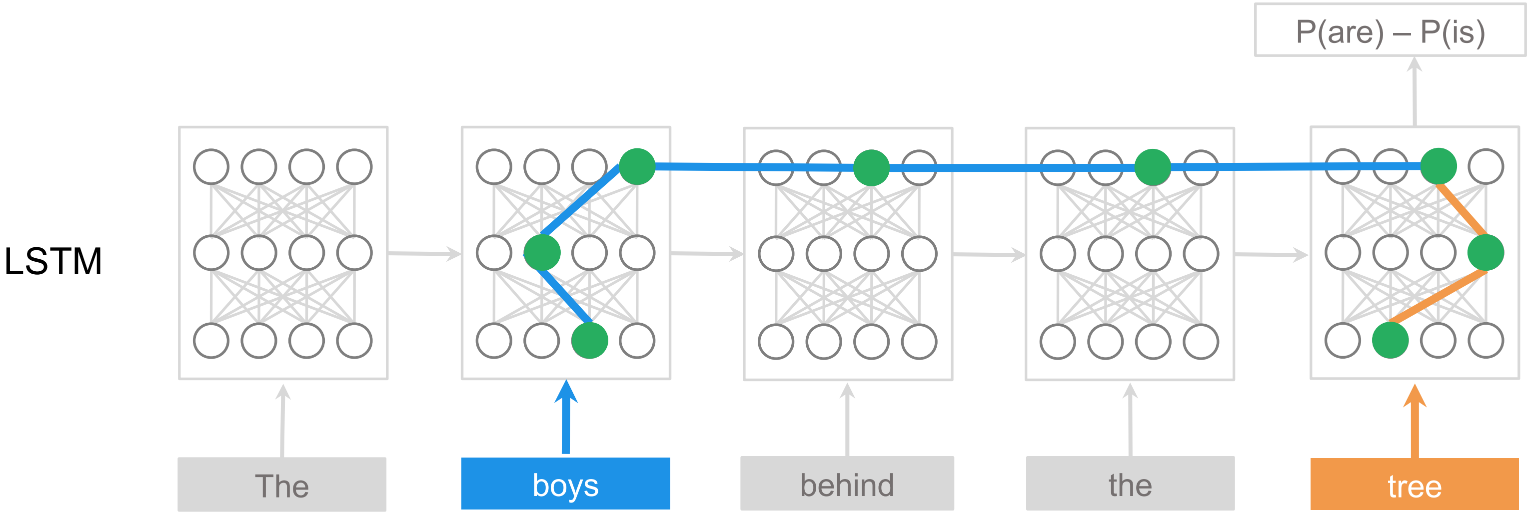
→ multiplying $\mathrm{I}_j^s(F,P)$ with $(x - x_b)$ → Integrated Gradient *[Sundararajan et al. 2017]*

- and $\mathcal{X}$ is a set of points in the Gaussian Distribution centered with the target input

→ Smooth Gradient *[Smilkov et al. 2017]*

⋮

Internal Explanations via Influence Paths



- Influence paths provide insights into misclassifications
- Model can be compressed down the influential paths without changing the utility of the model

Influence Paths
Lu, Mardziel, Leino, Fedrikson, Datta, ACL '20

Model Compression with Influence Paths

- Primary path from the subject alone provides strong signal for SVA; removing it breaks the model
- Removing primary path from the intervening noun
    - Decreases performance if it is a helpful noun
    - Increases performance if it is an attractor

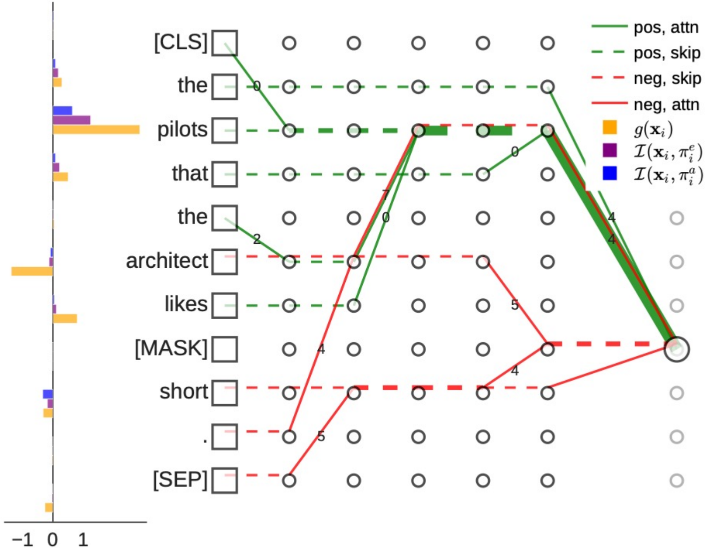
Task	C	Compression Scheme						
		$\overline{C_{si}}$	$\overline{C_s}$	$\overline{C_i}$	C_{si}	C_s	C_i	C
nounPP	SS	.66	.77	.95	.93	.71	.77	.95
nounPP	SP	.64	.36	.94	.64	.75	.40	.74
nounPP	PS	.34	.24	.92	.40	.69	.18	.80
nounPP	PP	.39	.66	.91	.76	.68	.58	.97
nounPP	mean	.51	.51	.93	.68	.70	.48	.87

Ci: Only keep primary from intervening noun
Cs: Only keep primary path from subject
Csi: combination of Ci and Cs
C: The original model
$\bar{C}$: complements

Influence Graphs for BERT

BERT v.s. LSTM

- Scaling up method to identify influential paths
- Prevalence of "copy" and "transfer" operations to carry context



Influence Graphs for BERT
Lu, Wang, Mardziel, Datta, 2020

Axiomatic Foundations of Explanations

Dummy　Symmetry　Linearity　Monotonicity　Distributional Faithfulness

Weight Faithfulness　Robustness　Proportionality　Efficiency　(…)

ConceptSHAP

TreeSHAP

…

SHAP

Shapley Value

QII

LIME

Saliency Map

Uniform Gradient

Smooth Gradient

LRP

DeepLIFT

TSP

Distributional Influence → ***Pathway Influence***

Integrated Gradient → *Conductance*

Occlusion-N

Guided BP

Uncategorized methods…

CAM

Grad-CAM

Information Bottleneck

Feature Visualization

If an axiom is not noted on a method, it is either not validated yet or violated

Related Work

	Explanation Framework Properties			*Influence Properties*	
	Quantity	**Distribution**	**Internal**	**Marginality**	**Sensitivity**
Influence-Directed Explanation *[Leino et al. ITC '18]*	✓	✓	✓	✓	✓*
Conductance *[Dhamdhere et al. ICLR '19]*		✓⁻	✓	✓	✓
Integrated Gradient *[Sundararajan et al. ICML '17]*		✓⁻		✓	✓
Smooth Gradient *[Smilkov et al. 2017]*		✓⁻		✓	✓
Simple Taylor *[Bach et al. 2015 PLOS ONE]*		✓⁻		✓	
Deconvolution *[Zeiler et al. ECCV '14]*			✓†		
Guided Backpropagation *[Springenberg et al. 2015 ICLR Workshop]*			✓†	✓	
Layer-wise Relevance Propagation *[Bach et al. 2015 PLOS ONE]*		✓⁻	✓†	✓*	✓*

✓ Supports

✓⁻ Limited flexibility

✓* Supports under some parameterizations

✓† Internal influence as an intermediate step

Takeaways

“Good” explanations

- Answer rich set of queries
- Capture causal influence
- Reflect “power” of a feature (axiomatic foundations)
- Are accurate

Applies consistently to

- Traditional statistical ML and neural networks
- Structured, image, text data

Demo TruLens

Library containing attribution and interpretation methods for deep nets.

```
pip install trulens
```

Explain and visualize models built with

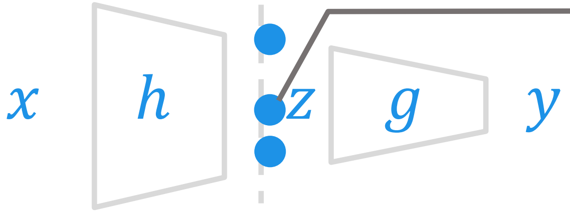
TensorFlow 2.0

PyTorch

github.com/truera/trulens

Recap | Distributional Influence

Influence = average gradient over distribution of interest



$$y = F(x) = g(z), z = h(x)$$

$$\mathrm{I}_j^s(F,P) := \int_{x\in\mathcal{X}} \frac{\partial g(z)}{\partial z_j} P(x)dx$$

Slice with *neuron z_j*

Gradient of **Quantity of Interest** *g(.)*

For input x [note z = h(x)]

