Decoding Misclassification Errors through Model Prediction Diversity

Pablo Piantanida
pablo.piantanida@cnrs.fr

International Laboratory on Learning Systems (ILLS)
CNRS CentraleSupélec - Université Paris-Saclay

Bellairs Workshop, January 22th, 2024

Outline

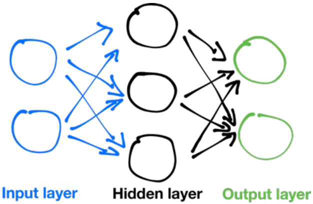
1. Introduction and Background
   - Uncertainty in Machine Learning
   - Information and Diversity Measures
2. A Novel Approach to Misclassification Detection
   - Measuring the Diversity of Predicted Categories
   - A Data-Driven Approach for Measuring the Diversity of Predicted Categories
3. Discussion and Research Perspectives

ILLS
International Laboratory on Learning Systems

Outline

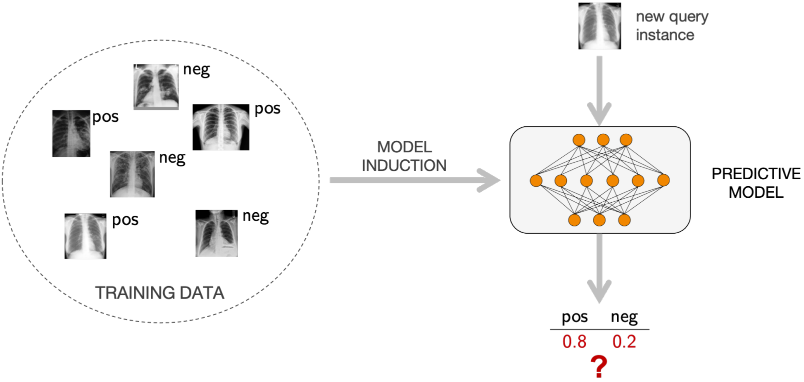
ILLS
International Laboratory on Learning Systems

Uncertainty in Machine Learning

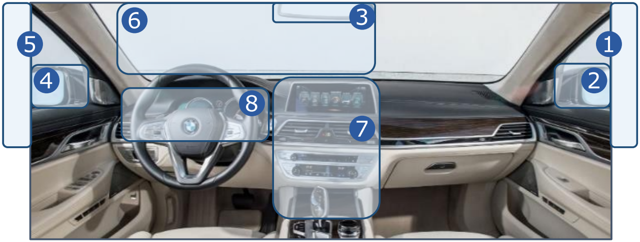


ILLS
International Laboratory on Learning Systems

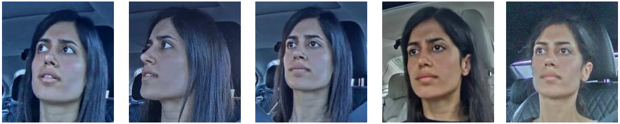
Need for Uncertainty-Awareness of ML Systems



Need for Uncertainty-Awareness of ML Systems



Regions of interest: right shoulder, right mirror, inner mirror, left mirror, left shoulder, front, center stack, speedometer.

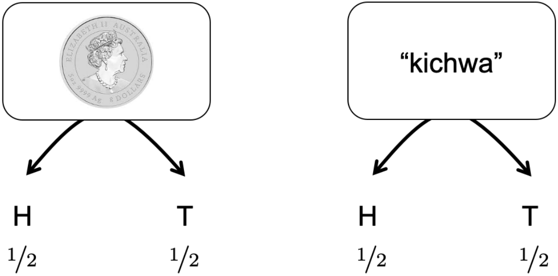


a) b) c) d) e)

Uncertainty-Aware ML Systems

- Ideally, when making a **prediction**, the learner knows what it knows and, perhaps more importantly, **what it does not know**
- This requires an adequate **representation** of predictions (e.g., in terms of distributions or sets) and **quantification** (e.g., in terms of entropy) of their uncertainty, ...
- ... as well as suitable **learning algorithms** to produce p-valued predictors (e.g., conformal learning)
- In this regard, a distinction between different **sources and types of uncertainty** turns out to be meaningful, notably
  - **aleatoric** (inherent randomness, irreducible)
  - **epistemic** (lack of knowledge, reducible).

Aleatoric versus Epistemic Uncertainty



"Not knowing the chance of mutually exclusive events and knowing the chance to be equal are two quite different states of knowledge"

Ronald Fisher (1890-1962)

Aleatoric vs. Epistemic: Two Main Types of Uncertainty

Aleatoric Uncertainty

Uncertainty stemming from inherent variability in the data itself

- **Examples:** Noisy sensor data, variability in human behavior
- **Modeling aleatoric uncertainty:** Incorporating noise models in the learning process.

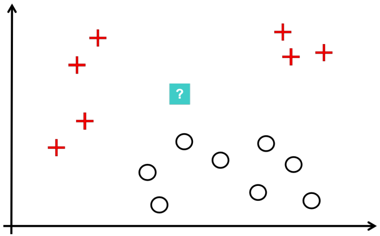
Epistemic Uncertainty

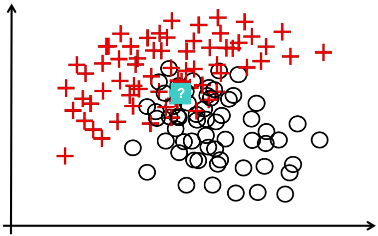
Uncertainty arising from a lack of knowledge or information

- **Bayesian perspective:** Treating model parameters as random variables (computationally expensive in large models)
- **Quantifying epistemic uncertainty:** Through model ensembles, dropout methods, or Bayesian inference.

