Available online at www.easdprocedia.org
EASD Procedia EURODYN (2020) 1958-1967

EASD Procedia

EURODYN 2020
XI International Conference on Structural Dynamics
M. Papadrakakis, M. Fragiadakis, C. Papadimitriou (eds.)
Athens, Greece, 23–26 November 2020

A STUDY ON THE EVALUATION OF WIND INDUCED VIBRATION OF LONG-SPAN SUSPENSION BRIDGES WITH ARTIFICIAL NEURAL NETWORKS

Dario R. Fernandez[1], **Aksel Fenerci**[1], **and Ole A. Øiseth**[1]
[1] Department of Structural Engineering, NTNU
{dario.r.f.castellon,aksel.fenerci,ole.oiseth}@ntnu.no

Keywords: Long span bridges, Wind Induced Vibrations, Machine Learning.

Abstract. *The Hardanger bridge is a long span suspension bridge located in western Norway and it is subjected to wind-induced-vibrations (WIV). Since the existing methods for WIV simulation demand high computational effort, it is appealing to use Machine Learning models to perform faster analyses. Therefore, this study applies Artificial Neural Network (ANN) and Support Vector Regression (SVR) algorithms to the analytical buffeting response predictions of the Hardanger bridge. The aim is to develop a strategy for training Machine Learning models that can be extended to real datasets. The samples of WIV features were analyzed with three different models: the single hidden layer perceptron, the multilayer perceptron and the support vector regression. Using the Normalized Root Mean Square Error as the performance metric, it was possible to obtain similar results with the different models.*

1 INTRODUCTION

The Hardanger bridge is by now the longest suspension in Norway and among the slenderest bridges in the world. It has been the subject to several research campaigns focusing on wind-induced-vibrations (WIV), on which high computational effort in analytical predictions was a constant [1] [2] [3] [4]. Thus, the deployment of a surrogate model for faster WIV assessment in real datasets is required to extend the experience gained with the Hardanger bridge not only for structural health monitoring applications, but also, for design of structures with similar characteristics. Nevertheless, the application of Machine Learning to datasets of WIV on long span bridges has not been well documented yet, and the comparable studies do not cover the specific category of structure and/or dynamic effect [5] [6].Therefore, this document presents the application of Machine Learning algorithms, specifically Artificial Neural Network (ANN) and Support Vector Regression (SVR), to the Hardanger Bridge WIV simulations with the purpose of developing a training strategy that could be extended to real datasets. This development involved the construction of a training dataset with simulated WIV of the Hardanger bridge which was fed into a group of Machine Learning models. Successively, the models' predictions were tested using a new dataset. The quality of the predictions was verified and compared using the Normalized Root Mean Squared Error (NRMSE). Finally, the training and testing procedure were repeated for different noise conditions added to the structural response fields.

ISSN:2311-9020
Peer-review under responsibility of the Organizing Committee of EURODYN 2020.
doi: 10.47964/1120.9159.18554

2 MATERIALS & METHODS

2.1 Artificial neural network

An ANN is an assembly of connected and layered sets of units, where: the unit (also known as neuron or node) is an operational entity that stores and distributes information, the layers are the set of neurons, and the connections are the arrays governing the transformation rule between nodes in the contiguous layers. The neurons in a generic layer are stimulated by numeric values or signals coming from those of the previous layer. Then, a neuron is said to be activated or "fired" whenever the algebraic sum of the incoming signals overcomes a certain threshold [8]. To generalize the concept of the linear discriminant involved in the algebraic sum [9], a non-linear activation function, $f(\blacksquare)$, is applied to the result as shown in Equation (1).

The type of neural network used in this study was the so-called Multilayer Perceptron, schematically shown in Figure 1. This type of model overcomes the limitations of the single layered network by allowing the representation of higher complexity functions with their consecutive transformations of the input signals [9]. Nevertheless, to avoid feedback loops, they must be restricted to a feed-forward information flow only.