Weighted by probability of input x from **Distribution of Interest** P

Influence-Directed Explanations

Leino, Sen, Fredrikson, Datta, Li, ITC '18

Demo TruLens

Library containing attribution and interpretation methods for deep nets.

```
pip install trulens
```

Explain and visualize models built with

TensorFlow 2.0

PyTorch

github.com/truera/trulens

Q & A

Break I [We will be back at 1:20 pm PT]

Section II

From Explainability to Model Quality

Privacy

Explanations

Fairness

Part One

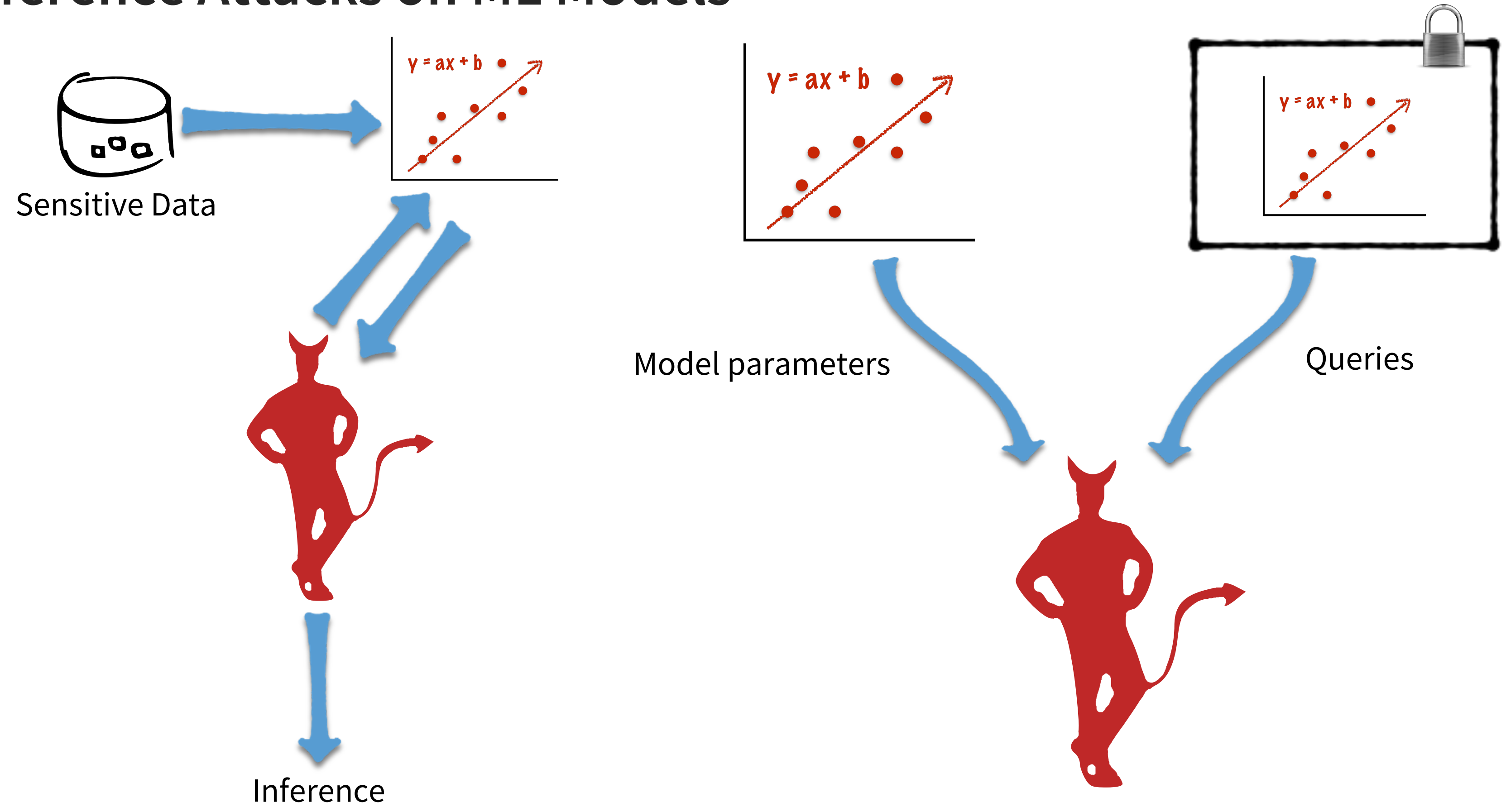
Model Quality & Privacy

Machine learning models can potentially violate societal privacy norms

- Misuse protected information when making predictions
- Automate, enhance surveillance activities
- Leak confidential information about subjects or training data

These outcomes are usually unintentional, symptomatic of model quality issues!

Inference Attacks on ML Models



Leaky Language Models

Carlini et al., "The Secret Sharer: Evaluating and Testing Unintended Memorization in Neural Networks". USENIX Security '19

"users may find that the input 'my social-security number is ...' gets auto-completed to an obvious secret"

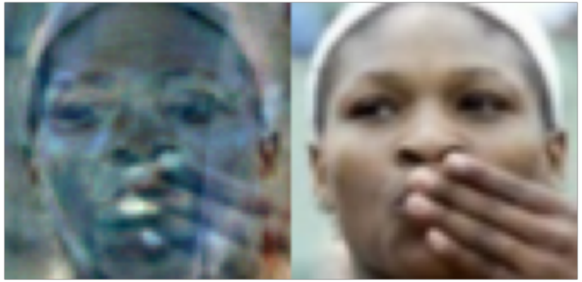
User	Secret Type	Exposure	Extracted?
A	CCN	52	✓
B	SSN	13	
C	SSN	16	
	SSN	10	
	SSN	22	
D	SSN	32	✓
F	SSN	13	
G	CCN	36	
	CCN	29	
	CCN	48	✓

Table 2: Summary of results on the Enron email dataset. Three secrets are extractable in < 1 hour; all are heavily memorized.

Reconstructing Training mages

Model Inversion *[Fredrikson et al., CCS'15]*

- Looked at facial recognition models
- Turkers matched reconstructed images to training data overwhelmingly often
- Limitation: models were simple



Howto: Reconstruct Training Images

Algorithm 1 Inversion attack for facial recognition models.

function $\text{MI-FACE}(label, \alpha, \beta, \gamma, \lambda)$

$\quad c(\mathbf{x}) \stackrel{\text{def}}{=} 1 - \tilde{f}_{label}(\mathbf{x}) + \text{AUXTERM}(\mathbf{x})$

$\quad \mathbf{x}_0 \leftarrow \mathbf{0}$

$\quad$ **for** $i \leftarrow 1 \ldots \alpha$ **do**

$\quad\quad \mathbf{x}_i \leftarrow \text{PROCESS}(\mathbf{x}_{i-1} - \lambda \cdot \nabla c(\mathbf{x}_{i-1}))$

$\quad\quad$ **if** $c(\mathbf{x}_i) \geq \max(c(\mathbf{x}_{i-1}), \ldots, c(\mathbf{x}_{i-\beta}))$ **then**

$\quad\quad\quad$ **break**

$\quad\quad$ **if** $c(\mathbf{x}_i) \leq \gamma$ **then**

$\quad\quad\quad$ **break**

$\quad$ **return** $[\arg\min_{\mathbf{x}_i}(c(\mathbf{x}_i)), \min_{\mathbf{x}_i}(c(\mathbf{x}_i))]$

- Basic idea: gradient descent on *model input*, towards targeted class
  - Processing, regularization for image quality
  - Often vanilla GD works just as well
- Attack is "whitebox“
  - Blackbox variant thwarted by quantizing output

Key quantity is the gradient wrt the input

This is given by many explanation methods!

Reconstruction and Explanations

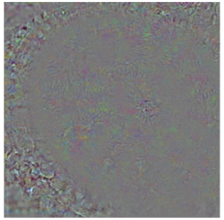
VGG

Resnet

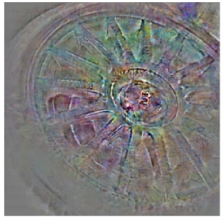
Robust models are also more prone to model inversion!

Recent observation: robust models are more explainable (see Part 3 of this tutorial)

Saliency Map on Regular Model
ResNet50



Saliency Map on Robust Model
ResNet50



[Meija et al. NeurIPS PriML'19]

Membership Inference *[Shokri et al. Oakland'17, Yeom et al. CSF'18]*

Attacker's goal: determine whether given point was in training data

1. Sample dataset S from population distribution D, train model F on S
2. Choose uniform-random b from {0,1}
3. Draw $z = (x, y)$ from S if $b = 0$, otherwise draw z from D
4. Give attacker A following information: F, z, D
5. Attacker "wins" if $A(F, z, D) = b$

Why is this a privacy risk?

