Vector-Valued Kernel Ridge Regression for the Modeling of the Scattering Parameters of a Slotted Ground PCB Structure

(Article begins on next page)

4th URSI AT-RASC, Gran Canaria, 19 – 24 May 2024

Vector-Valued Kernel Ridge Regression for the Modeling of the Scattering Parameters of a Slotted Ground PCB Structure

N. Soleimani(1), T. Bradde(1), P. Manfredi(1), I. S. Stievano(1) and R. Trinchero(1)
(1) Politecnico di Torino, Torino, Italy; e-mail:
{nastaran.soleimani,tommaso.bradde,paolo.manfredi,igor.stievano,riccardo.trinchero}@polito.it

Abstract

This paper presents a preliminary study on the performance of the vector-valued kernel Ridge regression in complex-valued regression problems. The proposed approach is applied for the modeling of the scattering parameters of a slotted ground PCB structure as a function of 2 parameters. The effectiveness of the proposed modeling scheme in terms of both training time and accuracy is assessed for an increasing number of training samples.

1 Introduction

Machine Learning (ML) methods have achieved notable success in various electrical and electronic applications. Despite demonstrating high performance in practical problems, the majority of conventional formulations for these techniques have primarily been designed to handle real-valued data [1].

However, complex-valued data find widespread use in electronic applications, particularly in AC simulations and frequency-domain analysis [2]–[5]. In such scenarios, the most straightforward approach to extend the applicability of real-valued ML regression techniques to complex-valued data is through the dual-channel formulation [6]–[9]. In this formulation, the complex-valued problem is transformed into two uncorrelated real-valued ones by concatenating the real and imaginary parts of the complex input and output values. While this method allows for the direct adoption of plain real-valued ML techniques, without the need for generalization or modification, its application to vector-valued problem via scalar regression approaches requires the training of a large number of single output models. Moreover such scheme overlooks potential correlations between the real and imaginary parts of the complex-valued output [6]–[9].

Due to these limitations, alternative formulations dedicated to pure complex-valued data have been proposed for various ML techniques. These formulations aim to account for the inherent correlation between the real and imaginary parts of complex variables in methods such as Artificial Neural Networks (ANN) [7], Support Vector Machine (SVM) regression [8], kernel Least Squares regression [3, 6], Least-Squares Support Vector Machine regression (LS-SVM) [9, 10], and Gaussian Process regression (GPR) [11].

Unfortunately, for the specific case of complex-valued kernel-based regressions (e.g., SVM regression, LS-SVM regression, and GPR), a critical challenge arises in determining the appropriate complex-valued kernel. In contrast to the well-established real-valued formulation of kernel regressions, where Gaussian or polynomial kernels are default choices, the selection and/or construction of complex-valued kernels is inherently more challenging and mathematically intricate [9, 10]. Furthermore, depending on their complexity and structure, these kernels may necessitate the tuning of several hyperparamters. Due to the aforementioned constraints, the adoption of complex-valued formulations in kernel regressions has not gained extensive traction in engineering applications, particularly in electronic and electrical domains for which the dual-channel formulation typically emerges as the favored choice (see [9] and the references therein for further insights).

This work explores the capabilities and the viability of an alternative solution for complex-valued regression problems based on the vector-valued kernel ridge regression (VV-KRR). Similar to the dual-channel formulation, the complex- and vector-value output data are recast into real ones by stacking their real and imaginary parts. This results in a new real- and vector-valued regression with an output dimension that is the double of the original. The above regression problem can be learnt by a single vector-valued model trained via the VV-KRR. The resulting model allows accounting for the coupling between real and imaginary part of the complex-valued problem by exploring the correlation in the output dimensions [12]–[16]. Specifically, this works exploits the effectiveness of a pure data-driven output kernel built by looking at the correlation on the output dimension in the training set.

The proposed modeling approach is validated by considering the modeling of the scattering parameters characterizing the port behavior of a slotted ground PCB structure as a function of 2 parameters. The performance of the proposed technique are investigated in both deterministic and statistical sense.

