Information-Theoretic Methods in Data Sciences

Model Uncertainty, Robustness and Model Drift

Pablo Piantanida
pablo.piantanida@centralesupelec.fr

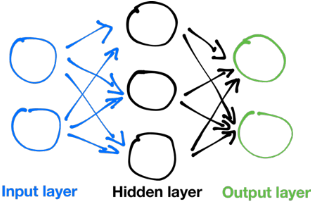
Laboratoire des Signaux et Systèmes (L2S)
Université Paris-Saclay CNRS CentraleSupélec

7th workshop "Journée Statistique et Informatique à Paris Saclay",
January 26th, 2022

Outline

Outline

The birth of AI and Deep Learning



A Brief History of AI - Dartmouth Conference (1956)

We propose that a 2 month, 10 man study of artificial intelligence be carried out during the summer of 1956 at Dartmouth College in Hanover, New Hampshire.

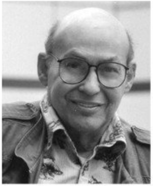
The study is to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it. An attempt will be made to find how to make machines use language, form abstractions and concepts, solve kinds of problems now reserved for humans, and improve themselves.

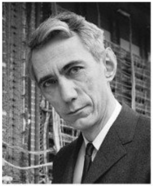
- Dartmouth AI Project Proposal; J. McCarthy et al.; Aug. 31, 1955.

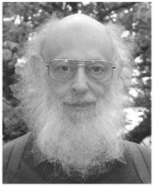
Photographer: Joe Mehling

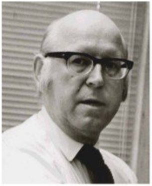
A Brief History of AI - Dartmouth Conference (1956)


John MacCarthy


Marvin Minsky

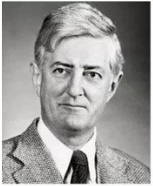

Claude Shannon


Ray Solomonoff


Alan Newell


Herbert Simon

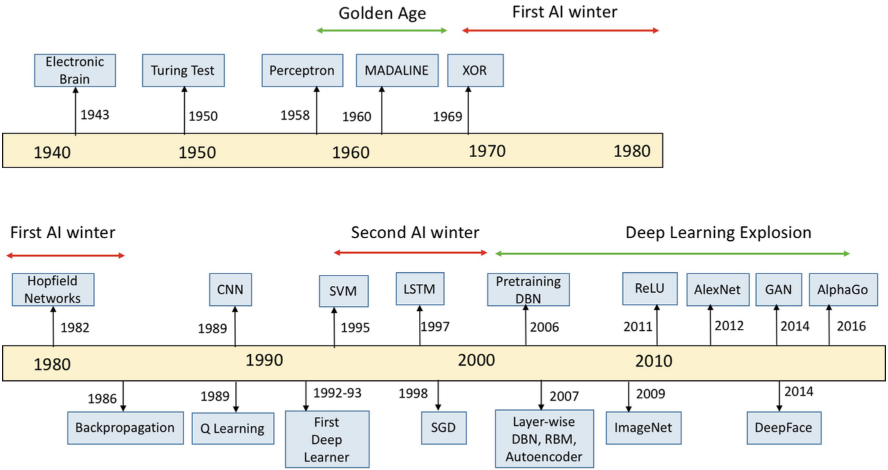

Arthur Samuel


Oliver Selfridge

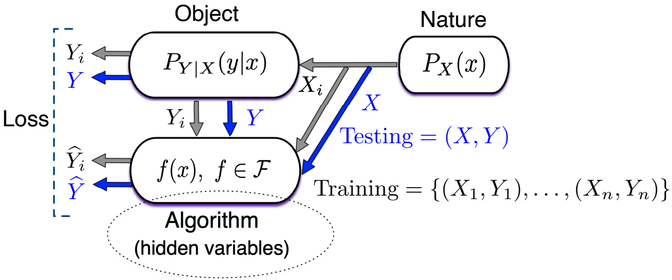

Nathaniel Rochester


Trenchard More

The Founding Fathers of AI


Golden Age
First AI winter
Electronic Brain
Turing Test
Perceptron
MADALINE
XOR
1943
1950
1958
1960
1969
1940
1950
1960
1970
1980
First AI winter
Second AI winter
Deep Learning Explosion
Hopfield Networks
CNN
SVM
LSTM
Pretraining DBN
ReLU
AlexNet
GAN
AlphaGo
1982
1989
1995
1997
2006
2011
2012
2014
2016
1980
1990
2000
2010
1986
1989
1992-93
1998
2007
2009
2014
Backpropagation
Q Learning
First Deep Learner
SGD
Layer-wise DBN, RBM, Autoencoder
ImageNet
DeepFace

A Probabilistic Model of Learning (1960)



- **Imitation of the object:** try to construct a predictor which provides the best predictions to the supervisor output
- **Approximation of the object:** try to approximate the object (nature) itself based on a model (uncertainty and calibration)

Learning is data compression: To separate structure from noise, the regularities present in the data by choosing appropriately $f \in \mathcal{F}$.

Statistical Learning Theory (1960 - 1990)

$$P\left(\sup_{f\in\mathcal{F}}\left|\hat{R}_n(f)-R(f)\right|>\varepsilon\right)\leq 8S(\mathcal{F},n)e^{-n\varepsilon^2/32}$$

$$\mathbb{E}\left[\sup_{f\in\mathcal{F}}\left|\hat{R}_n(f)-R(f)\right|\right]\leq 2\sqrt{\frac{\log S(\mathcal{F},n)+\log 2}{n}}$$

Vapnik–Chervonenkis theory (1960) addresses key questions:

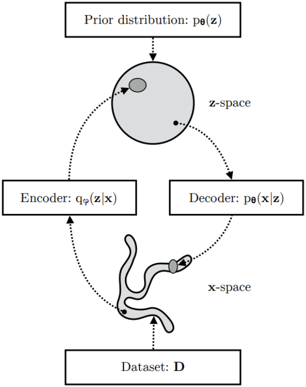
- What are the conditions for **consistency of a learning rule** based on the empirical risk minimization principle?
- How fast is the **rate of convergence** of the learning process?
- How can one control the **generalization ability** (convergence rate) of the learning process?

Vapnik and Chervonenkis' ingenious formulation led to the characterization of **necessary and sufficient conditions** (finite VC-dimension) for the minimizing of a risk $R(f)$ using data.

Deep Learning (2006/2007)

Good representations learn to **disentangle manifolds:**

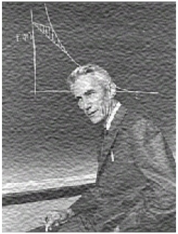
- Enc/Dec map between low and high representations of data,
- Encoders **perform inference** to **interpret data**, **flatten** and to **disentangle** the data manifold,
- Decoders can introduce changes in reconstructing data features,
- How **goodness** should be defined is an **open problem**.



My research focus on **developing and bringing new mathematical tools and methodological principles from information theory** to machine learning and deep learning.

Legacy of Shannon's work

Shannon's Model of a Communication System



Shannon proposed (1948) an asymptotic approach:

- A k-symbol sequence U is mapped by an encoder into an n-symbol input sequence X
- The received channel output sequence Y is mapped by a decoder into an estimate Û
- What is the the **maximum communication rate** $R = k/n$ (bits per transmission) such that $\mathrm{P}\{\mathrm{U} \neq \hat{\mathrm{U}}\}$ **can be made arbitrarily small** when (k, n) are sufficiently large?

Shannon's ingenious formulation led to the characterization of necessary and sufficient conditions for reliable communication.

Shannon proposed (1948) an (k,n)-asymptotic approach:

- **Channel coding theorem:** The capacity is the maximum of the mutual information between the channel input and output

$$C = \sup_{\mathrm{p_X}} \mathrm{I}(\mathsf{X};\mathsf{Y}) \quad \text{in bit/transmission.}$$

- **Lossy source coding theorem:** The optimal tradeoff between the rate $R = n/k$ and the distortion D is

$$R(D) = \inf_{\mathrm{p}_{\hat{\mathsf{U}}|\mathsf{U}} : \mathbb{E}[d(\hat{\mathsf{U}},\mathsf{U})] \le D} \mathrm{I}\big(\mathsf{U};\hat{\mathsf{U}}\big) \quad \text{in bits/symbol.}$$

- **Separation theorem:** Shannon's ingenious formulation led to

$$R(D) < C,$$

necessary and sufficient conditions for reliable communication.

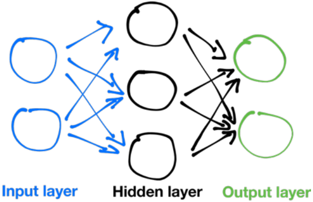
Legacy of Shannon's Work

"Using bits as a universal representation between sources and channels is essentially optimal"

Nothing is more practical than a good theory:

- Analogue **data can be represented by discrete symbols** and compressed before transmission
- **Representation of information is at the heart of modern communications** (codes that can squish messages, saving time resources and codes that can protect data from noise)
- Information theory provides valuable insight, highlighting key properties of good codes, **leading to optimal designs.**

Information, Uncertainty and Learning



Shannon Entropy

Entropy $\mathrm{H}(\mathsf{X})$ of a discrete random variable (RV) $\mathsf{X} \sim \mathrm{p}$:

- **1. Measure of uncertainty** → "surprise" function $s(x)$, $x \in \mathcal{X}$, and $\mathrm{H}(\mathsf{X}) = \mathbb{E}[s(\mathsf{X})]$
- **2. Independent of alphabet** → $s(x) = s\big(\mathrm{p}(x)\big)$
- **3. Additivity:**

$$s(\mathrm{p}(x)\mathrm{q}(y)) = s(\mathrm{p}(x)) + s(\mathrm{q}(y)) \quad \to \quad s(x) = \log \mathrm{p}(x)$$

- Lower probability implies higher surprise → $s(x) = -\log \mathrm{p}(x)$

$$\begin{aligned} \mathrm{H}(\mathsf{X}) &= -\sum_{x \in \mathcal{X}} \mathrm{p}(x) \log \mathrm{p}(x) \\ &= -\mathbb{E}[\log \mathrm{p}(\mathsf{X})] \end{aligned}$$

- $\mathrm{H}(\mathsf{X})$ is nonnegative, continuous, and strictly concave function of p, and $0 \le \mathrm{H}(\mathsf{X}) \le \log|\mathcal{X}|$.