- Think: medical data, political surveys, …
- Sometimes viewed as a general indicator of training data leakage

Why is this even possible?

Seems to contradict the purpose of ML: learn general trends from many examples

Key idea: overfitting (poor generalization in loss) is sufficient for membership vulnerability

Theorem. There exists a membership adversary whose advantage is proportional to the model's generalization error [Yeom et al., CSF'18].

Surprise: overfitting is *not necessary* for membership vulnerability

Theorem. Given an ε*(n)*-ARO-stable learning rule *L*, there exists a related *L'* that is ε*'(n)*-ARO-stable, where |ε*(n)*-ε*'(n)*|is negligible in *n*, and *L'* admits a membership adversary that achieves advantage near 1 with high probability. [Yeom et al., CSF'18].

Membership inference from feature use *[Usenix Security'20]*

Hypothesis: feature use provides *evidence* of membership

training set

Celebrity A

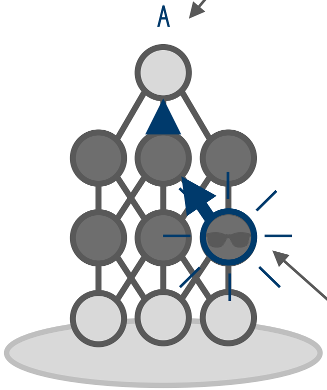
celebrity A has sunglasses in 50% of training instances

training set

Celebrity B

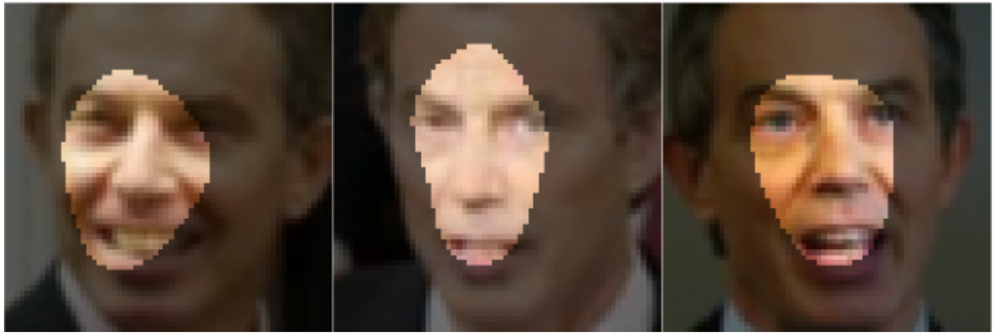
celebrity B has sunglasses in 25% of training instances

influence of "sunglasses" feature indicates membership

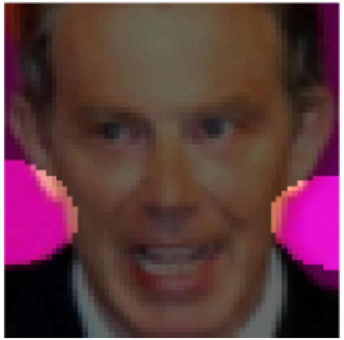


sunglasses are predictive in training set

Sample of LFW training instances



Typical explanations on test instances of Tony Blair



Attribution map on training instance of Tony Blair with distinctive pink background, which is influential on the model's correct prediction.

Leveraging Explanations to Fix Representations

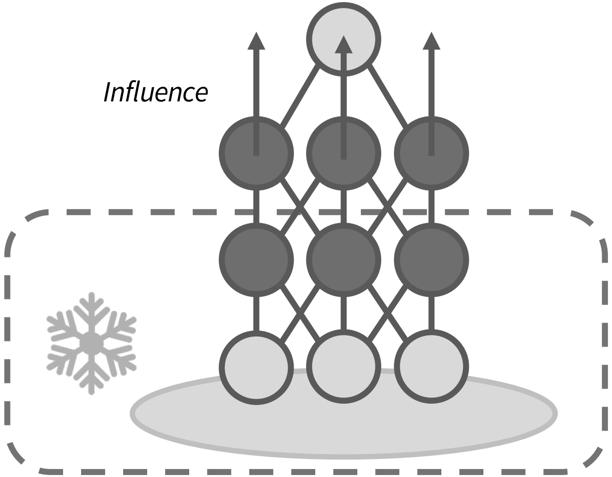
Internal influence gives us the information we need

Step 1: estimate “normal” distribution of feature importance

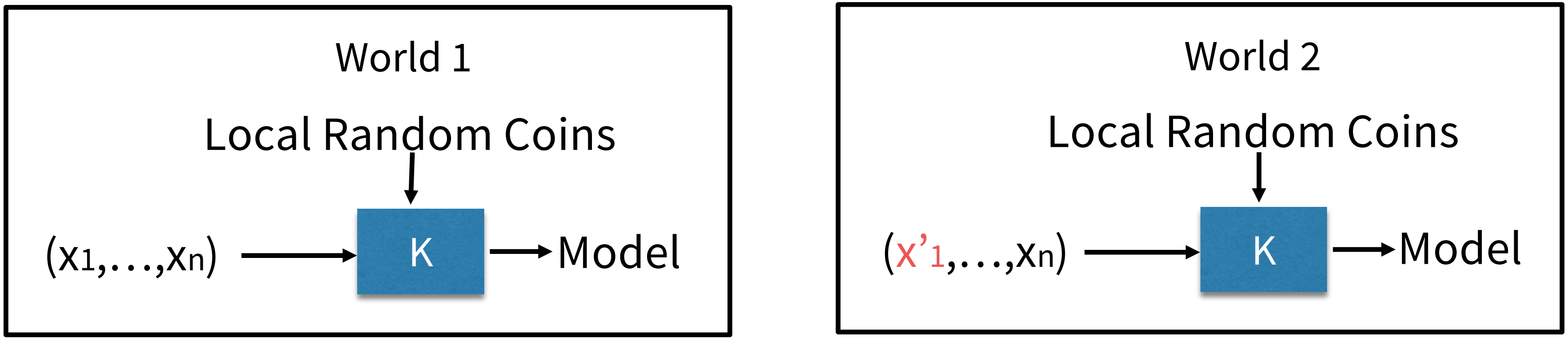
- Freeze network up to a given layer
- Train “proxy” models above that layer
- Measure feature importance on proxies

Step 2: estimate of how useful a feature is as evidence of membership

Step 3: build “attack model” to predict membership



Differential Privacy: A Rigorous Defense



Differential privacy says:

For all x1, x1', s . $\Pr[K(x1,\ldots,xn) = s] \leq \exp(\boldsymbol{\varepsilon}) \times \Pr[K(x1',\ldots,xn) = s]$

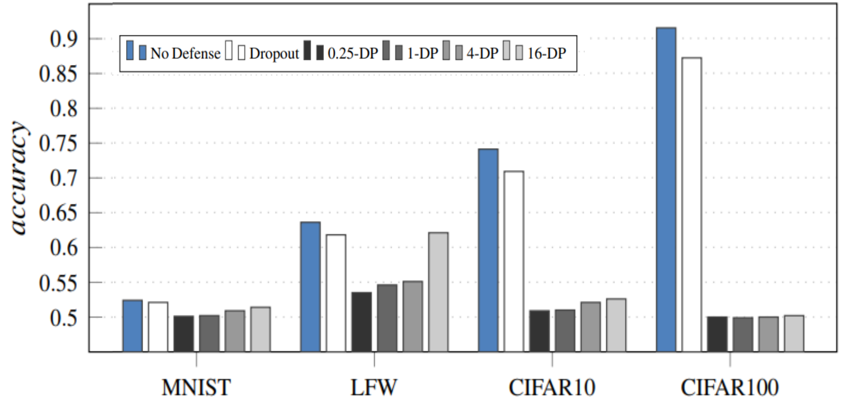
Bounds the relative advantage of *any* breach!

