DESIGN OF ROBUST COMPLEX-VALUED FEED-FORWARD NEURAL NETWORKS

Ana Neacșu[1,2], *Răzvan Ciubotaru*[1,2],*Jean-Christophe Pesquet*[2], *and Corneliu Burileanu*[1]

[1] Speech and Dialogue Laboratory
University Politehnica of Bucharest, Bucharest, Romania
[2] Université Paris-Saclay, CentraleSupélec, Inria
Centre de Vision Numérique, Gif-sur-Yvette, France

ABSTRACT

This paper addresses the problem of designing robust complex-valued neural networks in order to reduce their sensitivity to adversarial perturbations. The robustness is guaranteed by imposing a bound on the Lipschitz constant of the network. We present a new architecture (RCFF-Net), for which we derive tight Lipschitz constant bounds. A constrained learning strategy is then developed to train the proposed structure, while controlling its global Lipschitz constant. The proposed approach is evaluated in an audio signal denoising task. The achieved results demonstrate the effectiveness of the aforementioned design method.

Index Terms— audio denoising, robustness, complex-valued neural networks

1. INTRODUCTION

Despite growing popularity of deep neural networks (DNNs), used in an ever-increasing number of fields (image classification [1], speech enhancement [2], natural language processing [3], etc.), the majority of the existing solutions operate on real-valued inputs. Complex valued neural networks (CVNNs) [4, 5, 6] have however been a topic of ongoing interest in the data science community. Indeed, being able to deal with complex-valued data is of high interest in the signal processing domain. Even when analyzing real-valued signals (e.g., audio or images), one of the most commonly used approaches is based on frequency analysis, which immediately leads us to the complex plane. Magnitude and phase carry different information types, which offer insight on the information content of the signals [7]. Handling complex-valued data may be difficult since it requires more computational power, but it has also been shown that CVNNs may be more expressive than classical neural networks [8]. For example, [9] shows that an end-to-end complex-valued neural network outperforms a 2-channel real-valued network in the context of accelerated MRI reconstruction. In [10], CVNNs are employed for non-stationary RADAR data analysis. Also, it was shown that smaller CVNNs outperform larger real networks when dealing with physical data [11].

Apart from developing high-performance systems, another critical aspect to be considered nowadays is the safety of AI-based systems. Recent results show that DNNs are sensitive to adversarial perturbations of their inputs [12, 13]. A known metric used to assess the robustness of a model is the Lipschitz constant of the network. It provides a quantitative stability measurement: the norm of the output perturbation cannot be higher than the norm of the input perturbation multiplied by the Lipschitz constant [14]. However, due to the complexity of neural network structures, computing the exact Lipschitz constant is a non-deterministic polynomial-time (NP-hard) problem. Several algorithms have been proposed to approximate this constant in case of real-valued feed-forward networks [15, 16, 17]. Extending these techniques to CVNNs is however quite challenging.

In this work, we introduce a new class of neural networks operating in the complex domain, called *Robust Complex Feed-forward Network (RCFF-Net)*. The structure of the network is inspired by CapsNets [18, 19]. The weight matrices process the real and imaginary part distinctly. The activation functions handle the correlation between the real and imaginary parts of complex pairs. We demonstrate that, for the proposed structure, the Lipschitz constant can be efficiently computed. In addition, we develop a learning strategy to control this constant and ensure the robustness of the network against adversarial perturbations. We validate our approach in the context of denoising audio signals corrupted with a variable level of white Gaussian noise.

The rest of the paper is structured as follows. The theoretical background of our work is presented in Section 2. Section 3 describes our solution for designing robust complex-valued neural networks, while in Section 4 the constrained learning strategy is detailed. Experimental results are reported in Section 5, and Section 6 gives concluding remarks.

This project was supported by the BRIDGEABLE ANR Research and Teaching Chair in AI.