2 Vector-Valued Kernel Ridge Regression & Complex-Valued Problems

In this work the VV-KRR will be used to learn a generic vector-value map, hereafter referred to as surrogate model, $\hat{\boldsymbol{f}} : \mathcal{X} \rightarrow \mathcal{Y}$, starting from the complex-valued output data collected in the training set $\mathcal{D} = \{(\boldsymbol{x}_l, \boldsymbol{y}_l\}_{l=1}^{L_s}$, where $\boldsymbol{x}_l \in \mathcal{X} \subseteq \mathbb{R}^p$ and the vectors $\boldsymbol{y}_l = [\text{Re}\{\boldsymbol{y}_l^C\}^T, \text{Im}\{\boldsymbol{y}_l^C\}^T]^T \in \mathcal{Y} \subseteq \mathbb{R}^{2D}$ defined by stacking the real and imaginary parts of the realizations of the vector- and complex-valued output $\boldsymbol{y}_l^C \in \mathbb{C}^D$. It is important to remark that the above manipulation of the training set leads to a real-valued regression problem in which the dimensionality of the real-valued output space (i.e., $2D$) is two times the one of the original complex one (i.e., D).

The above learning problem turns out to be equivalent to learn $2D$ scalar real-valued functions $\hat{f}^{(d)} : \mathcal{X} \rightarrow \mathbb{R}$ with $d = 1, \ldots, 2D$ minimizing the following empirical risk functional:

$$\hat{\boldsymbol{f}} = \arg\min_{\tilde{\boldsymbol{f}} \in \mathcal{H}} \sum_{d=1}^{2D} \sum_{l=1}^{L_s} (y_l^{(d)} - \tilde{f}^{(d)}(\boldsymbol{x}_l))^2 + \lambda \|\tilde{\boldsymbol{f}}\|_{\mathcal{H}}^2, \quad (1)$$

where λ is the regularizer hyperparameter and, $y_l^{(d)}$ and $\tilde{f}^{(d)}(\boldsymbol{x}_l)$ represent the d-th component of the $l-$th training output and the corresponding model prediction, respectively.

As per the theorem introduced in [14], any optimal solution $\hat{\boldsymbol{f}}$ for (1) can be formulated as:

$$\hat{\boldsymbol{f}}(\boldsymbol{x}) = \sum_{l=1}^{L_s} \mathbf{K}(\boldsymbol{x}, \boldsymbol{x}_l)\boldsymbol{c}_l, \quad (2)$$

where $\hat{\boldsymbol{f}}(\boldsymbol{x}) = [\hat{f}^{(1)}(\boldsymbol{x}), \ldots, \hat{f}^{(2D)}(\boldsymbol{x})]^T$ is a vector collecting the model prediction for any $\boldsymbol{x} \in \mathcal{X}$, $\mathbf{K}(\cdot,\cdot) : \mathbb{R}^{p \times p} \rightarrow \mathbb{R}^{2D \times 2D}$ is the matrix kernel function and $\boldsymbol{c}_l = [c_{1,l}, \ldots, c_{2D,l}]^T \in \mathbb{R}^{2D}$ with $l = 1, \ldots, L_s$ are column vectors collecting the regression unknowns, which must be estimated during the training phase.

The multi-output kernel needs to address correlation in both parameter space and output components, which in this case includes also the correlation among the real and imaginary parts of the output. However, different from the scalar case, there is a lack of ready-made kernel functions suitable for direct application in a vector-valued context [12]. A straightforward approach is to focus on a particular category of multi-output kernels, such as the separable kernel or the sum of separable kernels [14].

Hereafter we will consider a separable matrix kernel function $\mathbf{K}(\boldsymbol{x}, \boldsymbol{x}')$ derived as the product of two scalar kernels operating on either the input space or the output dimensions, which writes:

$$[\mathbf{K}(\boldsymbol{x}, \boldsymbol{x}')]_{[d,d']} = k_{\boldsymbol{x}}(\boldsymbol{x}, \boldsymbol{x}') k_o(d, d'), \quad (3)$$

where $k_{\boldsymbol{x}}$ and k_o are scalar kernels acting independently on the input space (i.e., $k_{\boldsymbol{x}} : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$) and on the output dimensions (i.e., $k_o : \{1, \ldots, 2D\} \times \{1, \ldots, 2D\} \rightarrow \mathbb{R}$), respectively.

Therefore the separable kernel matrix $\mathbf{K}(\mathbf{x}, \mathbf{x}')$ can be expressed as:

$$\mathbf{K}(\boldsymbol{x}, \boldsymbol{x}') = k_{\boldsymbol{x}}(\boldsymbol{x}, \boldsymbol{x}')\mathbf{B}, \quad (4)$$

where $\mathbf{B} \in \mathbb{R}^{2D \times 2D}$ is a symmetric semi-definite matrix accounting for the correlation among the output dimensions.