Close Match for Membership Inference

Membership inference is closely tied to differential privacy

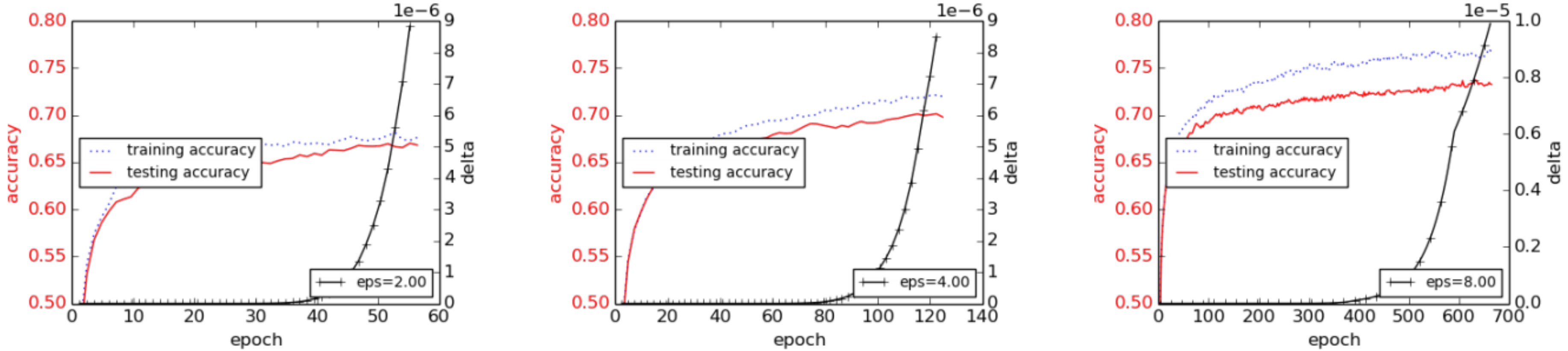
> **Theorem** [Yeom et al., CSF'18]. If F is ε-differentially private, then any membership adversary A will have advantage bounded by $e^{\varepsilon} - 1$.

The "proven" ε is a (probably loose) upper-bound on the property satisfied by a model



The Downside: Accuracy Tradeoff

Source: Abadi et al., Deep Learning with Differential Privacy. CCS'16



(1) $\varepsilon = 2$ (2) $\varepsilon = 4$ (3) $\varepsilon = 8$

CIFAR10, pre-trained convolutional filters, with tensorflow-privacy

Summary

Model quality issues can lead to unintentional privacy issues

In some cases, these can be identified using explanation techniques

There are many open questions around balancing privacy, utility, and explainability

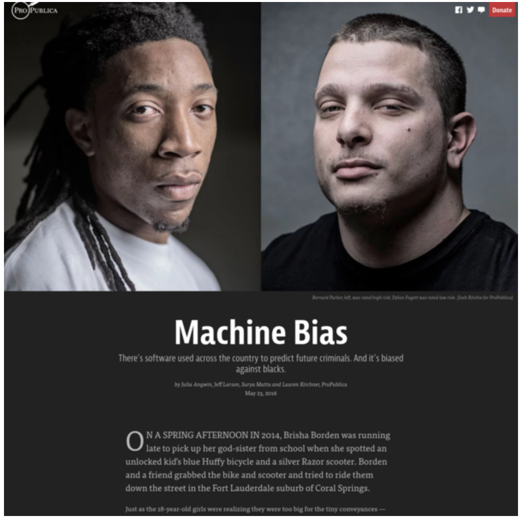
Privacy

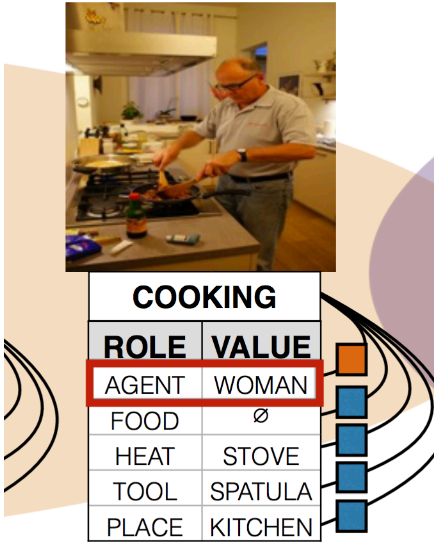
Explanations

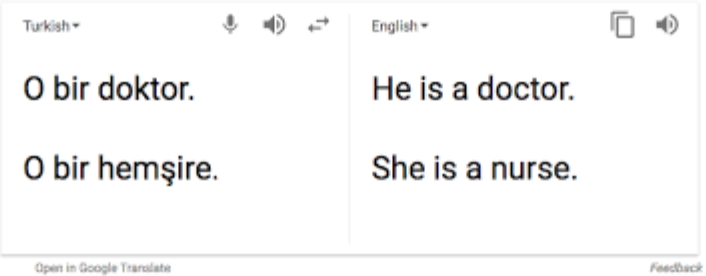
Fairness

Part Two

Bias in ML Applications

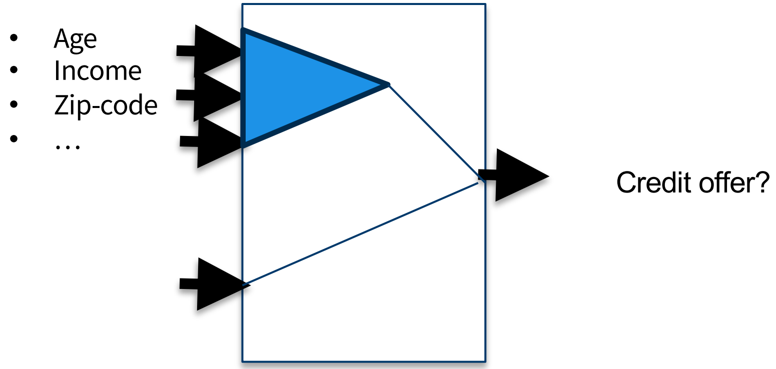






Proxy Use & Fairness

Protected information type: Race



Proxy use

- Interpretation (Strong predictor; associated)
- Influence (high QII)

Proxy Use

Datta, Fredrikson, Ko, Mardziel, Sen CCS 2017
Yeom, Datta, Fredrikson NIPS 2018

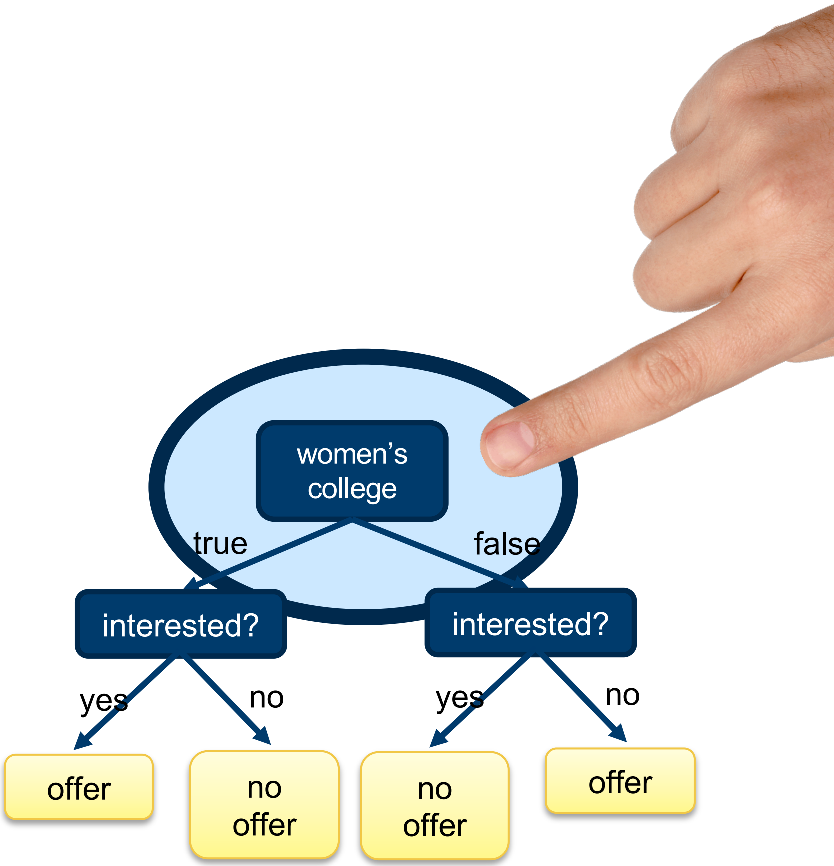
Proxy Use in Tree Models

Decomposition is:

- p_1: subtree of model's AST
- p_2: enclosing context

Finding of proxy use includes a *witness:* a subtree that causes the use

Can function as an explanation for some discriminatory behaviors in the model!



Proxies in Linear Models

$$Y(\boldsymbol{X}) = a_1X_1 + a_2X_2 + \ldots + a_nX_n$$

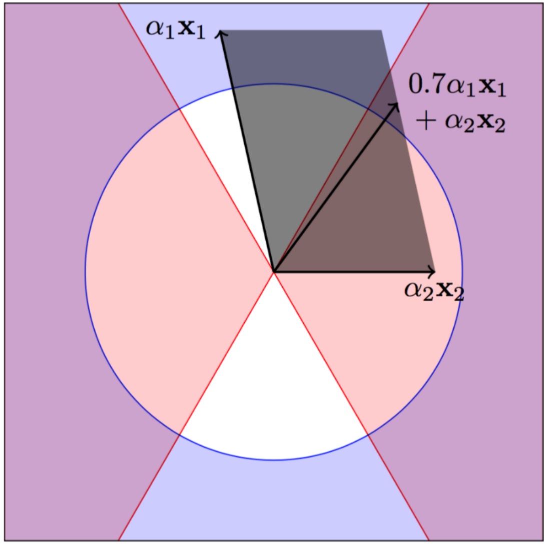
What are the decompositions?