Shannon Entropy

Entropy $\mathrm{H}(\mathsf{X})$ of a discrete random variable (RV) $\mathsf{X} \sim \mathrm{p}$:

- **1. Measure of uncertainty** → "surprise" function $s(x)$, $x \in \mathcal{X}$, and $\mathrm{H}(\mathsf{X}) = \mathbb{E}[s(\mathsf{X})]$
- **2. Independent of alphabet** → $s(x) = s\big(\mathrm{p}(x)\big)$
- **3. Additivity:**

$$s(\mathrm{p}(x)\mathrm{q}(y)) = s(\mathrm{p}(x)) + s(\mathrm{q}(y)) \quad \to \quad s(x) = \log \mathrm{p}(x)$$

- Lower probability implies higher surprise → $s(x) = -\log \mathrm{p}(x)$

$$\begin{aligned}\mathrm{H}(\mathsf{X}) &= -\sum_{x \in \mathcal{X}} \mathrm{p}(x) \log \mathrm{p}(x) \\ &= -\mathbb{E}[\log \mathrm{p}(\mathsf{X})]\end{aligned}$$

- $\mathrm{H}(\mathsf{X})$ is nonnegative, continuous, and strictly concave function of p, and $0 \leq \mathrm{H}(\mathsf{X}) \leq \log |\mathcal{X}|$.

Rényi Entropy

- Rényi entropy for a discrete r.v. X with probability $\mathrm{p}(x)$:

$$H_\alpha(\mathsf{X}) = \frac{1}{1-\alpha} \log \sum_{x\in\mathcal{X}} \mathrm{p}(x)^\alpha = \frac{1}{1-\alpha} \log \mathbb{E}\left[\mathrm{p}(\mathsf{X})^{\alpha-1}\right],$$

for $\alpha > 0$; $H_\alpha(\mathsf{X}) \to \mathrm{H}(\mathsf{X})$ as $\alpha \to 1$.

- Conditional Rényi entropy for discrete RVs $(\mathsf{X},\mathsf{Y}) \sim \mathrm{p}(x,y)$:

$$H_\alpha(\mathsf{X}|\mathsf{Y}) = \sum_{y\in\mathcal{Y}} \mathrm{p}(y)\left(\frac{1}{1-\alpha} \log \sum_{x\in\mathcal{X}} \mathrm{p}(x|y)^\alpha\right) = \frac{1}{1-\alpha}\mathbb{E}\left[\log \sum_{x\in\mathcal{X}} \mathrm{p}(x|Y)^\alpha\right].$$

- There are many other information measures.

Rényi Entropy

- Rényi entropy for a discrete r.v. X with probability $\mathrm{p}(x)$:

$$H_\alpha(\mathsf{X}) = \frac{1}{1-\alpha}\log\sum_{x\in\mathcal{X}}\mathrm{p}(x)^\alpha$$
$$= \frac{1}{1-\alpha}\log\mathbb{E}\left[\mathrm{p}(\mathsf{X})^{\alpha-1}\right],$$

  for $\alpha > 0$; $H_\alpha(\mathsf{X}) \to \mathrm{H}(\mathsf{X})$ as $\alpha \to 1$.

- Conditional Rényi entropy for discrete RVs $(\mathsf{X},\mathsf{Y}) \sim \mathrm{p}(x,y)$:

$$H_\alpha(\mathsf{X}|\mathsf{Y}) = \sum_{y\in\mathcal{Y}}\mathrm{p}(y)\left(\frac{1}{1-\alpha}\log\sum_{x\in\mathcal{X}}\mathrm{p}(x|y)^\alpha\right)$$
$$= \frac{1}{1-\alpha}\mathbb{E}\left[\log\sum_{x\in\mathcal{X}}\mathrm{p}(x|Y)^\alpha\right].$$

- There are many other information measures.

Information and Learning

An emerging interface?

- Shannon's entropy provides a **measure of uncertainty** about the amount of information that a learner possesses **relative to a given concept** when only the probability distribution is given.
- **But** the basic problem of learning consists in that **one has to separate the relevant information** from patterns.

Two questions naturally arise:

- Are information measures **fundamental measures of the random properties of data** for learning problems?
- What are the instances of learning problems for which **information measures can play a key role?**

The study of these questions has played an important role and, undoubtedly, it will play a central role in future learning methods.

A Long-Lasting Partnership

Learning is data compression:

- The goal is to learn the laws and regularities present in the data, that is, **to separate structure from noise.**
- **Data compression is fundamentally related to statistical generalization** as shown by a number of sample complexity bounds (e.g., VC-dimension, PAC-Bayes, and others). ☺
- The celebrated **Minimum Description Length (MDL)** principle, to approach model selection in statistical inference.
- In unsupervised learning, **Variational Autoencoders (VAEs)** are motivated by compression methods and the **Information Bottleneck** method for supervised learning as well.

This talk focuses on two related problems:

- **How to measure uncertainty from model predictions?**
- **How to detect uncertainty induced from data drift?**

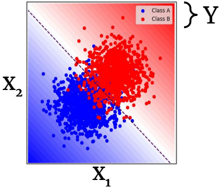
Outline

What Does Model Uncertainty Means?

Return a distribution over predictions rather than a single prediction.

- ***Classification***: Output label along with its confidence.
- ***Regression***: Output mean along with its variance.

Good uncertainty estimates quantify ***when we can trust the model's predictions***.



$$p(\mathbf{y}|\mathbf{x})$$

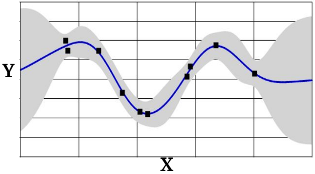
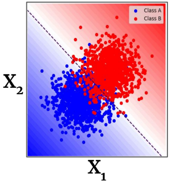


Image credit: Eric Nalisnick

What Does Out-of-Distribution Robustness Means?

I.I.D. $p_{TEST}(y,x) = p_{TRAIN}(y,x)$

(*Independent and Identically Distributed*)



O.O.D. $p_{TEST}(y,x) \neq p_{TRAIN}(y,x)$

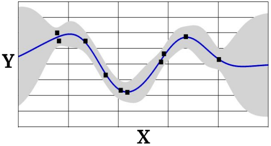


Image credit: Eric Nalisnick

What Does Out-of-Distribution Robustness Means?

I.I.D. $p_{TEST}(y,x) = p_{TRAIN}(y,x)$

O.O.D. $p_{TEST}(y,x) \neq p_{TRAIN}(y,x)$

Examples of dataset shift:

- ***Covariate shift.*** Distribution of features p(x) changes and p(y|x) is fixed.
- ***Open-set recognition.*** New classes may appear at test time.
- ***Label shift.*** Distribution of labels p(y) changes and p(x|y) is fixed.

ImageNet-C: Varying Intensity for Dataset Shift

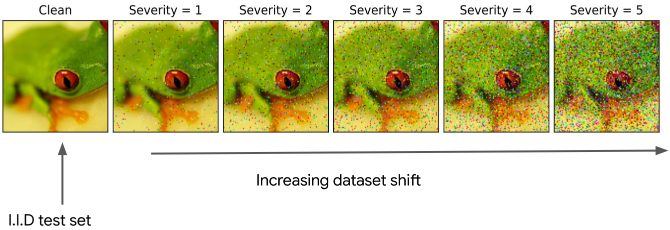
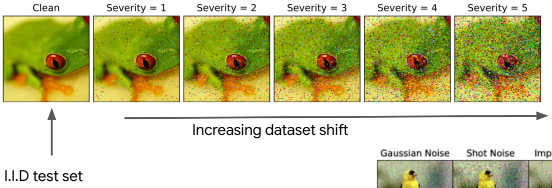


Image source: Benchmarking Neural Network Robustness to Common Corruptions and Perturbations, [Hendrycks & Dietterich, 2019](#).

ImageNet-C: Varying Intensity for Dataset Shift



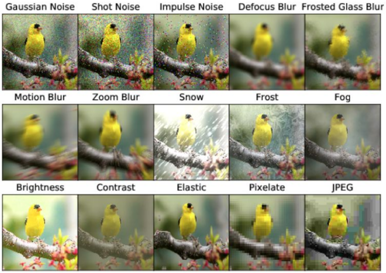
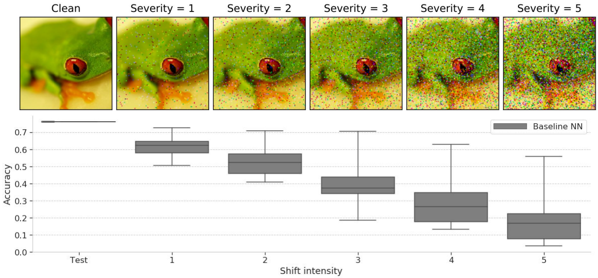


Image source: Benchmarking Neural Network Robustness to Common Corruptions and Perturbations, [Hendrycks & Dietterich, 2019](#).

Neural Networks Do Not Generalize Under Distribution Shift

- **Accuracy drops** with increasing shift on Imagenet-C

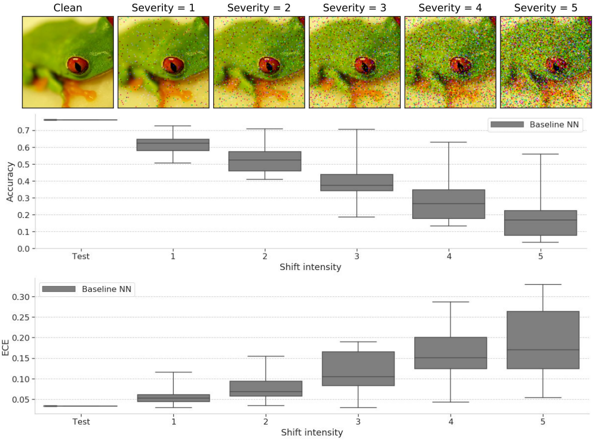


- But do the models know that they are less accurate?

Can You Trust Your Model's Uncertainty? Evaluating Predictive Uncertainty Under Dataset Shift?, [Ovadia et al. 2019]

Neural Networks Do Not Know When They Are Wrong

- **Accuracy drops** with increasing shift on Imagenet-C
- **Quality of uncertainty degrades with shift** -> "overconfident mistakes"



Models Assign High Confidence Predictions to OOD Inputs

Example images where model assigns >99.5% confidence.