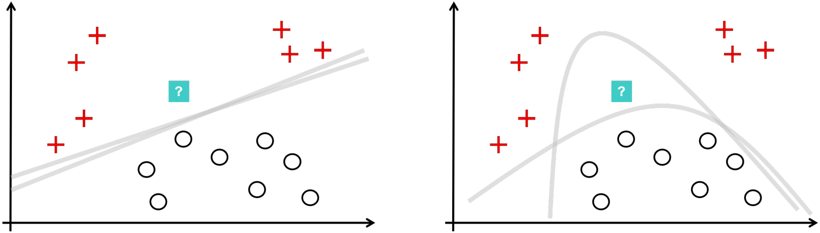
Aleatoric versus Epistemic Uncertainty

- Both types of uncertainty also play an important role in ML, where the learner's state of knowledge strongly depends on the amount of data seen so far ...



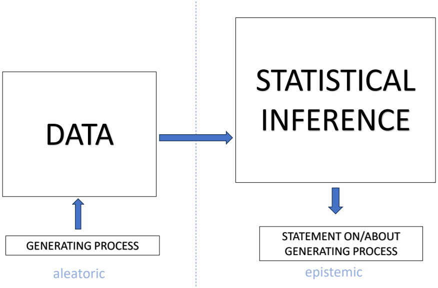


- … but also on the underlying model assumptions



strong prior (linear model) weaker prior (nonlinear model)

Aleatoric versus Epistemic Uncertainty



- In statistics both **aletoric** and **epistemic** uncertainty have always played an important role; often without explicitly using these terms
- Concepts from statistical inference are **still relevant** and **further developed** in modern ML approaches.

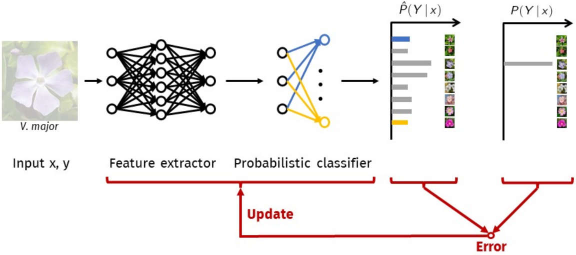
Methods for Uncertainty Quantification

- **Bayesian Methods:** Modeling uncertainty through probabilistic approaches (e.g., conformal learning)
- **Dropout-Based Methods:** Using dropout layers during training for uncertainty estimation
- **Ensemble Methods:** Utilizing multiple models to capture different aspects of uncertainty.

Challenges and considerations:

- **Modeling challenges:** Balancing complexity and interpretability
- **Impact on decisions:** Understanding how uncertainty affects decision-making processes.

Classification Tasks Using Deep Neural Networks



$$H(\boldsymbol{p}, \hat{\boldsymbol{p}}) = H(\boldsymbol{p}) + D_{KL}(\boldsymbol{p}, \hat{\boldsymbol{p}})$$

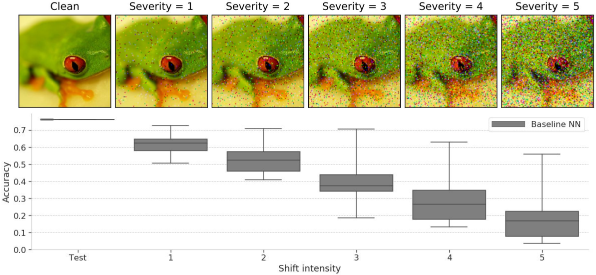
- Minimization of a risk using empirical data

$$\hat{p}_\theta(y|\mathbf{x}) = \arg \inf_{\theta\in\Theta} \mathbb{E}_{\mathbf{X}Y}[-\log p_\theta(Y|\mathbf{X})]$$

- Ideally, when making a **prediction**: $f(\mathbf{x}) = \arg\max_y \hat{p}_\theta(y|\mathbf{x})$, the learner knows what it knows and **what is does not know.**

Lack of Uncertainty-Awareness of ML Systems

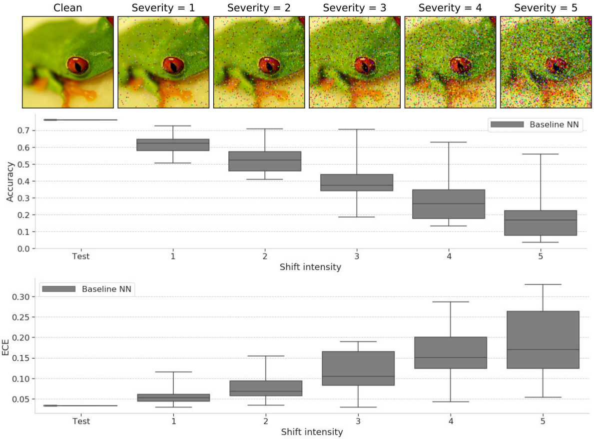
- **Accuracy drops** with increasing shift on Imagenet-C



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

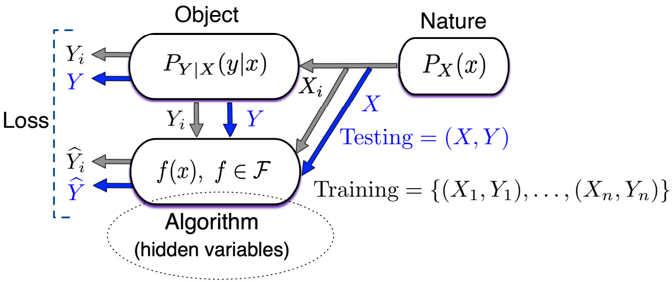
Can You Trust Your Model's Uncertainty? Evaluating Predictive Uncertainty Under Dataset Shift?, [Ovadia et al. 2019]

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



But, what causes these models to **inadequately capture the distributions** of categories?