- Individual terms a_nX_n? Or groups like $a_1X_1 + a_2X_2$?
- What about $0.5*a_1X_1 + a_2X_2$?

$$\boxed{\begin{gathered}\text{Component } P(\boldsymbol{X}) = \beta_1 a_1X_1 + \beta_2 a_2X_2 + \ldots + \beta_n a_nX_n \\ \text{for } \beta_1, \ldots, \beta_n \in [0, 1]\end{gathered}}$$

Proxies in Linear Models

$$Y(\boldsymbol{X}) = a_1X_1 + a_2X_2 + \ldots + a_nX_n$$



View random variables as vectors in inner product space

- Covariance is an inner product
- Influence is proportional to magnitude (i.e. variance)
- Association measured by the angle between variables

This gives us:

$$\iota(X, X') = \mathbf{E}_{X,X'}[\ (Y(\boldsymbol{X}) - Y(\boldsymbol{X}, P(\boldsymbol{X}')))^2\ ] \propto \mathrm{Var}(\ P(\boldsymbol{X}))$$

$$\mathrm{Asc}(Y, Z) \propto \mathrm{Cov}(Y, Z)$$

Finding Linear Proxies

Encode as Quadratic Program

- Maximize influence
- Subject to association threshold

Linear Relaxation

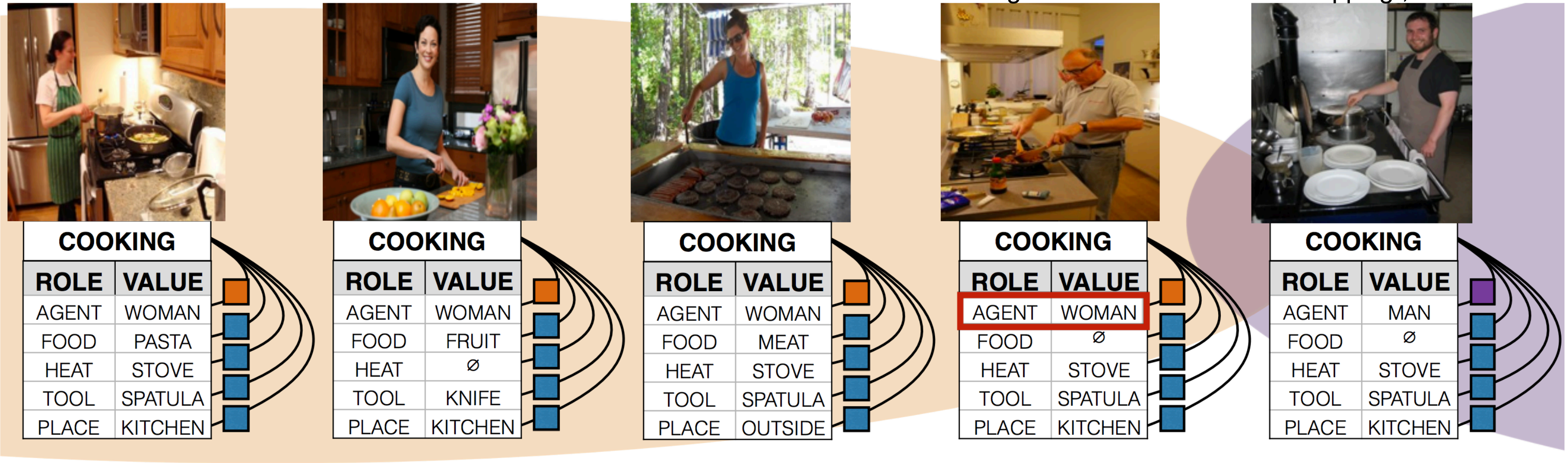
- Lower-bound influence
- Solutions over-approximate

Solution iff proxy exists

- QP tractable in some cases
- LP relaxation gives good results in practice

Bias Amplification *[Zhao et al., EMNLP'17]*

Image source: "Men also like shopping", Zhao et al.



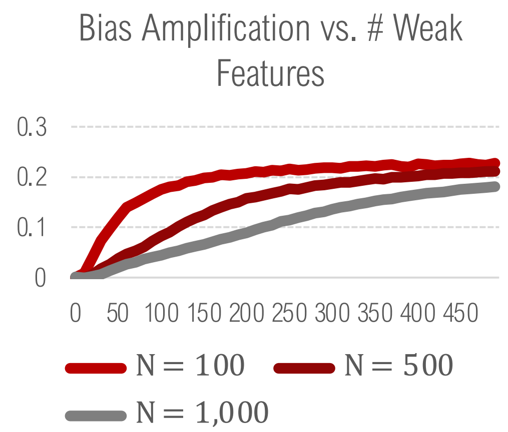
In training data, 66% of "cooking" images have women in them

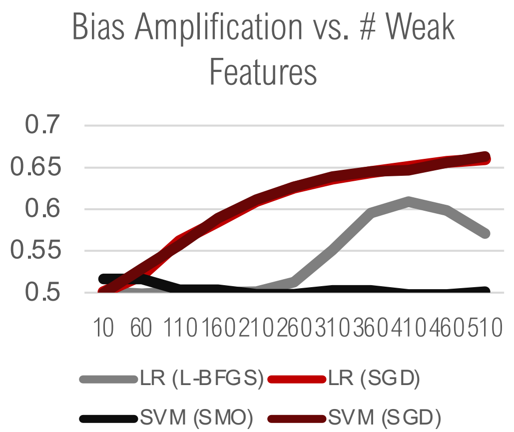
In predictions, 84% of "agent" roles in cooking images are labeled "woman"

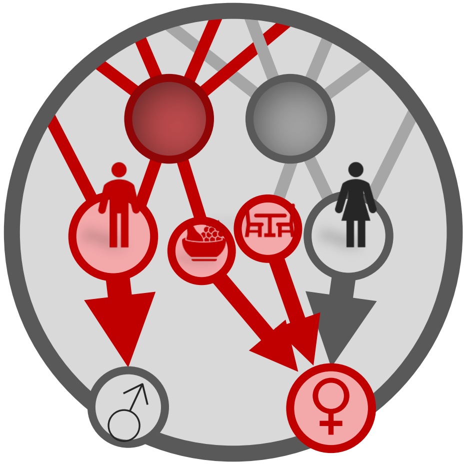
Feature-wise Bias Amplification *[ICLR'19]*

Intuition: "kitchen features" are weak proxies for gender in dataset

- Weak features have too much influence in predictions
- Prevalent weak features for class → biased predictions
- Consistent outcome with gradient descent



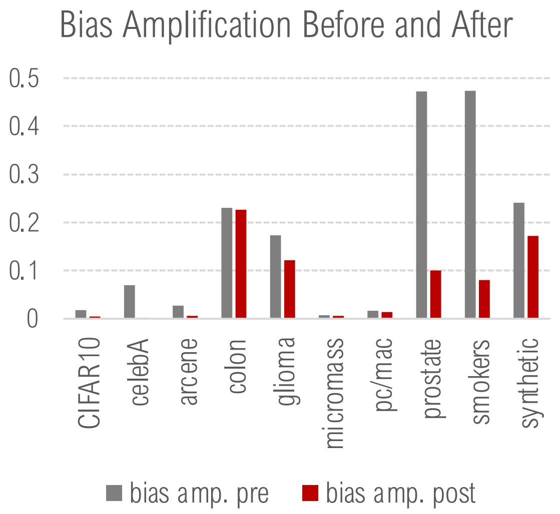


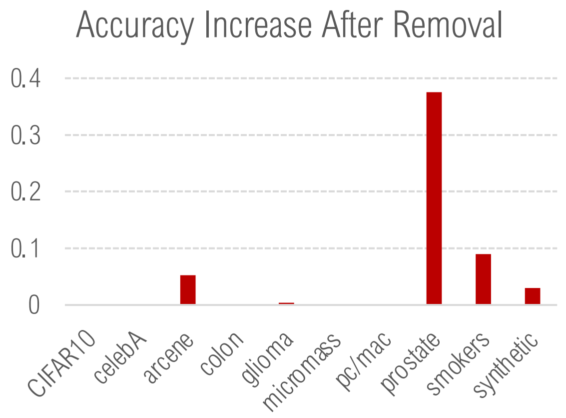


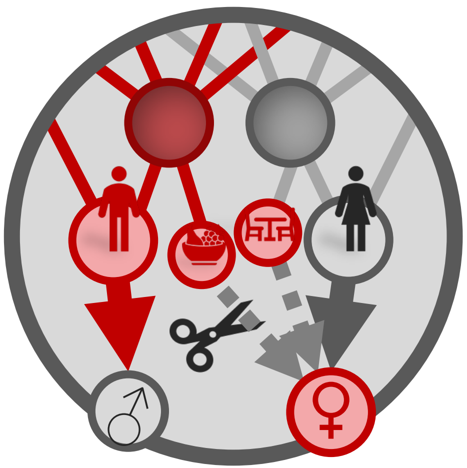
Quick Fix: Feature Pruning

Intuition: balance weak features across classes

- Measure internal influence to identify weak features
- Optimize “cut set” to mitigate bias while preserving accuracy
- Remove selected features from model