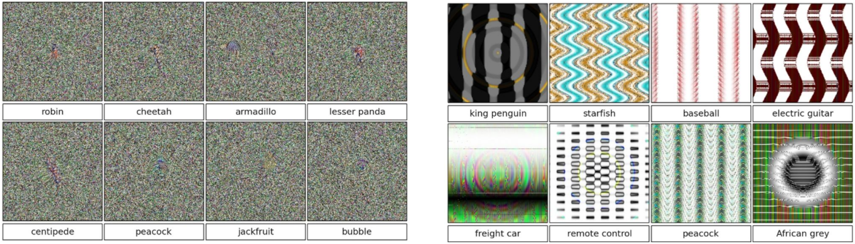
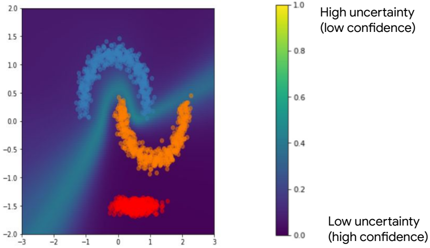


Image source: "Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images" [Nguyen et al. 2014](#)

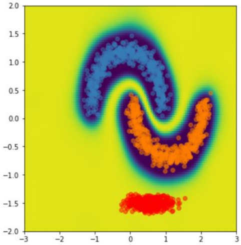
Models Assign High Confidence Predictions to OOD Inputs



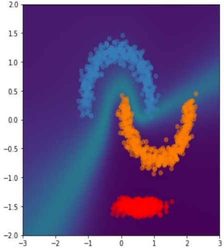
Deep neural networks

Image source: "Simple and Principled Uncertainty Estimation with Deterministic Deep Learning via Distance Awareness" [Liu et al. 2020](#)

Models Assign High Confidence Predictions to OOD Inputs



Ideal behavior



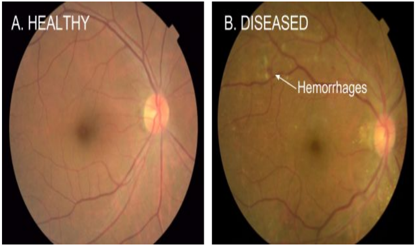
Deep neural networks

Trust model when x* is close to $p_{TRAIN}(x,y)$

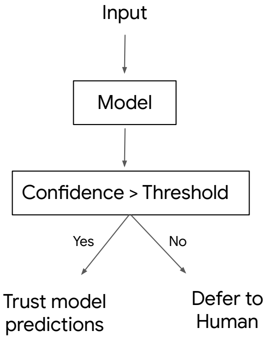
Image source: "Simple and Principled Uncertainty Estimation with Deterministic Deep Learning via Distance Awareness" [Liu et al. 2020](#)

Applications to Healthcare

- Use model uncertainty to decide when to trust the model or to defer to a human.
- Reject low-quality inputs.

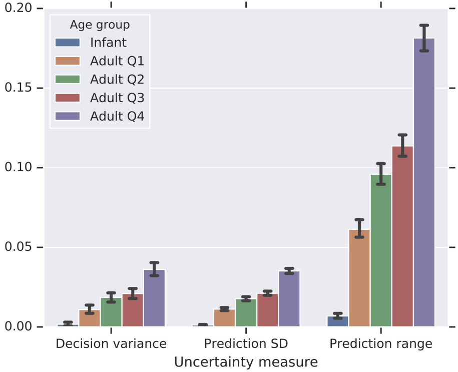


Diabetic retinopathy detection from fundus images
[Gulshan et al, 2016]



Applications to Healthcare

- Model accuracy and uncertainty across patient sub-groups



Mortality prediction from electronic health records
[Dusenberry et al, 2020]

Applications to Self-driving Cars

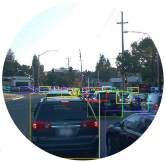
Dataset shift:

- Time of day / Lighting
- Geographical location (City vs suburban)
- Changing conditions (Weather / Construction)

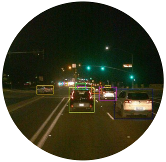
Weather

Construction

Image credit: Sun et al, Waymo Open Dataset



Daylight



Night

Downtown

Suburban

- Example: Classification of genomic sequences

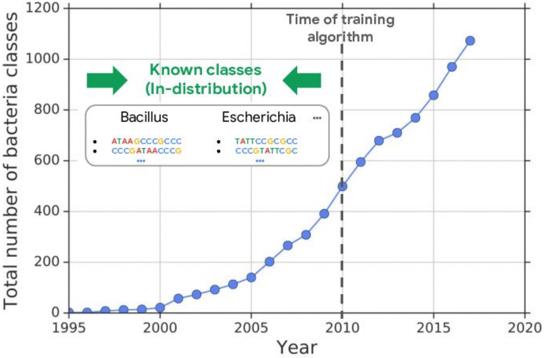


Image source: https://ai.googleblog.com/2019/12/improving-out-of-distribution-detection.html

Outline

DOCTOR: A Simple Method for Detecting Misclassification Errors

Joint work with Federica Granese, Marco Romanelli, Daniele Gorla and Catuscia Palamidessi

NEURAL INFORMATION PROCESSING SYSTEMS

(https://neurips.cc/virtual/2021/spotlight/28017)

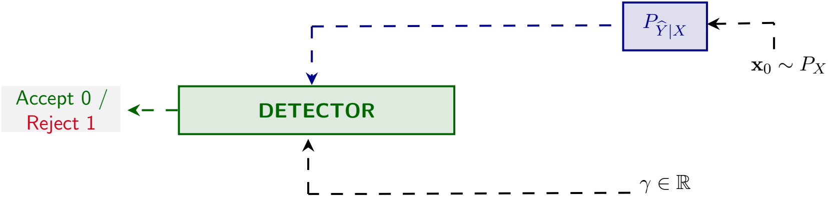
Main Definitions

Let

- $\mathcal{X} \subseteq \mathbb{R}$ be the **feature space**;
- $\mathcal{Y} = \{1, \ldots, C\}$ be the **label space**;
- p_{XY} be the underlying (unknown) probability density function over $\mathcal{X} \times \mathcal{Y}$;
- $\mathcal{D}_n = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\} \sim p_{XY}$ be a random realization of n i.i.d. samples according to p_{XY} denoting the **training set**;
- $f_{\mathcal{D}_n} : \mathcal{X} \to \mathcal{Y}$ be the **predictor**,

$$f_{\mathcal{D}_n}(\mathbf{x}) \equiv f_n(\mathbf{x}; \mathcal{D}_n) \triangleq \arg\max_{y \in \mathcal{Y}} P_{\widehat{Y}|X}(y|\mathbf{x}; \mathcal{D}_n).$$

Problem Definition

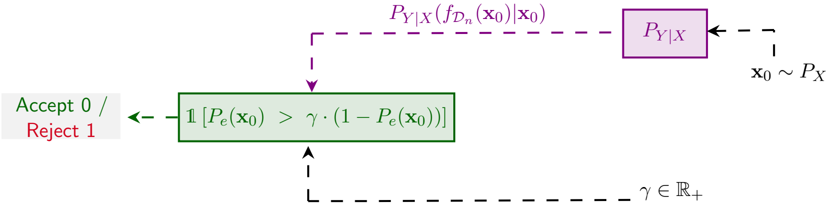


Ideal (Oracle) Detector

Definition (Error probability per sample)

For a given testing feature $\mathbf{x}_0 \in \mathcal{X}$,

- $E(\mathbf{x}_0) \triangleq \mathbb{1}[Y \neq f_{\mathcal{D}_n}(\mathbf{x}_0)]$ is the **error variable** corresponding to a predetermined predictor $f_{\mathcal{D}_n}$ (based on $P_{Y|X}$);
- $P_e(\mathbf{x}_0) \triangleq \mathbb{E}[E(\mathbf{x}_0)|\mathbf{x}_0] = 1 - P_{Y|X}(f_{\mathcal{D}_n}(\mathbf{x}_0)|\mathbf{x}_0)$ is the **probability of error classification w.r.t.** $P_{Y|X}$.



In practice, $P_e(\mathbf{x}_0)$ **is not available**, but can we approximate it?

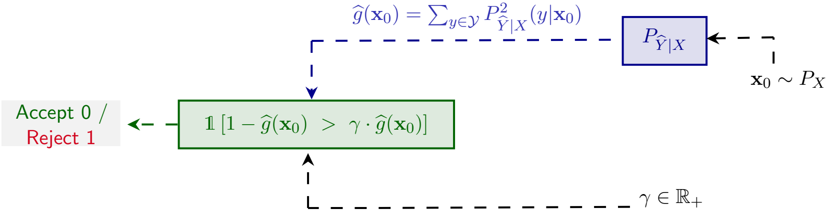
DOCTOR: D_α

Proposition (DOCTOR: D_α)

For a given testing feature $\mathbf{x}_0 \in \mathcal{X}$,

- $1 - \widehat{g}(\mathbf{x}_0) \triangleq \sum_{y \in \mathcal{Y}} P_{\widehat{Y}|X}(y|\mathbf{x}_0)\, \Pr(\widehat{Y} \neq y|\mathbf{x}_0) = 1 - \sum_{y \in \mathcal{Y}} P^2_{\widehat{Y}|X}(y|\mathbf{x}_0)$ *approximates the* **probability of incorrectly classifying** $\mathbf{x}_0$*;*
- $(1 - \sqrt{\widehat{g}(\mathbf{x}_0)}) - \Delta(\mathbf{x}_0) \leq P_e(\mathbf{x}_0) \leq (1 - \sqrt{\widehat{g}(\mathbf{x}_0)}) + \Delta(\mathbf{x}_0)$ *where*

$$\Delta(\mathbf{x}_0) \triangleq 2\sqrt{2\ \mathbf{KL}(P_{Y|X}(\cdot|\mathbf{x}_0) \| P_{\widehat{Y}|X}(\cdot|\mathbf{x}_0))}.$$



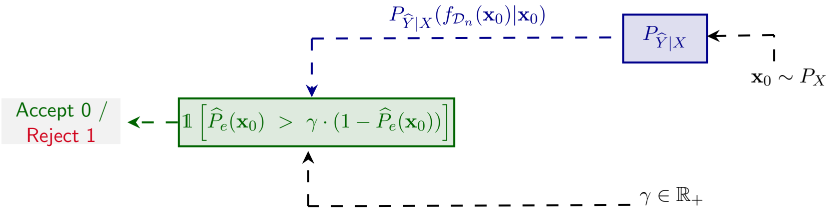
$$D_\alpha(\mathbf{x}_0, \gamma) \triangleq \mathbb{1}\left[1 - \widehat{g}(\mathbf{x}_0) \ > \ \gamma \cdot \widehat{g}(\mathbf{x}_0)\right]$$

DOCTOR: D_β

Definition (DOCTOR: D_β)

For a given testing feature $\mathbf{x} \in \mathcal{X}$,

- $\widehat{E}(\mathbf{x}_0) \triangleq \mathbb{1}[\widehat{Y} \neq f_{\mathcal{D}_n}(\mathbf{x}_0)]$ is the **self-error variable** corresponding to $f_{\mathcal{D}_n}$ (based on the model $P_{\widehat{Y}|X}$);
- $\widehat{P}_e(\mathbf{x}_0) \triangleq \mathbb{E}[\widehat{E}(\mathbf{x}_0)|\mathbf{x}_0] = 1 - P_{\widehat{Y}|X}(f_{\mathcal{D}_n}(\mathbf{x}_0)|\mathbf{x}_0)$ is the **probability of error classification w.r.t.** $P_{\widehat{Y}|X}$.



$$D_\beta(\mathbf{x}_0, \gamma) \triangleq \mathbb{1}\left[\widehat{P}_e(\mathbf{x}_0) > \gamma \cdot (1 - \widehat{P}_e(\mathbf{x}_0))\right]$$

Evaluation Metrics

Definition (FRR versus TRR)

The false rejection rate (FRR) represents the probability that a hit (sample correctly classified) is rejected, while the true rejection rate (TRR) is the probability that a miss (sample wrongly classified) is rejected.

