eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea
FACULTY OF COMPUTER SCIENCE
UNIVERSITY OF THE BASQUE COUNTRY
Intelligent Systems Group

Broadening the Horizon of Adversarial Attacks in Deep Learning

Jon Vadillo

Work coauthored by Roberto Santana and Jose A. Lozano

April 2023

Overview

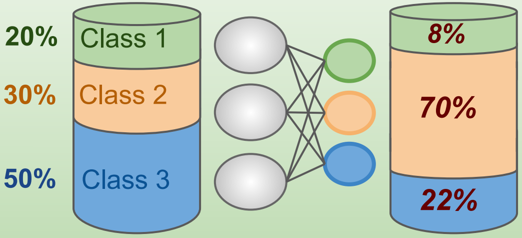
Conventional scenarios

Single instance scenarios

Classification models

Part 1

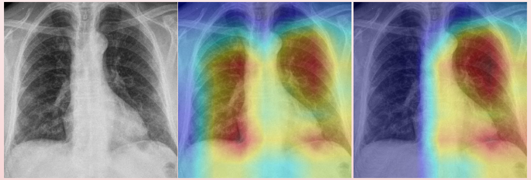
Multiple-instance attacks paradigms

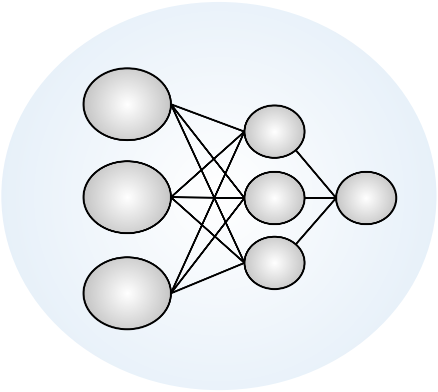


$\{x_1, \ldots, x_n\}$

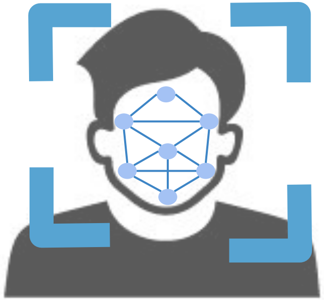
Part 2

Attacks against **explainable models**

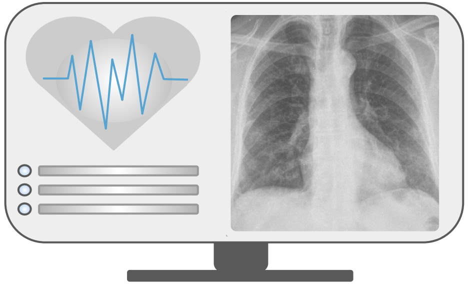




Deep Learning



Identity recognition



Healthcare

Speech Processing

Self-driving vehicles

Adversarial Examples

Prediction: *Police van*

Original Input

+

Adversarial Perturbation

=

Prediction: *Printer*

Adversarial Example

Adversarial Examples

Taxonomy

Type of
misclassification

Scope of the
perturbation

Resources available
to the adversary

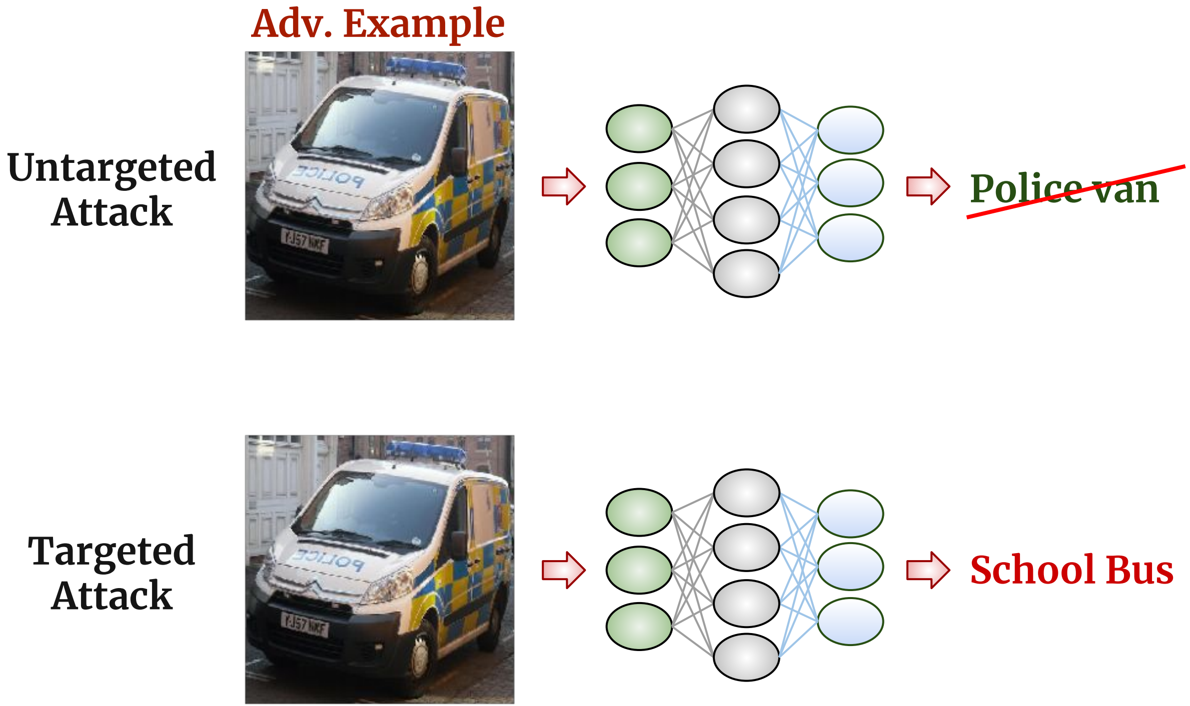
Adversarial Examples

Taxonomy

Type of misclassification

Scope of the perturbation

Resources available to the adversary

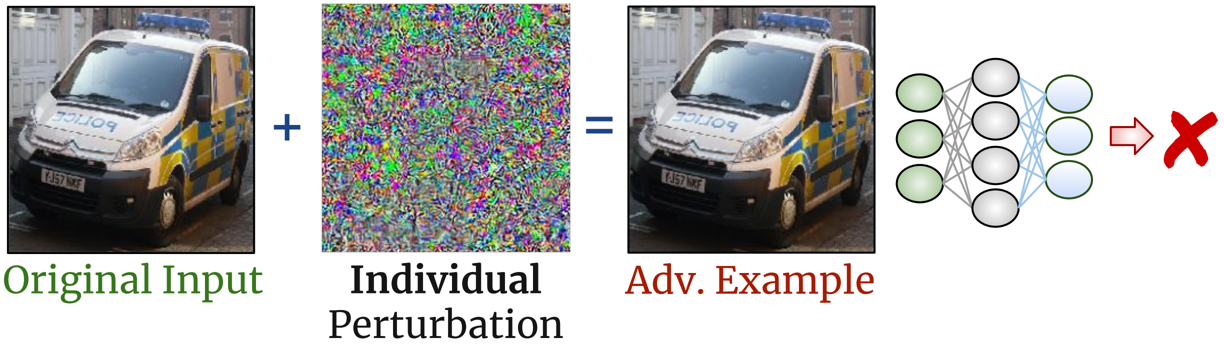


Adversarial Examples

Taxonomy

Type of misclassification

Scope of the perturbation



Resources available to the adversary

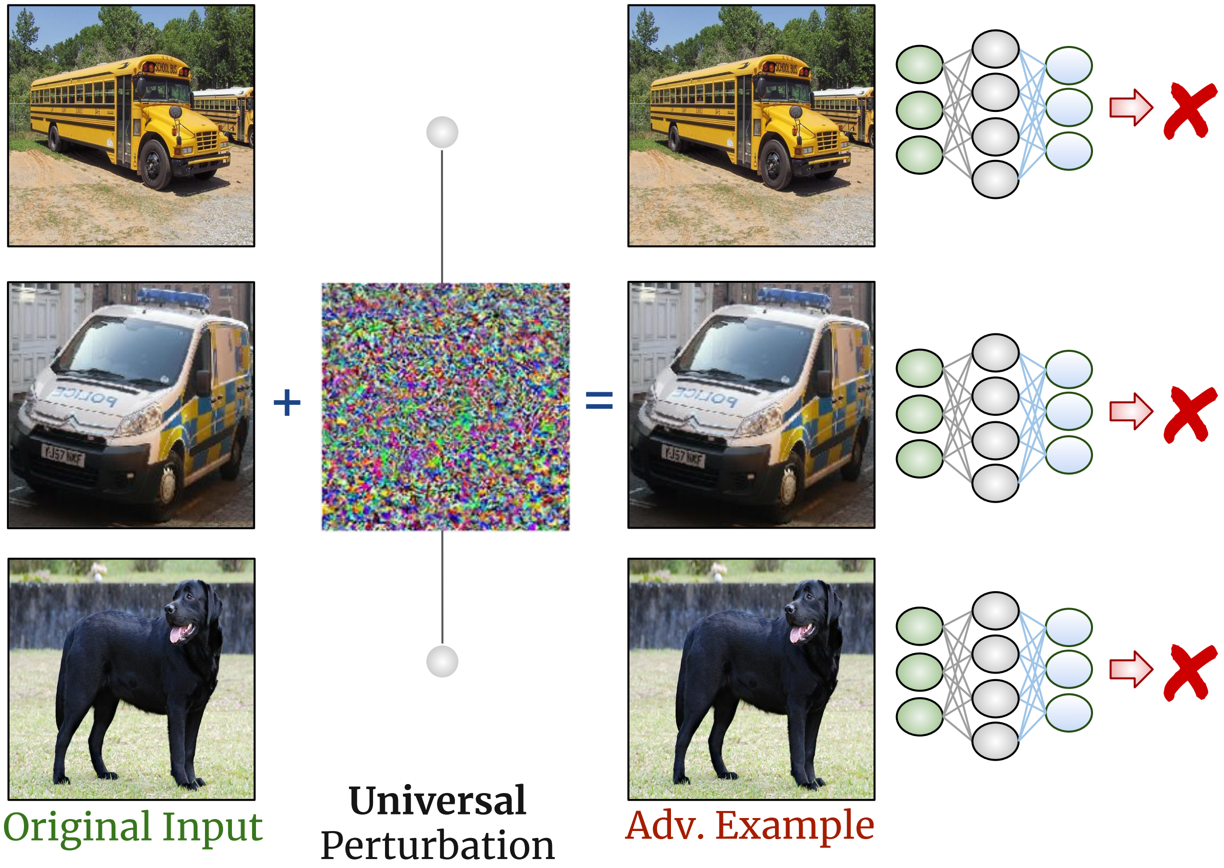
Adversarial Examples

Taxonomy

Type of misclassification

Scope of the perturbation

Resources available to the adversary



Adversarial Examples

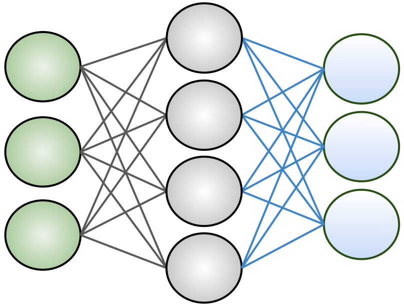
Taxonomy

Type of misclassification

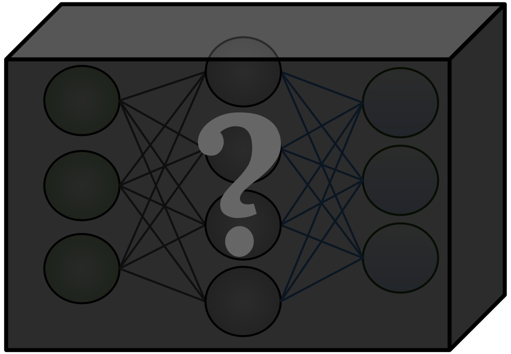
Scope of the perturbation

Resources available to the adversary

White-box scenario



Black-box scenario



Adversarial Examples

Taxonomy

Type of misclassification	**Untargeted** **Targeted**
Scope of the perturbation	**Individual** **Universal**
Resources available to the adversary	**Black-box** **White-box**

