Modelling the Uncertainty of a Deep Neural Network Enhances its Adversarial Robustness

Marine PICOT[1], Pablo PIANTANIDA[1,2], Francisco MESSINA[3], Fabrice LABEAU[3]

[1]Laboratoire des Signaux et Systèmes, CentraleSupélec-CNRS-Université Paris Sud, Gif-sur-Yvette, France
[2]Montreal Institute for Learning Algorithms (Mila), Université de Montréal, QC, Canada

[3]Telecommunications and Signal Processing Lab, McGill University, McConnell Engineering Building, QC, Canada
{marine.picot,pablo.piantanida}@l2s.centralesupelec.fr

Résumé – Les réseaux de neurones profonds (DNN) ont atteint des performances à la pointe de la technologie dans plusieurs applications, mais ce sont des systèmes extrêmement peu robustes, en ce sens qu'ils sont vulnérables aux légères perturbations adversaires de leurs entrées. Dans cet article, nous étudions la relation entre la probabilité d'erreur, la qualité de d'ajustement et l'incertitude du classifieur. Nous proposons un nouvel objectif (loss) basé sur l'entropie conditionnelle et la divergence de Rényi. Nos résultats numériques, sur les jeux de données MNIST, CIFAR-10 et SVHN, montrent que, sans autre modification, la fonction coût proposée conduit à une amélioration significative de la robustesse des DNN face aux exemples adversaires par rapport à l'entropie croisée standard.

Abstract – Deep Neural Networks (DNNs) have achieved state-of-the-art performance in several applications, whereas they are extremely vulnerable to adversarial perturbations of inputs. This work first investigates bounds on the misclassification error as a funcion of the goodness of the fit and the uncertainty of the classifier. Then, these bounds are used to define a novel loss function based on the conditional entropy and the Rényi divergence. Our empirical studies, on MNIST, CIFAR-10 and SVHN datasets, show that, with no further modifications, the proposed loss leads to a significant enhancement in the robustness of DNNs to adversarial examples with respect to the standard categorical cross-entropy.

1 Introduction

Deep Neural Networks (DNNs) have achieved several breakthroughs on different fields like computer vision, speech recognition, natural language processing, etc. Nevertheless, it is well-known that these systems are extremely sensitive to small perturbations on the inputs [10]. For instance, it is possible to design additive perturbations that will slightly modify input images (in the sense that they are indistinguishable to the naked eye) so that they will be misclassified by a DNN with high probability. These are known as adversarial examples and they exist in different domains, which has led to the emergence of the field of adversarial machine learning (see [11] for further details). The effectiveness of adversarial examples has been attributed to the linear regime of DNNs [5] and the data manifold geometrical structure itself [4]. Although the problem of adversarial examples is relevant for several areas of deep learning, in this paper we only focus on training robust classifiers in the framework of supervised learning.

An important property of adversarial examples is that they are transferable, which means that adversarial examples generated with a given DNN can be used as adversarial examples for another neural network even if the architectures are rather different [9]. This implies that keeping the DNN model private is not a robustness guarantee, i.e., the so-called black box attacks are feasible [8]. As a consequence, improving robustness of DNNs to adversarial (universal) attacks is necessary for critical and safety related applications. The literature on adversarial machine learning is extensive and can be mainly divided in three overlapping groups which study generative models, detection and defence aspects of the field. Adversarial examples can be generated in a targeted or untargeted manner. A popular formulation of the problem of generating adversarial examples is as follows [5] : find an additive perturbation δ (with an ℓ_p norm bounded by some parameter ε) to an input x, in such a way that the loss is maximized. The most simple generation algorithm is the Fast Gradient Sign Method (FGSM) [5], which is based on a single step in the direction of the sign of the gradient of the loss with respect to the input. Surprisingly, this method is already quite effective in fooling a well-trained DNN. In the literature, most of the work explores the ubiquitous Cross-Entropy (CE) loss but some papers have proposed different alternatives. In [7], the authors consider the reverse CE loss function which encourages uniformity among the elements of the softmax output to improve robustness. In [3], a connection is made between adversarial training and total variation regularization and between worst-case adversarial training and Lipschitz regularization of the loss.