Definition (AUROC)

The area under the Receiver Operating Characteristic curve (ROC) depicts the relationship between TRR and FRR. The perfect detector corresponds to a score of 100%.

Definition (FRR at 95% TRR)

This is the probability that a hit is rejected when the TRR is at 95%.

Scenarios: Totally Black Box & Partially Black Box

Definition (Totally Black Box (TBB) Scenario)

In TBB only the output of the last layer of the network is available, hence gradient-propagation to perform input pre-processing is not allowed.

Definition (Partially Black Box (PBB) Scenario)

In PBB we allow method-specific inputs perturbations and the possibility of doing temperature scaling.

Competitors (SOTA Methods) for TBB and PBB

1) ODIN [Liang et al., 2018]

$$\mathbf{SODIN}(\widetilde{\mathbf{x}}) = \max_{i=[1:C]} \frac{\exp(f_i(\widetilde{\mathbf{x}})/T)}{\sum_{j=1}^{C} \exp(f_j(\widetilde{\mathbf{x}})/T)}$$

$$\mathbf{ODIN}(\widetilde{\mathbf{x}}; \delta, T, \epsilon) = \begin{cases} \text{out}, & \text{if } \mathbf{SODIN}(\widetilde{\mathbf{x}}) \leq \delta \\ \text{in}, & \text{if } \mathbf{SODIN}(\widetilde{\mathbf{x}}) > \delta \end{cases}$$

- $f(\widetilde{\mathbf{x}})$ the vector of logits;
- $\widetilde{\mathbf{x}}$ represents a magnitude ϵ perturbation of the original $\mathbf{x}$;
- T is the temperature scaling parameter;
- $\delta \in [0, 1]$ is the threshold value;
- in indicates the acceptance decision;
- out indicates the rejection decision.

Competitors (SOTA Methods) for PBB

2) Mahalanobis distance [Lee et al., 2018]

$$\mathbf{M}(\widetilde{\mathbf{x}}) = \max_{c \in \mathcal{Y}} \; -(f(\widetilde{\mathbf{x}}) - \widehat{\mu}_c)^\top \widehat{\Sigma}^{-1} (f(\widetilde{\mathbf{x}}) - \widehat{\mu}_c)$$

$$\mathbf{MHLNB}(\widetilde{\mathbf{x}}; \zeta, \epsilon) = \begin{cases} \text{out}, & \text{if } \mathbf{M}(\widetilde{\mathbf{x}}) > \zeta \\ \text{in}, & \text{if } \mathbf{M}(\widetilde{\mathbf{x}}) \leq \zeta \end{cases}$$

- $\widehat{\mu}_c$ is the *empirical class mean* for each class c (training set);
- $\widehat{\Sigma}$ is the *empirical covariance* (trainig set);
- $f(\widetilde{\mathbf{x}})$ the vector of logits;
- $\widetilde{\mathbf{x}}$ represents a magnitude ϵ perturbation of the original $\mathbf{x}$;
- $\zeta \in \mathbb{R}_+$ is the threshold value;
- in indicates the acceptance decision;
- out indicates the rejection decision For a given $\mathbf{x} \in \mathcal{X}$.

TBB versus PBB

1) Softmax Response (SR) [Hendrycks and Gimpel, 2017, Geifman and El-Yaniv, 2017]
ODIN with $T = 1$ and $\epsilon = 0$.

2) Mahalanobis distance (MHLNB) [Lee et al., 2018]
Mahalanobis distance without input pre-processing and with the softmax output in place of the logits.

TBB

- Temperature scaling, $T = 1$
- Input pre-processing, $\epsilon = 0$

PBB

- D_α, $T_\alpha = 1$ and $\epsilon_\alpha = 0.00035$
- D_β, $T_\beta = 1.5$ and $\epsilon_\beta = 0.00035$
- ODIN, $T_{\textbf{ODIN}} = 1.3$ and $\epsilon_{\textbf{ODIN}} = 0$
- MHLNB, $T_{\textbf{MHLNB}} = 1$ and $\epsilon_{\textbf{MHLNB}} = 0.0002$

Discrimination performance for TBB

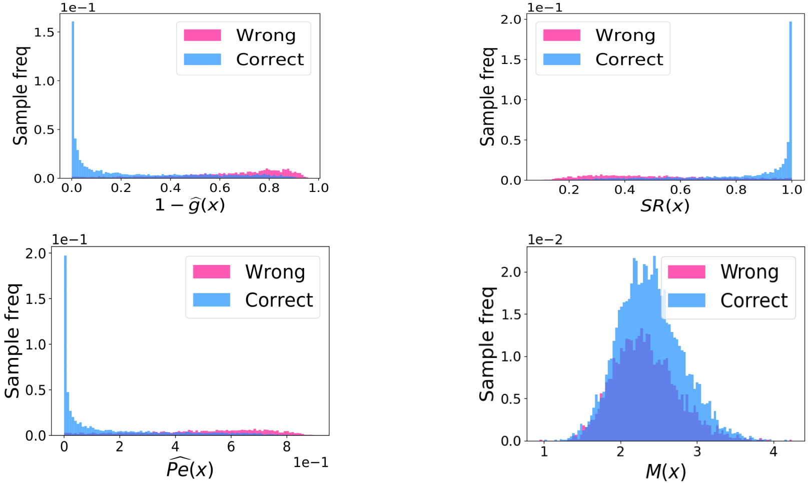


Figure 1. **DOCTOR**, **SR** and **MHLNB** to split data samples in TinyImageNet under TBB. Histograms for wrongly classified samples and correctly classified samples.

Discrimination performance for PBB

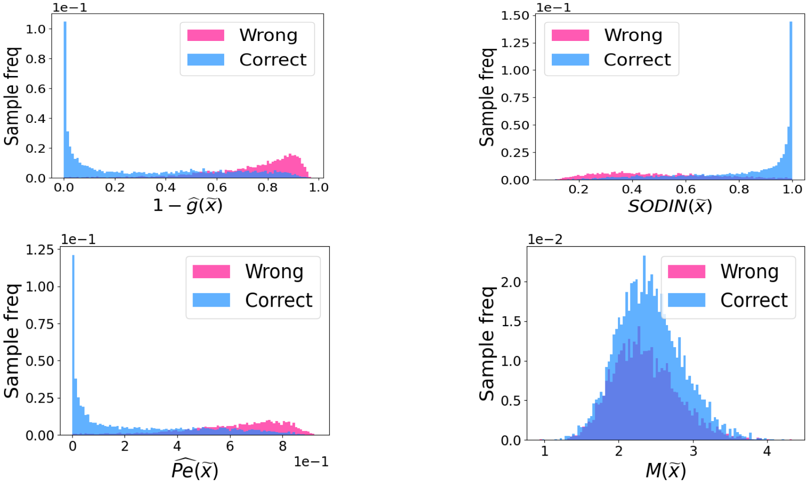


Figure 2. DOCTOR, **ODIN** and **MHLNB** to split data samples in TinyImageNet under PBB. Histograms for wrongly classified samples and correctly classified samples.

PBB: ROCs

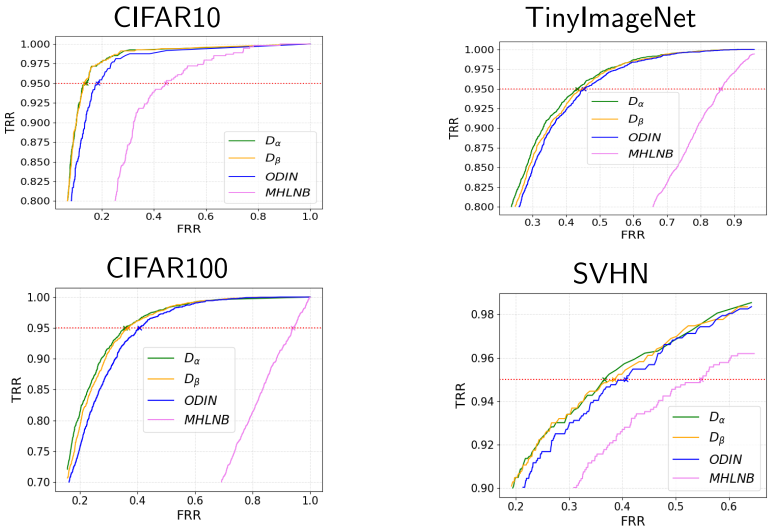


Figure 3. ROC curves. Comparison between **DOCTOR**, **ODIN** and **MHLNB**. The red dashed line marks the 95% threshold of TRR.

Overall Results: TBB & PBB

Table 1. Collection of the results in both **TBB** and **PBB**. For all methods, in TBB, we set $T = 1$ and $\epsilon = 0$; in PBB we set : $\epsilon_\alpha = 0.00035$ and $T_\alpha = 1$, $\epsilon_\beta = 0.00035$ and $T_\beta = 1.5$, $\epsilon_{\text{ODIN}} = 0$ and $T_{\text{ODIN}} = 1.3$, $\epsilon_{\text{MHLNB}} = 0.0002$ and $T_{\text{MHLNB}} = 1$. In TBB for ODIN we report same results as in SR, since both methods coincide when $T = 1$ and $\epsilon = 0$.

DATASET	METHOD	AUROC %		FRR % (95 % TRR)	
		TBB	PBB	TBB	PBB
CIFAR10 Acc. 95%	D_α	**94**	**95.2**	**17.9**	13.9
	D_β	68.5	94.8	18.6	**13.4**
	ODIN	93.8	94.2	18.2	18.4
	SR	93.8	-	18.2	-
	MHLNB	92.2	84.4	30.8	44.6
CIFAR100 Acc. 78%	D_α	**87**	**88.2**	40.6	**35.7**
	D_β	84.2	87.4	40.6	36.7
	ODIN	86.9	87.1	40.5	40.7
	SR	86.9	-	**40.5**	-
	MHLNB	82.6	50	66.7	94
TINY IMAGENET Acc. 63%	D_α	**84.9**	**86.1**	**45.8**	**43.3**
	D_β	**84.9**	85.3	**45.8**	45.1
	ODIN	84.9	84.9	45.8	45.3
	SR	**84.9**	-	**45.8**	-
	MHLNB	78.4	59	82.3	86

DATASET	METHOD	AUROC %		FRR % (95 % TRR)	
		TBB	PBB	TBB	PBB
SVHN Acc. 96%	D_α	**92.3**	**93**	**38.6**	**36.6**
	D_β	92.2	92.8	39.7	38.4
	ODIN	92.3	92.3	38.6	40.7
	SR	**92.3**	-	**38.6**	-
	MHLNB	87.3	88	85.8	54.7
AMAZON FASHION Acc. 85%	D_α	**89.7**	-	27.1	-
	D_β	**89.7**	-	**26.3**	-
	SR	87.4	-	50.1	-
AMAZON SOFTWARE Acc. 73%	D_α	**68.8**	-	**73.2**	-
	D_β	**68.8**	-	**73.2**	-
	SR	67.3	-	86.6	-
IMDB Acc. 90%	D_α	**84.4**	-	**54.2**	-
	D_β	**84.4**	-	54.4	-
	SR	83.7	-	61.7	-

Misclassification Detection in Presence of OOD Samples

- DOCTOR is not tuned for OOD detection (differently from ODIN).
- We test ODIN and DOCTOR when one sample to reject out of five (♣), three (♢), or two (♠) is OOD.