Adversarial Examples

Attack Methods

Fast Gradient Sign Method

$$x' = x + \epsilon \cdot \underbrace{\mathrm{sign}\big(\nabla \underbrace{\mathcal{L}(x, y_c, f)}_{\text{Prediction loss}}\big)}_{\text{Gradient sign}}$$

Where $f(x) = y_c$

$$f : \mathbb{R}^d \to \{y_1, y_2, \dots, y_k\}$$

(Goodfellow et al., 2014). Explaining and harnessing adversarial examples. ICLR

Adversarial Examples

Attack Methods

Projected Gradient Descent

$$x'_{[i+1]} = \underbrace{\mathcal{B}^x_\epsilon}_{\text{Projection operator}}\left(x'_{[i]} + \alpha \cdot \underbrace{\mathrm{sign}(\nabla \underbrace{\mathcal{L}(x'_{[i]}, y_c, f)}_{\text{Prediction loss}})}_{\text{Gradient sign}}\right)$$

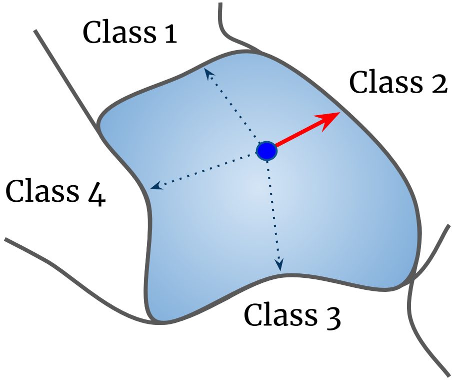
Where $f(x) = y_c$

$$f : \mathbb{R}^d \rightarrow \{y_1, y_2, \ldots, y_k\}$$

(Madry et al., 2018). Towards deep learning models resistant to adversarial attacks. ICLR

Generating Adversarial Examples

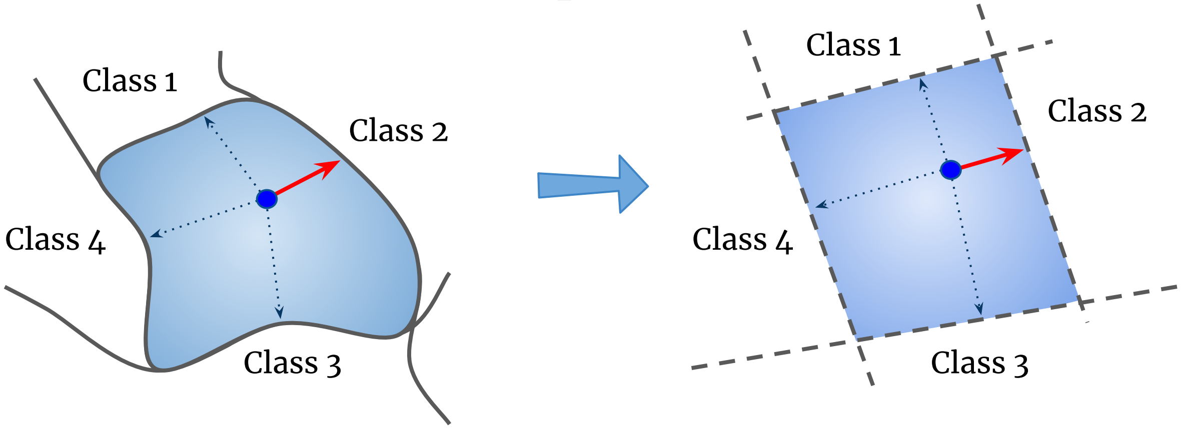
DeepFool



(Moosavi-Dezfooli et al., 2016). DeepFool: a simple and accurate method to fool deep neural networks. IEEE CVPR.

Generating Adversarial Examples

DeepFool



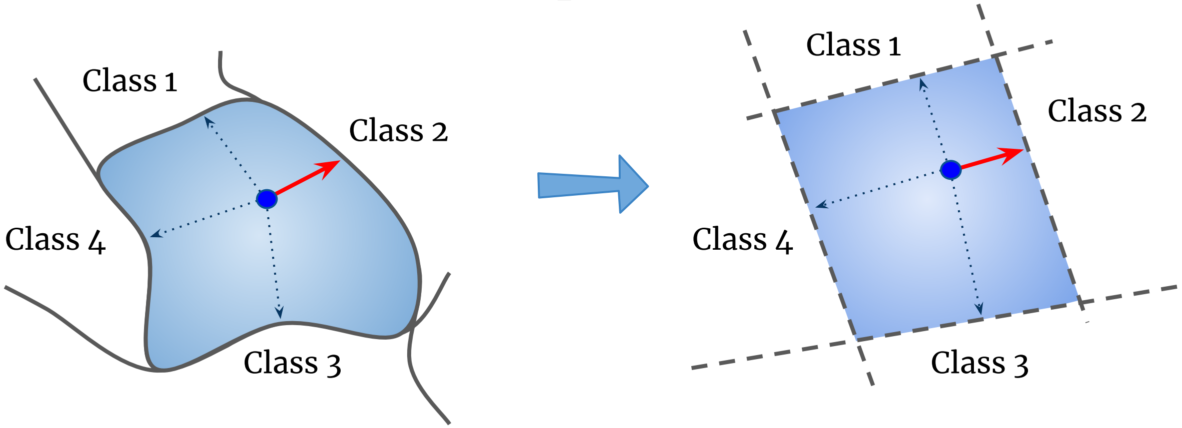
Source class: y_c

Boundary estimation (class y_j):

$$\textit{Distance:}\quad \frac{|f'_j|}{||w'_j||_2} = \frac{\left|\hat{f}(x'_{[i]})_j - \hat{f}(x'_{[i]})_c\right|}{||\underbrace{\nabla \hat{f}(x'_{[i]})_j - \nabla \hat{f}(x'_{[i]})_c}_{\textit{Direction}}||_2}$$

Generating Adversarial Examples

DeepFool



Source class: y_c

Boundary estimation (class y_j):

Distance: $$\frac{|f'_j|}{||w'_j||_2} = \frac{\left|\hat{f}(x'_{[i]})_j - \hat{f}(x'_{[i]})_c\right|}{\underbrace{||\nabla \hat{f}(x'_{[i]})_j - \nabla \hat{f}(x'_{[i]})_c||_2}_{\textit{Direction}}}$$

Closest boundary: $$l = \underset{j \neq c}{\operatorname{argmin}} \frac{|f'_j|}{||w'_j||_2}$$

Update rule: $$x'_{[i+1]} \leftarrow x'_{[i]} + \frac{|f'_l|}{||w'_l||_2^2} w'_l$$

Extending Adversarial Attacks to Produce Adversarial Class Probability Distributions

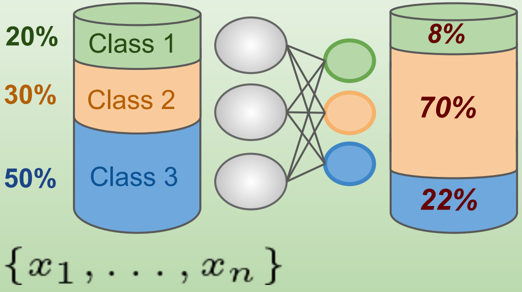
J. Vadillo, R. Santana, J. A. Lozano. (2023). **Journal of Machine Learning Research, volume 23, pp. 1-42.**

Single instance scenarios

Classification models

1

Multiple-instance attacks paradigms



2

Attacks against explainable models

Introduction

'Single-instance' **attack paradigm** Focus on individual inputs (isolatedly): x

Introduction

'Single-instance' **attack paradigm** Focus on individual inputs (isolatedly): x

'Multiple-instance' **attack paradigm** Consider multiple-inputs (coordinatedly): $\left\{x^{(1)},\ x^{(2)},\ \ldots,\ x^{(n)}\right\}$

Introduction

'Single-instance' **attack paradigm** Focus on individual inputs (isolatedly): x

'Multiple-instance' **attack paradigm** Consider multiple-inputs (coordinatedly): $\left\{x^{(1)},\ x^{(2)},\ \ldots,\ x^{(n)}\right\}$

Objective: Develop an attack method $\Phi(x)$ capable of:

1.

2.

Introduction

'Single-instance' **attack paradigm** Focus on individual inputs (isolatedly): x

'Multiple-instance' **attack paradigm** Consider multiple-inputs (coordinatedly): $\left\{x^{(1)},\ x^{(2)},\ \ldots,\ x^{(n)}\right\}$

Objective: Develop an attack method $\Phi(x)$ capable of:

1. Producing misclassifications: $f\left(\Phi(x)\right) \neq f(x)$

2.