In this paper, we first show a strong theoretical relation between the misclassification probability of the classifier, the goodness of fit of the model (measured by the Rényi divergence between the true distribution of the data and the one induced by the softmax distribution) and the uncertainty of the classifier (measured by the conditional entropy of the softmax distribution). This is used to define a novel loss function. We then evaluate the robustness of a DNN trained with the resulting new loss and show that it offers considerable improvements over a DNN trained based on the CE loss. Our loss is as simple as the CE while achieving the same accuracy when evaluated with natural (i.e., non-adversarial) examples. Moreover, in some cases, it converges faster than CE loss.

2 A Novel Loss Function to Train DNNs

2.1 Bounds on the misclassification probability

Consider a standard supervised learning framework where $X \in \mathcal{X}$ denotes the input vector on the feature space $\mathcal{X}$, and let $Y \in \mathcal{Y}$ be the discrete concept defined without loss of generality as : $\mathcal{Y} := \{1, \ldots, M\}$. The data distribution is denoted by p_{XY}. A soft classifier is represented by the family of conditional probability distributions $p_{\widehat{Y}|X}$, where $\widehat{Y}$ is the soft decision. The soft classifier is used to induce a hard decision : $f : \mathcal{X} \to \mathcal{Y}$ with $f(X) := \arg\max_y p_{\widehat{Y}|X}(y|X)$. We also define the misclassification probability as $\mathrm{P}_e := \mathbb{P}(Y \neq f(X))$, and the uncertainty of the classifier as $\mathrm{P}_u := \mathbb{P}(\widehat{Y} \neq f(X))$. The conditional entropy of $\widehat{Y}$ given X is indicated by $H(\widehat{Y}|X)$. The Rényi divergence of order α between two distributions p and q is denoted as $D_\alpha(p\|q)$.

The following bounds on the error probability P_e hold for an arbitrary classifier and are a consequence of the so-called logarithmic probability comparison bounds (LPCBs) which relate the probability of an event using two different probability measures with the Rényi divergence between them (see Appendix A for definitions and details).

Proposition 1 *The error probability P_e satisfies the following inequalities :*

$$\log P_e \leq \frac{\alpha-1}{\alpha}\log[1-\exp(-H(\widehat{Y}|X))] + (\alpha-1)D_\alpha(p_{X,Y}\|p_{X,Y}), \quad \forall\alpha > 1, \tag{1}$$

$$\log P_e \geq \beta\,\frac{\alpha-1}{\alpha-2}\log[1-\exp(-H(\widehat{Y}|X))] - (\alpha-1)D_{\alpha-1}(p_{X,\widehat{Y}}\|p_{X,Y}), \quad \forall\alpha > 2, \tag{2}$$

where D_α denotes the Rényi divergence of order α and β is a constant dependent only on M. For the second inequality, the mild assumption $P_u \leq 1 - 1/M$ is required.

Proof 1 *See Appendix A.*

This result shows that the error probability P_e is controlled by the conditional entropy of the softmax distribution $H(\widehat{Y}|X)$, which is a measure of the uncertainty of the classifier, and a Rényi divergence between the data distribution p_{XY} and the distribution induced by the softmax $p_{X\widehat{Y}} = p_X\, p_{\widehat{Y}|X}$, which is a measure of the goodness of fit of the model.

2.2 A New Loss

The upper bound presented in Proposition 1 could be directly used to define a new loss function to train a DNN. The problem with (1), however, is that it diverges as $H(\widehat{Y}|X) \to 0$, which can lead to numerical issues at the end of the learning process. Thus, the bound was relaxed to define the new loss function, but preserving the monotonic relation between $H(\widehat{Y}|X)$ and P_e, leading to the following result.

Corollary 1 *From the results of Proposition 1, we can define a new loss function as follows :*

$$\mathcal{L}_{HR}(\theta) = H(\widehat{Y}|X) + \alpha D_\alpha(p_{XY}\|p_{X\widehat{Y}}), \tag{3}$$

where $\alpha > 1$ is considered as an hyperparameter.

Proof 2 *See Appendix B.*

It should be noted that α controls the way in which the goodness of fit of the softmax is measured. In the limit as $\alpha \to 1$, it converges to the KL divergence : $\alpha D_\alpha \to D$. On the other hand, as α becomes larger, the loss $\mathcal{L}_{\mathrm{HR}}$ penalizes more heavily the mismatch between p_{XY} and $p_{X\widehat{Y}}$, since αD_α is a monotonically non-decreasing function of α [1]. By a simple application of Jensen's inequality [2], it can also be shown that the CE loss and the Rényi divergence are related by the following inequality : $\mathrm{CE} \leq \alpha D_\alpha(p_{XY}\|p_{X\widehat{Y}}) + H(Y|X)$. Thus, the most significant difference between our loss $\mathcal{L}_{\mathrm{HR}}$ and the CE loss lies on the term $H(\widehat{Y}|X)$.