ISBN: 978-1-6654-6798-8

2. THEORETICAL ASPECTS

A complex-valued m-layer feedforward neural network is mathematically defined as an operator $T = T_m \circ \cdots \circ T_1$, where, for every $i \in \{1, \ldots, m\}$, $T_i = R_i(W_i \cdot + b_i)$. At layer i, $W_i \in \mathbb{C}^{N_i \times N_{i-1}}$ is the weight matrix, $b_i \in \mathbb{C}^{N_i}$ the bias vector, and $R_i \colon \mathbb{C}^{N_i} \to \mathbb{C}^{N_i}$ the activation function. $(N_i)_{0 \leqslant i \leqslant m}$ are positive integers defining each layer width.

In the following, we will make the technical assumption that the activation operators $(R_i)_{1 \leqslant i \leqslant m}$ satisfy some nonexpansiveness properties and that all of them (except possibly for the last layer) are separable:

Assumption 2.1 R_m is a nonexpansive (i.e., 1-Lipschitz) operator and, for every $i \in \{1, \ldots, m-1\}$, R_i is separable, i.e.

$$\big(\forall z = (\zeta_k)_{1 \leqslant k \leqslant N_i} \in \mathbb{C}^{N_i}\big) \quad R_i z = \big(\varrho_{i,k}(\zeta_k)\big)_{1 \leqslant k \leqslant N_i}, \tag{2.1}$$

where, for every $k \in \{1, \ldots, N_i\}$, $\varrho_{i,k} \colon \mathbb{C} \to \mathbb{C}$.
In addition, for every $i \in \{1, \ldots, m-1\}$ and $k \in \{1, \ldots, N_i\}$, $\varrho_{i,k}$ is α_i-averaged where $\alpha_i \in]0, 1]$, that is

$$\begin{aligned}(\forall (\zeta, \zeta') \in \mathbb{C}^2) \quad & |\varrho_{i,k}(\zeta) - \varrho_{i,k}(\zeta')|^2 + \\ & + \frac{1 - \alpha_i}{\alpha_i} |\zeta - \varrho_{i,k}(\zeta) - \zeta' + \varrho_{i,k}(\zeta')|^2 \leqslant |\zeta - \zeta'|^2.\end{aligned} \tag{2.2}$$

Note that, when $\alpha_i = 1$ in (2.2), we recover the definition of a nonexpansive function. More generally, nonexpansive functions form a superset of α_i-averaged functions. When $\alpha_i = 1/2$, $\varrho_{i,k}$ is a firmly nonexpansive function. It can be shown that if $\alpha_i > 1/2$, then $\varrho_{i,k}$ is α_i-averaged if and only if it is an overrelaxation of a firmly nonexpansive function [15].

There are two main recipes for building activation functions satisfying (2.2). The first one is to use split-complex activation functions of the form

$$(\forall \zeta \in \mathbb{C}) \quad \varrho_{i,k}(\zeta) = \varrho^{\mathrm{R}}_{i,k}(\mathrm{Re}\zeta) + \imath \varrho^{\mathrm{I}}_{i,k}(\mathrm{Im}\zeta), \tag{2.3}$$

where $\varrho^{\mathrm{R}}_{i,k} \colon \mathbb{R} \to \mathbb{R}$ and $\varrho^{\mathrm{I}}_{i,k} \colon \mathbb{R} \to \mathbb{R}$ are α_i-averaged activation functions. It is shown in [15] that most real-valued activation functions are averaged. The majority of them are proximity operators of some proper lower-semicontinuous functions and are thus firmly nonexpansive. An example of such function within this class is the split-complex ReLU function:

$$(\forall \zeta \in \mathbb{C}) \quad \varrho_{i,k}(\zeta) = \mathrm{ReLU}(\mathrm{Re}\zeta) + \imath\, \mathrm{ReLU}(\mathrm{Im}\zeta). \tag{2.4}$$

The second recipe is based on the observation that (2.2) is satisfied by many activation functions having the form

$$(\forall \zeta \in \mathbb{C}) \quad \varrho_{i,k}(\zeta) = \omega(|\zeta|)\zeta, \tag{2.5}$$

with $\omega \colon \mathbb{R}_+ \to \mathbb{R}$. Such activation functions operating jointly on the real and imaginary parts of their input are thus averaged. Examples of (2.5) are the Georgiou-Katsageras activation function [20] and the squashing function used in CapsNets [19], which are both proximity operators of convex functions [21, Example 2.15] [15, Example 3.2(ii)].