Similar to the scalar case, Gaussian or polynomial kernels can be seen as default choices for the kernel $k_{\boldsymbol{x}}$, since it acts on the input parameters only. On the other hand, the choice of the output kernel k_o or of the corresponding the matrix $\mathbf{B}$ is more problematic. Several structures have been proposed in the literature (see [15, 16] and the references therein).

In this work, we investigate a purely data driven approach in which the output kernel matrix $\mathbf{B}$ in (4) is built from the correlation matrix computed on the output dimension:

$$\hat{\mathbf{B}} = \text{corr}(\mathbf{Y}), \quad (5)$$

where $\mathbf{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{L_s}]^T$ is a $L_s \times 2D$ matrix associated to the training output, in which the rows correspond to observations, and the columns correspond to output dimensions. The resulting correlation matrix $\hat{\mathbf{B}}$ is a $2D \times 2D$ matrix, such that:

$$\hat{B}_{ij} = \text{E}\left[\boldsymbol{y}_{(:,i)}^T \boldsymbol{y}_{(:,j)}\right], \quad (6)$$

in which the expected valued is computed on the inner product between the i- and j-column of the matrix $\mathbf{Y}$, respectively. It is important to notice that after standardization the realizations collected in the columns of the matrix $\mathbf{Y}$ have zero mean and standard deviation equal to 1. The output kernel $\mathbf{B}$ is obtained as a low-rank approximation of the matrix $\hat{\mathbf{B}}$. Further details will be provided in a future report.

The separable kernel structure of the matrix kernel in (4) allows reformulating the training problem in (1) in terms of the following discrete-time Sylvester equation:

$$\mathbf{K}_{\boldsymbol{x}}\mathbf{C}\mathbf{B} + \lambda\mathbf{C} = \mathbf{Y}, \quad (7)$$

where $\mathbf{K}_{\boldsymbol{x}}$ is a $L_s \times L_s$ Gram matrix of the input kernel (i.e., $[\mathbf{K}_{\boldsymbol{x}}]_{ij} = k_{\boldsymbol{x}}(\boldsymbol{x}_i, \boldsymbol{x}_j)$), $\mathbf{B}$ is the the output kernel defined in (5), $\mathbf{C} = [\boldsymbol{c}_1, \ldots, \boldsymbol{c}_L]^T \in \mathbb{R}^{L_s \times 2D}$ is a matrix collecting the model unknowns and $\mathbf{Y}$ is the $L_s \times 2D$ matrix associated to the training output used in (5).

The above discrete-time Sylvester equation is then efficiently solved via the diagonalization procedure presented in [13].

3 Application Example: PCB Interconnect with Slotted Ground Plane

The application example concerns the PCB interconnect with slotted ground plane considered in [5] and shown in Fig. 1. A copper microstrip line with width $t = 0.12\,\text{mm}$ and a thickness of $35\,\mu\text{m}$ runs over a square dielectric substrate of size $a \times b$, with $a = b = 100\,\text{mm}$, thickness $h = 0.3\,\text{mm}$, and relative permittivity $\varepsilon_r = 4.3$. The bottom ground plane has a transversal slot of width $w = 0.12\,\text{mm}$, with a nominal length L and offset d from the midpoint of 15 mm each. These two parameters are considered as independent Gaussian random variables with a 10% relative standard deviation.

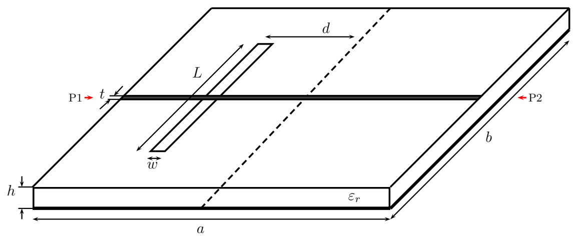

Figure 1. PCB interconnect with slotted ground plane [5].

S-parameter samples for the considered two-port structure in Fig. 1 are computed with CST Studio Suite from dc to 10 GHz at $D = 1389$ linearly spaced frequency points $\{f_d\}_{d=1}^{D}$ for 1000 random configurations of the uncertain parameters, drawn according to a latin hypercube design. The data obtained for each S-parameter are then normalized via standardization and split into the training and test set by stacking their real and imaginary part.

