

# Stochastic Complex-valued Neural Networks for Radar

1<sup>st</sup> Othmane-Latif Ouabi  
CentraleSupélec, Université Paris-Saclay,  
LSS, UMI 2958 Georgia Tech-CNRS,  
Metz, France  
ououabi@georgiatech-metz.fr

2<sup>nd</sup> Radmila Pribić  
Sensors Advanced Developments  
Thales Nederland BV  
Delft, the Netherlands  
radmila.pribic@nl.thalesgroup.com

3<sup>rd</sup> Sorin Olaru  
Université Paris-Saclay, CNRS,  
CentraleSupélec, LSS, 91190,  
Gif-sur-Yvette, France  
sorin.olaru@centralesupelec.fr

**Abstract**—Neural networks (NNs) prove to be performant in learning nonlinear models, but their mechanisms are yet to be fully understood. Since signal models in radar are inherently nonlinear with respect to unknown range, Doppler or angles, and moreover, radar processing is intrinsically stochastic, stochastic NNs which tie the numerical capability of NNs with the probabilistic inferences can enhance model-based radar processing. Indeed, radar data are complex-valued while most algorithms based on NNs are real-valued and furthermore, lack of uncertainty assessment. To address these issues, we elaborate, in the present paper, a stochastic complex-valued NNs framework for radar. We show that these networks can achieve parameter estimation with refined learned models from radar measurements and provide an indicator of the uncertainty on the estimation. We also build a stopping criterion based on the detection principles, so that the NNs training stops when there is noise only in data. Finally, the performances of the networks are illustrated in simulation.

**Index Terms**—models, neural networks, radar, raw data

## I. INTRODUCTION

Neural Networks (NNs) are having a huge success in many diverse big-data fields (e.g. [1]). In radar, when nonlinear data models are not analytically well known, or when standard approaches have been exhausted, NNs outcomes can be useful (e.g. [2]) but their uncertainty is also to be assessed. Radar processing is intrinsically stochastic [3], [4]. The rules of probability theory are used to infer the unobserved unknowns given the observed data, probability distribution and a model. Over the decades, radar knowledge about data modeling and processing has grown considerably.

Bayesian inferences can be applied to NNs [3]–[5] through the transformation of the prior to posterior distributions by using data. In addition, information distances (IDs) from information geometry (IG, e.g. [8]–[10]) have been applied to approximate posteriors [7]. Finally, since measurements from modern radar are complex-valued (e.g. [3], [4]), NNs shall be complex-valued as well [11]. However, in the literature, NNs have been proposed for real-valued radar data and without assessing the uncertainty (e.g. [2]).

Hence, to the best of the authors' knowledge, no stochastic complex-valued NNs have been applied for radar systems. The aim of this paper is to answer this need and elaborate a framework by completing the ideas in [12] studied in [13] with the emphasis on the uncertainty assessment, the

complex analysis and radar detection. In Section II, complex-valued NNs and the Bayesian inferences through the IDs are explained and merged into stochastic complex-valued NNs for radar (scNNr). Radar data models are clarified in Section III together with the link of NNs with radar detection and parameter estimation. The potential of scNNr is exemplified with a concrete scheme in Section IV and with numerical results in Section V. The conclusions and future works complete the paper. Our main contributions are : a study on complex-valued NNs for parameter estimation and the reveal of their potential for data typical but not limited to radar, the design of a stopping criterion for the training which avoids learning from the noise, and the uncertainty assessment enabled by the approximation of posteriors using IDs.

## II. STOCHASTIC NEURAL NETWORKS

NNs combined with the probabilistic inferences based on the information distances [10] and Bayesian methods [7] enable assessing the uncertainty of NNs outcomes in radar.

### A. Complex-valued Neural Networks

The use of architectures of NNs is widespread in Machine Learning. NNs utilize multiple levels of representation which help to make sense of (big) data and to fit nonlinear functions to (big) data. In radar, data and NNs are to be complex-valued (e.g. [11]). In a neural network with  $L$  layers, a learning atom is an  $n_l$ -th artificial neuron,  $l = 1, \dots, L$ , and  $n_l = 1, \dots, N_l$ . The principal processing elements relate the neurons' output vector  $\mathbf{y}_l$  to input data  $\mathbf{y}_{l-1} \in \mathbb{C}^{N_{l-1}}$ , as follows (e.g. [1]):