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 (typically ill-posed problem)

Uncertainty of model predictions is related to the approximation $\hat{p}_\theta(y|\mathbf{x})$ of the objet.

A Probabilistic Model of Imitation Learning (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?

Their ingenious formulation led to the characterization of **necessary and sufficient conditions** (e.g., finite VC-dimension) for distribution-free minimization of a risk $R(f)$ using data.

Is Distribution-Free Inference Possible for Object Learning?

- We study the extension of the distribution-free framework **beyond the imitation task** for binary classification $\mathcal{Y} = \{0, 1\}$

- **Objective:** Provide distribution-free inference on the conditional label probability $\pi_P(\mathbf{x}) = \Pr(Y = 1|\mathbf{x})$. Particularly, in scenarios where $\pi_P(\mathbf{x})$ is not close to 0 or 1

- **Research question:** Given training samples $\{\mathbf{x}_i, y_i\}_{i=1}^n$, can we construct an algorithm from mapping a new data point $\mathbf{x} \in \mathbb{R}^d$ to an $(1 - \alpha)$-confidence interval $\hat{C}_n(\mathbf{x}) \subseteq \mathbb{R}$ such that

$$\Pr_{(\mathbf{X}_i, Y_i) \overset{\text{iid}}{\sim} P} \Big(\pi_P(\mathbf{X}_{n+1}) \in \hat{C}_n(\mathbf{X}_{n+1}) \Big) \geq 1 - \alpha,$$

for all probability distributions P on $\mathbb{R}^d \times \{0, 1\}$?

Is Distribution-Free Inference Possible for Object Learning?

- We begin with a few definitions. First, for $t \in [0, \frac{1}{2}]$ and $a \in [0,1]$, we define

$$\ell(t,a) = \begin{cases} 2(1-a)t, & a \geq \frac{1}{2}, \\ \frac{t}{2a}, & a \geq t \text{ and } 0 < a < \frac{1}{2}, \\ 1 - \frac{t}{2a}, & a < t, \\ 0, & a = t = 0 \end{cases}$$

  and for $t \in (\frac{1}{2}, 1]$ and let $\ell(t,a) = \ell(1-t,a)$

- For any distribution Q on $[0,1]$ and any $\alpha \in [0,1]$, define

$$L_\alpha(Q) = \inf_{\substack{\text{Measurable fns.} \\ a:[0,1]\to[0,1]}} \left\{ \mathbb{E}_{T\sim Q}[\ell(T, a(T))] : \mathbb{E}_{T\sim Q}[a(T)] \leq \alpha \right\}$$

- Next, we will **establish lower bounds** on the length of a distribution-free confidence interval for object learning.

Distribution-Free Object Learning is Not Feasible

Theorem (Object learning is not feasible)

Let $\hat{C}_n$ be any algorithm that provides a $(1-\alpha)$-distribution-free confidence interval for $\pi_P(\mathbf{x})$. Then, for any nonatomic distribution P on $\mathbb{R}^d \times \{0,1\}$, it holds that

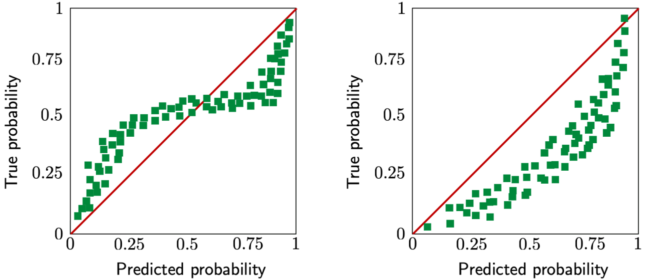
$$\mathbb{E}_{(\mathbf{X}_i,Y_i)\, \underset{\sim}{iid}\, P}\Big[\mathit{leb}(\hat{C}_n(\mathbf{X}_{n+1}))\Big] \geq L_\alpha(\Pi_P),$$

where Π_P is the (unknown) distribution of the random variable $\pi_P(\mathbf{X}) \in [0,1]$; leb() denotes the Lebesgue measure on $\mathbb{R}$.

- Proof follows by using similar arguments to (Donoho 1988)
- In the distribution-free setting, **parameter estimation is fundamentally as imprecise as prediction!**
- Confidence intervals for estimating the label probabilities $\pi_P(\mathbf{x}) = \Pr(Y=1|\mathbf{x})$ have a **lower bound on their length that does not vanish** even with sample size $n \to \infty$ ☹

Improving Calibration

- Ideally, true and estimated probabilities should coincide:



bias on extreme probabilities (left), systematic overestimation (right)

- We say that a (binary) classifier is **calibrated** if

$$\Pr\Big(y = 1 | \mathbf{X} \in \{\mathbf{x} \in \mathcal{X} : \hat{P}(y = 1|\mathbf{x}) = \alpha\}\Big) = \alpha, \quad \forall\ \alpha \in (0,1).$$

Conformal Learning

- Instead of point predictions, make **set-valued predictions** covering the true outcome with higher probability
- **Conformal prediction** (Vovk *et al.*, 2004) is a framework for reliable prediction that is rooted in classical frequentist statistics and hypothesis testing
- Instead of point predictions, CP makes **set-valued predictions covering** the true outcome with high probability

$$\longrightarrow \quad P\Big(y \in Y = \{2, 3, 9\}\Big) \text{ w.h.p.}$$

- **Guaranteed validity:** probability of an invalid prediction ($y \notin Y$) is (asymptotically) bounded by $\alpha > 0$.