Introduction

'Single-instance' **attack paradigm** Focus on individual inputs (isolatedly): x

'Multiple-instance' **attack paradigm** Consider multiple-inputs (coordinatedly): $\left\{x^{(1)},\ x^{(2)},\ \ldots,\ x^{(n)}\right\}$

Objective: Develop an attack method $\Phi(x)$ capable of:

1. Producing misclassifications: $f\left(\Phi(x)\right) \neq f(x)$
2. Controlling the frequency with which each class is predicted: $P_{x\sim\mathcal{P}(X)}\left[f\left(\Phi(x)\right)=y_i\right] = \tilde{p}_i,\ 1\leq i\leq k$

$$\widetilde{\mathcal{P}}(Y) = (\tilde{p}_1, \ldots, \tilde{p}_k)$$

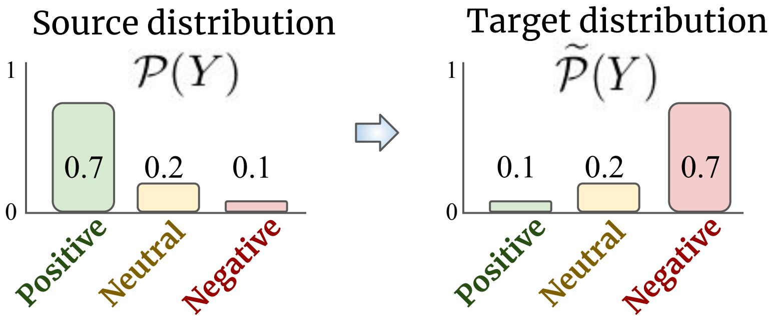
Target distribution of the output classes

Motivation

Representative use-cases:

1. **Aggregated predictions are highly relevant** (*quantification…*)
   a. **Collective information retrieval** (*opinion mining…*)
   b. **Prevalence of a disease** (*epidemiology…*)

Opinion mining:

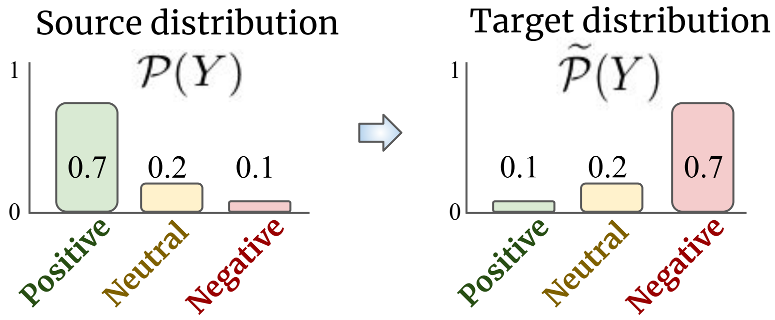


Motivation

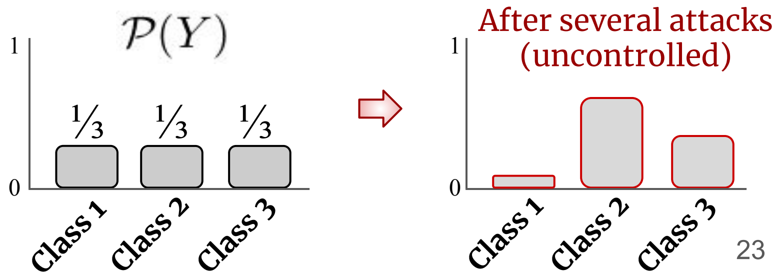
Representative use-cases:

1. **Aggregated predictions are highly relevant** (*quantification…*)
   a. **Collective information retrieval** (*opinion mining…*)
   b. **Prevalence of a disease** (*epidemiology…*)

Opinion mining:



2. **Fool the model several times preserving the source distribution**

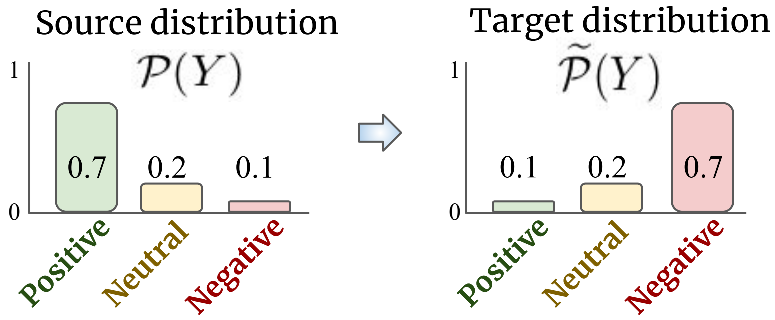


Motivation

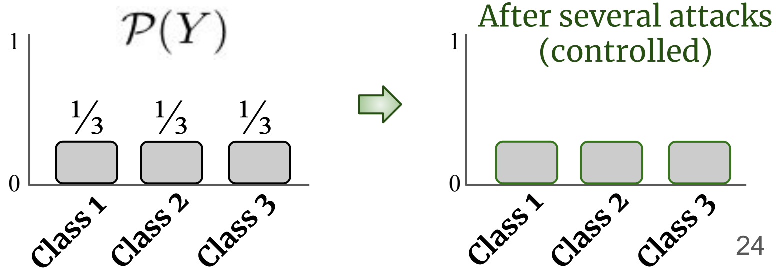
Representative use-cases:

1. **Aggregated predictions are highly relevant** (*quantification…*)
   a. **Collective information retrieval** (*opinion mining…*)
   b. **Prevalence of a disease** (*epidemiology…*)

Opinion mining:



2. **Fool the model several times preserving the source distribution**



Approach

Requirement: a targeted adversarial attack algorithm

Approach

Requirement: a targeted adversarial attack algorithm

Main objective:

$$T = \begin{pmatrix} t_{1,1} & t_{1,2} & \cdots & t_{1,k} \\ t_{2,1} & t_{2,2} & \cdots & t_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ t_{k,1} & t_{k,2} & \cdots & t_{k,k} \end{pmatrix}$$

Transition matrix

Approach

Requirement: a targeted adversarial attack algorithm

Main objective:

$$\underset{(p_1,\ldots,p_k)}{\mathcal{P}(Y)} \overset{T}{\begin{pmatrix} t_{1,1} & t_{1,2} & \cdots & t_{1,k} \\ t_{2,1} & t_{2,2} & \cdots & t_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ t_{k,1} & t_{k,2} & \cdots & t_{k,k} \end{pmatrix}} = \overset{\widetilde{\mathcal{P}}(Y)}{(\tilde{p}_1,\ldots,\tilde{p}_k)}$$

Source distribution

Transition matrix

Target distribution

Approach

Requirement: a targeted adversarial attack algorithm

Main objective:

$$\underbrace{\mathcal{P}(Y)}_{\substack{\text{Source}\\ \text{distribution}}} \; (p_1, \ldots, p_k) \underbrace{\overbrace{\begin{pmatrix} t_{1,1} & t_{1,2} & \cdots & t_{1,k} \\ t_{2,1} & t_{2,2} & \cdots & t_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ t_{k,1} & t_{k,2} & \cdots & t_{k,k} \end{pmatrix}}^{T}}_{\text{Transition matrix}} = \underbrace{\widetilde{\mathcal{P}}(Y)}_{\substack{\text{Target}\\ \text{distribution}}} \; (\tilde{p}_1, \ldots, \tilde{p}_k)$$

Bounded perturbation $||x' - x|| \leq \epsilon$

Approach

Requirement: a targeted adversarial attack algorithm

Main objective:

$$\underset{\text{Source distribution}}{\underset{(p_1,\ldots,p_k)}{\mathcal{P}(Y)}} \; \underset{\text{Transition matrix}}{\overset{T}{\begin{pmatrix} t_{1,1} & t_{1,2} & \cdots & t_{1,k} \\ t_{2,1} & t_{2,2} & \cdots & t_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ t_{k,1} & t_{k,2} & \cdots & t_{k,k} \end{pmatrix}}} = \underset{\text{Target distribution}}{\overset{\widetilde{\mathcal{P}}(Y)}{(\tilde{p}_1,\ldots,\tilde{p}_k)}}$$

Bounded perturbation $||x' - x|| \leq \epsilon$ ⇨ Some class transitions might not be feasible

Approach

Requirement: a targeted adversarial attack algorithm

Main objective:

$$\underset{\text{Source distribution}}{\underset{(p_1,\ldots,p_k)}{\mathcal{P}(Y)}} \underset{\text{Transition matrix}}{\overset{T}{\begin{pmatrix} t_{1,1} & t_{1,2} & \cdots & t_{1,k} \\ t_{2,1} & t_{2,2} & \cdots & t_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ t_{k,1} & t_{k,2} & \cdots & t_{k,k} \end{pmatrix}}} = \underset{\text{Target distribution}}{\underset{(\tilde{p}_1,\ldots,\tilde{p}_k)}{\widetilde{\mathcal{P}}(Y)}}$$

Source distribution | **Transition matrix** | **Target** distribution

Bounded perturbation $||x' - x|| \leq \epsilon$ ⇨ Some class transitions might not be feasible

Attack process:

Given:

x

$f(x) = y_i$

Approach

Requirement: a targeted adversarial attack algorithm

Main objective:

$$\underset{(p_1,\ldots,p_k)}{\mathcal{P}(Y)} \overset{T}{\begin{pmatrix} t_{1,1} & t_{1,2} & \cdots & t_{1,k} \\ t_{2,1} & t_{2,2} & \cdots & t_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ t_{k,1} & t_{k,2} & \cdots & t_{k,k} \end{pmatrix}} = \underset{(\tilde{p}_1,\ldots,\tilde{p}_k)}{\widetilde{\mathcal{P}}(Y)}$$

Source distribution | **Transition matrix** | **Target** distribution

Bounded perturbation $||x' - x|| \leq \epsilon$ ⇨ Some class transitions might not be feasible

Attack process:

Given: x, $f(x) = y_i$ ⇨ 1. Compute the set of "reachable" classes $\mathcal{Y}$

Approach

Requirement: a targeted adversarial attack algorithm

Main objective:

$$\underbrace{\mathcal{P}(Y)}_{(p_1, \ldots, p_k)} \overbrace{\begin{pmatrix} t_{1,1} & t_{1,2} & \cdots & t_{1,k} \\ t_{2,1} & t_{2,2} & \cdots & t_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ t_{k,1} & t_{k,2} & \cdots & t_{k,k} \end{pmatrix}}^{T} = \underbrace{\widetilde{\mathcal{P}}(Y)}_{(\tilde{p}_1, \ldots, \tilde{p}_k)}$$

Source distribution — **Transition matrix** — **Target** distribution

Bounded perturbation $||x' - x|| \leq \epsilon$ ⇨ Some class transitions might not be feasible

Attack process:

Given: x, $f(x) = y_i$ ⇨ 1. Compute the set of "reachable" classes $\mathcal{Y}$ ⇨ $(t_{i,1}, \ldots, t_{i,k})$

Approach

Requirement: a targeted adversarial attack algorithm

Main objective:

$$\underset{\text{(Source distribution)}}{\overset{\mathcal{P}(Y)}{(p_1, \dots, p_k)}} \overset{T}{\begin{pmatrix} t_{1,1} & t_{1,2} & \cdots & t_{1,k} \\ t_{2,1} & t_{2,2} & \cdots & t_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ t_{k,1} & t_{k,2} & \cdots & t_{k,k} \end{pmatrix}} = \overset{\widetilde{\mathcal{P}}(Y)}{(\tilde{p}_1, \dots, \tilde{p}_k)}$$

Source distribution | **Transition matrix** | **Target** distribution

Bounded perturbation $||x' - x|| \leq \epsilon$ ⇨ Some class transitions might not be feasible

Attack process:

Given:

x

$f(x) = y_i$

⇨ 1. Compute the set of "reachable" classes

$\mathcal{Y}$

⇨ 2. Normalize the probabilities

$(t_{i,1}, \dots, t_{i,k})$

$$t'_{i,j} = \begin{cases} \frac{t_{i,j}}{\sum_{y_r \in \mathcal{Y}} t_{i,r}} & \textit{if } y_j \in \mathcal{Y} \\ 0 & \text{else.} \end{cases}$$

Approach

Requirement: a targeted adversarial attack algorithm

Main objective:

$$\underset{(p_1,\ldots,p_k)}{\mathcal{P}(Y)} \overset{T}{\begin{pmatrix} t_{1,1} & t_{1,2} & \cdots & t_{1,k} \\ t_{2,1} & t_{2,2} & \cdots & t_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ t_{k,1} & t_{k,2} & \cdots & t_{k,k} \end{pmatrix}} = \overset{\widetilde{\mathcal{P}}(Y)}{(\tilde{p}_1,\ldots,\tilde{p}_k)}$$