Summary

Fairness in learning is a complex issue, with no one-size-fits-all solution or technique

Explaining a model's use of protected information, and its features, can shed light on discriminatory outcomes

Q & A [2:00pm – 2:20pm Pacific Time]

Break II

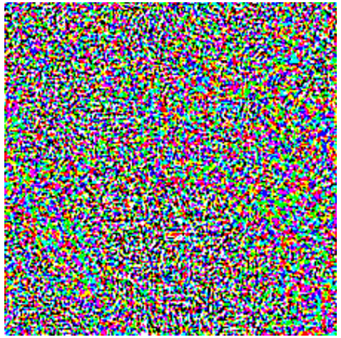
Section IV will start on 2:30 pm, Pacific Time

Section III

From Model Quality to Explainability

Fooling a DNN is easy

$+ .007 \times$



$=$

“panda”

adversarial perturbation

“gibbon”

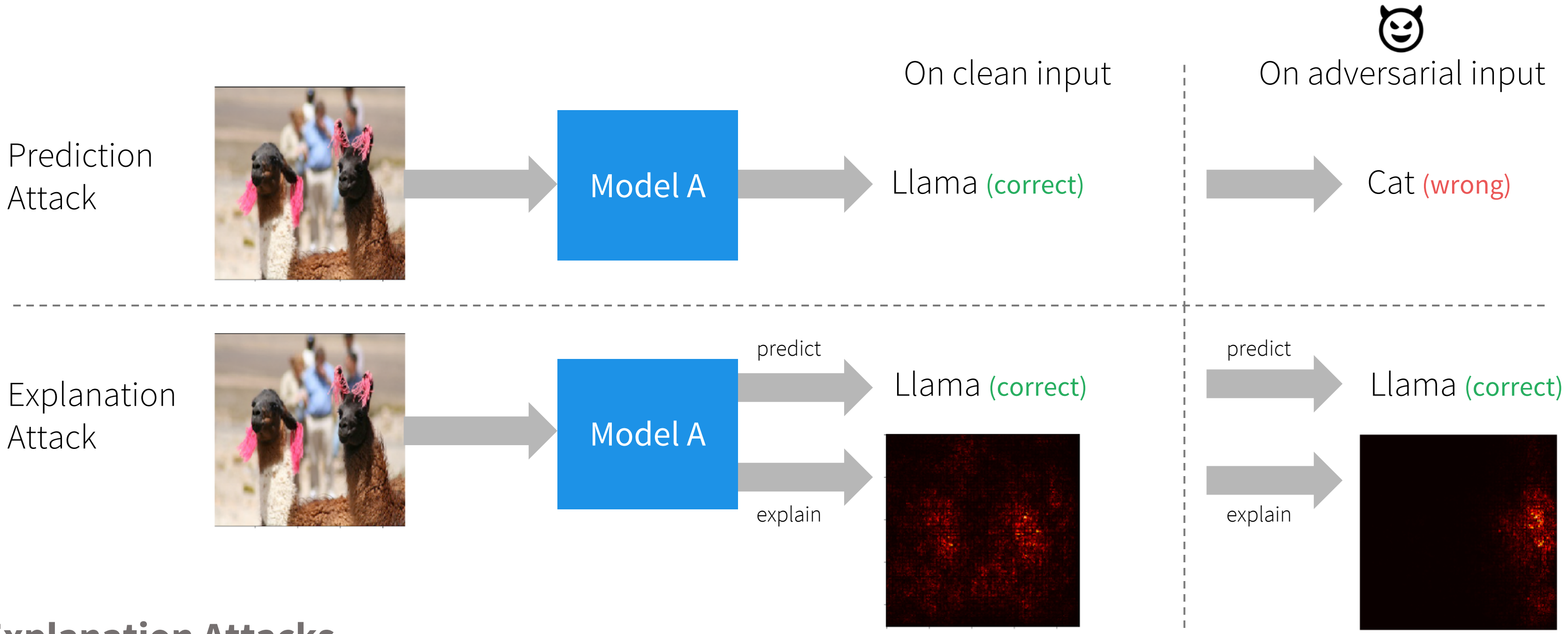
Adversarial Examples

Szegedy et al. 2014

*Goodfellow et al. 2015**

Papernot et al. 2016

Explanations can also be manipulated adversarially



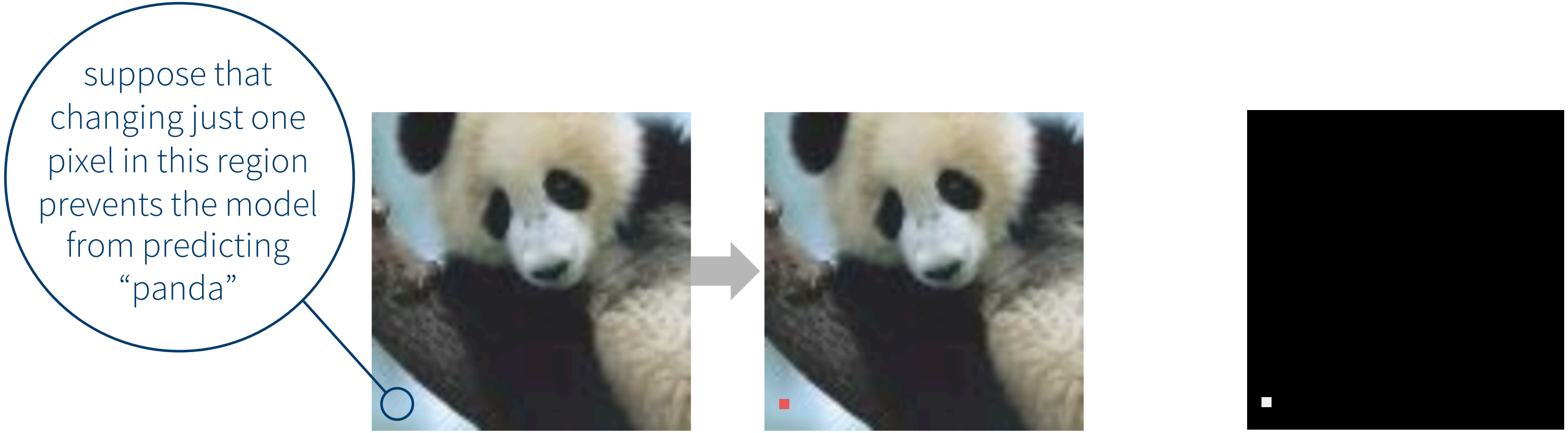
Explanation Attacks
*Ghorbani et al. AAAI 2019**
Dombrowski et al. NIPS 2019
Wang et al. NIPS 2020

! attribution map changes significantly

Can we trust explanations?

- If explanations can be manipulated, can we trust them?
- Is there something wrong with the explanation method that produces these anomalies?

Can we trust explanations?



“panda” not “panda” possible explanation

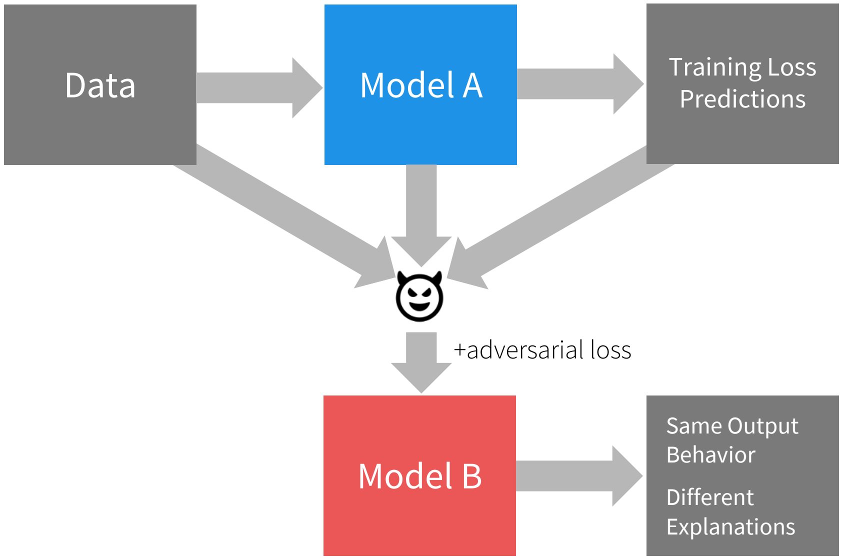
Is it really wrong to assign influence to the pixel that can be modified to change the model’s prediction?