DATASET-In	DATASET-Out	AUROC %				FRR % (95 % TRR)			
		D_α	D_β	ODIN	ENERGY	D_α	D_β	ODIN	ENERGY
CIFAR10 ♣	iSUN	**95.4** / 0.1	95.1 / 0.1	94.6 / 0.1	92.4 / 0	14 / 0.5	**13.5** / 0.4	17.2 / 0.3	32.2 / 0.1
	Tiny (res)	**95.2** / 0.1	94.9 / 0	94.6 / 0.1	92.3 / 0.1	**14** / 0.4	**14** / 0.5	17.8 / 0.4	32.2 / 0.1
CIFAR10 ♢	iSUN	**95.5** / 0.1	95.3 / 0.1	94.9 / 0.1	92.9 / 0	14.4 / 0.6	**13.4** / 0.2	16.8 / 0.5	27 / 1
	Tiny (res)	**95.4** / 0.1	95 / 0.1	94.8 / 0.1	92.8 / 0	15 / 0.1	**14.8** / 0.7	17 / 0.5	28.8 / 1.9
CIFAR10 ♠	iSUN	**95.6** / 0.1	**95.6** / 0	95.4 / 0	93.6 / 0.1	15.1 / 0.1	**13.6** / 0.5	16.1 / 0.2	25.1 / 0.2
	Tiny (res)	**95.5** / 0.1	95.2 / 0.1	95.1 / 0.1	93.5 / 0	**14.7** / 0.3	14.8 / 0.5	17.1 / 0.4	25.6 / 0.3

Table 2. Results in terms of *mean / standard deviation*.

Takeaways from DOCTOR

* DOCTOR provides a flexible framework for miscalssification error detection that applies to any pre-trained DNN classifier.
* We leverage information-theoretic tools to better discriminate between trusted and untrusted model predictions.
* Our method adapts to various scenarios depending on the level of information access of the DNN, uses only the pre-trained model.

On-going work:

* Formalize statistical learning mechanisms that enable error detection and adaptation from few resources.
* Characterize their capabilities and limitations.
* Extension to semantic image segmentation, object detection and regression problems.

Supplementary: Optimal (Oracle) Discriminator

- $E \triangleq \mathbb{1}[Y \neq f_{\mathcal{D}_n}(\mathbf{X})]$ denotes the error variable corresponding to $f_{\mathcal{D}_n}$
- $\mathbf{x} \in \mathcal{X}$ and $y \in \mathcal{Y}$ drawn from the unknown distribution p_{XY}
- $p_{XY}(\mathbf{x}, y) \equiv P_E(1)p_{XY|E}(\mathbf{x}, y|1) + P_E(0)p_{XY|E}(\mathbf{x}, y|0)$
- $p_X(\mathbf{x}) \equiv P_E(1)p_{X|E}(\mathbf{x}|1) + P_E(0)p_{X|E}(\mathbf{x}|0)$
- $\mathsf{Pe}(\mathbf{x}) \triangleq \mathbb{E}[E(\mathbf{x})|\mathbf{x}] = 1 - P_{Y|X}(f_{\mathcal{D}_n}(\mathbf{x})|\mathbf{x})$ is the probability of error classification w.r.t. $P_{Y|X}$

$$
\begin{aligned}
D(\mathbf{x}, \gamma) &= \mathbb{1}[p_{X|E}(\mathbf{x}|1) > \gamma \cdot p_{X|E}(\mathbf{x}|0)] \\
&= \mathbb{1}[P_{E|X}(1|\mathbf{x})P_E(0) > \gamma \cdot (1 - P_{E|X}(1|\mathbf{x}))P_E(1)] \\
&= \mathbb{1}[\mathsf{Pe}(\mathbf{x})P_E(0) > \gamma \cdot (1 - \mathsf{Pe}(\mathbf{x}))P_E(1)] \\
&= \mathbb{1}[\mathsf{Pe}(\mathbf{x}) > \gamma' \cdot (1 - \mathsf{Pe}(\mathbf{x}))],
\end{aligned}
$$

where $\gamma' = \frac{P_E(1)}{P_E(0)}$.

References I

Geifman, Y. and El-Yaniv, R. (2017).
Selective classification for deep neural networks.
In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4878–4887.

Hendrycks, D. and Gimpel, K. (2017).
A baseline for detecting misclassified and out-of-distribution examples in neural networks.
In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

Lee, K., Lee, K., Lee, H., and Shin, J. (2018).
A simple unified framework for detecting out-of-distribution samples and adversarial attacks.
In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 7167–7177.

References II

Liang, S., Li, Y., and Srikant, R. (2018).
Enhancing the reliability of out-of-distribution image detection in neural networks.
In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.

IGEOOD: An Information Geometry Approach to Out-of-Distribution Detection

Joint work with Eduardo D. C. Gomes, Florence Alberge and Pierre Duhamel

ICLR
International Conference On Learning Representations

(https://openreview.net/pdf?id=mfwdY3U_9ea)

Motivation

- We introduce IGEOOD, an effective method for detecting **Out-of-Distribution (OOD)** samples.
- IGEOOD applies to any pre-trained neural network, works under different degrees of access to the ML model, does not require OOD samples or assumptions on the OOD data but can also benefit (if available) from OOD samples.
- By building on the geodesic (**Fisher-Rao**) distance between the underlying data distributions, our discriminator combines confidence scores from the logits outputs and the learned features of a deep neural network.

Background

- Let $\mathcal{X} \subseteq \mathbb{R}^d$ be the feature space and $\mathcal{Y}$ a label space and let p_{XY} be the underlying unknown probability density function (pdf) over $\mathcal{X} \times \mathcal{Y}$.
- In order to model the underlying problem, we introduce an artificial binary random variable $Z \in \{0, 1\}$ indicating with $z = 1$ that the test sample $\boldsymbol{x}$ is OOD and $z = 0$ otherwise.
- The open-world data can then be modeled as a *mixture* distribution $p_{X|Z}$ defined by

$$p_{X|Z}(\boldsymbol{x}|z = 0) \triangleq p_X(\boldsymbol{x}), \quad p_{X|Z}(\boldsymbol{x}|z = 1) \triangleq q_X(\boldsymbol{x}).$$

- The intrinsic difficulty arises from the fact that very little can be assumed about the unknown distributions p_X and q_X, in particular for out-of-distribution.
- **Alternative:** distance based criteria w.r.t an in-distribution probability reference.

Statistical Model

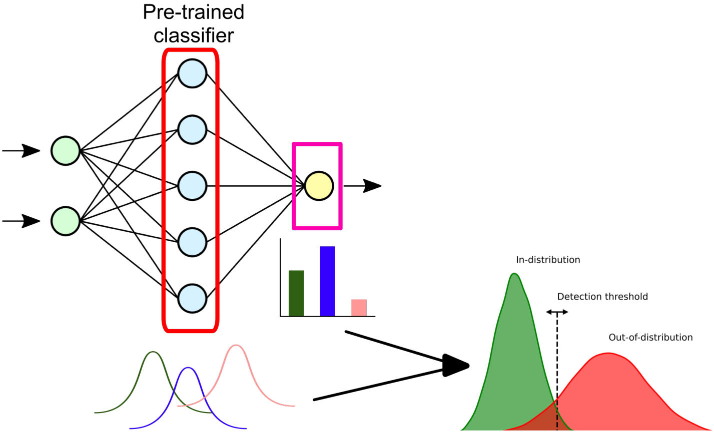


Figure: We model the hidden layers' outputs as class conditional Gaussian distributions and the DNN's outputs as softmax probability distributions.

We propose an OOD detector based on the geodesic Fisher-Rao distance between probability density functions:

$$d_{\mathrm{FR}}(q_{\boldsymbol{\theta}}, q'_{\boldsymbol{\theta}}) \triangleq \inf_{\gamma} \int_0^1 \sqrt{\frac{d\gamma^\top(t)}{dt} \boldsymbol{G}(\gamma(t)) \frac{d\gamma(t)}{dt}} dt$$

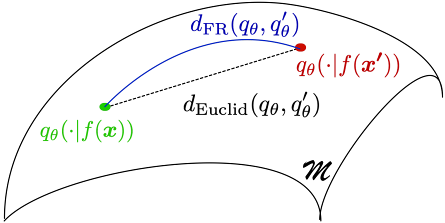


Figure: Illustration of the shortest path between distributions in a statistical manifold.