$$\mathbf{y}_l = \chi(\mathbf{W}_{l-1}^H \mathbf{y}_{l-1} + \mathbf{b}_{l-1}).$$

where  $\mathbf{W}_{l-1} \in \mathbb{C}^{N_{l-1} \times N_l}$  are the weights,  $\chi$  a nonlinear excitation function and  $\mathbf{b}_{l-1} \in \mathbb{C}^{N_l}$  a bias. The input  $\mathbf{y}_{l-1}$  is the output of a previous layer. The weights  $\mathbf{W}_l$  are estimated by fitting the related data  $\mathbf{y}_l \in \mathbb{C}^{N_l}$  according to a loss function  $L(\mathbf{y}_l, \mathbf{W}_l)$  as its gradient  $\nabla L$  w.r.t  $\mathbf{W}_l$  (at  $\hat{\mathbf{W}}_l^{t-1}$ ) as follows:

$$\hat{\mathbf{W}}_l^t = \hat{\mathbf{W}}_l^{t-1} - \lambda \hat{\nabla} L. \quad (1)$$

where  $\lambda$  is a learning rate and  $t$  a learning step.

In (1), the gradients  $\nabla L$  of the real-valued loss  $L$  are w.r.t complex-valued  $\mathbf{W}_l$ . The Wirtinger derivatives [14] are extensions of the complex derivative (which cannot be employed for

real-valued functions with complex-valued arguments). Given the real and imaginary part of  $\mathbf{W}_l$ ,  $\hat{\mathbf{W}}_l = \hat{\mathbf{W}}_{l,\mathcal{R}} + j\hat{\mathbf{W}}_{l,\mathcal{I}}$ , the conjugate-real Wirtinger derivative is defined by:

$$\hat{\nabla}L = \frac{1}{2} \left( \frac{\partial L}{\partial \hat{\mathbf{W}}_{l,\mathcal{R}}} + j \frac{\partial L}{\partial \hat{\mathbf{W}}_{l,\mathcal{I}}} \right).$$

Since such a derivative provides the optimal direction for increasing  $L$  [14], it is used for the training process. Moreover, there are more efficient training techniques than gradient descent (1), such as Adam optimizer (e.g. [7]) widely used in Machine Learning due to its practical results. It can be straightforwardly adapted to the complex-valued case [13].

### B. Information Distances

The Fisher-Rao metric  $ds$  is the infinitesimal distance on manifolds of data distributions  $p(\mathbf{y}|\theta)$  studied in information geometry (IG, [8]–[10]), defined as:

$$ds^2 = d\theta^H \mathbf{J}_\theta d\theta, \quad \mathbf{J}_\theta = -\mathbb{E} \left[ \frac{\partial^2 \log p(\mathbf{y}|\theta)}{\partial \theta \partial \theta^H} \right]. \quad (2)$$

$\mathbf{J}_\theta$  is the Fisher information matrix which quantifies information from data  $\mathbf{y}$  about unknown  $\theta$  (e.g. [15]). In the accuracy analysis,  $\mathbf{J}_\theta^{-1}$  gives the Cramer-Rao Lower Bound (CRLB) of the mean squared error (MSE) of an unbiased estimator  $\hat{\theta}$ , as:  $MSE(\hat{\theta}) \geq \mathbf{J}_\theta^{-1}$ .

IG enables computing an information (pseudo-)distance (ID) between data distributions by integrating  $ds$  from (2) over curves on the manifold. Many different IDs from IG and information theory (e.g. [16]) have been applied, e.g. in:

- Machine Learning with the natural gradient whose  $\lambda$  in (1) equals  $\mathbf{J}_\theta^{-1}$  based on the constant Kullback-Leibler (KL) divergence between  $p(\mathbf{y}|\mathbf{W})$  and  $p(\mathbf{y}|\mathbf{W} + \delta\mathbf{W})$  [9], [10],
- Bayesian variational inferences by minimizing an ID between posterior distributions [7], [17], [18], etc.

### C. Bayesian Methods

The core of Bayesian methods is the posterior distribution  $p(\theta|\mathbf{y})$  of an unknown parameter  $\theta$  given data  $\mathbf{y}$  defined as:

$$p(\theta|\mathbf{y}) = p(\theta)p(\mathbf{y}|\theta)/p(\mathbf{y}).$$

where  $p(\theta)$  is a prior,  $p(\mathbf{y}|\theta)$  is a likelihood (assumed or learned) and  $p(\mathbf{y}) = \int p(\mathbf{y}|\theta)p(\theta)d\theta$  is the evidence whose computation is hard and therefore, mostly approximated in practice. Variational autoencoders (VAEs, [7]) enable the approximation  $q(\theta|\mathbf{y})$  of  $p(\theta|\mathbf{y})$  with:

$$\min_{q(\theta|\mathbf{y})} KL(q(\theta|\mathbf{y})||p(\theta|\mathbf{y})) - \mathbb{E}_{q(\theta|\mathbf{y})} [\log p(\mathbf{y}|\theta)]. \quad (3)$$

where their KL divergence is  $\mathbb{E}_{q(\theta|\mathbf{y})} [\log(q(\theta|\mathbf{y})/p(\theta|\mathbf{y}))]$ , and (3) maximizes the Evidence Lower Bound (ELBO).