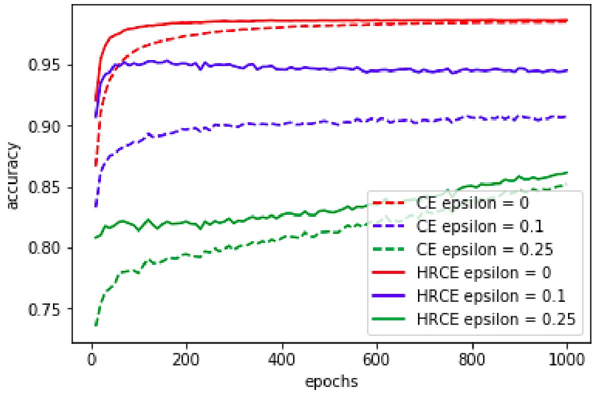

FIGURE 1 – Validation accuracy evolution for MNIST based on CE and HR with light and medium attacks using CE.

3 Experimental results

We evaluate our proposed loss function with three different data sets : MNIST, CIFAR-10, and SHVN. In each case, the value of α was optimized on a validation set, and we found that α

ε	Attack-based CE	Attack-based CE	Attack-based HR	Attack-based HR
	CE (MNIST | CIFAR-10 | SVHN)	HRCE (MNIST | CIFAR-10 | SVHN)	HR (MNIST | CIFAR-10 | SVHN)	CEHR (MNIST | CIFAR-10 | SVHN)
0	0.985 | 0.656 | 0.865	**0.986 | 0.685 | 0.901**	**0.986 | 0.685 | 0.901**	0.985 | 0.656 | 0.865
0.1	0.907 | 0.296 | 0.797	**0.945 | 0.337 | 0.836**	**0.977 | 0.324 | 0.835**	**0.977** | 0.288 | 0.801
0.25	0.852 | 0.282 | 0.672	**0.861 | 0.320 | 0.704**	**0.974 | 0.306 | 0.698**	0.972 | 0.271 | 0.674

TABLE 1 – Test accuracy on different datasets (MNIST, CIFAR-10 and SVHN) with ($\varepsilon > 0$) and without ($\varepsilon = 0$) adversarial attacks based on all combinations of training and attack on CE and HR losses.

= 1.6 works well for all datasets. We averaged over 5 different realizations for each simulation. The details of the DNNs architectures that we used for the simulations are relegated to Appendix C.

Due to its simplicity and effectiveness, we decided to compute adversarial examples using the FGSM algorithm [5], which generates an adversarial example x_{adv} from a normal example (x, y) according to $x_{\text{adv}} = x + \varepsilon \ \text{sgn}\left(\nabla_x \mathcal{L}^{(x,y)}(\Theta)\right)$, where sgn is the sign function, $\nabla_x \mathcal{L}^{(x,y)}(\Theta)$ denotes the gradient w.r.t. x of the loss function evaluated at (x, y), and ε is a parameter controlling the magnitude of the perturbation. Note that we omit the subscript intentionally in $\mathcal{L}$ since the attack can be performed with either the HR or the CE losses.

In Fig. 1, we show the validation set accuracy of both classifiers on MNIST as the training evolves across epochs for natural training ($\varepsilon = 0$), light ($\varepsilon = 0.1$) and medium ($\varepsilon = 0.25$) CE attacks. Results with the HR attacks are not reported since, in our experiments, during training, the accuracy for adversarial examples generated with a DNN using HR was almost constant and better than the accuracy using CE. The results for CE attack were therefore more interesting to visualize. We observe that DNNs trained on the HR are more robust to small and medium perturbations than trained on CE, no matter how well each of them is trained. It should emphasized that we wanted to compare CE and HR in a similar framework and this is why we fixed α for the entire training. Moreover, we can point out the fact that training convergence on MNIST using HR seems to be faster than using CE.