Information and Diversity Measures

ILLS
International Laboratory on Learning Systems

Shannon Entropy

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

- **1. Measure of uncertainty** → "surprise" function $s(x)$, $x \in \mathcal{X}$, and $\mathrm{H}(\mathrm{X}) = \mathbb{E}[s(\mathrm{X})]$
- **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 → $s(x) = -\log \mathrm{p}(x)$

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

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

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 $0 \leq \mathrm{H}(\mathsf{X}) \leq \log|\mathcal{X}|$.

Mutual Information

- Mutual Information for discrete RVs $(X, Y) \sim p$ is defined as

$$\begin{aligned} I(X;Y) &\triangleq H(X) - H(X|Y) \\ &= H(Y) - H(Y|X) \\ &= H(X) + H(Y) - H(XY) \\ &= \sum_{(x,y)\in\mathcal{X}\times\mathcal{Y}} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \end{aligned}$$

  Mutual information is a **measure of dependency**

- It is a non-negative function of p_{XY}, concave in p_X for fixed $p_{Y|X}$, and convex in $p_{Y|X}$ for fixed p_X
- $I(X;Y) \geq 0$ with equality iff X and Y are independent
- Entropy and mutual information can be extended to continuous alphabets, **but care must be exercised** in applications.

Rényi Entropy

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

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

  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.

Measures of Diversity

- Diversity is a fundamental concept found in various scientific disciplines, including statistics, ecology, and machine learning
- Extensive literature on measures of diversity within populations and dissimilarity or similarity between populations
- **Examples of applications:** in anthropology, genetics, economics, sociology, and biology
- We will show that Rao's measures of diversity (Rao *et al.* 1982) are **essential tools** that provide insights into the predicted distributions and uncertainty.

- **Shannon Diversity Index** quantifies the uncertainty associated with a random variable representing the distribution of different categories:

$$\mathrm{H}(\mathrm{Y}) = -\sum_{y \in \mathcal{X}} \mathrm{p}(y) \log \mathrm{p}(y)$$

$\mathrm{p}(y)$ is the probability of observing category $\mathrm{p}(y)$

- **Simpson's Diversity Index** focuses on the probability that two randomly selected individuals belong to different categories:

$$s_{\mathrm{Gini}} = 1 - \sum_{y \in \mathcal{Y}} \mathrm{p}(y)^2,$$

it considers the **proportion of individuals of each type** $\mathrm{p}(y)$

- It is particularly **useful in ecology to measure biodiversity and species dominance.**

Rao's Diversity Measure (1982)

- Proposed by Rao in 1982, Rao's Diversity Measure introduces a unique perspective by focusing on the **distribution of distances between pairs of individuals:**

$$s_d = \sum_{y \in \mathcal{Y}} \sum_{y' \in \mathcal{Y}} d(y, y')\mathrm{p}(y)\mathrm{p}\big(y'\big)$$

  $d(\cdot,\cdot) \geq 0$ is a distance measure, and $\mathrm{p}(y)$ is the probability measure of a discrete random variable

- Note that the Simpson Diversity Index coefficient $s_{\mathbf{Gini}}$ **is a special case** of s_d when $d_{ij} = 1$ if $i \neq j$ and $d_{ii} = 0$. Thus, $s_d = s_{\mathrm{Gini}}$ when choosing d to be the Hamming distance
- Extensions to the **continuous random variables** are available.

Definition (Fisher-Rao Distance (FRD))

- Given a **family of probability distributions:**

$$\mathcal{C} = \left\{ q(\cdot|\theta) : \theta \in \Theta \right\}$$

- **Metric tensor** (Fisher information):

$$G(\theta) = \mathbb{E}_{Y \sim q(\cdot|\theta)} \left[\nabla_\theta \log q(Y|\theta) \nabla_\theta^\intercal \log q(Y|\theta) \right]$$

  is positive definite for any $\theta \in \Theta$

- Infinitesimal squared length element:

$$ds^2 = \langle d\theta, d\theta \rangle_{G(\theta)} = d\theta^\intercal G(\theta) d\theta$$

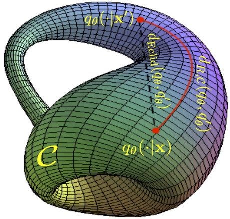
- The **FRD between** $q_\theta(\cdot|\theta)$ and $q_\theta(\cdot|\theta')$ is:

$$d_{R,\mathcal{C}}(q_\theta, q_{\theta'}) = \inf_\gamma \int_0^1 \sqrt{\frac{d\gamma^\intercal(t)}{dt} G(\gamma(t)) \frac{d\gamma(t)}{dt}}$$

  the inf is over all piecewise smooth curves

- FRD is the length of the **geodesic** between (θ, θ') using $G(\theta)$ as the metric tensor.

R. Rao and R. Fisher, 1956



Applications of Diversity Measures

- **Ecological studies:** In ecology, raw diversity measures help in understanding species distribution and ecosystem health
- **Statistical analysis:** These measures are valuable in statistical analysis, especially when dealing with categorical data
- **Image and signal processing:** In image and signal processing, diversity measures aid in pattern recognition and understanding the distribution of features
- **Our focus:** Rao's Diversity Measure finds applications in **detecting misclassifications** by assessing the distribution of distances between predicted categories.