Source distribution | **Transition matrix** | **Target** distribution

Bounded perturbation $||x' - x|| \leq \epsilon$ ⇨ Some class transitions might not be feasible

Attack process:

Given: x, $f(x) = y_i$ ⇨ 1. Compute the set of “reachable” classes $\mathcal{Y}$ ⇨ 2. Normalize the probabilities $(t_{i,1},\ldots,t_{i,k})$

$$t'_{i,j} = \begin{cases} \frac{t_{i,j}}{\sum_{y_r \in \mathcal{Y}} t_{i,r}} & \text{if } y_j \in \mathcal{Y} \\ 0 & \text{else.} \end{cases}$$

⇨ 3. Select a target class $(t'_{i,1},\ldots,t'_{i,k})$

Generating transition matrices

$$\min \quad z = \sum_{i=1}^{k} t_{i,i}$$

$$\underbrace{\begin{aligned} \text{s.t.} \quad & \sum_{j=1}^{k} t_{i,j} = 1 && \forall i \in \{1, \ldots, k\} \\ & 0 \leq t_{i,j} \leq 1 && \forall i, j \in \{1, \ldots, k\} \end{aligned}}_{\text{T is a transition matrix}}$$

$$\mathcal{P}(Y) \cdot T = \widetilde{\mathcal{P}}(Y)$$

Generating transition matrices

$$\min \quad z = \sum_{i=1}^{k} t_{i,i}$$

$$\begin{aligned}
\text{s.t.} \quad & \sum_{j=1}^{k} t_{i,j} = 1 && \forall i \in \{1, \dots, k\} \\
& 0 \le t_{i,j} \le 1 && \forall i, j \in \{1, \dots, k\} \\
& \mathcal{P}(Y) \cdot T = \widetilde{\mathcal{P}}(Y)
\end{aligned}$$

T is a transition matrix

T produces the target distribution

Generating transition matrices

$$\min \quad z = \sum_{i=1}^{k} t_{i,i}$$

Maximize the fooling rate

$$\text{s.t.} \quad \sum_{j=1}^{k} t_{i,j} = 1 \qquad \forall i \in \{1, \ldots, k\}$$

$$0 \leq t_{i,j} \leq 1 \qquad \forall i, j \in \{1, \ldots, k\}$$

T is a transition matrix

$$\mathcal{P}(Y) \cdot T = \widetilde{\mathcal{P}}(Y)$$

T produces the target distribution

Generating transition matrices

$$\min \quad z = \sum_{i=1}^{k} t_{i,i}$$

Maximize the fooling rate

$$\text{s.t.} \quad \sum_{j=1}^{k} t_{i,j} = 1 \qquad \forall i \in \{1, \ldots, k\}$$

$$0 \leq t_{i,j} \leq 1 \qquad \forall i, j \in \{1, \ldots, k\}$$

T is a transition matrix

$$\mathcal{P}(Y) \cdot T = \widetilde{\mathcal{P}}(Y)$$

T produces the target distribution

Different solutions might produce different results in practice

Generating transition matrices

$$\min \quad z = \sum_{i=1}^{k} t_{i,i}$$

Maximize the fooling rate

$$\text{s.t.} \quad \sum_{j=1}^{k} t_{i,j} = 1 \qquad \forall i \in \{1, \ldots, k\}$$

$$0 \leq t_{i,j} \leq 1 \qquad \forall i, j \in \{1, \ldots, k\}$$

T is a transition matrix

$$\mathcal{P}(Y) \cdot T = \widetilde{\mathcal{P}}(Y)$$

T produces the target distribution

Different solutions might produce different results in practice

Additional constraints to include information about the problem

Four different methods proposed (+2 baselines)

Creating more informed transition matrices

Example: Upper-Bound Method (UBM)

Intuition: Prioritize those transitions that are feasible with higher frequency.

N inputs per class

$$\Rightarrow R = \begin{pmatrix} r_{1,1} & r_{1,2} & \ldots, & r_{1,k} \\ r_{2,1} & r_{2,2} & \ldots, & r_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ r_{k,1} & r_{k,2} & \ldots, & r_{k,k} \end{pmatrix} \qquad \Rightarrow R' = \frac{1}{N} R$$

$r_{i,j}$: Number of samples that can be moved from the class ***i*** to the class ***j***

$r'_{i,j}$: **"Maximum proportion" for each transition**

Upper bound for the highest probability:

$$t_{i,j} \leq r'_{i,j} \qquad \forall i, j \in \{1, \ldots, k\}$$

Creating more informed transition matrices

Example: Upper-Bound Method (UBM)

$$\min \quad z = \sum_{i=1}^{k} t_{i,i}$$

$$\left.\begin{array}{lr} \text{s.t.} \quad \sum_{j=1}^{k} t_{i,j} = 1 & \forall i \in \{1, \ldots, k\} \\ \quad 0 \leq t_{i,j} \leq 1 & \forall i, j \in \{1, \ldots, k\} \end{array}\right\} \text{T is a transition matrix}$$

$$\left.\mathcal{P}(Y) \cdot T = \widetilde{\mathcal{P}}(Y)\right\} \text{T produces the target distribution}$$

Creating more informed transition matrices

Example: Upper-Bound Method (UBM)

$$\min \quad z \quad = \quad \sum_{i=1}^{k} t_{i,i}$$

$$\left.\begin{aligned} \text{s.t.} \quad & \sum_{j=1}^{k} t_{i,j} = 1 && \forall i \in \{1, \ldots, k\} \\ & 0 \leq t_{i,j} \leq 1 && \forall i, j \in \{1, \ldots, k\} \end{aligned}\right\} \text{T is a transition matrix}$$

$$\left.\mathcal{P}(Y) \cdot T = \widetilde{\mathcal{P}}(Y)\right\} \text{T produces the target distribution}$$

$$\left.t_{i,j} \leq r'_{i,j} \quad \forall i, j \in \{1, \ldots, k\}\right\} \textbf{Avoid “excessively high” probabilities}$$

Creating more informed transition matrices

Example: Upper-Bound Method (UBM)

$$\min \quad z = \sum_{i=1}^{k} t_{i,i} + \sum_{i=1}^{k}\sum_{j=1}^{k} \eta_{i,j}$$

$$\left.\begin{aligned} \text{s.t.} \quad & \sum_{j=1}^{k} t_{i,j} = 1 && \forall i \in \{1,\dots,k\} \\ & 0 \le t_{i,j} \le 1 && \forall i,j \in \{1,\dots,k\} \end{aligned}\right\} \text{T is a transition matrix}$$

$$\left.\mathcal{P}(Y)\cdot T = \widetilde{\mathcal{P}}(Y)\right\} \text{T produces the target distribution}$$

$$\left. t_{i,j} \le r'_{i,j} + \eta_{i,j} \qquad \forall i,j \in \{1,\dots,k\}\right\} \textbf{Avoid “excessively high” probabilities}$$

Creating more informed transition matrices

Example: Upper-Bound Method (UBM)

$$\min \quad z = \sum_{i=1}^{k} t_{i,i} + \sum_{i=1}^{k}\sum_{j=1}^{k} \eta_{i,j}$$

$$\left.\begin{array}{lll} \text{s.t.} & \sum_{j=1}^{k} t_{i,j} = 1 & \forall i \in \{1,\ldots,k\} \\ & 0 \leq t_{i,j} \leq 1 & \forall i,j \in \{1,\ldots,k\} \end{array}\right\} \text{T is a transition matrix}$$

$$\left.\mathcal{P}(Y) \cdot T = \widetilde{\mathcal{P}}(Y)\right\} \text{T produces the target distribution}$$

$$\left.t_{i,j} \leq r'_{i,j} + \eta_{i,j} \qquad \forall i,j \in \{1,\ldots,k\}\right\} \textbf{Avoid “excessively high” probabilities}$$

$$\left.\begin{array}{ll} t_{i,j} \geq l_{i,j} & \forall i,j \in \{1,\ldots,k\}, i \neq j \\ 0 \leq l_{i,j} \leq \xi & \forall i,j \in \{1,\ldots,k\} \end{array}\right\} \text{Avoid null probabilities}$$

Creating more informed transition matrices

Example: Upper-Bound Method (UBM)

$$\min \quad z = \sum_{i=1}^{k} t_{i,i} + \sum_{i=1}^{k}\sum_{j=1}^{k} \eta_{i,j} - \sum_{i=1}^{k}\sum_{\substack{j=1\\ j\neq i}}^{k} l_{i,j}$$

$$\begin{aligned}
\text{s.t.} \quad & \sum_{j=1}^{k} t_{i,j} = 1 && \forall i \in \{1,\ldots,k\} \\
& 0 \leq t_{i,j} \leq 1 && \forall i,j \in \{1,\ldots,k\}
\end{aligned} \quad \Bigg\} \text{ T is a transition matrix}$$

$$\mathcal{P}(Y) \cdot T = \widetilde{\mathcal{P}}(Y) \quad \Big\} \text{ T produces the target distribution}$$

$$t_{i,j} \leq r'_{i,j} + \eta_{i,j} \qquad \forall i,j \in \{1,\ldots,k\} \quad \Big\} \textbf{ Avoid ``excessively high'' probabilities}$$

$$\begin{aligned}
& t_{i,j} \geq l_{i,j} && \forall i,j \in \{1,\ldots,k\}, i \neq j \\
& 0 \leq l_{i,j} \leq \xi && \forall i,j \in \{1,\ldots,k\}
\end{aligned} \quad \Bigg\} \text{ Avoid null probabilities}$$

Main results and conclusions

Evaluation:

➢ 2 classification problems (**speech commands**, TSA):

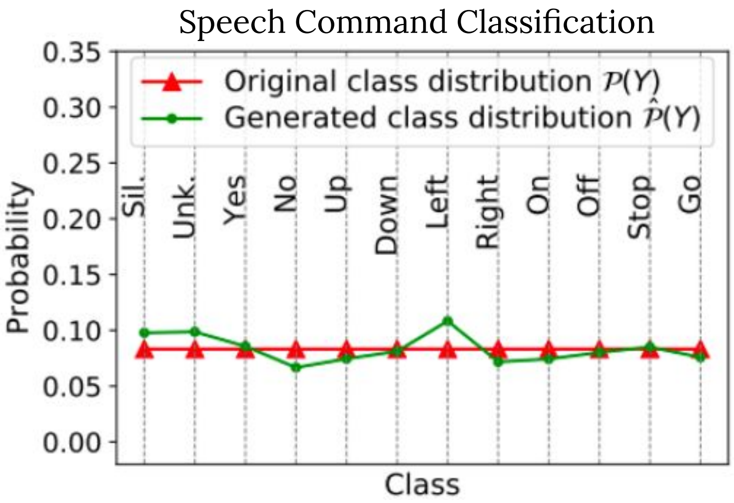
Speech Command Classification
12 classes: {“*yes*”, “*no*”, “*stop*”, “*go*”...}

Tweet Sentyment Analysis
6 classes: {*anger*, *fear*, *joy*...}

Main results and conclusions

Evaluation:

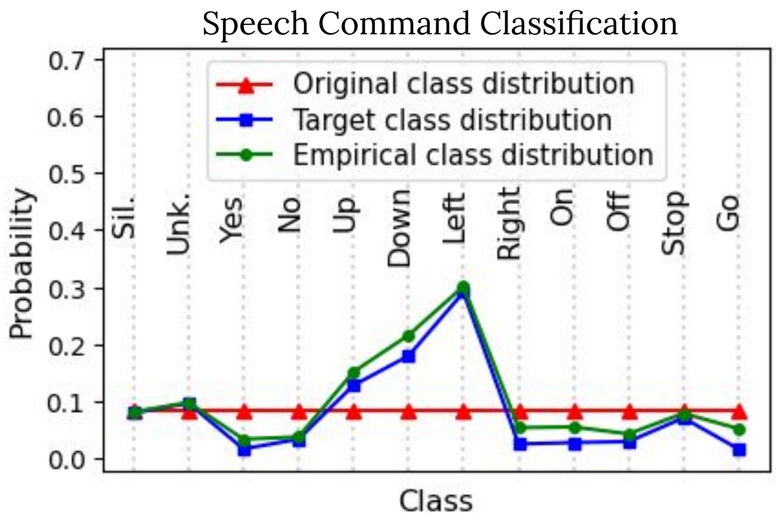
➢ 2 classification problems (**speech commands**, TSA)
➢ Different setups for the target distribution (**original**, random...)