Stochastic NNs employ complex-valued Machine Learning (section II.A) whose loss function  $L$  as well as Bayesian inferences  $p(\theta|\mathbf{y})$  about  $\theta$  from data  $\mathbf{y}$  (section II.C) are computed by optimizing IDs (section II.B).

## III. PRINCIPLES OF MACHINE LEARNING FOR RADAR

In radar, nonlinear signal models based on the physics of radar sensing are well-known. Moreover, the processing of radar measurements with point targets in complex-Gaussian receiver noise is well-established in theory and practice [3], [4]. However, realistic data models may deviate a bit from the basic data models due to radar-system errors such as e.g. calibration or production errors, or due to more complicated targets and environments. In such cases when the analytical approach might be burdensome, strong numerical tools of NNs may help to learn such deviated data models and accordingly, may help to estimate unknown parameters more accurately. Hence, we start exploring the learning and estimation potential of NNs by assuming an unknown model.

In a basic radar data model, sensing of an input signal  $\alpha \in \mathbb{C}$  at unknown  $\theta$  results in raw data  $\mathbf{y} \in \mathbb{C}^M$  from an array of size  $M$ , modeled as (e.g [19]):

$$\mathbf{y} = \alpha e^{j\beta\theta} + \mathbf{z} = \alpha \mathbf{a}(\theta) + \mathbf{z}. \quad (4)$$

where  $\beta \in \mathbb{R}^M$  is an observation vector (centered, i.e.  $\sum_m \beta_m = 0$ ),  $\mathbf{z} \in \mathbb{C}^M$  is a complex-Gaussian receiver noise with zero mean and equal variances  $\gamma$ ,  $\mathbf{z} \sim \mathcal{CN}(\mathbf{0}, \gamma \mathbf{I}_M)$  and  $\mathbf{a}(\theta)$  is a sensing vector. In a spatial array,  $\beta_m$  and  $\theta$  yield the antenna element position and unknown angle, respectively.

In practical scenarios, the sensing vector  $\mathbf{a}(\theta)$  might not be perfectly known. Therefore, NNs shall estimate unknown parameters  $\alpha \in \mathbb{C}$  and  $\theta \in \mathbb{R}$ ,  $\rho = [\alpha, \theta]$ , from data  $\mathbf{y}$  as in (4) by employing a Machine Learning scheme to learn the sensing model, and which is to be supervised due to the extensive radar knowledge and the clear physical meaning of  $\rho$ .

### A. Parameter Estimation with Neural Networks

In a first approach, a neural network  $\mathcal{H}_{\{\mathbf{W}, \mathbf{b}\}}$  serves as an inverse solver by providing an estimation of the target unknowns from radar measurements:  $\mathcal{H}_{\{\mathbf{W}, \mathbf{b}\}}(\mathbf{y}) = \tilde{\rho}$ . Classically, it is trained by minimizing the MSE loss over a training dataset  $\mathcal{D}$  to match the real  $\rho$  as in Fig.1-a):

$$L_{\mathcal{D}}(\mathbf{W}, \mathbf{b}) = \frac{1}{\text{Card}(\mathcal{D})} \sum_{(\mathbf{y}, \rho) \in \mathcal{D}} \|\mathcal{H}_{\{\mathbf{W}, \mathbf{b}\}}(\mathbf{y}) - \rho\|^2.$$

Alternatively,  $\rho$  can be estimated from the learned radar sensing through data reconstruction. First, a radar sensing such as  $\mathbf{a}(\theta)$  in (4) is learned from training data  $\mathbf{y}$  and the known true  $\rho$  in a complex-valued neural network  $\mathcal{F}_{\{\mathbf{W}_0, \mathbf{b}_0\}}$ . It serves as a decoder and provides signal reconstruction with  $\mathcal{F}_{\{\mathbf{W}_0, \mathbf{b}_0\}}(\rho) = \tilde{\mathbf{y}}$ . Next, an encoder  $\mathcal{H}_{\{\mathbf{W}_1, \mathbf{b}_1\}}$  is trained through the minimization of the reconstruction loss (also known as residuals) obtained using the trained decoder:

$$L_{\mathcal{D}}(\psi, \phi) = \frac{1}{\text{Card}(\mathcal{D})} \sum_{\mathbf{y} \in \mathcal{D}} \|\mathcal{F}_\psi(\mathcal{H}_\phi(\mathbf{y})) - \mathbf{y}\|^2.$$

We use the notation  $\psi$  for the training parameters of the decoder  $\{\mathbf{W}_0, \mathbf{b}_0\}$  and  $\phi$  for  $\{\mathbf{W}_1, \mathbf{b}_1\}$ , the training parameters of the encoder. This structure results in an auto-encoder (AE) scheme whose principle is also illustrated in Fig.1-b).