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)

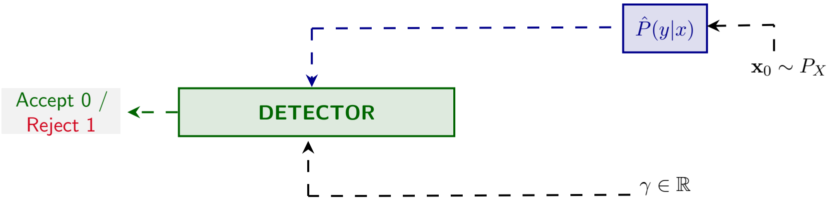
Main Definitions

We use the following notation:

- $\mathcal{X} \subseteq \mathbb{R}^{d_x}$ be the **feature space**
- $\mathcal{Y} = \{1, \ldots, C\}$ be the **label space**
- p_{XY} be the underlying p.d.f. 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**

$$f_{\mathcal{D}_n}(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} \widehat{P}(y|\mathbf{x}; \mathcal{D}_n).$$

Problem Formulation

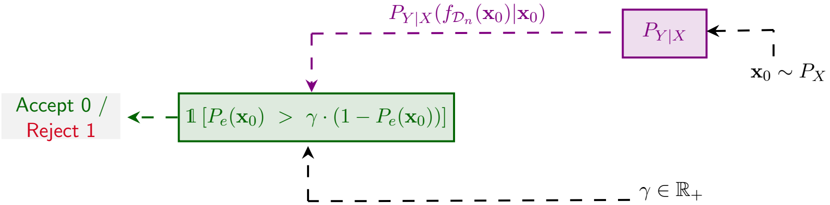


Ideal (Oracle) Setting

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, $P_e(\mathbf{x}) : \mathcal{X} \to [0,1]$ is **not available**, but can we approximate it?

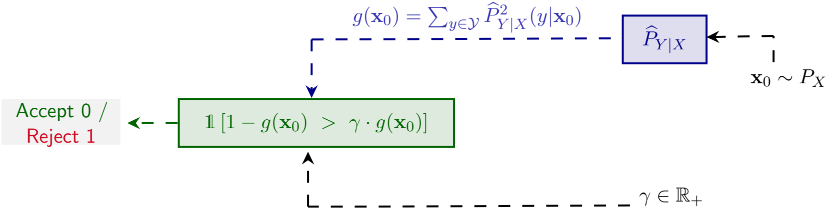
Simpson Index of Diversity: D_α

Proposition (Simpson Index of Diversity: D_α)

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

- $1 - g(\mathbf{x}_0) = 1 - \sum_{y \in \mathcal{Y}} \widehat{P}^2_{Y|X}(y|\mathbf{x}_0)$, *which approximates* $P_e(\mathbf{x})$
- $(1 - \sqrt{g(\mathbf{x}_0)}) - \Delta(\mathbf{x}_0) \leq P_e(\mathbf{x}_0) \leq (1 - \sqrt{g(\mathbf{x}_0)}) + \Delta(\mathbf{x}_0)$, *where*

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



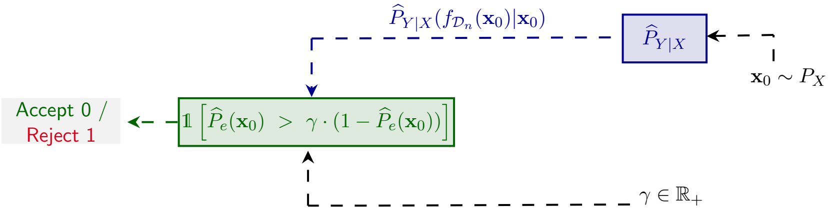
$$D_\alpha(\mathbf{x}_0, \gamma) \triangleq \mathbb{1}\left[1 - g(\mathbf{x}_0) > \gamma \cdot g(\mathbf{x}_0)\right]$$

Self-error Approximation: D_β

Definition (SELF-ERROR APPROXIMATION: 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 $\widehat{P}_{Y|X}$)
- $\widehat{P}_e(\mathbf{x}_0) \triangleq \mathbb{E}[\widehat{E}(\mathbf{x}_0)|\mathbf{x}_0] = 1 - \widehat{P}_{Y|X}(f_{\mathcal{D}_n}(\mathbf{x}_0)|\mathbf{x}_0)$ is the **probability of error classification w.r.t.** $\widehat{P}_{Y|X}$



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

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

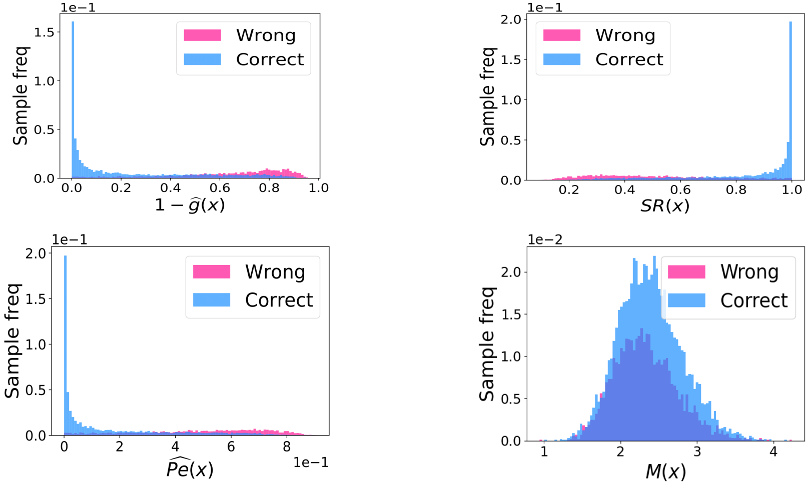


Figure 1. **DOCTOR**, **SR** and **MHLNB** to split data samples in TinyImageNet under TBB. Histograms for wrongly classified samples and correctly classified samples.