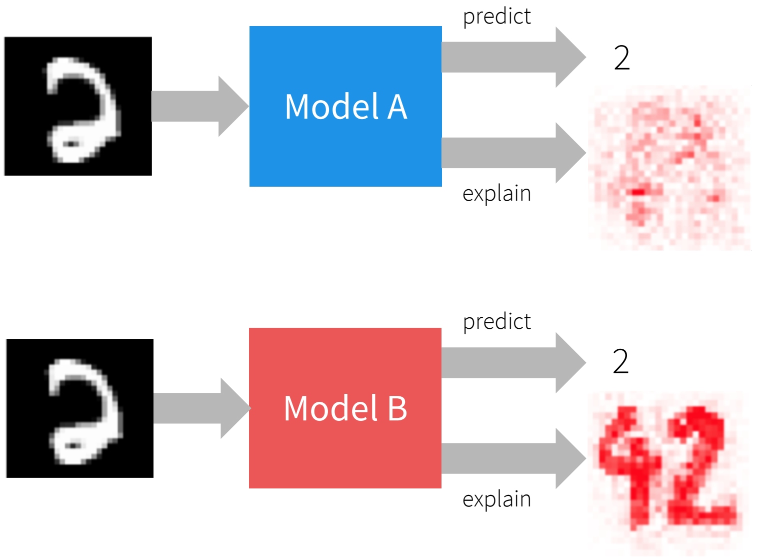
If it weren’t for this pixel, this point would not be classified as “panda”

Proposition

Key Idea
“bugs” in *faithful* explanations are evidence of model quality issues

Model-based attacks on explanations





Model-based Explanation Attacks
Anders et al. 2020

Now what?

- **Key Idea**: “bugs” in faithful explanations are evidence of model quality issues
- On well-behaved models, we shouldn’t see these anomalies
- How do we improve model quality?

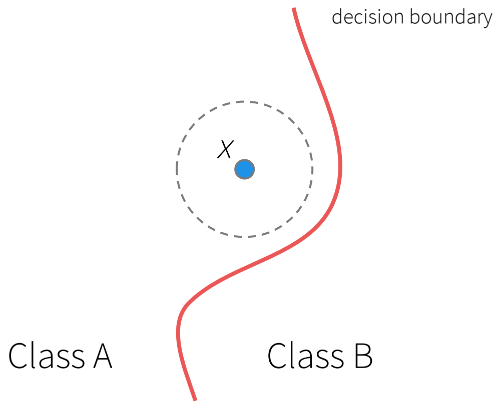
Local robustness

Definition

A model, F, is ϵ-*locally-robust* at x if $\forall x'$,

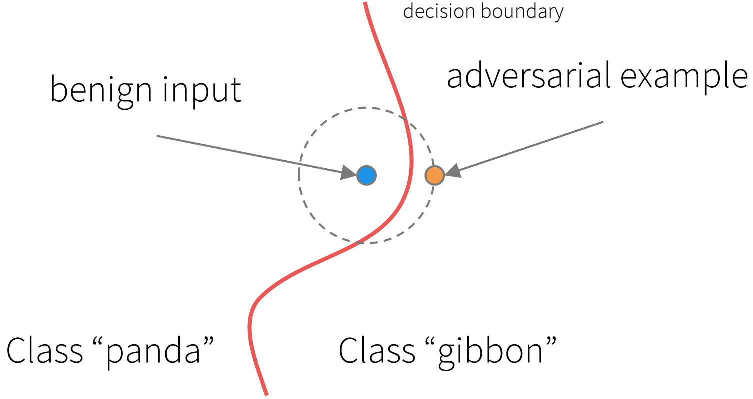
$$||x - x'|| \leq \epsilon \implies F(x) = F(x')$$

i.e., the model makes the same prediction on all points in the ε-ball centered at *x*



Adversarial examples are a violation of local robustness

“panda”

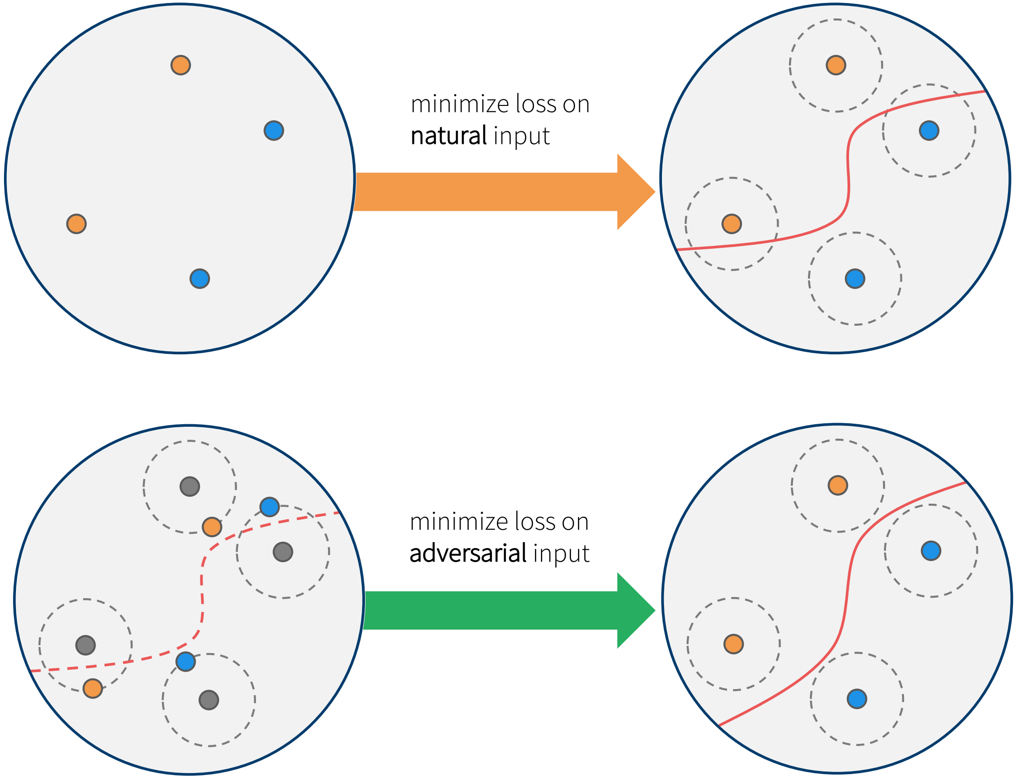


“gibbon”