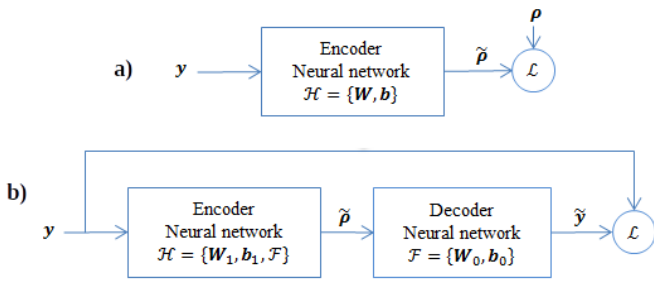


Fig. 1. Two Machine Learning schemes for solving  $\rho = [\alpha, \theta]$  from data  $\mathbf{y}$  as in (4) using complex-valued neural networks and a mismatch function  $L$ .

### B. Stopping Criterion and Radar Detection

Machine Learning for radar involves noisy data as in (4). Accordingly, the training of NNs must be stopped when further training involves learning from the noise. The AE approach for NNs parameter estimation described in the previous subsection enables the design of a simple stopping criterion. Indeed, since the residuals would contain only noise when the NNs estimation and data reconstruction are exact (i.e when  $\mathcal{H}_\psi(\mathbf{y}) = \rho$  and  $\mathcal{F}_\psi(\rho) = \alpha\mathbf{a}(\theta)$ ), we propose to stop the training when the average reconstruction error approaches  $M\gamma$  as follows, assuming the variance  $\gamma$  known:

$$\begin{aligned} L_{\mathcal{D}}(\psi, \phi) &\approx \frac{1}{\text{Card}(\mathcal{D})} \sum_{\mathbf{y} \in \mathcal{D}} \mathbb{E} [\|\mathbf{y} - \mathcal{F}_\psi(\mathcal{H}_\psi(\mathbf{y}))\|^2], \\ &= \frac{1}{\text{Card}(\mathcal{D})} \sum_{\mathbf{z} \in \mathcal{D}} \mathbb{E}[\mathbf{z}^H \mathbf{z}] = M\gamma. \end{aligned} \quad (5)$$

In other words, NNs shall distinguish a signal in data, i.e. information about unknowns. In the radar detection theory, the Neyman-Pearson (NP) lemma (e.g. [20]) provides a statistical test for optimal signal detection at pre-imposed false alarms. Thus, NNs with the stopping criterion (5) for the training must achieve this detection purpose. This detection approach avoids also overfitting, i.e. learning from the noise.

Generalized Likelihood Ratio Test (GLRT) can perform the NP-detection test with the test statistic  $\lambda_{\text{GLRT}}(\mathbf{y}) = |\mathbf{a}(\theta)^H \mathbf{y}| / \|\mathbf{a}(\theta)\|$  compared to a threshold  $\eta = \sqrt{-\gamma \log P_{\text{fa}}}$  at a fixed probability of false alarms  $P_{\text{fa}}$ . The auto-encoder structure enables the implementation of such a test with NNs with the test statistic:

$$\lambda_{\text{AE}}(\mathbf{y}) = |\mathcal{F}_\psi(\mathcal{H}_\psi(\mathbf{y}))^H \mathbf{y}| / \|\mathcal{F}_\psi(\mathcal{H}_\psi(\mathbf{y}))\|.$$

which equals  $\lambda_{\text{GLRT}}(\mathbf{y})$  when the parameter estimation and data reconstruction by the neural networks are exact.

### IV. BAYESIAN ESTIMATION WITH VARIATIONAL AUTO-ENCODERS

In radar, the preferred stochastic approach is exemplified by a VAE (e.g. [7]). A VAE can learn a stochastic decoder  $p_\psi(\mathbf{y}|\rho)$  and the posterior  $q_\phi(\rho|\mathbf{y})$  in complex-valued NNs. This enables the estimation of probability density functions (pdfs) instead of single-point estimates which cannot give any information about the confidence to have in the estimation.