Discrimination Performance for PBB

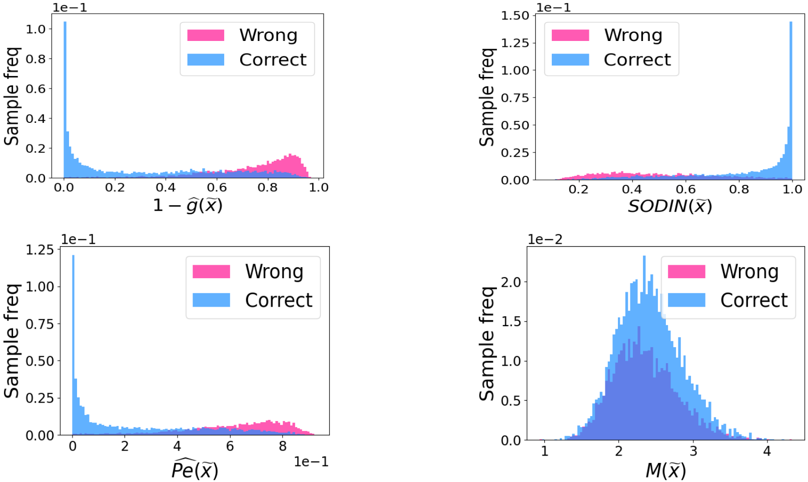


Figure 2. **DOCTOR**, **ODIN** and **MHLNB** to split data samples in TinyImageNet under PBB. Histograms for wrongly classified samples and correctly classified samples.

PBB: ROCs

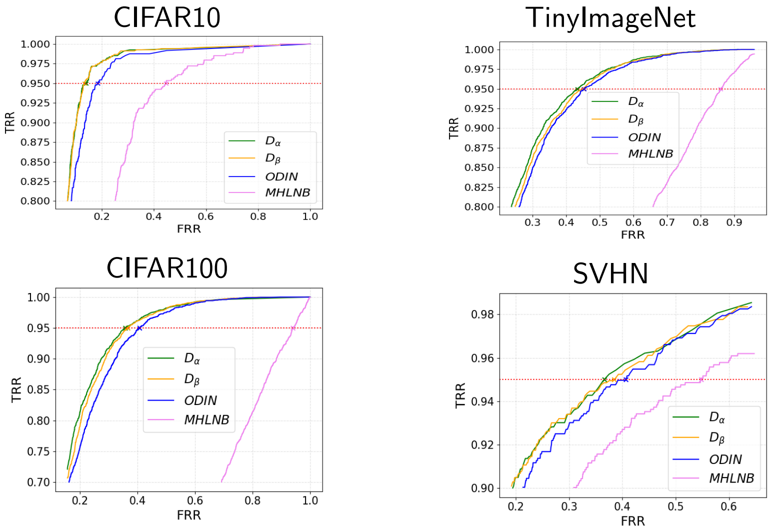


Figure 3. ROC curves. Comparison between **DOCTOR**, **ODIN** and **MHLNB**. The red dashed line marks the 95% threshold of TRR.

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 **very simple tool for detecting misclassification errors** which applies to any pre-trained classifier
- We leverage simple diversity measures 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.

Limitations and open issues:

- Statistical capabilities and limitations are not known
- It does not perform well in presence of a large number of classes
- It cannot incorporate validation samples.

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*.

Outline

ILLS International Laboratory on Learning Systems

A Data-Driven Measure of Relative Uncertainty for Misclassification Detection

Joint work with Eduardo Dadalto, Marco Romanelli, and Georg Pichler

(https://openreview.net/pdf?id=ruGY8v10mK)

Misclassification Detection Problem

- Misclassification detection is a standard binary classification problem, where the random binary error event

$$E = 1[f_{\mathcal{D}_n}(\mathbf{X}) \neq Y]$$

needs to be predicted from a given $\mathbf{x}$

- The underlying pdf p_X can be expressed as a mixture of two random variables:

$$\mathbf{X}_+ \sim p_{X|E}(\mathbf{x}|0) \text{ (positive instances } E = 0)$$
$$\mathbf{X}_- \sim p_{X|E}(\mathbf{x}|1) \text{ (negative instances } E = 1)$$

- **Our focus:** How can we enhance the performance of Doctor when provided with both positive and negative examples?

Rao's Measure of Diversity

- We propose to construct a class of uncertainty measures, inspired by the measure of diversity investigated in (Rao 1982)
- The quantity $\hat{\mathbf{p}}(\mathbf{x})$ denotes the posterior distribution output $(\hat{p}(y=1|\mathbf{x}),\dots,\hat{p}(y=C|\mathbf{x}))$ by the model given the input $\mathbf{x}$
- We define an **uncertainty measure** $s_d\colon \mathcal{X}\to\mathbb{R}$ that assigns a score $s_d(\mathbf{x})$ to every feature $\mathbf{x}$ in the input space $\mathcal{X}$ as

$$s_d(\mathbf{x}) = \mathbb{E}[d(\widehat{Y},\widehat{Y}')|\mathbf{X}=\mathbf{x}] = \sum_{y\in\mathcal{Y}}\sum_{y'\in\mathcal{Y}} d(y,y')\hat{\mathbf{p}}(\mathbf{x})_y\hat{\mathbf{p}}(\mathbf{x})_{y'}$$

  where $d\colon \mathcal{Y}\times\mathcal{Y}\to\mathbb{R}$ is a **symmetric matrix of positive values**
- Given a feature $\mathbf{x}$, the random variables $\widehat{Y},\widehat{Y}'\sim\hat{\mathbf{p}}(\mathbf{x})$ are i.i.d. according to $\hat{\mathbf{p}}(\mathbf{x})$.