IGEOOD Score Using the Soft-Predictions

- **Igeood score using the softmax probability:** Let $q_{\boldsymbol{\theta}}(\cdot|f(\boldsymbol{x}))$ be the softmax probability distribution of the outputs. We can define the Fisher-Rao distance between softmax distributions as:

$$d_{\mathrm{FR-Logits}}(q_{\boldsymbol{\theta}}, q'_{\boldsymbol{\theta}}) \triangleq 2 \arccos\left(\sum_{y \in \mathcal{Y}} \sqrt{q_{\boldsymbol{\theta}}(y|f(\boldsymbol{x}))\, q_{\boldsymbol{\theta}}(y|f(\boldsymbol{x}'))}\right)$$

- From which we derive our IGEOOD score for the logits:

$$\mathrm{FR}_0(\boldsymbol{x}) \triangleq \sum_{y \in \mathcal{Y}} d_{\mathrm{FR-Logits}}\big(q_{\boldsymbol{\theta}}(\cdot|f(\boldsymbol{x})), q_{\boldsymbol{\theta}}(\cdot|\boldsymbol{\mu}_y)\big)$$

- Where $\boldsymbol{\mu}_y$ are the class conditional centroids given by:

$$\boldsymbol{\mu}_y \triangleq \min_{\boldsymbol{\mu} \in \mathbb{R}^{|\mathcal{Y}|}} \frac{1}{N_y} \sum_{\forall\, i\,:\, y_i = y} d_{\mathrm{FR-Logits}}\big(q_{\boldsymbol{\theta}}(\cdot|f(\boldsymbol{x}_i)), q_{\boldsymbol{\theta}}(\cdot|\boldsymbol{\mu})\big).$$

IGEOOD Score Leveraging Latent Features

- **Igeood score leveraging latent features:** For each layer, we model the features as a set of class-conditional Gaussian distributions with diagonal standard deviation matrix:

$$\boldsymbol{\mu}_y^{(\ell)} = \frac{1}{N_y} \sum_{\forall i : y_i = y} f^{(\ell)}(\boldsymbol{x}_i)$$

$$\boldsymbol{\sigma}^{(\ell)} = \operatorname{diag}\left(\sqrt{\frac{1}{N} \sum_{y \in \mathcal{Y}} \sum_{\forall i : y_i = y} \left(f_j^{(\ell)}(\boldsymbol{x}_i) - \mu_{y,j}^{(\ell)} \right)^2} \right).$$

- We derive a confidence score by calculating the Fisher-Rao distance between the test sample $\boldsymbol{x}$ and the closest class-conditional diagonal Gaussian distribution:

$$\mathrm{FR}_\ell(\boldsymbol{x}) = \min_{y \in \mathcal{Y}} d_{\mathrm{FR-Gauss}}\left(\left(\boldsymbol{x}, \boldsymbol{\sigma}^{(\ell)} \right), \left(\boldsymbol{\mu}_y^{(\ell)}, \boldsymbol{\sigma}^{(\ell)} \right) \right).$$

Feature Ensemble

- **Feature ensemble:** we combine the confidence scores of the logits and low-level features through a linear combination. If OOD data is available, we can also calculate $\mathrm{FR}'_\ell\left(\boldsymbol{x};\boldsymbol{\mu}^{(\ell)\prime},\boldsymbol{\sigma}^{(\ell)\prime}\right)$ with OOD statistics, obtaining $\textsc{IGEOOD}+$:

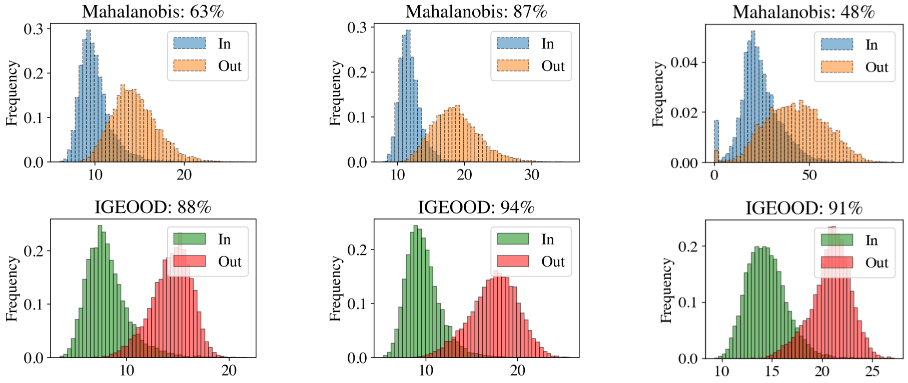
$$\mathrm{FR}(\boldsymbol{x}) \triangleq \alpha_0 \mathrm{FR}_0(\boldsymbol{x}) + \sum_\ell \alpha_\ell \cdot \mathrm{FR}_\ell(\boldsymbol{x}) + \alpha'_\ell \cdot \mathrm{FR}'_\ell(\boldsymbol{x}).$$

- *Therefore, we have derived a unified OOD detection framework that combines a single distance for both the softmax outputs and the latent features of a neural network.*

Experimental Setup

- The experimental setup follows the setting established by [1, 2, 4].
- We use two *pre-trained* deep neural networks architectures for image classification tasks: a Dense Convolutional Network (DenseNet-BC-100) and a Residual Neural Network (ResNet-34).
- *in-distribution data*: images from CIFAR-10, CIFAR-100 and SVHN datasets.
- *out-of-distribution data*: natural image examples from Tiny-ImageNet, LSUN, Describable Textures Dataset, Chars74K, Places365, iSUN and a synthetic dataset generated from Gaussian noise.

- The IGEOOD score increases the separation between in- and out-of-distribution data.



(a) Block 1. (b) Block 2. (c) Block 3.

Figure: Histograms of the Mahalanobis and IGEOOD scores for the outputs of each hidden block of a DenseNet model.

Black-Box Results

Table: Average and standard deviation OOD detection performance across various OOD datasets for each model and in-distribution dataset in a Black-Box setting. IGEOOD is compared to Baseline [1], ODIN [2], and Energy [3] methods.

		TNR at TPR-95%	AUROC
Model	In-dist.	Baseline / ODIN / Energy / IGEOOD (ours)	
DenseNet	C-10	52.5$\pm$16/**66.8**$\pm$20/65.3$\pm$23/65.6$\pm$23	91.8$\pm$3.2/**92.8**$\pm$4.6/92.1$\pm$5.3/92.3$\pm$5.1
	C-100	15.9$\pm$6.8/20.5$\pm$9.5/20.3$\pm$9.6/**20.7**$\pm$9.8	69.1$\pm$15/71.6$\pm$20/71.6$\pm$20/**73.2**$\pm$17
	SVHN	68.4$\pm$14/68.8$\pm$20/70.2$\pm$17/**72.1**$\pm$15	**92.3**$\pm$4.0/87.3$\pm$14/90.1$\pm$5.9/90.9$\pm$5.3
ResNet	C-10	41.7$\pm$16/51.9$\pm$15/56.3$\pm$13/**56.7**$\pm$13	89.6$\pm$3.1/90.4$\pm$3.1/90.4$\pm$3.0/**90.5**$\pm$3.0
	C-100	15.0$\pm$5.5/16.0$\pm$6.3/16.3$\pm$7.1/**16.4**$\pm$6.8	74.0$\pm$1.9/75.2$\pm$1.7/**75.5**$\pm$1.9/**75.5**$\pm$1.7
	SVHN	76.2$\pm$7.8/77.7$\pm$7.9/78.0$\pm$7.9/**78.3**$\pm$8.0	**92.2**$\pm$2.9/91.4$\pm$3.2/91.4$\pm$3.2/91.7$\pm$3.2
Average and Std.		44.9$\pm$24/50.3$\pm$24/51.1$\pm$24/**51.6**$\pm$24	84.8$\pm$9.5/84.8$\pm$8.3/85.2$\pm$8.4/**85.7**$\pm$8.0

White-Box Results

- We increase the average TNR-95% by 11.8% and 2.5% with validation on OOD and adversarial data, respectively.

Table: Average and standard deviation of OOD detection performance for the White-Box settings. The abbreviation TNR-95%, C-10 and C-100 stands for TNR at TPR-95%, CIFAR-10 and CIFAR-100, respectively.

		Validation on OOD data		Validation on adversarial data	
		TNR-95%	AUROC	TNR-95%	AUROC
Model	In-dist.	Mahalanobis / IGEOOD+ (ours)		Mahalanobis / IGEOOD (ours)	
DenseNet	C-10	76.6±31/**92.6**±14	92.1±12/**98.4**±3.0	75.9±30/**77.9**±29	91.7±12/**94.0**±9.0
	C-100	67.2±28/**90.2**±21	90.2±13/**97.7**±5.0	60.4±34/**70.9**±35	85.3±19/**90.8**±13
	SVHN	93.3±8.0/**98.0**±2.0	98.6±1.0/**99.6**±0.1	**93.7**±10/92.2±9.0	**98.6**±2.0/98.4±1.0
ResNet	C-10	82.5±23/**91.6**±16	96.5±4.0/**98.4**±3.0	**78.6**±24/77.3±32	**95.3**±6.0/90.0±15
	C-100	70.4±30/**86.4**±23	91.9±10/**97.1**±5.0	57.4±36/**65.1**±33	86.9±13/**88.6**±15
	SVHN	96.8±6.0/**98.9**±2.0	99.2±1.0/**99.7**±0.1	**96.3**±8.0/93.6±14	**99.1**±1.0/98.4±3.0
Average and Std.		81.1±11/**92.9**±4.0	94.8±4.0/**98.5**±1.0	77.0±15/**79.5**±10	92.8±5.4/**93.4**±3.9

Further Comparison to the Literature

Table: TNR at TPR-95% (%) performance comparison in a White-Box setting considering the original results from [1,2,3,4]. Methods with an (*) were tuned without OOD data.

	OOD dataset	CIFAR-10	CIFAR-100	SVHN
		Mahalanobis [4] / Gram Matrix* [5] / DeConf-C* [6] / Res-Flow [7] / IGEOOD / IGEOOD+		
DenseNet	iSUN	95.3/99.0/ - / - /97.7/**99.8**	87.0/95.9/ - / - /93.8/**99.7**	**99.9**/99.4/ - / - /98.3/**99.9**
	LSUN	97.2/99.5/99.4/98.2/98.5/**99.9**	91.4/97.2/98.7/96.3/95.2/**99.9**	**99.9**/99.5/ - /**100**/97.1/**99.9**
	TinyImgNet	95.0/98.8/99.1/96.4/95.7/**99.8**	86.6/95.7/98.6/93.0/94.5/**99.5**	**99.9**/99.1/ - /**100**/98.2/**99.9**
	SVHN/C-10	90.8/96.1/98.8/94.9/98.9/**99.9**	82.5/89.3/95.9/84.9/93.3/**99.6**	96.8/80.4/ - /**99.0**/91.6/98.3
ResNet	iSUN	97.8/99.3/ - / - /97.2/**99.9**	89.9/94.8/ - / - /93.4/**99.8**	99.7/99.4/ - / - /99.8/**100**
	LSUN	98.8/99.6/ - /99.0/98.4/**100**	90.9/96.6/ - /96.2/94.3/**100**	**99.9**/99.6/ - /**100**/99.7/**99.9**
	TinyImgNet	97.1/98.7/ - /97.8/96.3/**99.6**	90.9/94.8/ - /94.6/90.1/**99.6**	**99.9**/99.3/ - /**100**/99.7/**99.9**
	SVHN/C-10	87.8/97.6/ - /96.5/98.8/**99.8**	91.9/80.8/ - /93.0/91.6/**99.7**	98.4/85.8/ - /99.4/97.7/**99.7**

Takeaways from IGEOOD

- IGEOOD provides a flexible framework for OOD detection that applies to any pre-trained DNN classifier.
- We leverage information geometry tools to better discriminate between probability distributions.
- Our method adapts to various scenarios depending on the level of information access of the DNN, uses only in-distribution samples but can also benefit (if available) of OOD samples.

On-going work:

- Formalize hypothetical learning mechanisms that enable OOD generalization and adaptation.
- Characterize their capabilities and limitations.
- Extension to time-series and progressive distribution/model drifts.