Main results and conclusions

Evaluation:

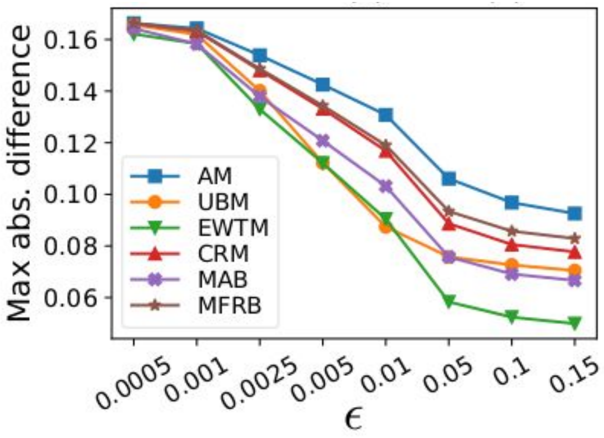
- 2 classification problems (**speech commands**, TSA)
- Different setups for the target distribution (original, **random**...)



Main results and conclusions

Evaluation:

- ➢ 2 classification problems (**speech commands**, TSA)
- ➢ Different setups for the target distribution (original, **random**...)
- ➢ Multifactorial (fooling rate, KL-divergence, correlation...)



Fooling rate (%)

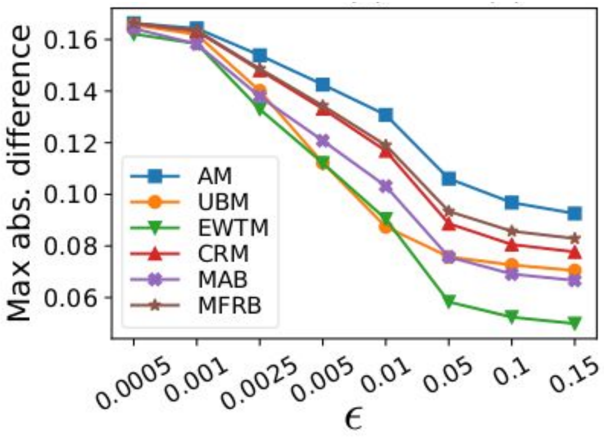
		Maximum distortion amount (ϵ)							
		0.0005	0.001	0.0025	0.005	0.01	0.05	0.1	0.15
Methods	AM	3.80	11.17	31.58	46.98	62.36	87.29	92.31	94.69
	UBM	0.45	2.88	19.06	38.03	57.89	87.05	92.28	94.68
	EWTM	1.88	6.87	23.59	38.65	53.60	79.66	85.21	87.84
	CRM	3.90	11.29	31.55	46.88	62.23	87.26	92.31	94.70
Baselines	MAB	2.06	6.55	21.33	33.72	46.87	71.02	76.96	79.64
	MFRB	3.93	11.47	32.02	47.44	62.80	87.54	92.48	94.86
	Max. FR	3.93	11.47	32.02	47.44	62.80	87.54	92.48	94.86

Results averaged for 100 random target distributions.

Main results and conclusions

Evaluation:

- ➢ 2 classification problems (**speech commands**, TSA)
- ➢ Different setups for the target distribution (original, **random**...)
- ➢ Multifactorial (fooling rate, KL-divergence, correlation...)
- ➢ Multiple adversarial attack algorithms as component



Fooling rate (%)

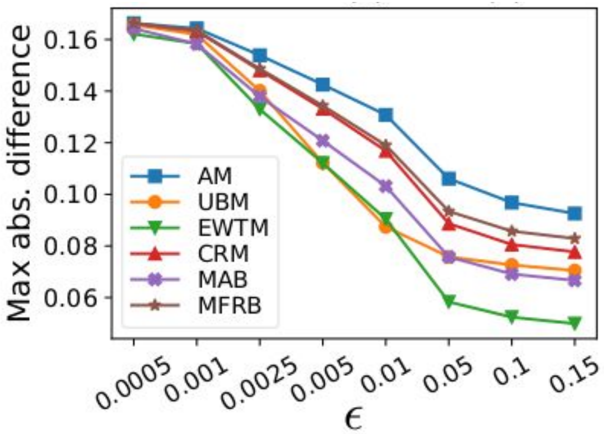
		Maximum distortion amount (ϵ)							
		0.0005	0.001	0.0025	0.005	0.01	0.05	0.1	0.15
Methods	AM	3.80	11.17	31.58	46.98	62.36	87.29	92.31	94.69
	UBM	0.45	2.88	19.06	38.03	57.89	87.05	92.28	94.68
	EWTM	1.88	6.87	23.59	38.65	53.60	79.66	85.21	87.84
	CRM	3.90	11.29	31.55	46.88	62.23	87.26	92.31	94.70
Baselines	MAB	2.06	6.55	21.33	33.72	46.87	71.02	76.96	79.64
	MFRB	3.93	11.47	32.02	47.44	62.80	87.54	92.48	94.86
	Max. FR	3.93	11.47	32.02	47.44	62.80	87.54	92.48	94.86

Results averaged for 100 random target distributions.

Main results and conclusions

Evaluation:

- ➢ 2 classification problems (**speech commands**, TSA)
- ➢ Different setups for the target distribution (original, **random**...)
- ➢ Multifactorial (fooling rate, KL-divergence, correlation...)
- ➢ Multiple adversarial attack algorithms as component



Our methods were capable of:

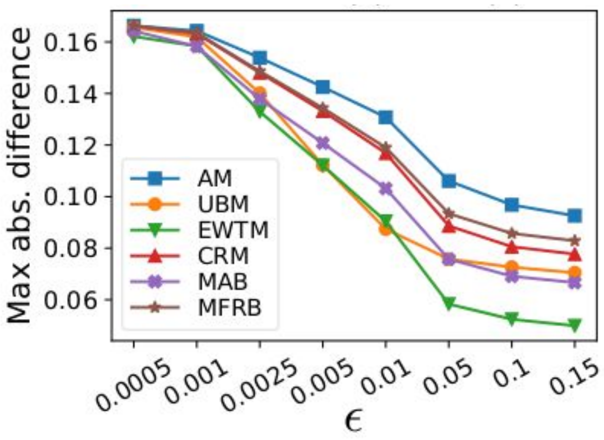
- ➢ **Closely approximating the target distributions**
- ➢ **Maintain a high fooling rate**

Fooling rate (%)

		Maximum distortion amount (ϵ)							
		0.0005	0.001	0.0025	0.005	0.01	0.05	0.1	0.15
Methods	AM	3.80	11.17	31.58	46.98	62.36	87.29	92.31	94.69
	UBM	0.45	2.88	19.06	38.03	57.89	87.05	92.28	94.68
	EWTM	1.88	6.87	23.59	38.65	53.60	79.66	85.21	87.84
	CRM	3.90	11.29	31.55	46.88	62.23	87.26	92.31	94.70
Baselines	MAB	2.06	6.55	21.33	33.72	46.87	71.02	76.96	79.64
	MFRB	3.93	11.47	32.02	47.44	62.80	87.54	92.48	94.86
	Max. FR	3.93	11.47	32.02	47.44	62.80	87.54	92.48	94.86

Results averaged for 100 random target distributions.

Main results and conclusions



Evaluation:

- ➢ 2 classification problems (**speech commands**, TSA)
- ➢ Different setups for the target distribution (original, **random**…)
- ➢ Multifactorial (fooling rate, KL-divergence, correlation…)
- ➢ Multiple adversarial attack algorithms as component

Our methods were capable of:

- ➢ Closely approximating the target distributions
- ➢ Maintain a high fooling rate

No *"best method"* for all the factors considered:

- ➢ **Fooling rate vs Similarity**

Fooling rate (%)

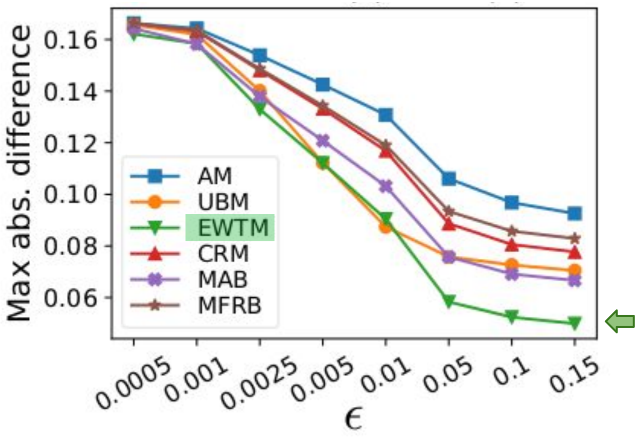
	Maximum distortion amount (ϵ)							
	0.0005	0.001	0.0025	0.005	0.01	0.05	0.1	0.15
AM	3.80	11.17	31.58	46.98	62.36	87.29	92.31	94.69
UBM	0.45	2.88	19.06	38.03	57.89	87.05	92.28	94.68
EWTM	1.88	6.87	23.59	38.65	53.60	79.66	85.21	87.84
CRM	3.90	11.29	31.55	46.88	62.23	87.26	92.31	94.70
MAB	2.06	6.55	21.33	33.72	46.87	71.02	76.96	79.64
MFRB	3.93	11.47	32.02	47.44	62.80	87.54	92.48	94.86
Max. FR	3.93	11.47	32.02	47.44	62.80	87.54	92.48	94.86

Results averaged for 100 random target distributions.