Optimization Problem

Definition (Objective function)

- Given the hyperparameter $\lambda \in [0,1]$,

$$\mathcal{L}(D) = \bar{\lambda}\mathbb{E}\Big[\hat{\mathbf{p}}(\mathbf{X}_+)\, D\, \hat{\mathbf{p}}(\mathbf{X}_+)^\top\Big] - \lambda\mathbb{E}\Big[\hat{\mathbf{p}}(\mathbf{X}_-)\, D\, \hat{\mathbf{p}}(\mathbf{X}_-)^\top\Big]$$

- For a fixed $K \in \mathbb{R}^+$, we define our optimization problem as:

$$\begin{cases} \text{minimize}_{D \in \mathbb{R}^{C\times C}}\ \mathcal{L}(D) & & \\ \text{subject to} & d_{ii} = 0, & \forall i \in \mathcal{Y} \\ & d_{ij} \geq 0, & \forall i,j \in \mathcal{Y} \\ & d_{ij} = d_{ji}, & \forall i,j \in \mathcal{Y} \\ & T(DD^\top) \leq K & \end{cases}$$

Closed Form Solution

Proposition (Closed form solution)

- The constrained optimization problem defined above admits a closed form solution

$$D^* = \frac{1}{Z}\left(d^*_{ij}\right),$$

where

$$d^*_{ij} = \begin{cases} \mathrm{ReLU}\left(\lambda \mathbb{E}\left[\hat{\mathbf{p}}(\mathbf{X}_-)_i^\top \hat{\mathbf{p}}(\mathbf{X}_-)_j\right] - \bar{\lambda}\mathbb{E}\left[\hat{\mathbf{p}}(\mathbf{X}_+)_i^\top \hat{\mathbf{p}}(\mathbf{X}_+)_j\right]\right) & i \neq j \\ 0 & i = j \end{cases}$$

- The multiplicative constant Z is chosen such that D^* satisfies the condition $T(D^*(D^*)^\top) = K$

The proof is based on a Lagrangian approach.

Relative Uncertainty Score

Definition (Relative uncertainty)

For a given feature $\mathbf{x}$, the Relative Uncertainty (Rel-U) score as

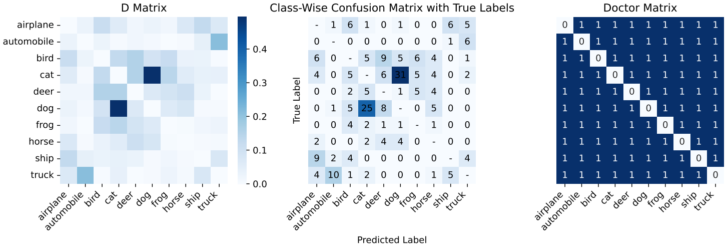
$$s_{\text{Rel-U}}(\mathbf{x}) = \hat{\mathbf{p}}(\mathbf{x})\, D^{*}\, \hat{\mathbf{p}}(\mathbf{x})^{\top}$$

- We can derive a misclassification detector g by fixing a threshold $\gamma \in \mathbb{R}$,

$$g(\mathbf{x}; s, \gamma) = 1[s_{\text{Rel-U}}(\mathbf{x}) \leq \gamma],$$

  where $g(\mathbf{x}) = 1$ implies $\hat{E} = 0$
- Note that the Gini coefficient $s_{\text{gini}}(\mathbf{x}) = H_2(\widehat{Y}|\mathbf{x})$ proposed by Doctor is a special case of $s_{\text{Rel-U}}(\mathbf{x})$ when $d_{ij} = 1$ if $i \neq j$ and $d_{ii} = 0$
- Thus, $s_{1-d}(\mathbf{x}) = s_{\text{gini}}(\mathbf{x})$ when choosing d to be the Hamming distance.

What Does the Diversity Matrix Uncover?



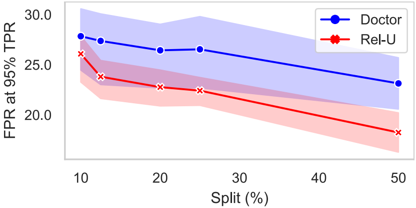
- Intuitive example illustrating the advantage of this method compared to entropy-based methods
- This method (left-end side heatmap) **captures the real uncertainty** (central heatmap) much better than Doctor.

Misclassification Detection Results

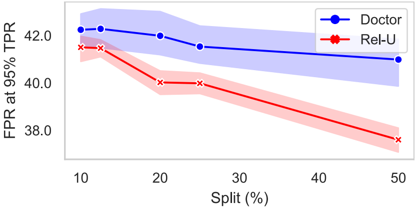
Model	Training	Accuracy	MSP [2]	ODIN [3]	Doctor [1]	REL-U
ResNet-34 (CIFAR-10)	CrossEntropy	95.4	25.8 (4.8)	19.4 (1.0)	14.3 (0.2)	**14.1** (0.1)
	LogitNorm	94.3	30.5 (1.6)	**26.0** (0.6)	31.5 (0.5)	31.3 (0.6)
	Mixup	96.1	60.1 (10.7)	38.2 (2.0)	26.8 (0.6)	**19.0** (0.3)
	OpenMix	94.0	40.4 (0.0)	39.5 (1.3)	**28.3** (0.7)	28.5 (0.2)
	RegMixUp	97.1	34.0 (5.2)	26.7 (0.1)	21.8 (0.2)	**18.2** (0.2)
ResNet-34 (CIFAR-100)	CrossEntropy	79.0	42.9 (2.5)	38.3 (0.2)	34.9 (0.5)	**32.7** (0.3)
	LogitNorm	76.7	58.3 (1.0)	55.7 (0.1)	**65.5** (0.2)	**65.4** (0.2)
	Mixup	78.1	53.5 (6.3)	43.5 (1.6)	**37.5** (0.4)	**37.5** (0.3)
	OpenMix	77.2	46.0 (0.0)	43.0 (0.9)	41.6 (0.3)	**39.0** (0.2)
	RegMixUp	80.8	50.5 (2.8)	45.6 (0.9)	40.9 (0.8)	**37.7** (0.4)

- Misclassification detection performance in terms of average FPR at 95% TPR **(lower is better)** in percentage with one standard deviation over ten different seeds in parenthesis.