### A. Generative and Inference Models

A generative model is a joint probability distribution over all the variables of the problem. It is decomposed in such a way that it exhibits the forward model of the observed variable given hidden conditional variables. For radar, we would write:

$$\log p_\psi(\mathbf{y}, \rho) = \log p_\psi(\mathbf{y}|\rho) + \log p(\rho).$$

using the logarithm as it is more convenient to optimize numerically when dealing with values close to zero.  $p(\rho)$  is a prior distribution thus, it is assumed a priori.  $p_\psi(\mathbf{y}|\rho)$  is learned by a neural network with learning parameters  $\psi$ . The encoding of a pdf is done by first selecting its type of distribution (Gaussian, complex-Gaussian, uniform...), and the parameters relative to this type are outputs of a neural network. The decoder  $p_\psi(\mathbf{y}|\rho)$  would then be encoded by:

$$p_\psi(\mathbf{y}|\rho) \sim \mathcal{CN} \left( \alpha e^{j\mu_\beta(\theta, \psi)}, \gamma \mathbf{I}_M \right).$$

where  $\mu_\beta(\theta, \psi)$  is the real part of the output of a complex-valued NN that takes  $\theta$  as input, and whose learning parameters are contained in  $\psi$ , so that  $e^{j\mu_\beta(\theta, \psi)}$  is the learned sensing vector which aims at approximating  $\mathbf{a}(\theta)$  as in (4). The noise power  $\gamma$  is assumed known. Besides, the inference model corresponds to the pdf of all the hidden variables given the observed ones. Hence, for radar, it is designed as:

$$\log q_\phi(\rho|\mathbf{y}) = \log q_\phi(\alpha|\mathbf{y}) + \log q_\phi(\theta|\mathbf{y}).$$

which relies on the assumption that, given the data, the amplitude and the non-linear parameter are independent. Gaussian distributions are selected for the pdfs of the decoder:  $q_\phi(\tilde{\varphi}) \sim \mathcal{N}(\mu_\varphi(\mathbf{y}, \phi), \sigma_\varphi^2(\mathbf{y}, \phi))$ , estimating now  $\tilde{\varphi}$ , the phase of a unitary amplitude:  $\tilde{\alpha} = e^{j\tilde{\varphi}}$ , and  $q_\phi(\tilde{\theta}) \sim \mathcal{N}(\mu_\theta(\mathbf{y}, \phi), \sigma_\theta^2(\mathbf{y}, \phi))$ .

### B. Training a Variational Auto Encoder

In radar, a VAE is trained in such a way that it learns simultaneously the forward model of the data and the inverse model that provides estimates of the target unknown. This results in the joint optimization of two different cost functions.

The decoder of the VAE learns the true sensing model of data through the maximization of the likelihood of data in the training dataset, i.e. for  $(\mathbf{y}, \rho) \in \mathcal{D}$ :

$$L_1(\mathbf{y}, \rho, \psi) = \log p_\psi(\mathbf{y}, \rho).$$

as  $p_\psi(\mathbf{y}|\rho)$  is optimized with the known true  $\rho$ . Besides, the encoder of the VAE is trained by maximizing the ELBO :

$$\begin{aligned} L_2(\mathbf{y}, \psi, \phi) &= \mathbb{E}_{q_\phi(\tilde{\rho}|\mathbf{y})} [\log p_\psi(\mathbf{y}, \tilde{\rho}) - \log q_\phi(\tilde{\rho}|\mathbf{y})], \\ &= \log p_\psi(\mathbf{y}) - KL(q_\phi(\tilde{\rho}|\mathbf{y}) || p_\psi(\tilde{\rho}|\mathbf{y})). \end{aligned} \quad (6)$$

It can be seen that maximizing  $L_2$  will maximize the average  $\log p_\psi(\mathbf{y}, \tilde{\rho})$  for  $\mathbf{y} \in \mathcal{D}$  w.r.t  $\tilde{\rho} \sim q_\phi(\tilde{\rho}|\mathbf{y})$ , meaning that the joint probability of data in the training dataset with the estimated  $\tilde{\rho}$  is maximized. Simultaneously, the encoder  $q_\phi(\tilde{\rho}|\mathbf{y})$  approximates the true learned posterior  $p_\psi(\tilde{\rho}|\mathbf{y})$  because their KL-divergence is minimized as shown by (6).

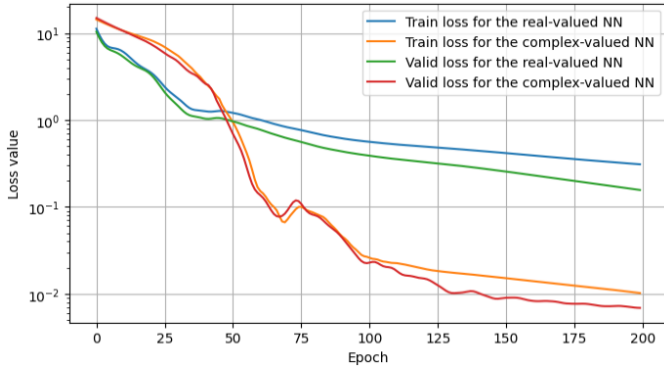


Fig. 2. Comparison of the training and validation losses for the real-valued and complex-valued networks during the learning.