Main results and conclusions

Evaluation:

- ➢ 2 classification problems (**speech commands**, TSA)
- ➢ Different setups for the target distribution (original, **random**...)
- ➢ Multifactorial (fooling rate, KL-divergence, correlation...)
- ➢ Multiple adversarial attack algorithms as component



Our methods were capable of:

- ➢ Closely approximating the target distributions
- ➢ Maintain a high fooling rate

Fooling rate (%)

	Maximum distortion amount (ϵ)							
	0.0005	0.001	0.0025	0.005	0.01	0.05	0.1	0.15
AM	3.80	11.17	31.58	46.98	62.36	87.29	92.31	94.69
UBM	0.45	2.88	19.06	38.03	57.89	87.05	92.28	94.68
EWTM	1.88	6.87	23.59	38.65	53.60	79.66	85.21	87.84
CRM	3.90	11.29	31.55	46.88	62.23	87.26	92.31	94.70
MAB	2.06	6.55	21.33	33.72	46.87	71.02	76.96	79.64
MFRB	3.93	11.47	32.02	47.44	62.80	87.54	92.48	94.86
Max. FR	3.93	11.47	32.02	47.44	62.80	87.54	92.48	94.86

No *"best method"* for all the factors considered:

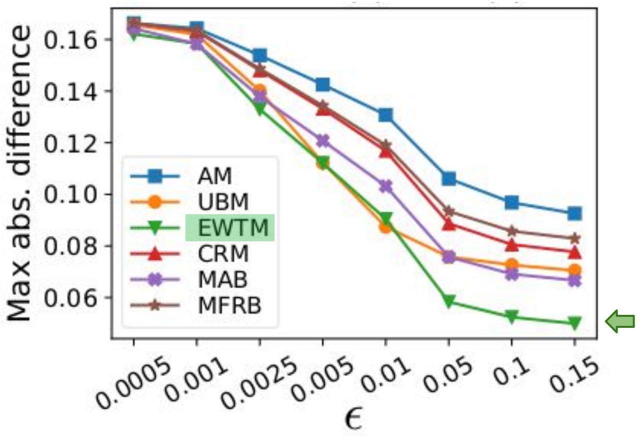
- ➢ **Fooling rate vs Similarity**

Results averaged for 100 random target distributions.

Main results and conclusions

Evaluation:

- 2 classification problems (**speech commands**, TSA)
- Different setups for the target distribution (original, **random**...)
- Multifactorial (fooling rate, KL-divergence, correlation...)
- Multiple adversarial attack algorithms as component



Our methods were capable of:

- Closely approximating the target distributions
- Maintain a high fooling rate

Fooling rate (%)

	Maximum distortion amount (ϵ)							
	0.0005	0.001	0.0025	0.005	0.01	0.05	0.1	0.15
AM	3.80	11.17	31.58	46.98	62.36	87.29	92.31	94.69
UBM	0.45	2.88	19.06	38.03	57.89	87.05	92.28	94.68
EWTM	1.88	6.87	23.59	38.65	53.60	79.66	85.21	87.84
CRM	3.90	11.29	31.55	46.88	62.23	87.26	92.31	94.70
MAB	2.06	6.55	21.33	33.72	46.87	71.02	76.96	79.64
MFRB	3.93	11.47	32.02	47.44	62.80	87.54	92.48	94.86
Max. FR	3.93	11.47	32.02	47.44	62.80	87.54	92.48	94.86

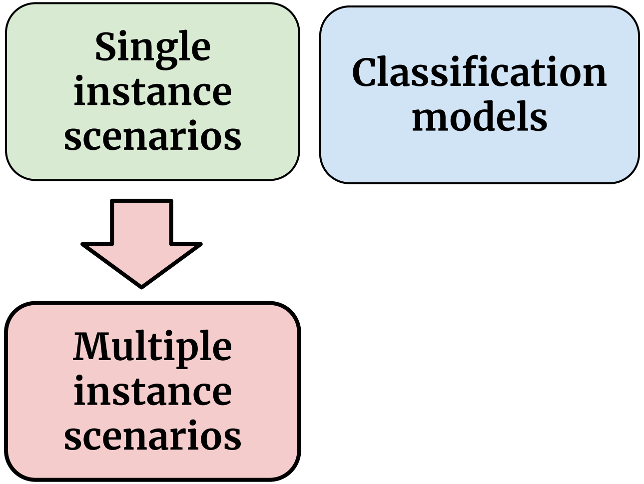
No *"best method"* for all the factors considered:

- **Fooling rate vs Similarity**

Results averaged for 100 random target distributions.

Contributions

- Novel multiple-instance attack paradigm:
  - Produce misclassifications for the incoming inputs
  - Control the probability distribution for the output classes
- Four different methods proposed
- Expose novel vulnerabilities in multiple scenarios and use-cases:
  - Adversarial label-drifts
  - Attacks less detectable in the long run



When and How to Fool Explainable Models (and Humans) With Adversarial Examples

J. Vadillo, R. Santana, J. A. Lozano. Under Review.

Single instance scenarios

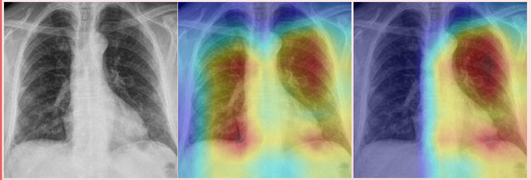
Classification models

1

Multiple-instance attacks paradigms

2

Attacks against **explainable models**



Motivation

Observed factors

Input	*Output*	*Explanation*	*Scenario*
			• Regular attacks

Scenario 1: Only the input is observed

Adversarial Example

Undetectable Threats

Motivation

Observed factors

Input	Output	Explanation	Scenario
			• Regular attacks
			• Observing the input and the output

Scenario 1: Only the input is observed

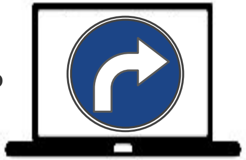
Adversarial Example

Undetectable
Threats

Scenario 2: The output is shown to the user

Adversarial Example

+



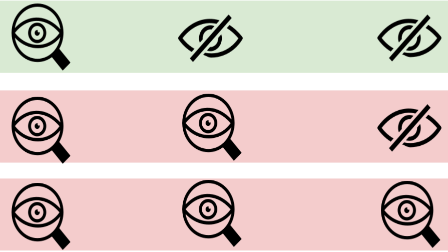
Model's Output

Undetectable?
Threats?

Motivation

Observed factors

Input	*Output*	*Explanation*	*Scenario*



- Regular attacks
- Observing the input and the output
- **Attacks against explainable models**

Scenario 1: Only the input is observed

Adversarial Example

Undetectable Threats

Scenario 2: The output is shown to the user

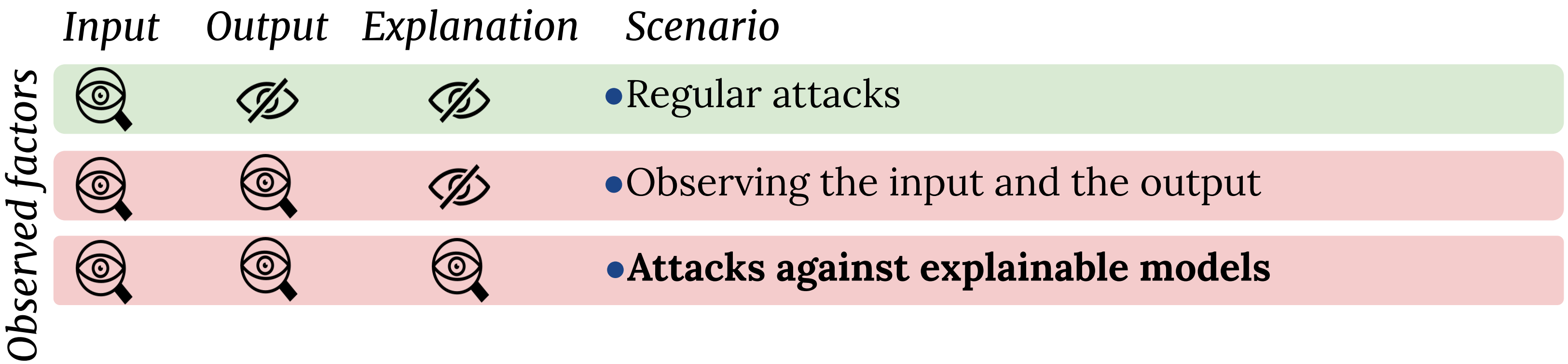
Adversarial Example

Model's Output

Explanation

Undetectable? Threats?

Motivation



Objective

How to generate **stealthy** and **realistic** adversarial attacks against explainable models (under human supervision):

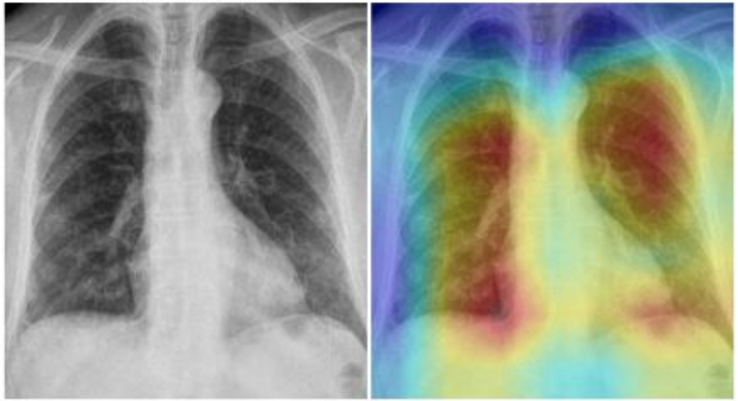
- Requirements
- Attack types
- Critical scenarios

Explanation methods

Local feature-based explanations

Prediction:
Great Pyrenees

Prediction:
COVID-19



Prediction: *Negative*

The movie was absolutely awful!