For each S-parameter, the corresponding training set is used to train a vector-valued model with output dimension $D = 2 \times 1389 = 3718$ via the VV-KRR presented in Sec. 2 by using a Gaussian kernel as input kernel and the correlation matrix in (5) as output kernel. The model hyperparameters are tuned via a 3-fold cross-validation.

Table 1 provides a quantitative analysis of the performance of the regression models built for S_{11}, S_{22} and S_{21} (which is equal to S_{12} due to reciprocity of the considered structure) in terms of training time and relative $L2$-norm error computed on 500 test samples for an increasing number of the training samples (i.e., $L_s = 50$, 150 and 300).

The results highlight the capability of the proposed modeling strategy of accurately learning the actual parametric behavior of the S-parameters. An average error $< 5\,\%$ is achieved for all the considered S-parameters over the considered frequency bandwidth with $L_s = 300$ sample. It is important to remark that a Gaussian distribution of the input parameters has been considered, while for the case of the parametric modeling a uniform distribution is usually preferred. Concerning the computational cost, the overall training time turns out to be less than 5 min for $L_s = 300$ and the overall evaluation time is around 1 s, while a single CST simulation for a given configuration of the input parameters requires 2 min.

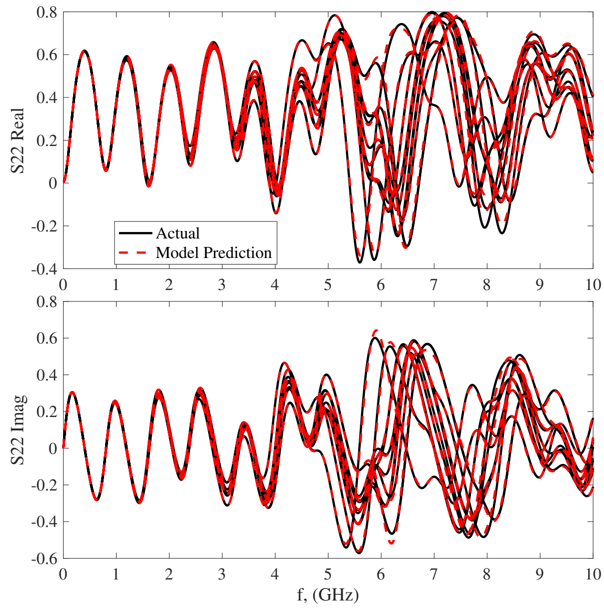

Figure 2. Parametric plots comparing the real and imaginary parts of S_{22} predicted by the proposed model trained with 300 samples with the corresponding values in the test-set for 10 different realizations of the input parameters.

Table 1. Training time and relative L2-norm error computed from the regression models built for the S-parameter of the structure in Fig. 1 with an increasing number of training samples (i.e., L_s).

Param.	$L_s = 50$		$L_s = 150$		$L_s = 300$	
	ε_{L2}	t_{train}	ε_{L2}	t_{train}	ε_{L2}	t_{train}
S_{11}	8.8%	40s	4.6%	59s	3.9%	90s
S_{12} / S_{21}	3.3%	45s	2.4%	69s	2.4%	104s
S_{22}	8.6%	40s	5.6%	60s	4.1%	92s

Figure 2 shows a parametric plot providing a comparison between the real and imaginary parts of S_{22} predicted by the proposed model trained via $L_s = 300$ training samples with the corresponding responses in the test set for 10 random configurations of the input parameters. The plots highlight the complexity and the richness of the responses in the considered testcase, as well as the remarkable accuracy achieved by the proposed model.

Moreover, Fig. 3 compares the probability density functions (PDFs) of the magnitude of S_{22} at the frequency $f = 9\,\text{GHz}$ predicted by the proposed model trained with $L_s = 300$ training samples with the corresponding one computed from 500 test samples. The results provide a further statistical validation of the proposed modeling framework.

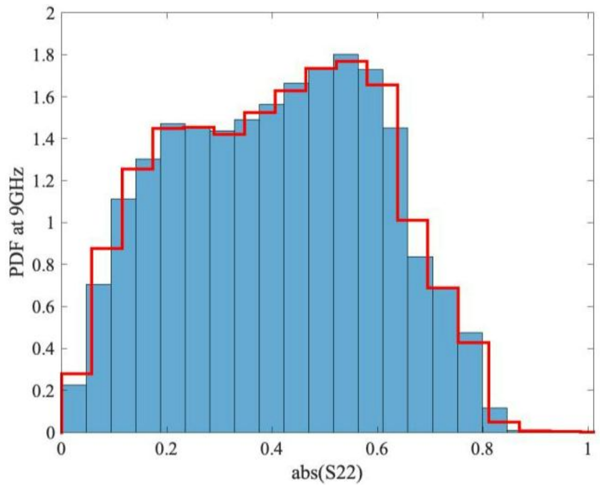

Figure 3. Comparison of the PDF of the S_{22} at 9 GHz computed from 500 test samples with the corresponding one obtained from the predictions of the proposed model trained with 300 training samples.