A major issue comes from the fact that  $L_2$  is intractable, thus it cannot be directly optimized. Instead, an unbiased estimation of  $L_2$  is used for the training as suggested in [7]:

$$\tilde{L}_2(\mathbf{y}, \psi, \phi) = \log p_{\psi}(\mathbf{y}, \tilde{\rho}) - \log q_{\phi}(\tilde{\rho}|\mathbf{y}).$$

where  $\tilde{\rho}$  is a sample from the distribution  $q_{\phi}(\tilde{\rho}|\mathbf{y})$ . In order to train the VAE with unbiased estimations of the gradients of  $L_2$ , we use the reparametrization trick [7] which here would result in the following sampling :  $\tilde{\varphi} = \mu_{\varphi}(\mathbf{y}, \phi) + \sigma_{\varphi}(\mathbf{y}, \phi)\varepsilon_1$  and  $\tilde{\theta} = \mu_{\theta}(\mathbf{y}, \phi) + \sigma_{\theta}(\mathbf{y}, \phi)\varepsilon_2$  where  $\varepsilon_1$  and  $\varepsilon_2$  are independently sampled from a reduced Gaussian distribution  $\mathcal{N}(0, 1)$ . Finally, the overall function maximized for the training is :

$$\tilde{L}_{\mathcal{D}}(\psi, \phi) = \frac{1}{\text{Card}(\mathcal{D})} \sum_{(\mathbf{y}, \rho) \in \mathcal{D}} L_1(\mathbf{y}, \rho, \psi) + \tilde{L}_2(\mathbf{y}, \psi, \phi). \quad (7)$$

Hence, the proposed scNNr scheme involves the invaluable numerical ability for learning nonlinear sensing models from data and estimating target parameters directly in the continuous domain. In realistic scenarios with partly unknown sensing, scNNr can refine sensing models directly by using data and accordingly, can outperform existing processing in the accuracy of the estimation  $\tilde{\rho}$ .

## V. NUMERICAL RESULTS

In this section, we illustrate in simulation, the performances of the proposed scNNr scheme for radar.

### A. Complex-valued NNs vs Real-valued NNs

We first illustrate the relevance of using complex-valued NNs for radar signal processing. We train a real-valued and a complex-valued network to estimate the phase  $\phi$  from discrete measurements vectors  $s$ ,  $s(t_k) = e^{j(\omega t_k + \phi)}$  for  $k = 1, 2, \dots, 10$ . Both networks have only one hidden layer which has the same number of units than the input layer, an output layer with a single neuron which outputs an estimation of  $\phi$ , and use the hyperbolic tangent activation function. The weights are initialized randomly with complex-Gaussian distributions. The real-valued network is fed with the real and imaginary parts of the signals and thus has more training parameters, and both networks are trained using the same learning rate by

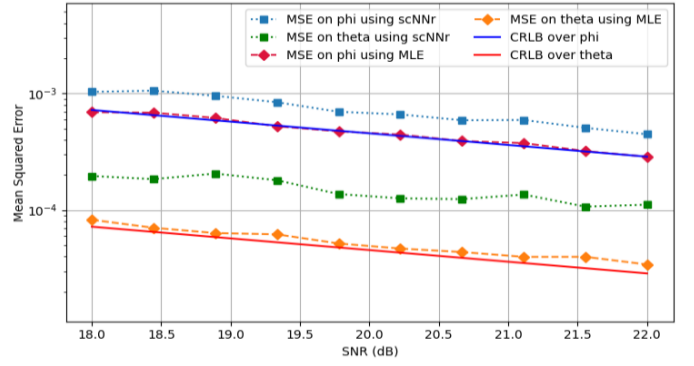


Fig. 3. MSE of the estimations provided by MLE, by the posteriors of the target parameters given by the VAE, and CRLB w.r.t SNR of test data.

minimizing the MSE loss function. Fig. 2 shows the evolution of the train and validation losses during the learning phase. The results clearly indicate that the complex-valued network is much more efficient in learning due to the structural adequacy of complex numbers for wave-related signals.

### B. Neural Networks Architectures

In the following, we design a complex-valued VAE and assess its performances on simulated data. The designed VAE contains in total 5 complex-valued NNs : one outputs the mean of the distribution  $p_{\psi}(\mathbf{y}|\rho)$  (its variance  $\gamma \mathbf{I}_M$  is assumed known), two NNs output the mean and variance of  $q_{\phi}(\varphi|\mathbf{y})$  and two last ones output the mean and variance of  $q_{\phi}(\theta|\mathbf{y})$ .