For completeness and comparison purposes, Table 1 presents the test accuracy for three data sets : MNIST, CIFAR-10, SHVN, and different magnitude of the attacks (none, light, medium) for both classifiers (HR and CE) and both attack methods (CE and HR). Note that HRCE stands for the classifier trained with the HR loss but attacked with the CE loss while CEHR stands for the other asymmetrical case. Results show that for MNIST, the worst-case attacks are generated by the CE, whereas for SVHN and CIFAR-10, they are generated by the HR.

For all datasets, no matter how the attack is generated, training on the HR loss will be beneficial (0.1% to 3.6% without any attacks, 3.4% to 4.1% for light attacks, 0.2% to 3.8% for medium attacks).

Finally, we computed the confusion matrices, i.e., the semi-empirical estimation of the probabilities $p_{\widehat{Y}|Y}(\widehat{y}|y)$ where $\widehat{y}$ is the row index and y is the column index, corresponding to the classifiers under the CE and HR loss for MNIST and under the CE attack with $\varepsilon = 0.1$. These matrices represent the uncertainty of the DNNs. Then, we computed the Frobenius norm of the difference between the identity matrix and each of these confusion matrices, yielding : $\|I_{10} - C_{\text{HR}}\|_F = 0.23$ and $\|I_{10} - C_{\text{CE}}\|_F = 0.40$. This result clearly shows that there is less uncertainty for the model trained with our loss.

4 Summary and Concluding Remarks

We have introduced a new loss for training DNNs. Our loss was shown to be a surrogate of the misclassification probability and consists of two terms : the conditional entropy of the softmax distribution and the Rényi divergence between the data generating distribution and the joint distribution of the input and the soft decisions. As was shown through experimental results, this new loss function offers better robustness to adversarial examples than the standard cross-entropy loss. The results presented in this paper are promising but, of course, preliminary. As future work, we will consider more powerful attacks than the FGSM algorithm. An advantageous property of information measures is that they are amenable to the analysis of nonlinear perturbations, thus offering the possibility to extend the adversarial transformations and eventually generalize the current concept of robustness in deep neural networks.

A Proof of Proposition 1

The Rényi divergence is formally defined as follows :

$$D_\alpha(\mu\|\nu) = \frac{1}{\alpha(\alpha-1)} \log \ \mathbb{E}_\mu \left[\left(\frac{d\nu}{d\mu}\right)^{1-\alpha}\right], \qquad (4)$$

if $\nu \ll \mu$ (i.e., ν is absolutely continuous with respect to μ).

The LPCBs [1] are very general inequalities relating the probability of an arbitrary event for two probability measures with the Rényi divergence between the measures. Concretely, by considering $\mu = \mathbb{P}_{XY}$, $\nu = \mathbb{P}_{X\widehat{Y}}$, and $\mathcal{A}$ be the error event of the classifier f, i.e $\mathcal{A} = \{(x, y) \in \mathcal{X} \times \mathcal{Y} : y \neq f(x)\}$, then

$$\frac{1}{\alpha-1}\log \mathrm{P}_e \geq \frac{1}{\alpha-2}\log \mathrm{P}_u - D_{\alpha-1}(p_{X\widehat{Y}}\|p_{XY})$$
$$\frac{1}{\alpha-1}\log \mathrm{P}_e \leq \frac{1}{\alpha}\log \mathrm{P}_u + D_{\alpha}(p_{XY}\|p_{X\widehat{Y}}), \qquad (5)$$

where, the first inequality holds for any $\alpha > 2$ while the second one holds for any $\alpha > 1$.

Upper Bound. We start by writing

$$\mathrm{P}_u = \mathbb{P}(\widehat{Y} \neq f(X)) = 1 - \mathbb{E}_X[p_{\widehat{Y}|X}(f(X)|X)]. \quad (6)$$

Now, by the definition of f, it is easy to see that

$$\mathbb{E}_X[p_{\widehat{Y}|X}(f(X)|X)] \geq \mathbb{E}_X[\mathbb{E}_{\widehat{Y}|X}[p_{\widehat{Y}|X}(\widehat{Y}|X)]], \quad (7)$$

with equality if $\widehat{Y} = f(X)$ almost surely. Finally, using Jensen's inequality, we have

$$\mathbb{E}_X[\mathbb{E}_{\widehat{Y}|X}[p_{\widehat{Y}|X}(\widehat{Y}|X)]] \geq \exp\left(-H(\widehat{Y}|X)\right). \quad (8)$$

This completes the proof of inequality (1).