References

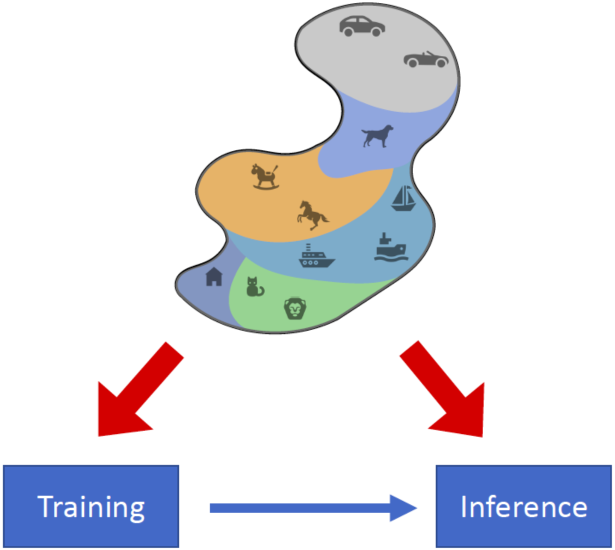
[1] Dan Hendrycks & Kevin Gimpel. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. 2017.
[2] Shiyu Liang et al. Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks. 2018.
[3] Liu et al. Energy-based Out-of-distribution Detection. 2020.
[4] Kimin L. et al. A simple unified framework for detecting out-of-distribution samples and adversarial attacks, 2018.
[5] Sastry & Oore. Detecting out-of-distribution examples with Gram matrices, 2020.
[6] Hsu et al. Generalized ODIN: Detecting out-of-distribution image without learning from out-of-distribution data, 2020.
[7] Zisselman & Tamar. Deep residual flow for novelty detection, 2020.

Adversarial Robustness via Fisher-Rao Regularization

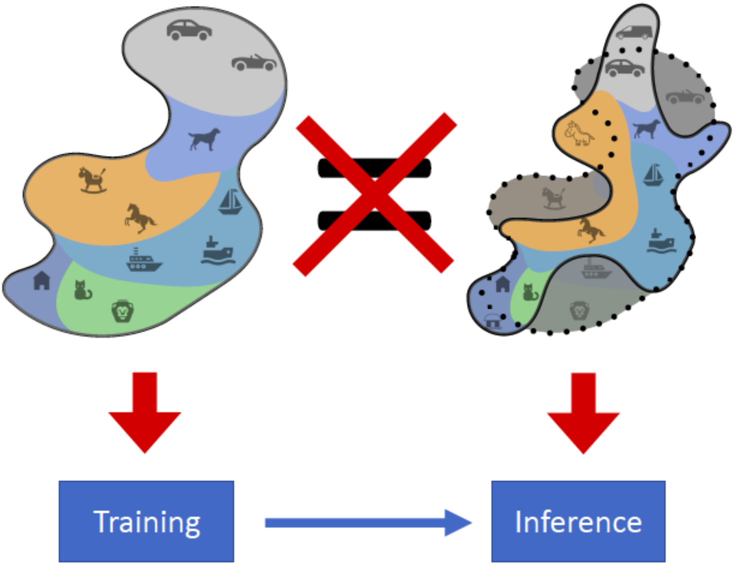
Joint work with Marine Picot, Francisco Messina, Malik Boudiaf, Fabrice Labeau, Ismail Ben Ayed

(https://arxiv.org/abs/2106.06685)

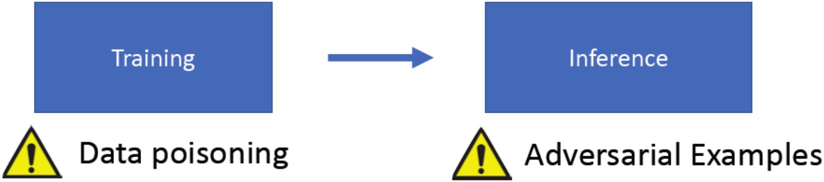
Deep Neural Networks

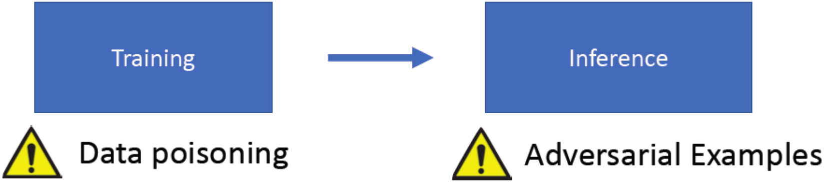


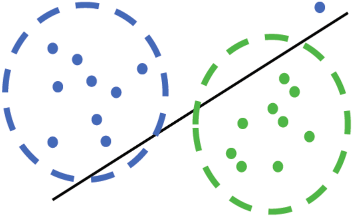
Deep Neural Networks



Attacking Deep Neural Networks







Data poisoning: modification of the boundaries

Adversarial Examples

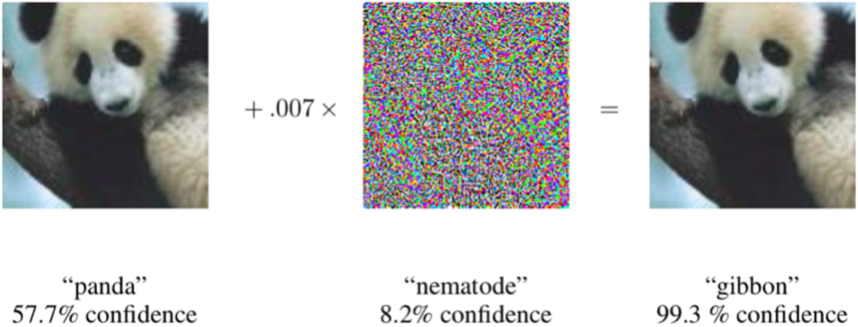


Figure: Building adversarial examples [Ian J Goodfellow et al. Arxiv 2014]

Adversarial Examples

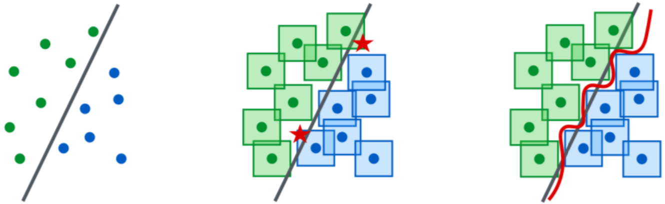


Figure: "Natural" vs "Adversarial" decision boundaries [A. Madry et al. ICLR 2018]

Motivation

- Security

Glasses that fool face recognition [Mahmood Sharif et al. CCS 2016]

Motivation

- Security and Safety

Fooling autonomous car [Nir Morgulis et al. arXiv 2019.]

Main Defintions

Let us consider the multi-class classification problem with:

- $\mathcal{X} \subseteq \mathbb{R}^n$ is the input space.
- $\mathcal{Y} = \{1, \ldots, M\}$ is the label (concept) space.
- q_θ is the general classification model, parametrized by $\theta \in \Theta$.
- $P_e(\theta)$ is the error probability of the model parametrized by $\theta \in \Theta$.
- $\ell(\theta; \boldsymbol{x}, y)$ is the loss of the model parametrized by θ, computed for the input $(\boldsymbol{x}, y)$ and its expectation is the risk $\mathcal{L}(\theta)$.
- ε is the maximal distortion allowed in the adversarial problem, according to a specific L^p-norm.
- $\boldsymbol{x}'$ refers to the adversarial version of any variable $\boldsymbol{x}$.

Problem Statement of Adversarial Attacks

Definition (Adversarial attacks)

The adversarial problem [1] is defined, according to L^p-norm, as:

$$\begin{aligned} \boldsymbol{x}^\star(\boldsymbol{x}) \equiv \quad & \arg \min_{\boldsymbol{x}' \in [0,1]^n \,:\, \|\boldsymbol{x}' - \boldsymbol{x}\|_p < \varepsilon} \quad \|\boldsymbol{x}' - \boldsymbol{x}\|_p \\ & \text{s.t.} \quad f_\theta(\boldsymbol{x}') = t \end{aligned}$$

where

- t is the target class or any class different from the original label y,
- $\boldsymbol{x}' \in [0,1]^n$ assures that $\boldsymbol{x}^\star(\boldsymbol{x})$ is close enough to the original image.

[1] Christian Szegedy et al. *Intriguing properties of neural networks* ICLR 2014.

Fast Gradient Sign Method (FGSM)

Definition (FGSM Algorithm [Ian J Goodfellow et al. 2014])

$$\boldsymbol{x}' = \boldsymbol{x} + \alpha \,\mathrm{sgn}\left(\nabla_{\boldsymbol{x}}\, \ell(\theta; \boldsymbol{x}, y)\right),$$

where

- $(\boldsymbol{x}, y)$: clean example
- $\boldsymbol{x}'$: adversarial example
- sgn : the sign function
- $\nabla_{\boldsymbol{x}}\, \ell(\theta; \boldsymbol{x}, y)$: the gradient w.r.t. $\boldsymbol{x}$ of the loss function $\ell(\theta; \boldsymbol{x}, y)$ evaluated at $(\boldsymbol{x}, y)$
- $\alpha \leq \varepsilon$: parameter controlling the magnitude of the perturbation.

Overview of PGD Attack

Definition (PGD Attack [A. Madry et al. ICLR 2018])

- It is the iterative extension of the FGSM method
- For a certain number of iterations k, we apply at each iteration i:

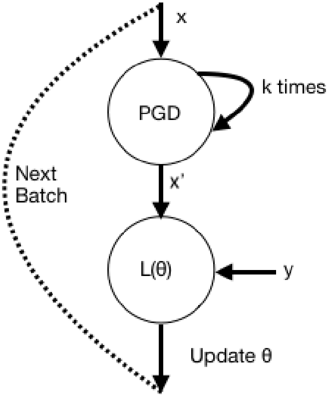
$$\boldsymbol{x}'^{(i+1)} = \boldsymbol{x}'^{(i)} + \delta \cdot \text{sgn}\left(\nabla_{\boldsymbol{x}} \, \ell(\theta; \boldsymbol{x}'^{(i)}, y)\right),$$

  where $\delta \leq \varepsilon$ is the noise norm at each step.
- $\boldsymbol{x}'^{(0)}$ is either equal to $\boldsymbol{x}$ or $\boldsymbol{x} + \boldsymbol{\eta}$ where $\boldsymbol{\eta}$ is a random noise of maximum amplitude ε.
- To ensure that the L^p-norm constraint is met, at each iteration, we have to force: $\|\boldsymbol{x}'^{(i)} - \boldsymbol{x}\|_p \leq \varepsilon$.