3. ROBUSTNESS QUANTIFICATION IN THE COMPLEX CASE

3.1. Lipschitz stability

Let $x \in \mathbb{R}^{N_0}$ represent the input and $T(x) \in \mathbb{R}^{N_m}$ the associated output of the neural model. If we add a small perturbation $z \in \mathbb{R}^{N_0}$, the error on the output can be quantified as follows:

$$\|T(x+z) - T(x)\| \leqslant \theta_m \|z\|, \tag{3.1}$$

where θ_m denotes the Lipschitz constant of the neural network. It is worth noting that θ_m is a crucial parameter that can be used as a feature to evaluate the robustness of the system against possible adversarial attacks. To the best of our knowledge, the problem of evaluating this bound in the case of complex-valued neural networks has not been addressed yet.

3.2. Deriving Lipschitz bounds in the complex domain

For every $M \in \mathbb{N} \setminus \{0\}$ and $\alpha \in]0, 1]$, let us define the following set

$$\begin{aligned}\mathscr{B}^M_\alpha = \{\mathrm{Diag}(\lambda_1, \ldots, \lambda_M) \mid (\forall i \in \{1, \ldots, M\}) \\ \lambda_i \in \mathbb{C} \ \text{ and } \ |\lambda_i - 1 + \alpha| = \alpha\}.\end{aligned}$$

In the complex plane, each of the diagonal elements of a matrix in $\mathscr{B}^M_\alpha$ lies on a circle with center $1-\alpha$ and radius α. The following result can then be proved where, for every complex valued matrix (or vector) A, $\|A\|_{\mathrm{S}}$ is its spectral norm, and $|A|$ the matrix formed with the moduli of the elements of A.

Proposition 3.1 *Suppose that Assumption 2.1 holds. Define*

$$\theta_m = \sup_{\substack{\Lambda_1 \in \mathscr{B}^{N_1}_{\alpha_1} \\ \vdots \\ \Lambda_{m-1} \in \mathscr{B}^{N_{m-1}}_{\alpha_{m-1}}}} \|W_m \Lambda_{m-1} \cdots \Lambda_1 W_1\|_{\mathrm{S}}. \tag{3.2}$$

Then θ_m is a Lipschitz constant of T. In addition,

$$\|W_m \cdots W_1\|_{\mathrm{S}} \leqslant \theta_m \leqslant \||W_m| \cdots |W_1|\|_{\mathrm{S}}. \tag{3.3}$$

Note that matrices $\Lambda_i = \mathrm{Diag}\big((\lambda_{i,k})_{1 \leqslant k \leqslant N_i}\big) \in \mathscr{B}^{N_i}_{\alpha_i}$, with $i \in \{1, \ldots, m-1\}$ are such that $\|\Lambda_i\|_{\mathrm{S}} \leqslant 1$. We can thus deduce an upper bound for the Lipschitz constant, given by the following inequality:

$$\theta_m \leqslant \|W_m\|_{\mathrm{S}} \cdots \|W_1\|_{\mathrm{S}}. \tag{3.4}$$

This upper bound corresponds to a classical, less accurate estimate of the Lipschitz constant of the network [16]. Although these results extend those given in [15, Theorem 5.2], there exists a fundamental difference between the complex-valued case and the real one. For the Lipschitz bound established for the real case in [15, Theorem 5.2], the supremum is calculated on a finite set of values. In contrast, the Lipschitz constant (3.2) requires to compute it on an infinite set of parameters since the sets $(\mathscr{B}_{\alpha_i}^{N_i})_{1\leqslant i\leqslant m-1}$ are not countable. However, we will show next that it is possible to circumvent this difficulty by focusing on a specific class of complex neural networks.