Obtaining robust models

Standard Training

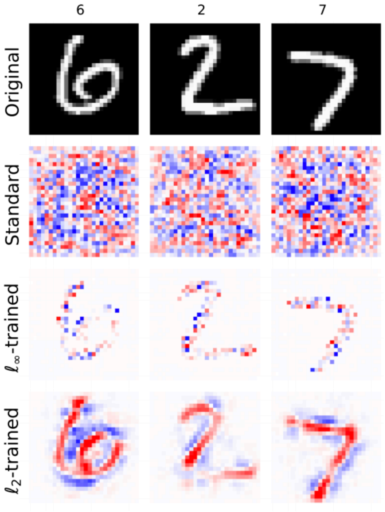
Adversarial Training

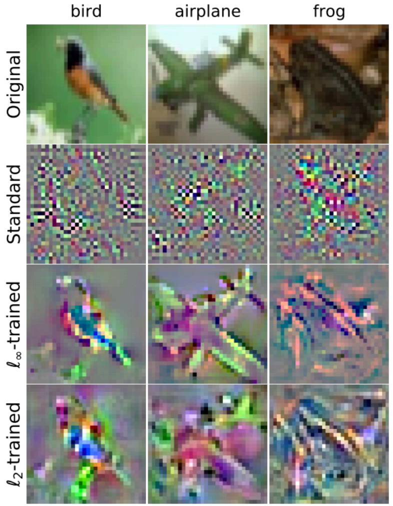


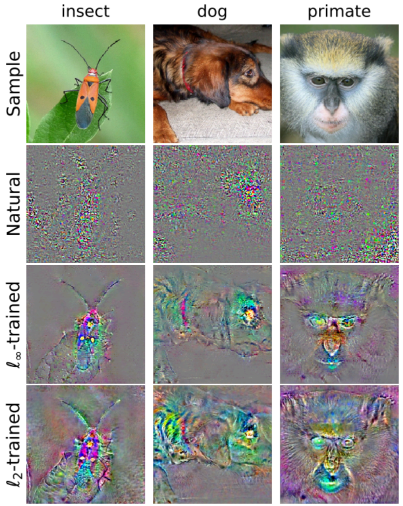
Adversarial Training
Madry et al. 2017

Robust models are more explainable

- Input gradients on robust models better align with the salient objects







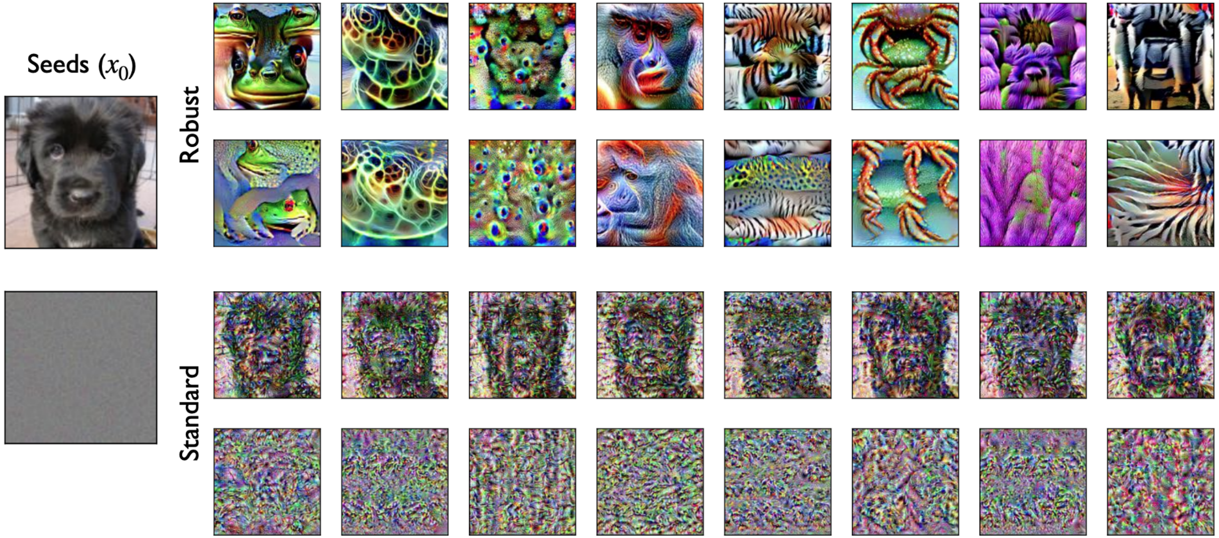
Explanations on Robust Models

*Tsipras et al. ICLR 2019**

Etmann et al. ICML 2019

Robust models are more explainable

- Feature visualization on robust models yields more recognizable results



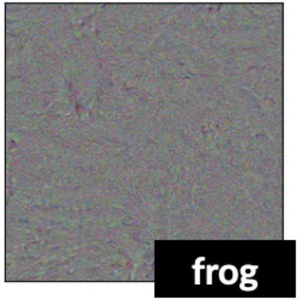
Feature Visualization
For classifier, f, and class , c, find δ that maximizes $f_c(x_0 + \delta)$

Visualizations on Robust Models
Tsipras et al. ICLR 2019

Why are robust models more explainable?

Hypothesis *(Ilyas et al. ICLR 2019)*
standard-trained models use *non-robust features* that are nonetheless predictive on the data distribution

example of non-robust features contained in an instance labeled “frog”



non-robust features only

Non-robust Features
Ilyas et al. ICLR 2019

Non-robust features

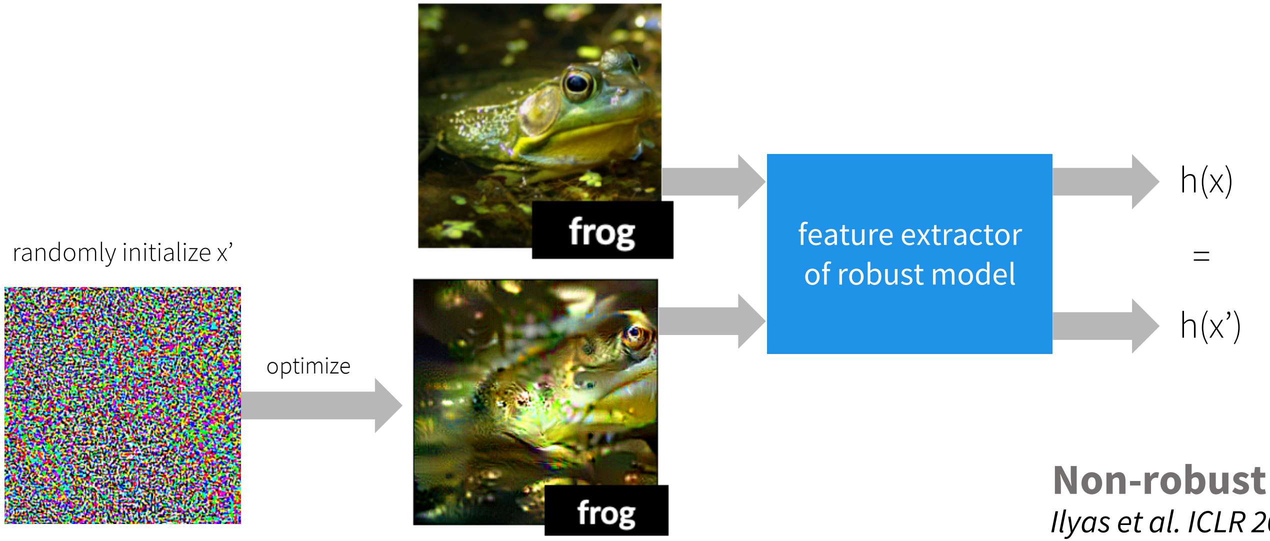
Definition
A *feature* is a neuron in a neural network,
which is a function, $f: \mathbb{R}^n \rightarrow \mathbb{R}$

Definition
A feature is *non-robust* on data points, (X, Y), if $f(X)$ correlates with Y,
but $f(X + \delta)$ does not correlate with Y for $||\delta|| \leq \epsilon$

Non-robust Features
Ilyas et al. ICLR 2019

Isolating robust features

- Non-robust features are not useful for a robust objective, thus we do not expect robust models to learn them (i.e., robust models should only learn robust features)



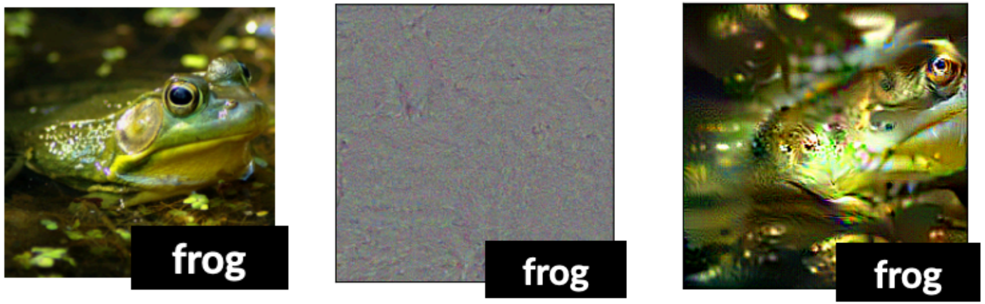
Non-robust Features
Ilyas et al. ICLR 2019

Why are robust models more explainable?

- Standard-trained models use *non-robust features* that are nonetheless predictive
- Non-robust features are not useful for a robust objective, thus we do not expect robust models to learn them
- Non-robust features are inherently less interpretable

not interpretable

more interpretable



non-robust features only

robust features only

Non-robust Features
Ilyas et al. ICLR 2019

Summary

"Bugs" in faithful explanations are evidence of model quality issues

Quality explanations require quality models

Robustness may be one way to achieve better model quality

Q & A

Thirty-Fifth AAAI Conference on Artificial Intelligence

From Explainability to Model Quality and Back Again

Anupam Datta, Matt Fredrikson, Klas Leino, Kaiji Lu, Shayak Sen and Zifan Wang

We appreciate your participance in this tutorial.

For More Resources:

- Tutorial Website
- Accountable Systems Lab
- TruLens and Demos
- Truera's Blog Posts on Explanability

Contact Us: shayak@truera.com, zifan@cmu.edu