Lower Bound. By Fano's inequality [2], considering the Markov chain $f(X) \leftrightarrow X \leftrightarrow \widehat{Y}$, we have that

$$q(\mathrm{P}_u) = h(\mathrm{P}_u) + \mathrm{P}_u \log M \geq H(\widehat{Y}|f(X)) \geq H(\widehat{Y}|X), \quad (9)$$

where the last inequality follows from the data-processing inequality. Here, $h(.)$ is the binary entropy [2]. Note that if $\mathrm{P}_u \leq 1 - 1/M$, the function $q(\mathrm{P}_u)$ is monotonically increasing. Therefore, under this assumption, we have that $\mathrm{P}_u \geq q^{-1}(H(\widehat{Y}|X))$. It can also be shown that

$$\log q^{-1}(H(\widehat{Y}|X)) \geq \beta \log[1 - \exp(-H(\widehat{Y}|X))], \quad (10)$$

where β is a constant dependent on M. Thus, we finally obtain the second inequality in (2)

B Derivation of Loss Function $\mathcal{L}_{\mathbf{HR}}$

Consider (1). Notice that the bound diverges as $H(\widehat{Y}|X) \to 0$, which is an undesirable behavior for a loss function. Thus, to obtain a suitable surrogate, we will use the so-called fundamental inequality in information theory which states that $\log a \leq a - 1$ for any $a > 0$ with equality if and only if $a = 1$. We also consider the inequality $-\exp(-a) \leq a - 1$ which holds for all $a \in \mathbb{R}$. This gives us

$$\log[1 - \exp(-H(\widehat{Y}|X))] \leq H(\widehat{Y}|X) - 1. \quad (11)$$

for any $H(\widehat{Y}|X) > 0$. This finally leads us to define the loss function presented in Section 2.2.

C Network Architecture

For MNIST, input images are scaled to be between 0 and 1. The model used is composed of two convolutional layers, each followed by a max pooling layer, and two fully connected layers. For the optimization, we use the SGD algorithm with a learning rate equal to 0.001. We train the model for 1000 epochs by using batches of size 100 (we observed that increasing the batch size doesn't have much effect on the results). Every 10 epochs, we generate adversarial examples from the test dataset using the FGSM method. The ε parameter is varied according to the experiment as described in Section 3.

For CIFAR-10, we use ResNet [6] with $n = 1$ and without data augmentation. We train for 200 epochs, with batches of 128 samples. Regarding SVHN, we use the same setup as for CIFAR-10 except that we do not perform data processing before training.

Acknowledgment

The work of Prof. Pablo Piantanida was supported by the European Commission's Marie Sklodowska-Curie Actions (MSCA), through the Marie Sklodowska-Curie IF (H2020-MSCAIF-2017-EF-797805-STRUDEL).

Références

[1] R. Atar, K. Chowdhary, and P. Dupuis. Robust Bounds on Risk-Sensitive Functionals via Rényi Divergence. *SIAM/ASA Journal on Uncertainty Quantification*, 3(1) :18–33, 2015.

[2] T.M. Cover and J.A. Thomas. *Elements of Information Theory*. Wiley, 2012.

[3] Chris Finlay, Adam M. Oberman, and Bilal Abbasi. Improved robustness to adversarial examples using lipschitz regularization of the loss. *CoRR*, abs/1810.00953, 2018.

[4] Justin Gilmer, Luke Metz, Fartash Faghri, Samuel S. Schoenholz, Maithra Raghu, Martin Wattenberg, and Ian J. Goodfellow. Adversarial spheres. *CoRR*, abs/1801.02774, 2018.

[5] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv :1412.6572*, 2014.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[7] Tianyu Pang, Chao Du, Yinpeng Dong, and Jun Zhu. Towards robust detection of adversarial examples. In *NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pages 4584–4594, 2018.

[8] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM*, ASIA CCS '17, pages 506–519, New York, NY, USA, 2017. ACM.

[9] Nicolas Papernot, Patrick D. McDaniel, and Ian J. Goodfellow. Transferability in machine learning : from phenomena to black-box attacks using adversarial samples. *CoRR*, abs/1605.07277, 2016.

[10] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv :1312.6199*, 2013.

[11] Y. Vorobeychik, M. Kantarcioglu, R. Brachman, P. Stone, and F. Rossi. *Adversarial Machine Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2018.