3.3. A special class of neural networks

We propose to build the weight operators of the network as follows:

$$\begin{aligned} &W_1 = W_1^+ \operatorname{Diag}\left(e^{\imath\beta_{1,1}}, \ldots, e^{\imath\beta_{1,N_0}}\right) \\ &(\forall i \in \{2,\ldots,m-1\}) \quad W_i = e^{\imath\beta_i} W_i^+ \\ &W_m = \operatorname{Diag}\left(e^{\imath\beta_{m,1}}, \ldots, e^{\imath\beta_{m,N_m}}\right) W_m^+, \end{aligned} \tag{3.5}$$

where

- for every $i \in \{1,\ldots,m\}$, $W_i^+ \in [0,+\infty[^{N_i\times N_{i-1}}$;
- $(\beta_{1,k})_{1\leqslant k\leqslant N_0} \in [0,2\pi[^{N_0}$, $(\beta_{m,k})_{1\leqslant k\leqslant N_m} \in [0,2\pi[^{N_m}$ and for every $i \in \{2,\ldots,m\}$, $\beta_i \in [0,2\pi[$.

A simple expression of the Lipschitz constant of the network is then deduced from Proposition 3.1.

Proposition 3.2 *Suppose that the network is defined as above and that Assumption 2.1 holds. Then*

$$\theta_m = \|W_m^+ \cdots W_1^+\|_{\mathrm{S}}. \tag{3.6}$$

In other words, in this specific case, the Lipschitz constant of the network reduces to the spectral norm of the product of positive-valued weight matrices. This result suggests a new neural network architecture which can be trained in a robust way, as explained in the following sections. The proof of the theoretical concepts presented above will be detailed in [22].

4. PROPOSED APPROACH

4.1. Robust CVNNs

We implement our architecture to meet the requirements of Proposition 3.2 and design a Robust Complex Feed-Forward Neural Network (RCFF-Net). The architecture is illustrated in Figure 1. The network processes complex-valued data by stacking their real and imaginary parts. The weights associated with the first and last layers, are *Diagonal layers* (DIAG), as they perform phase shifts in the complex plane, which will be optimized during the training phase. At the core of the architecture stands the *Complex Dense Layer* (CDL), detailed in Figure 1c. This layer encapsulates two nonnegative linear transforms that process distinctly the real and imaginary parts. In contrast, the activation functions operate on each pair of real-imaginary coefficients and they can be chosen as explained in Section 2. The output is then obtained by concatenating the real and imaginary parts. Between CDL layers, we apply a rotation operation (ROT), which induces a global phase-shift of all its arguments.

4.2. Training strategy

For the training stage, we propose to employ a projected version of the AdaMax optimizer [23]. More specifically, the gradient step is followed by a projection onto a constraint set. This set expresses the two constraints on which our approach is grounded. First of all, since our assumptions are valid under a nonnegativity condition for the weights, we need to ensure that $(\forall i \in \{1,\ldots,m\})$, $W_i^+ \in [0,+\infty[^{N_i\times N_{i-1}}$. Additionally, based on Proposition 3.2, to control the robustness we impose that $\|W_m^+ \ldots W_1^+\|_{\mathrm{S}} \leqslant \overline{\vartheta}$, where $\overline{\vartheta}$ is the target Lipschitz constant of the network. To do so, we implement a block-coordinate algorithm where a convex problem is solved at each step in the spirit of the approach proposed in [16].

5. APPLICATION TO AUDIO SIGNAL DENOISING

5.1. Experimental setup

The proposed methodology is applied to audio signal denoising. We use a 5 layer RFCC-Net ($m = 5$), with diverse activation functions, trained on a publicly available dataset[1] consisting of musical exercises and songs, detailed in [17]. We split the dataset into 3 distinct sets, as follows. The training set contains 67 min, the validation set comprises 7 mins, and 5 mins of music signal were used as a test set. The noisy signals are generated by adding zero-mean white Gaussian noise to the original samples. The noise standard-deviation is randomly chosen so that the resulting signal-to-noise ratio varies between 5 and 30 dB. A short-time Fourier transform (STFT) is performed by segmenting the signals using an overlapping (50%) `Hanning` sliding window. For each window, we compute the 1024-points discrete Fourier transform. Because of the Hermitian symmetry, we consider only the first 513 frequency bins. Our network then estimates the complex STFT coefficients and, in the post-processing phase, an inverse operation (ISTFT) is performed for signal reconstruction.