Impact of the Split Size on the Misclassification



(a) CIFAR-10



(b) CIFAR-100

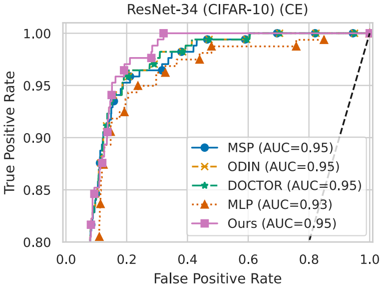
- Impact of the tuning split size on the misclassification performance on a ResNet-34 model trained with supervised CE loss for our method
- Doctor's hyperparameters are set to default values ($T = 1.0$, $\epsilon = 0.0$, and $\lambda = 0.5$), so that only the impact of the validation split size is observed.

Does Calibration Improve Detection?

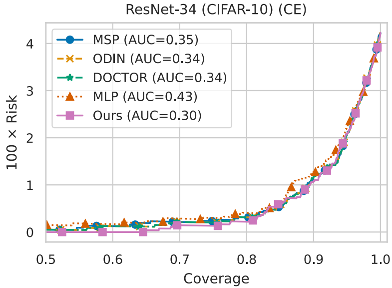
Architecture	Dataset	ECE_1	ECE_T	Uncal. Doctor	Cal. Doctor	Uncal. REL-U	Cal. REL-U
DenseNet-121	CIFAR-10	0.03	0.01	31.1 (2.4)	28.2 (3.8)	32.7 (1.7)	27.7 (2.1)
	CIFAR-100	0.03	0.01	44.4 (1.1)	45.9 (0.9)	45.7 (0.9)	46.6 (0.6)
ResNet-34	CIFAR-10	0.03	0.01	24.3 (0.0)	23.0 (1.4)	26.2 (0.0)	24.2 (0.1)
	CIFAR-100	0.06	0.04	40.0 (0.3)	38.7 (1.0)	40.6 (0.7)	38.9 (0.9)
ResNet-50	ImageNet	0.41	0.03	76.0 (0.0)	55.4 (0.7)	51.7 (0.0)	53.0 (0.3)

- Impact of model probability calibration on misclassification detection methods
- The uncalibrated and the calibrated performances are in terms of average FPR at 95% TPR **(lower is better)** and one standard deviation in parenthesis.

Misclassification Detection Results

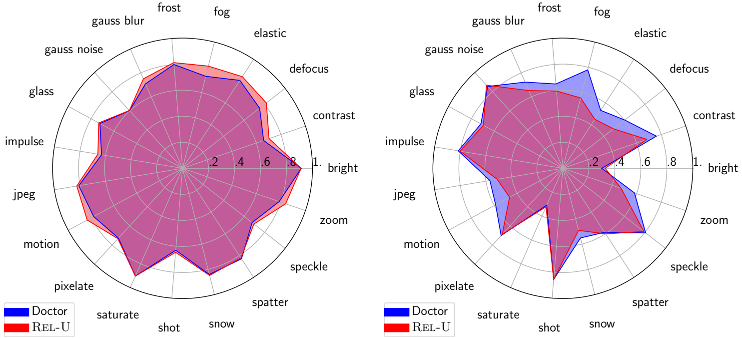


(a) ResNet-34 ROC curve.



(b) ResNet-34 RC curve.

- Equivalent performance of the detectors in terms of ROC **demonstrating lower FPR for high TPR regime**
- Risk and coverage curves also looks similar between methods, with a small advantage to our method in terms of AUROC.



(a) AUC

(b) FPR at 95% TPR

- CIFAR-10 vs CIFAR-10-C, ResNet-34, using 10% of the test split for validation.

References

[1] Federica Granese, Marco Romanelli, Daniele Gorla, Catuscia Palamidessi, and Pablo Piantanida.
DOCTOR: A simple method for detecting misclassification errors.
In *Advances in Neural Information Processing Systems*, 2021.

[2] Dan Hendrycks and Kevin Gimpel.
A baseline for detecting misclassified and out-of-distribution examples in neural networks.
In *International Conference on Learning Representations*, 2017.

[3] Shiyu Liang, Yixuan Li, and R. Srikant.
Enhancing the reliability of out-of-distribution image detection in neural networks.
In *International Conference on Learning Representations*, 2018.

[4] C Radhakrishna Rao.
Diversity and dissimilarity coefficients: a unified approach.
Theoretical population biology, 21(1):24–43, 1982.

Outline

ILLS
International Laboratory
on Learning Systems

Concluding Remarks

Understanding the nature of misclassification errors:

- 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 of paramount importance to understand what **drives a model to take certain decisions.**
- Rao's Diversity Measure finds applications in detecting misclassifications by assessing the distribution of distances between predicted categories.

Uncertainty and robustness are critical problems: **AI models that demonstrate self-awareness of their errors are highly valuable.**

Open Problems and Extensions

We need a better understanding of many aspects:

- Quantifying the link between distribution of distances of predicted categories and misclassification errors in a **theoretically sound manner**.
- The acquired distance metric D can be employed to capture **model interpretability and robustness**.
- We need **better benchmark models** for natural distribution drifts and calibration errors, uncertainty-robustness frontier.
- **Various extensions:** regression, segmentation, generalized settings (e.g., OOD data), evaluation, other forms of uncertainty, applications, etc.

Thank you for your attention