Adversarial attacks

Target class (y_t **):** $f(x') = y_t$

Target explanation (ξ_t **):** $g(x', f) = \xi_t$

Adversarial attacks

Target class (y_t): $f(x') = y_t$

Target explanation (ξ_t): $g(x', f) = \xi_t$

Projected Gradient Descent

$$x'_{[i+1]} = \underbrace{\mathcal{B}^x_\epsilon}_{\text{Projection operator}}\Big(x'_{[i]} - \alpha \cdot \mathrm{sign}\big(\nabla\underbrace{\mathcal{L}(x'_{[i]}, y_t, \xi_t, \tau, f)}_{\text{Attack loss}}\big)\Big)$$

Adversarial attacks

Target class (y_t): $f(x') = y_t$

Target explanation (ξ_t): $g(x', f) = \xi_t$

Projected Gradient Descent

$$x'_{[i+1]} = \mathcal{B}_\epsilon^x \left(x'_{[i]} - \alpha \cdot \operatorname{sign}\left(\nabla \mathcal{L}(x'_{[i]}, y_t, \xi_t, \tau, f) \right) \right)$$

Generalized attack loss

$$\mathcal{L}(x, y_t, \xi_t, \tau, f) = (1 - \tau) \cdot \mathcal{L}_{pred}(x, y_t, f) + \tau \cdot \mathcal{L}_{expl}(x, \xi_t, f)$$

Adversarial attacks

Target class (y_t): $f(x') = y_t$

Target explanation (ξ_t): $g(x', f) = \xi_t$

Projected Gradient Descent

$$x'_{[i+1]} = \mathcal{B}^{x}_{\epsilon}\left(x'_{[i]} - \alpha \cdot \mathrm{sign}\left(\nabla \underbrace{\mathcal{L}(x'_{[i]}, y_t, \xi_t, \tau, f)}\right)\right)$$

Generalized attack loss

$$\underbrace{\mathcal{L}(x, y_t, \xi_t, \tau, f) =} (1-\tau) \cdot \underbrace{\mathcal{L}_{pred}(x, y_t, f)}_{\text{Prediction loss}} + \tau \cdot \mathcal{L}_{expl}(x, \xi_t, f)$$

Adversarial attacks

Target class (y_t): $f(x') = y_t$

Target explanation (ξ_t): $g(x', f) = \xi_t$

Projected Gradient Descent

$$x'_{[i+1]} = \mathcal{B}_{\epsilon}^{x}\left(x'_{[i]} - \alpha \cdot \operatorname{sign}\left(\nabla \mathcal{L}(x'_{[i]}, y_t, \xi_t, \tau, f)\right)\right)$$

Generalized attack loss

$$\mathcal{L}(x, y_t, \xi_t, \tau, f) = (1 - \tau) \cdot \mathcal{L}_{pred}(x, y_t, f) + \tau \cdot \underbrace{\mathcal{L}_{expl}(x, \xi_t, f)}_{\text{Explanation loss}}$$

Adversarial attacks

Target class (y_t): $f(x') = y_t$

Target explanation (ξ_t): $g(x', f) = \xi_t$

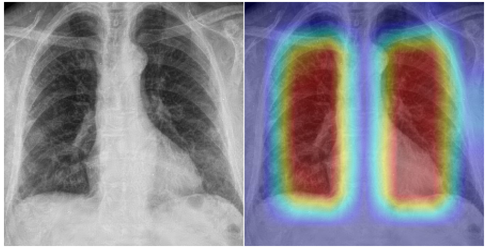
Projected Gradient Descent

$$x'_{[i+1]} = \mathcal{B}_\epsilon^x \left(x'_{[i]} - \alpha \cdot \mathrm{sign}\left(\nabla \mathcal{L}(x'_{[i]}, y_t, \xi_t, \tau, f) \right) \right)$$

Generalized attack loss

$$\mathcal{L}(x, y_t, \xi_t, \tau, f) = (1 - \tau) \cdot \mathcal{L}_{pred}(x, y_t, f) + \tau \cdot \underbrace{\mathcal{L}_{expl}(x, \xi_t, f)}_{\text{Explanation loss}}$$

ξ_t



$$\mathcal{L}_{expl}(x, \xi_t, f) = ||\xi_t - g(x, f)||_2$$

Adversarial attacks

Classification

Ground-truth class of x: y_x

Model's classification: $f(x)$

Human's classification: $h(x)$

Explanation

Adversarial attacks

Classification

Ground-truth class of x: y_x

Model's classification: $f(x)$

Human's classification: $h(x)$

Explanation

Explanation: $A(x)$ Model's: $A_f(x)$ Human's: $A_h(x)$

Adversarial attacks

Classification

Ground-truth class of x: y_x

Model's classification: $f(x)$

Human's classification: $h(x)$

Explanation

Explanation: $A(x)$ Model's: $A_f(x)$ Human's: $A_h(x)$

Agreement: $A_f(x) \approx A_h(x)$

Disagreement: $A_f(x) \not\approx A_h(x)$

Adversarial attacks

Classification

Ground-truth class of x: y_x

Model's classification: $f(x)$

Human's classification: $h(x)$

Explanation

Explanation: $A(x)$ Model's: $A_f(x)$ Human's: $A_h(x)$

Agreement: $A_f(x) \approx A_h(x)$ Disagreement: $A_f(x) \not\approx A_h(x)$

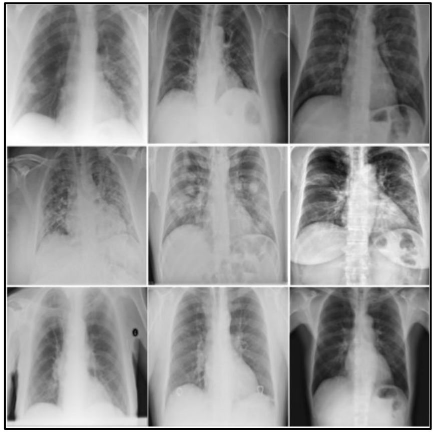
Consistency with class y: $A(x) \sim y$

Adversarial attacks

Case 1 $f(x) = h(x)$ $A_f(x) \not\approx A_h(x)$

Case 2 $f(x) \neq h(x)$ $A_f(x) \not\approx A_h(x)$

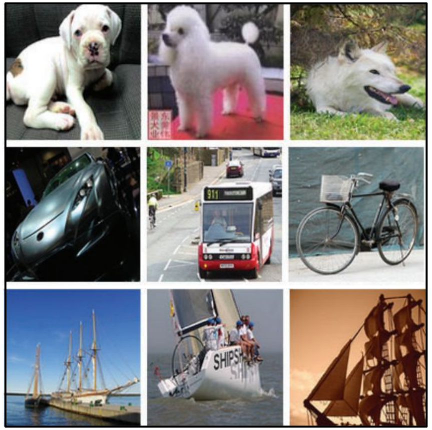
Case 3 $f(x) \neq h(x)$ $A_f(x) \approx A_h(x)$