5.2. Results and comparisons

We evaluate the performance of our RCFF-Net on 3 standard metrics: *Peak Signal-to-Noise Ratio (PSNR)*, *Cross-correlation (CC)*, and *Mean Squared Error (MSE)*, which was also employed as the training loss. The results on the test set are summarized in Table 1. We compare our solution with other standard denoising techniques, namely optimal Wiener filter and adaptive filter based on Normalised Least Mean Squares (NLMS) algorithm. As another baseline, we also trained a classical $m = 5$ layers Fully Connected Network (FCN) with ReLU activation. Furthermore, we trained RCFF-Net both using constrained and unconstrained weights, referred in Table 1 as `C` and `U`, respectively. For the unconstrained model, since the weights may have arbitrary signs, we only compute the upper bound θ_{upp} of the Lipschitz constant, given by right hand side of (3.4). In the constrained

[1] https://speed.pub.ro/downloads/music-datasets/

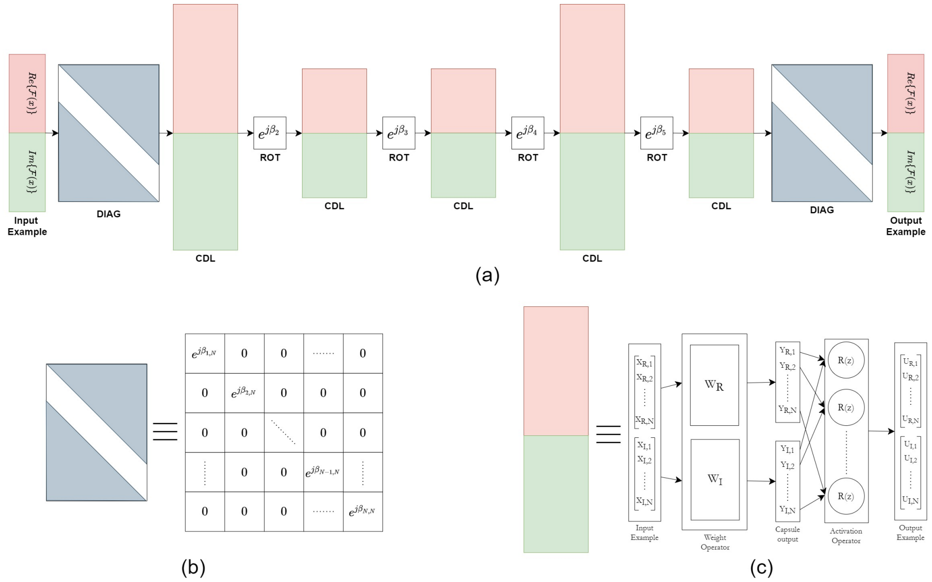

Fig. 1. Overview of the RCFF-Network. The red part denotes the real part, while the green accounts for the imaginary part. a) The proposed architecture: 5 CDLs (1024, 512, 512, 1024 and 513 neurons, respectively) followed by a Rotation layer (ROT) or a Diagonal layer (DIAG). b) The structure of a diagonal layer: the white band corresponds to the main diagonal which features non-zero coefficients. c) The structure of the dense complex layer: each group of neurons (capsule) will process jointly the real part and the imaginary part of the coefficients.