The NN which outputs  $\mu_{\beta}(\theta, \psi)$  has two hidden layers with  $M$  neurons which use the hyperbolic tangent activation function, while all the others NNs have one single hidden layer with the same number of units and also use the hyperbolic tangent activation function. For the NNs that output values that are real ( $\varphi, \theta, \mu_{\beta}$ ), we take the real part of the output layer's value or take the  $l_2$ -norm when the NN outputs a variance.

### C. Data Generation

We generate a training dataset  $\mathcal{D}$  of 1000 measurements. We use an aperture size of  $M = 11$  and an observation vector:

$$\beta = \left[ -\frac{M-1}{2}, -\frac{M-1}{2} + 1, \dots, \frac{M-1}{2} \right]^T.$$

The data are generated using (4) with  $\alpha = e^{j\varphi}$ ,  $\varphi$  taking 10 possible values regularly distributed between 0 and  $\pi/2$ ,  $\theta$  takes 10 possible values regularly distributed between 0 and 1. For each couple  $(\varphi, \theta)$ , 10 noisy measurements  $\mathbf{y}$  are generated with a SNR of + 20 dB (i.e. with a known  $\gamma = 0.01$ ). The VAE is trained by maximizing (7) with the Adam optimizer.

### D. Results

The performances are first assessed in terms of parameter estimation. The VAE is trained on the dataset designed in the previous subsection and the training is stopped due to the stopping criterion (5), using  $\tilde{\alpha} e^{j\mu_{\beta}(\tilde{\theta}, \tilde{\psi})}$  for  $\tilde{\mathbf{y}}$ . Fig. 3 shows the MSE on the estimation of the amplitude's phase  $\varphi$  and the parameter  $\theta$  w.r.t the SNR of test data. To consider a

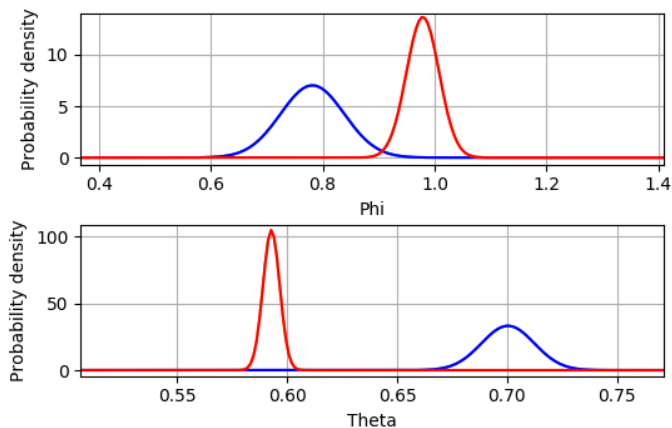


Fig. 4. Pdfs  $q_\phi(\tilde{\varphi}|\mathbf{y})$  and  $q_\phi(\tilde{\theta}|\mathbf{y})$  obtained from data  $\mathbf{y}$  generated with  $\varphi = 1$  and  $\theta = 0.6$  for an SNR = +20 dB (red curves) and an SNR = 0 dB (blue curves).

reference, we also display the MSE obtained using Maximum Likelihood Estimation (MLE) and the Cramer-Rao Lower Bounds (CRLB) for each parameter. The CRLB provides the least possible MSE for an unbiased estimator, and represents the best performances we could expect from data  $\mathbf{y}$  and known models. In theory, the CRLB is achieved by MLE. The results show that VAEs are appropriate for parameter estimation as the obtained MSEs remain close to their respective CRLB for values of SNR around +20 dB. The question for further work is whether the MSE of the VAEs estimation directly from data (with unknown models) could approach more closely the performances of MLE (which assumes known models).

With the aim to provide an insight of the VAE performance in both estimation and uncertainty assessment, we show the posteriors  $q_\phi(\tilde{\varphi}|\mathbf{y})$  and  $q_\phi(\tilde{\theta}|\mathbf{y})$  with examples obtained in two different cases of SNR. The posteriors are fed with measurements  $\mathbf{y}$  generated using  $\varphi = 1$ ,  $\theta = 0.6$  and SNRs of +20 dB and 0 dB. We can see on Fig.4 that not only the estimated means are more accurate in the case when the SNR is higher, but also the estimated variance is larger when the SNR decreases. This is the desired feature from our VAEs.

Finally, we show that the learned models are compatible with the principles of signal detection. We implement the detection test described in Section III.B using the models  $\alpha e^{j\mu_\beta(\theta, \psi)}$  for  $\mathcal{F}$  and  $\{\mu_\varphi(\mathbf{y}, \phi), \mu_\theta(\mathbf{y}, \phi)\}$  for  $\mathcal{H}$ , and evaluate empirically the probability of false alarms on test data w.r.t. the SNR for the fixed  $P_{fa} = 10^{-3}$ . Fig. 5 shows the results of the simulation. We can see that the learned signal model enables fixing the  $P_{fa}$  at the desired value for the considered range of SNR around +20 dB. This shows the potential of scNNr for radar with the designed stopping criterion based on the radar detection principles which naturally avoids overfitting, i.e. learning from the noise.