Medical Image Diagnosis

Dataset: *COVIDx (3 classes)*

Model: *Covid-Net (92.6% accuracy)*



Large-Scale Image Recognition

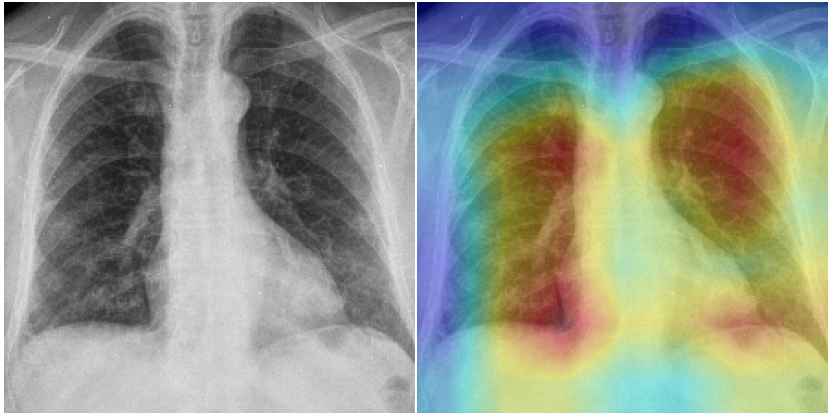
Dataset: *ImageNet (1000 classes)*

Model: *ResNet-50 (74.9% accuracy)*

Case 1

$$f(x) = h(x) \land A_f(x) \not\approx A_h(x)$$

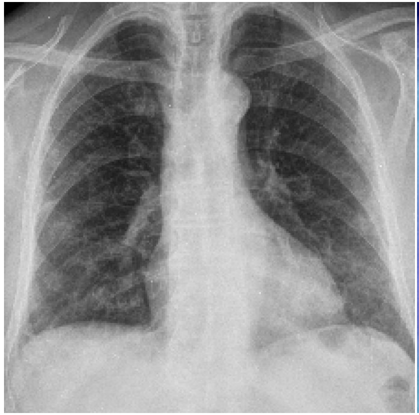
Clean input
Prediction: ***COVID-19***

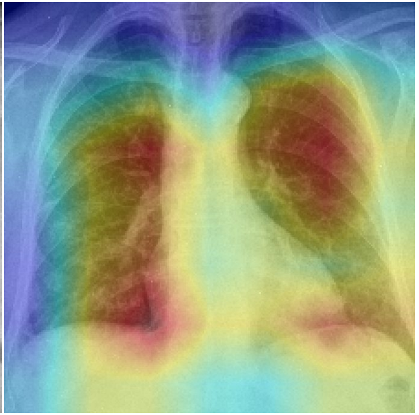


Case 1

$$f(x) = h(x) \wedge A_f(x) \not\approx A_h(x)$$

Clean input
Prediction: *COVID-19*





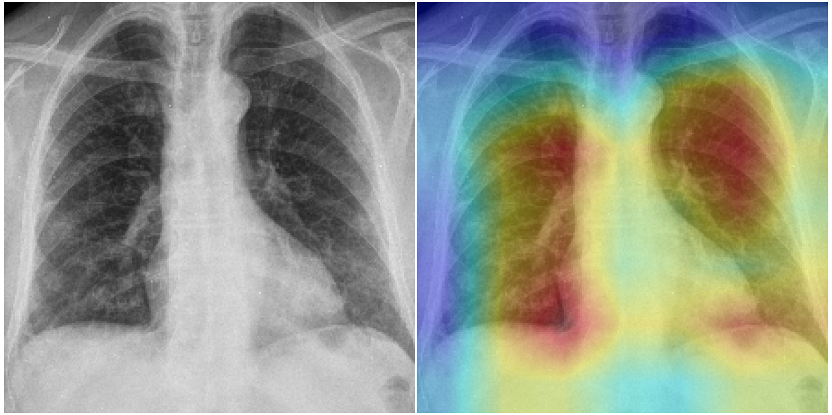
Case 1

$$\underbrace{f(x) = h(x)} \wedge \underbrace{A_f(x) \not\approx A_h(x)}$$

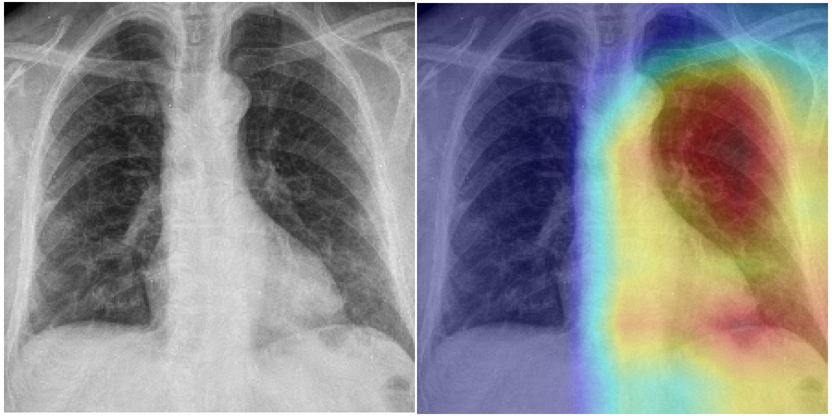
$$A_f(x) \sim y_x$$

Omit information
Misleading recommendations

Clean input
Prediction: *COVID-19*



Adversarial example
Prediction: *COVID-19*



Case 1

$$f(x) = h(x) \wedge A_f(x) \not\approx A_h(x)$$

$$A_f(x) \sim y_x$$

Omit information
Misleading recommendations
Produce/hide biases

Clean input

Output: **Reject credit loan:**

➢ **Income** $<$ 1200

➢ and **Gender** = **!**

Adversarial example

Output: **Reject credit loan:**

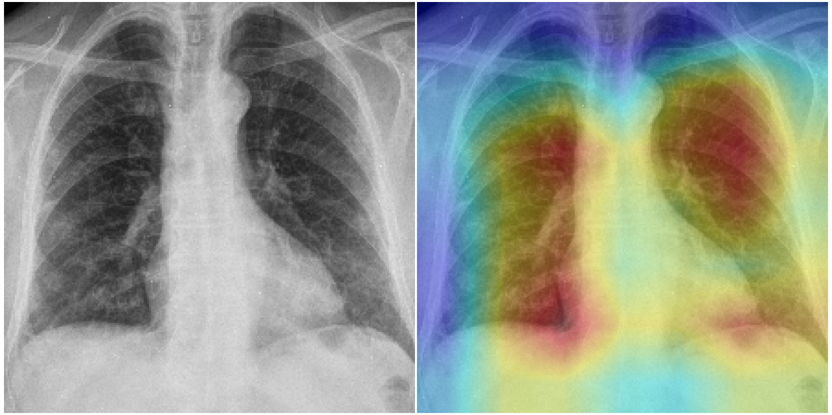
➢ **Income** $<$ 1500

➢ and **Job** = None

Case 2

$$f(x) \neq h(x) \wedge A_f(x) \not\approx A_h(x)$$

Clean input
Prediction: *COVID-19*



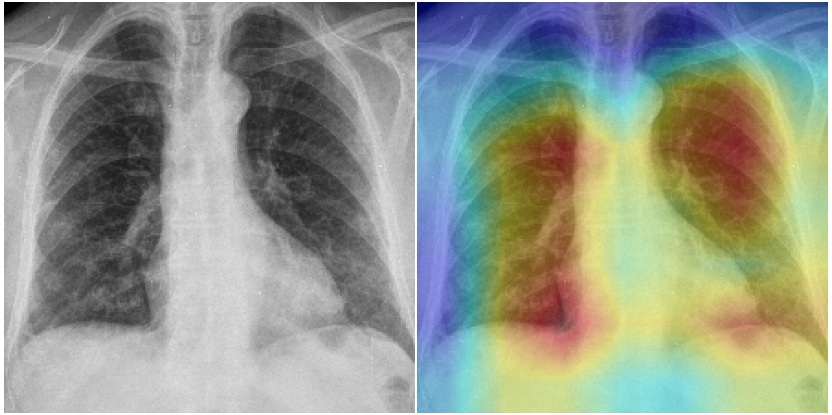
Case 2

$$f(x) \neq h(x) \wedge A_f(x) \not\approx A_h(x)$$

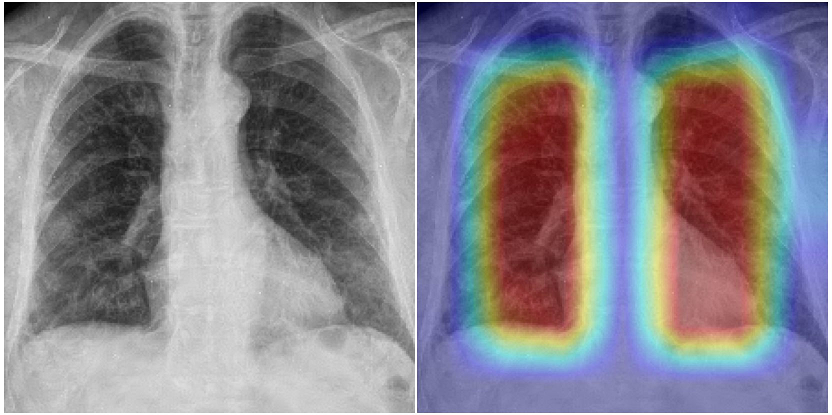
$$A_f(x) \sim f(x)$$

The model supports its (wrong) prediction

Clean input
Prediction: *COVID-19*



Adversarial example
Prediction: ***normal***



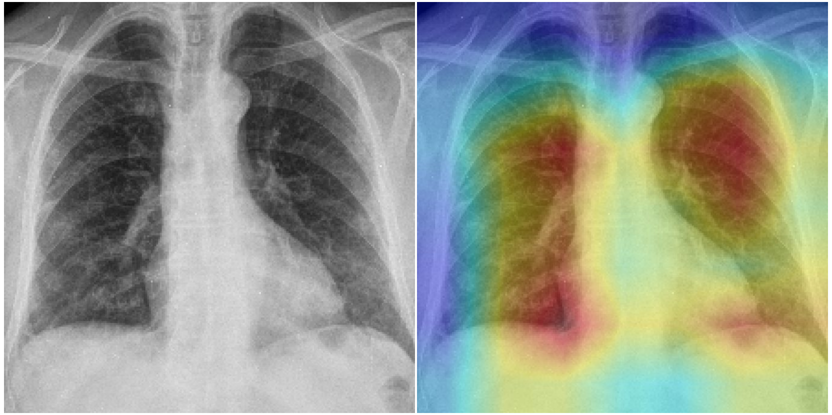
Case 2

$$f(x) \neq h(x) \wedge A_f(x) \not\approx A_h(x)$$

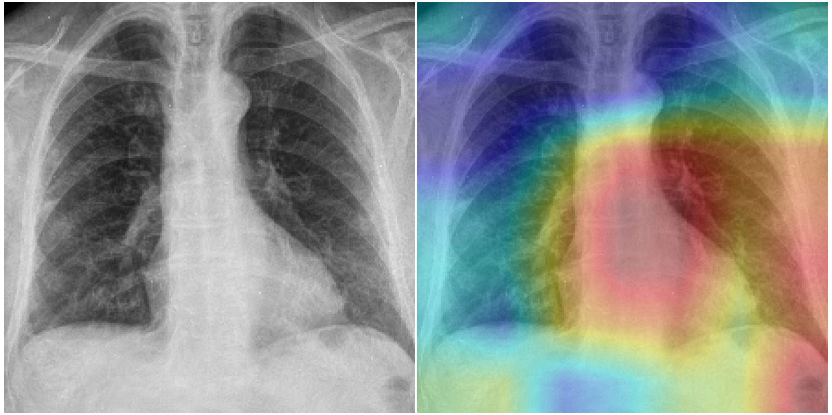
$$A_f(x) \sim f(x)$$

The model supports its (wrong) prediction

Clean input
Prediction: *COVID-19*



Adversarial example
Prediction: ***normal***



Case 2

$$f(x) \neq h(x) \wedge A_f(x) \not\approx A_h(x)$$

$$A_f(x) \sim f(x)$$

The model supports its (wrong) prediction

Shift the user's attention

Clean input
Prediction: *Curly-coated retriever*

Large-Scale Image Recognition

Case 2

$$f(x) \neq h(x) \wedge A_f(x) \not\approx A_h(x)$$

$$A_f(x) \sim f(x)$$

The model supports its (wrong) prediction

Shift the user's attention

Clean input
Prediction: *Curly-coated retriever*

Adversarial input
Prediction: *Suit*

Case 3

$$\boxed{f(x) \neq h(x)} \wedge \boxed{A_f(x) \approx A_h(x)}$$

Clean input

Prediction: *Curly-coated retriever*

Case 3

$$f(x) \neq h(x) \wedge A_f(x) \approx A_h(x)$$

$$\left.\begin{array}{l} A_f(x) \sim y_x \\ A_f(x) \sim f(x) \end{array}\right\} \text{Ambiguity}$$

Clean input

Prediction: ***Curly-coated retriever***

Case 3

$$f(x) \neq h(x) \wedge A_f(x) \approx A_h(x)$$

$$\left.\begin{aligned} A_f(x) &\sim y_x \\ A_f(x) &\sim f(x) \end{aligned}\right\} \text{Ambiguity}$$

Clean input

Prediction: *Curly-coated retriever*

Curly-coated retriever

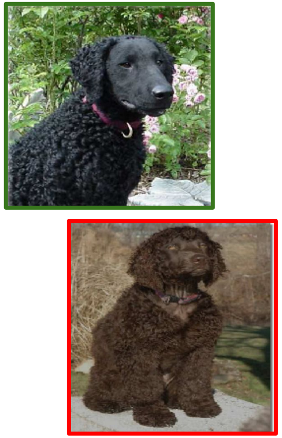
Irish water spaniel

Case 3

$$f(x) \neq h(x) \wedge A_f(x) \approx A_h(x)$$

$$\left.\begin{array}{l} A_f(x) \sim y_x \\ A_f(x) \sim f(x) \end{array}\right\} \text{Ambiguity}$$

Clean input
Prediction: *Curly-coated retriever*



Adversarial input
Prediction: *Irish water spaniel*

Cases

$$f(x) = h(x) \wedge A_f(x) \not\approx A_h(x) \wedge A_f(x) \sim y_x$$

$$f(x) \neq h(x) \wedge A_f(x) \not\approx A_h(x) \wedge A_f(x) \sim f(x)$$

$$f(x) \neq h(x) \wedge A_f(x) \approx A_h(x) \wedge A_f(x) \sim y_x \wedge A_f(x) \sim f(x)$$

Additional factors and scenarios

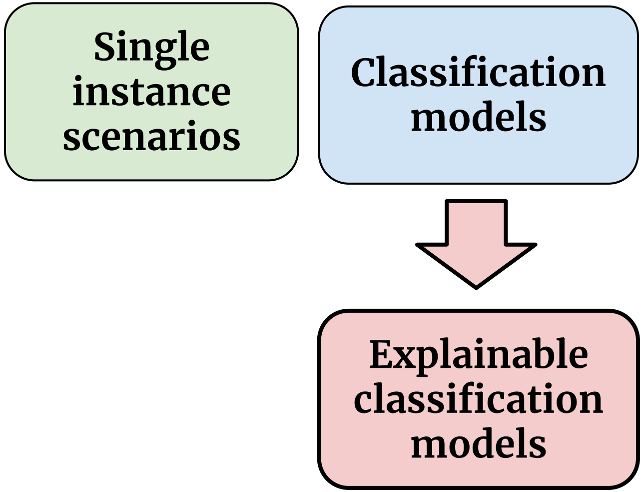
- Type of explanation? (feature-based, prototype-based...)
- User expertise? (none, medium, high...)
- Objective? (knowledge acquisition, debugging, ethics...)
- Impact?

Contributions

- Comprehensive roadmap for the design of realistic attacks against explainable ML:
    - Attack types
    - Requirements
    - Critical scenarios
    - Illustrative experiments

- More rigorous study of adversarial attacks in this domain

- Raise awareness about the possible threats that both models and humans may face



eman ta zabal zazu

Universidad del País Vasco

Euskal Herriko Unibertsitatea

FACULTY OF COMPUTER SCIENCE

UNIVERSITY OF THE BASQUE COUNTRY

Intelligent Systems Group

Questions?

Acknowledgements:

- Spanish Ministry of Science, Innovation and Universities
  - FPU19/03231 predoctoral grant
- Spanish Ministry of Economy and Competitiveness MINECO
  - PID2019-104966GB-I00
- Basque Government
  - ELKARTEK program

jon.vadillo@ehu.eus

github.com/vadel

www.vadel.github.io/