Table 1. Experimental results for audio denoising

					Unperturbed input			Attacked input			Deg.
					MSE	**PSNR [dB]**	**CC**	**MSE**	**PSNR [dB]**	**CC**	[%]
Noisy Signal					7.21×10^{-3}	21.02	0.83	7.30×10^{-3}	21.00	0.83	0.09
Denoised Signal	Baseline – Wiener Filter				3.45×10^{-3}	24.24	0.94	–	–	–	–
	Baseline – NLMS Adaptive Filter				2.52×10^{-3}	25.61	0.95	–	–	–	–
	Baseline – Standard FCN				2.78×10^{-3}	26.05	0.95	5.46×10^{-3}	22.87	0.90	12.24
	RCFF	$\rho(\zeta) = \mathbb{C}\text{ReLU}(\zeta)$	U	$\theta_{\text{upp}} = 335$	0.96×10^{-3}	30.00	0.99	4.84×10^{-3}	23.62	0.91	21.26
			C	$\theta_m = 0.99$	2.02×10^{-3}	27.64	0.96	1.96×10^{-3}	25.43	0.95	7.99
		$\rho(\zeta) = \frac{\zeta}{1+\lvert\zeta\rvert}$	U	$\theta_{\text{upp}} = 73.25$	1.04×10^{-3}	29.45	0.97	5.42×10^{-3}	23.31	0.90	20.84
			C	$\theta_m = 0.99$	2.11×10^{-3}	27.14	0.96	1.84×10^{-3}	25.72	0.95	5.23
		$\rho(\zeta) = \frac{8}{3\sqrt{3}} \frac{\lvert\zeta\rvert}{1+\lvert\zeta\rvert^2} \zeta$	U	$\theta_{\text{upp}} = 120$	0.96×10^{-3}	30.19	0.98	5.26×10^{-3}	22.05	0.90	26.96
			C	$\boldsymbol{\theta_m = 0.93}$	$\mathbf{1.22 \times 10^{-3}}$	**29.02**	**0.97**	$\mathbf{1.34 \times 10^{-3}}$	**28.68**	**0.97**	**1.17**
	ACNN [17]		C	$\theta_m = 1.00$	1.98×10^{-3}	26.24	0.96	2.46×10^{-3}	25.43	0.95	3.08

case, we were able to train nonexpansive models ($\theta_m \leqslant 1$). In accordance with the "no free lunch" theorem [24], this improved stability comes at the expense of a loss of performance that remains acceptable with our proposed method. We also compared our results with a competing approach for training robust denoisers [17] based on positive feed-forward neural networks, which only estimates the magnitude of the STFT coefficients. Moreover, to show that our solution is indeed robust against adversarial perturbations, we have tested the performance of our models when facing adversarial inputs, in the cases when it was possible and relevant. To the best of our knowledge, there are very few white-box attackers suited for regression problems and none currently available for complex-valued NNs. So, to create a worst-case input perturbation, we extended the gradient-based attacker proposed in [25] to operate in the complex domain. The attacked input was created by adding the aforementioned perturbation over the clean audio sample. To emphasize the effectiveness of our solution, the last column from Table 1 shows the degradation level (in terms of percentage of SNR) when the model faces adversarial inputs.

6. CONCLUSION

This paper proposes a novel solution for training robust CVNNs. By judiciously structuring the weight matrices, we derived a tight Lipschitz bound for the proposed architecture. We also showed how to control this bound numerically while training. We proved the effectiveness of our method in the context of audio denoising, but our method could be extended to other tasks such as source separation. In future works, it would be interesting to apply RCFF-Net to a larger panel of signal processing applications involving complex-valued data.

7. REFERENCES

[1] Y. Chen, K. Zhu, L. Zhu, X. He, P. Ghamisi, and J. A. Benediktsson, "Automatic design of convolutional neural network for hyperspectral image classification," *IEEE Geosci. Rem. Sens. Lett.*, vol. 57, no. 9, pp. 7048–7066, 2019.

[2] F. Weninger, H. Erdogan, S. Watanabe, E. Vincent, J. Le Roux, J. R. Hershey, and B. Schuller, "Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR," in *Int. Conf. Latent Variable Analysis and Signal Separation*. Springer International Publishing, 2015, pp. 91–99.

[3] N. Widiastuti, "Convolution neural network for text mining and natural language processing," *IOP Conf.: Mater. Sci. Eng.*, vol. 662, p. 052010, Nov. 2019.

[4] J. Bassey, L. Qian, and X. Li, "A survey of complex-valued neural networks," *arXiv:2101.12249*, 2021.

[5] A. Hirose, "Applications of complex-valued neural networks to coherent optical computing using phase-sensitive detection scheme," *Information Sciences-Applications*, vol. 2, no. 2, pp. 103–117, 1994.