Adversarial Training

Definition (Adversarial defense)

$$\theta^\star \equiv \arg\min_\theta \mathbb{E}_{\boldsymbol{X}Y}\left[\max_{\boldsymbol{x'}:\|\boldsymbol{x'}-\boldsymbol{x}\|_\infty\leq\varepsilon} \ell(\theta;\boldsymbol{x'},y)\right]$$



Relaxation hypothesis: We can approximate the max part with the generation of an adversarial example.

Losses for Adversarial Training

Definition (Madry's method for defense [A. Madry et al. ICLR 2018])

Consider the adversarial cross-entropy loss (ACE):

$$\ell(\theta; \boldsymbol{x'}, y) = -\log[q_\theta(y|\boldsymbol{x'})].$$

Definition (TRADES [Hongyang Zhang et al. ICML 2019])

Trade-off between natural and robust accuracies:

$$\ell(\theta; \boldsymbol{x'}, y) = -\log[q_\theta(y|\boldsymbol{x})] + \lambda \cdot d_{KL}\big(q_\theta(y|\boldsymbol{x}) \| q_\theta(y|\boldsymbol{x'})\big),$$

where λ is the hyperparameter controlling the trade-off between natural and adversarial accuracies.

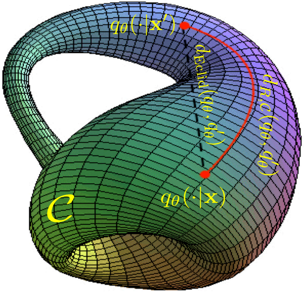
Robustness cannot be ensured against all adversarial (losses) attacks. Can we derive an universal defense?

Fisher-Rao Riemannian Geometry

Definition (Fisher-Rao Distance (FRD))

- Given a family of probability distributions:
$$\mathcal{C} = \{q_\theta(\cdot|\mathbf{x}) : \mathbf{x} \in \mathcal{X}\}.$$
- Metric tensor (Fisher information):
$$G(\mathbf{x}) = \mathbb{E}_{Y \sim q_\theta(\cdot|\mathbf{x})}\left[\nabla_{\mathbf{x}} \log q_\theta(Y|\mathbf{x}) \nabla_{\mathbf{x}}^{\intercal} \log q_\theta(Y|\mathbf{x})\right]$$
is positive definite for any $\mathbf{x}$ and $\theta \in \Theta$.
- Infinitesimal squared length element:
$$ds^2 = \langle d\mathbf{x}, d\mathbf{x}\rangle_{G(\mathbf{x})} = d\mathbf{x}^{\intercal} G(\mathbf{x}) d\mathbf{x}.$$
- The FRD between $q_\theta(\cdot|\mathbf{x})$ and $q_\theta(\cdot|\mathbf{x}')$ is:
$$d_{R,\mathcal{C}}(q_\theta, q'_\theta) = \inf_{\gamma} \int_0^1 \sqrt{\frac{d\gamma^{\intercal}(t)}{dt} G(\gamma(t)) \frac{d\gamma(t)}{dt}},$$
the inf is over all piecewise smooth curves.
- FRD is the length of the geodesic between $(\mathbf{x}, \mathbf{x}')$ using $G(\mathbf{x})$ as the metric tensor.

R. Rao and R. Fisher, 1956



FIRE: Fisher-Rao Regularization

Definition (Fisher-Rao Distance (FRD))

We define the FIRE loss function as the trade-off between the natural cross-entropy and the expected Fisher-Rao distance between natural and adversarial probability distributions:

$$\ell_{\mathrm{FIRE}}(\theta; \boldsymbol{x}, y) = -\log q_\theta(y|\boldsymbol{x}) + \lambda \cdot d_R^2(q_\theta(\,\cdot\,|\boldsymbol{x}), q_\theta(\cdot|\boldsymbol{x'})),$$

where λ is the hyperparameter controlling the trade-off between natural and adversarial performances with

$$d_R(q_\theta(.|\boldsymbol{x}), q_\theta(.|\boldsymbol{x'}) = 2 \arccos \left(\sum_{y \in \mathcal{Y}} \sqrt{q_\theta(y|\boldsymbol{x}) q_\theta(y|\boldsymbol{x'})} \right).$$

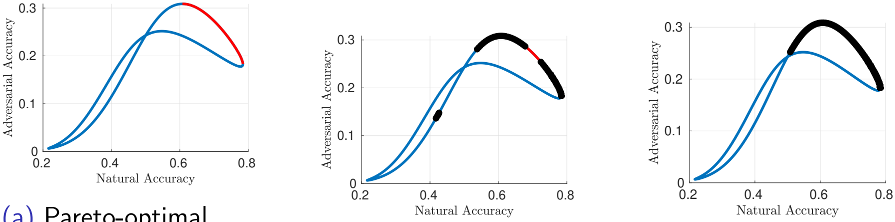
This metric has very interesting properties and **is related to well-known distances and Information divergences.**

Comparison to the Kullback-Leiber Distance

Definition (Binary logistic regression)

Assume two equally likely classes $\mathcal{Y} = \{-1, 1\}$ with conditional inputs given by $\mathbf{x}|y \sim \mathcal{N}(y\mu, \Sigma)$, and softmax probability

$$q_\theta(y|\mathbf{x}) = \frac{1}{1 + \exp(-y\,\theta^\intercal \mathbf{x})}.$$



(a) Pareto-optimal points

(b) TRADES

(c) FIRE

Figure: Plot of all possible pairs $(1 - P_e(\theta), 1 - P'_e(\theta))$ for Gaussian model $\varepsilon = 0.1$

Experimental Set-Up

- **Datasets:** MNIST, CIFAR-10 - with and without additional data (AD) - , CIFAR-100
- **Model architecture:** CNNs, ResNet
- **Training procedure without AD:** Number of epochs : 100, batch size : 256, optimizer: SGD with a 0.9 momentum, and 1.10^{-4} weight decay, l_r: 0.01 for MNIST, and to 0.1 for CIFAR-10 and CIFAR-100, l_r decay :divided by 10 at epochs 75 and 90.
- **Changes for AD simulations:** Number of epochs: 200, l_r decay: cosine.
- **Generation of adversarial examples:** PGD.
- **Additional data**: 500k additional images from 80M-TI[1], selected such that the l_2-norm between those images and the images from CIFAR-10 are below a threshold.

[1] Images available at https://github.com/yaircarmon/semisup-adv

Comparison between KL and Fisher-Rao Regularization

Table: Comparison between KL and Fisher-Rao based regularizer under white-box l_∞ threat model.

Defense	Dataset	ε	Structure	Natural	AutoAttack	AA	RunTime
TRADES	MNIST	0.3	CNN	99.35	92.91	96.13	2h22
FIRE			CNN	99.13	94.06	96.59	2h06
TRADES	CIFAR-10	8/255	WRN-34-10	86.01	50.26	68.13	13h49
FIRE			WRN-34-10	85.42	52.22	68.82	11h00
TRADES	CIFAR-100	8/255	WRN-34-10	59.76	26.09	42.92	13h49
FIRE			WRN-34-10	60.71	27.63	44.17	11h10

Comparison to SOTA Defense Mechanisms

Table: Test robustness on different datasets under white-box l_∞ attack. '*' indicates models were retrained. '-' indicates the result is unavailable.

Defense	Dataset	ε	Structure	Natural	AutoAttack	AA	Runtime
Without Additional Data							
Madry et al.	MNIST	0.3	CNN	98.53	88.50	93.51	2h03
Atzmon et al.			CNN	99.35	90.85	95.10	-
TRADES *			CNN	99.35	92.91	96.13	2h22
FIRE			CNN	99.14	94.06	96.60	2h06
Madry et al.	CIFAR-10	8/255	WRN-34-10	87.14	44.04	65.59	10h51
TRADES *			WRN-34-10	84.79	51.92	68.35	13h49
Self Adaptive			WRN-34-10	83.48	53.34	68.41	13h57
Overfitting *			WRN-34-10	86.85	51.74	69.29	42h01
FIRE			WRN-34-10	85.20	53.49	69.35	11h00
Overfitting	CIFAR-100	8/255	RN-18	53.83	18.95	36.39	-
Overfitting*			WRN-34-10	59.01	27.07	43.04	42h08
FIRE			WRN-34-10	60.71	27.63	44.17	11h10
With Additional Data Using 80M-TI							
Pre-training	CIFAR-10	8/255	WRN-28-10	87.10	54.92	71.01	13h51
UAT			WRN-106-8	86.46	56.03	71.24	-
MART			WRN-28-10	87.50	56.29	71.89	10h22
RST-adv			WRN-28-10	89.70	59.53	74.61	22h12
FIRE			WRN-28-10	89.77	59.93	74.85	18h30

Takeaways from FIRE

- FIRE is a novel method using tools from information geometry that encourages invariant softmax probabilities for natural and adversarial examples while maintaining high performances on natural samples.
- Theoretically, the optimization based on FIRE is well-behaved and gives all the desired Pareto-optimal points.
- Our empirical results showed that FIRE consistently enhances the robustness compared to TRADES.
- Compared to the state-of-the-art methods for adversarial defenses, FIRE increases the Average Accuracy (AA) while reducing the training time by 20%.

On-going work:

- Our framework might be used to devise novel detection methods of adversarial examples.
- Characterize capabilities and limitations of potential attacks.
- Auditing mechanisms for ML models, based on partial statistical knowledge of the underlying distribution.

Outline

Information Measures are Building Blocks of ML Systems

A long-lasting partnership:

- Learning is data compression.
- Concepts of data representations (e.g., encoders/decoders).
- Several information-based objectives (e.g., cross-entropy loss, mutual information,...), maximum information gain principle.
- Shannon entropy is a measure of randomness (or uncertainty).
- Minimum entropy principle is fundamental in statistical estimation and learning.

Nonetheless, there is a long way to go:

- It is fundamentally important **to study other measures of information having more appropriate properties** from the viewpoint of its own learning problems.
- How we find an appropriate and universal way **to measure and to detect model uncertainty?**

Open Research Questions

From empirical evidence to information and knowledge:

- Researchers often have a tendency to fixate on **model performance metrics**, e.g., accuracy, but metrics **only tell part of the story** of a model's predictive decisions.
- It is important to understand what **drives a model to make predictions** (learning nature, not only imitation).

Uncertainty & robustness are critical problems in modern AI:
Models are often wrong, but **AI models that know when they are wrong are more useful.**

Better understanding of the information-theoretic link between:

- **Data** (source of empirical evidence),
- **Information** (the part that is unpredictable from the data),
- **Redundancy** (structure in data that provides the knowledge),
- **Knowledge** (the explanations of the complex world).

Joint Work with PhD Students and Collaborators









- Federica Granese, Marine Picot, Eduardo D. C. Gomes, Francisco Messina, Malik Boudiaf, Marco Romanelli, Ismail Ben Ayed, Catuscia Palamidessi, Pierre Duhamel, Florence Alberge, Fabrice Labeau.

Thank you for your attention