## VI. CONCLUSION

We elaborated complex-valued Bayesian NNs for parameter estimation and model refinement in radar. We also included a stopping criterion based on detection to avoid overfitting.

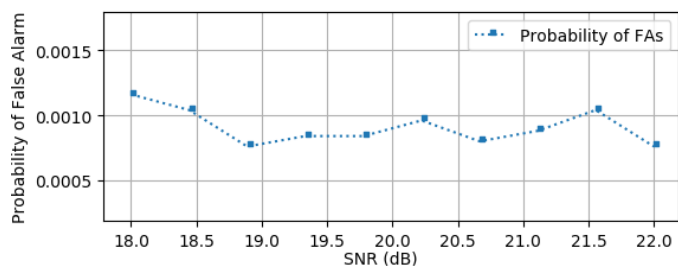


Fig. 5. Empirical probabilities of false alarms w.r.t. the SNR on test data.

The results show that they have potential for radar as they bring major benefits such as e.g. continuous parameter estimation, learning models directly from data and the uncertainty assessment. The initial results on learning data models with the Bayesian NNs of a VAE show the potential whose accuracy bounds are to be further investigated.

In future work, complex-valued VAEs using larger datasets with different SNRs are to be trained. Moreover, scNNr with fewer data from compressing-sensing radar is also to be explored, so that computational costs could decrease while the estimation performance could be maintained.

## REFERENCES

- [1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, 521(7553), pp. 436–444, 2015.
- [2] J. E. Mason, B. Yonel, B. Yazici, "Deep Learning for Radar", *IEEE Radar Conference (RadarConf)* 2017.
- [3] P.M. Woodward, *Probability and Information Theory with Applications to Radar*, Pergamon, 1953.
- [4] C. E. Cook and M. Bernfeld, *Radar signals: an introduction to theory and application*. Acad.Press, 1967.
- [5] M.Welling, "Exploiting the statistics of learning and inference". NIPS workshop, 2013.
- [6] M.Welling, "The Bayesian Bonus: Benefits of Being Bayesian in the Deep Learning Era", *IEEE Workshop on Machine Learning and Signal Processing* 2018.
- [7] D.P. Kingma, *Variational inference & deep learning: a new synthesis*. PhD thesis, UvA 2017.
- [8] C. R. Rao. Information and the accuracy attainable in the estimation of statistical parameters. *Bull. Calcutta Math.Soc.* 37, 1945.
- [9] S. Amari, "Natural Gradient Works Efficiently in Learning," *MIT Neural Computation* 10, 1998.
- [10] T. Liang, T. Poggio, A. Rakhlin and J. Stokes, "Fisher-Rao Metric, Geometry, and Complexity of Neural Networks", <https://arxiv.org/pdf/1711.01530.pdf>, 2017.
- [11] A. Hirose, "Complex-valued neural networks: The merits and their origins", *Conference on Neural Networks*, 2009.
- [12] R.Pribić, "Stochastic Deep Learning for Compressive-sensing Radar", *SEE Radar* 2019.
- [13] O-L. Ouabi, "Stochastic Complex-valued Deep Learning for Radar", *CentraleSupélec MSc thesis at Thales NL*, 2019.
- [14] Ken Kreutz-Delgado, "The Complex Gradient Operator and CR calculus", <https://arxiv.org/pdf/0906.4835.pdf>, 2009.
- [15] S.M Kay. *Fundamentals of Statistical Signal Processing Vol. II: Estimation Theory*. Prentice Hall, 1998.
- [16] T.M. Cover, J. A. Thomas, *Elements of Information Theory*, JohnWiley & Sons, Inc., 1991.
- [17] J.H. Huggins, M. Kasprzak, T. Campbell and T. Broderick, "Practical Bounds on the Error of Bayesian Posterior Approximations: A Nonasymptotic Approach", <https://arxiv.org/pdf/1809.09505.pdf>, 2018.
- [18] I. Tolstikhin, O. Bousquet, S. Gelly and B. Schölkopf, "Wasserstein Auto-Encoder", *ICLR workshop*, 2018.
- [19] H. L. Van Trees, *Optimum Array Processing*. Wiley, 2002.
- [20] R. Pribić, H.L. Yap, "False Alarms in Radar Detection within Sparse-signal Processing", *IEEE Workshop CoSeRa* 2016.