[6] T. Kim and T. Adalı, "Approximation by fully complex multilayer perceptrons," *Neural computation*, vol. 15, no. 7, pp. 1641–1666, 2003.

[7] I. Aizenberg, *Complex-valued neural networks with multi-valued neurons*. Springer, 2011, vol. 353.

[8] J. A. Barrachina, C. Ren, C. Morisseau, G. Vieillard, and J.-P. Ovarlez, "Complex-valued vs. real-valued neural networks for classification perspectives: An example on non-circular data," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Toronto, Canada, 6–11 Jun 2021, pp. 2990–2994.

[9] E. Cole, J. Cheng, J. Pauly, and S. Vasanawala, "Analysis of deep complex-valued convolutional neural networks for MRI reconstruction and phase-focused applications," *Magn. Reson. Med.*, vol. 86, no. 2, pp. 1093–1109, 2021.

[10] O.-L. Ouabi, R. Pribic, and S. Olaru, "Stochastic Complex-valued Neural Networks for Radar," in *Proc. IEEE European Signal Process. Conf.*, Amsterdam, Netherlands, 18 – 22 Jan. 2021, pp. 1442–1446.

[11] J. S. Dramsch, M. Lüthje, and A. N. Christensen, "Complex-valued neural networks for machine learning on non-stationary physical data," *Computers & Geosciences*, vol. 146, p. 104643, 2021.

[12] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial machine learning at scale," in *Int. Conf. Learn. Represent.*, Toulon, France, 24–26 Apr 2017.

[13] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Int. Conf. Learn. Represent.*, San Diego, CA, USA, 7–9 May 2015.

[14] K. Scaman and A. Virmaux, "Lipschitz regularity of deep neural networks: analysis and efficient estimation," in *Proc. Ann. Conf. Neur. Inform. Proc. Syst.*, Montreal, Canada, 3–8 Dec 2018, pp. 3839–3848.

[15] P. L. Combettes and J.-C. Pesquet, "Lipschitz certificates for layered network structures driven by averaged activation operators," *SIAM J. on Math. Data Sci.*, vol. 2, no. 4, pp. 529–557, 2020.

[16] A. Neacşu, J.-C. Pesquet, and C. Burileanu, "Accuracy-robustness trade-off for positively weighted neural networks," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Barcelona, Spain, 4–8 May 2020, pp. 8389–8393.

[17] A. Neacşu, K. Gupta, J.-C. Pesquet, and C. Burileanu, "Signal denoising using a new class of robust neural networks," in *Proc. IEEE European Signal Process. Conf.*, Amsterdam, Netherlands, 18–22 Jan. 2021, pp. 1492–1496.

[18] X. Cheng, J. He, J. He, and H. Xu, "Cv-CapsNet: Complex-valued Capsule Network," *IEEE Access*, vol. 7, pp. 85 492–85 499, 2019.

[19] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," *Adv. Neural Inform. Process. Syst.*, vol. 30, pp. 3856–3866, 2017.

[20] G. Georgiou and C. Koutsougeras, "Complex domain backpropagation," *IEEE Tran. on Circ. and Syst. II: Analog and Digital Signal Process.*, vol. 39, no. 5, pp. 330–334, 1992.

[21] P. Combettes and J.-C. Pesquet, "Deep neural network structures solving variational inequalities," Set-Valued Var. Anal., pp. 1–28, 2020.

[22] A. Neacșu, J.-C. Pesquet, and C. Burileanu, "Robust complex-valued neural networks with applications to audio processing," *to be submitted*.

[23] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Int. Conf. Learning Representations*, San Diego, CA, USA, 7 – 9 May 2015.

[24] D. Tsipras, S. Santurkar, L. Engstrom, A. Turner, and A. Madry, "There is no free lunch in adversarial robustness (but there are unexpected benefits)," *CoRR, abs/1805.12152*, 2018.

[25] K. Gupta, J.-C. Pesquet, B. Pesquet-Popescu, F. Kaakai, and F. Malliaros, "An adversarial attacker for neural networks in regression problems," in *IJCAI Workshop on Art.l Int. Safety*, 2021.