4 Conclusions

This paper investigates the performance of the VV-KRR in the context of complex-valued regression problems. The effectiveness of the proposed modeling scheme is assessed in terms of accuracy and training time by considering the modeling of scattering parameters for a slotted ground PCB structure as a function of 2 parameters. The results underscore the potential of the proposed method. A more comprehensive analysis, including a comparison with other state-of-the-art techniques, will be conducted in future works.

References

[1] S. Scardapane, S. Van Vaerenbergh, A. Hussain, and A. Uncini, "Complex-valued neural networks with nonparametric activation functions," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 4, no. 2, pp. 140-150, 2018.

[2] A. Papaioannou and S. Zafeiriou, "Principal Component Analysis With Complex Kernels."

[3] S. Scardapane, S. Van Vaerenbergh, D. Comminiello, and A. Uncini, "Widely Linear Kernels for Complex-Valued Kernel Activation Functions," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 8528-8532: IEEE.

[4] T. Adali, P. J. Schreier, and L. L. Scharf, "Complex-valued signal processing: The proper way to deal with impropriety," *IEEE Transactions on Signal Processing*, vol. 59, no. 11, pp. 5101-5125, 2011.

[5] P. Manfredi and S. Grivet-Talocia, "Fast Stochastic Surrogate Modeling via Rational Polynomial Chaos Expansions and Principal Component Analysis," in *IEEE Access*, vol. 9, pp. 102732-102745, 2021.

[6] R. Boloix-Tortosa, J. J. Murillo-Fuentes, and S. A. Tsaftaris, "The Generalized Complex Kernel Least-Mean-Square Algorithm," in *IEEE Transactions on Signal Processing*, vol. 67, no. 20, pp. 5213-5222, Oct. 15, 2019.

[7] A. Hirose, "Complex-Valued Neural Networks," Wiley-IEEE Press, 2013.

[8] P. Bouboulis, S. Theodoridis, C. Mavroforakis, and L. Evaggelatou-Dalla, "Complex Support Vector Machines for Regression and Quaternary Classification," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 6, pp. 1260-1274, June 2015.

[9] N. Soleimani and R. Trinchero, “Compressed complex-valued least squares support vector machine regression for modeling of the frequency-domain responses of electromagnetic structures,” *Electronics*, vol. 11, no. 4, 2022.

[10] F. Treviso, R. Trinchero, and F. G. Canavero, "Multiple delay identification in long interconnects via LS-SVM regression," *IEEE Access*, vol. 9, pp. 39028-39042, 2021.

[11] R. Boloix-Tortosa, J. J. Murillo-Fuentes, F. J. Payán-Somet, and F. Pérez-Cruz, "Complex Gaussian processes for regression," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 11, pp. 5499-5511, 2018.

[12] V. Sindhwani, H. Q. Minh, and A. C. Lozano, “Scalable matrix-valued kernel learning for high-dimensional nonlinear multivariate regression and granger causality”, in *Proc. Uncertainty in Artificial Intelligence (UAI)*, 2013.

[13] N. Soleimani and R Trinchero, “Efficient Implementation of the Vector-Valued Kernel Ridge Regression for the Parametric Modeling of the Frequency-Response of a High-Speed Link,” XXXVth URSI GASS, Sapporo, Japan, 2023, pp. 1-4.

[14] C.A. Micchelli and M. Pontil, “On learning vector–valued functions,” *Neural Computation*, vol. 17, pag. 177-–204, 2005.

[15] A. Mauricio, L. Rosasco, and N. D. Lawrence, “Kernels for vector-valued functions: A review,” *Foundations and Trends in Machine Learning* 4(3) pp. 195-266, 2012.

[16] N. Soleimani, R. Trinchero and F. Canavero, “Bridging the Gap Between Artificial Neural Networks and Kernel Regressions for Vector-Valued Problems in Microwave Applications,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 71, no. 6, pp. 2319-2332, June 2023.