Equation (1) shows the output of a *j-th* node in the arbitrary *k-th* layer of a Multilayer Perceptron (MLP), where y_j^k is related to an activation threshold, ϑ_j^k and with, $w_{i,j}^{k-1}$ are the connection weight and x_i^{k-1} is the incoming stimulus from the *i-th* node of the previous layer. Subsequently, Equation (2) presents the operations in matrix notation. Combining these operations until the *M-th* layer corresponding to the model physical output, it is possible to obtain an estimation of the of the same $\hat{\boldsymbol{y}}$ as shown in Equation (3).This process is known as feed-forward mapping.

$$y_j^k = f\left(\sum_{i=1,j=j}^{N} w_{i,j}^{k-1}\, x_i^{k-1} + \vartheta_j^k\right) \tag{1}$$

$$y^k = f\{(w^{k-1})^T * x^{k-1} + \vartheta^k\} \tag{2}$$

$$\hat{\boldsymbol{y}}^{k=M} = f\{(w^{M-1})^T \ldots f\{(w^1)^T x^1\} + \vartheta^M\} \tag{3}$$

In the training step of the analysis, the actual output of the network is known. Therefore, a measure in the accuracy of the network can be obtained as shown in Equation (4).

$$\varepsilon = L(\hat{y}_j, y_j) \tag{4}$$

Where ε denotes the actual error and $L(\blacksquare)$ denotes the "loss" or "cost" function.

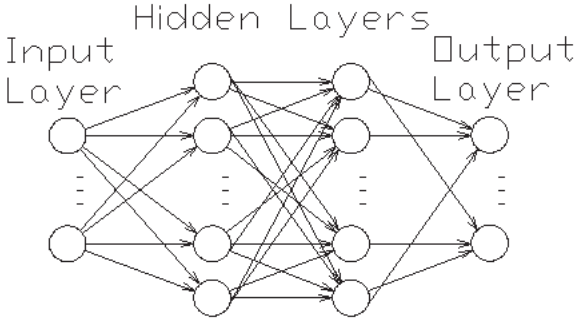

Figure 1: Architecture of ANN

The learning process consists of minimizing the loss function. For multilayered models, this is achieved by transmitting the error through all the layers within the network resulting in an

iterative optimization procedure which is known as backpropagation [10]. It is worth noting that the backpropagation procedure refers only to the error feeding and it is independent of the optimization algorithm used. Since the initial input signal, or origin, is fixed to the dataset, the only parameters to be updated in each iteration are the network weights and the neuron thresholds. Thus, the optimization problem can be written as Equation (5).

$$find\ (W_k\ \&\ \theta_k) \therefore L(\hat{y}_j, y_j) = \min(\varepsilon) \tag{5}$$

2.2 Supporting vector regression

Supporting vector regression is a machine learning tool developed by Vapnik et al. and its historical background can be found in [11]. It is the application of the supporting vector approximation to a regression problem using a ε-insensitive loss function [12]. Given a dataset with corresponding input and output features $\{x_i, y_i| \ldots |x_n, y_n\}$, the idea is to find a function $f(x)$ that approximates y_i with a certain tolerance ε. The regression estimation can be obtained with the linear function shown in Equation (6).

$$f(x) = x'w + \vartheta \tag{6}$$

Introducing the ε-insensitive loss function $L(\blacksquare)$ equal to zero when the difference of the estimation $f(x)$ and the target is less than ε, the constraint shown Equation (7) is added.

$$L(y_i - f(x,w) = \begin{cases} 0 & if|y_i - f(x,w)| \le \varepsilon \\ |y_i - f(x,w)| & otherwise \end{cases} \tag{7}$$

The best estimation minimizes the loss function and thus, the problem can be written as a convex optimization problem shown in Equation (8).

$$\begin{aligned} &minimize && \frac{1}{2}\|w\|^2 \\ &with\ constraints: && y_i - f(x,w) \le \varepsilon \quad f(x,w) - y_i \le \varepsilon \end{aligned} \tag{8}$$

To make the solution more feasible, the tolerance margin is softened by adding a set of slack variables ξ_i, ξ_i^*. The optimization then becomes as shown in Equation (9), where C is the so-called box constraint, a positive valued parameter that balances the accuracy of the model by imposing the penalty of the estimations outside the ε-margin.

$$\begin{aligned} &minimize && \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N} \xi_i + \xi_i^* \\ &with\ constraints: && \begin{cases} y_i - f(x,w) \le \varepsilon + \xi_i \\ f(x,w) - y_i \le \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \end{cases} \end{aligned} \tag{9}$$

Solving the posted optmization problem with inequality constraints is equivalient to finding the saddle point in the Lagrange functional of Equation (10).

$$L(\alpha) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\, x_i' x_j + \varepsilon\sum_{i=1}^{N}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{N} y_i(\alpha_i^* - \alpha_i) \tag{10}$$

Where α_n and α_n^* are the Lagrange multipliers with following constraints:

$$\sum_{n=1}^{N}(\alpha_n^* - \alpha_n) = 0;\ 0 \le \alpha_n \le \mathrm{C}\ \&\ 0 \le \alpha_n^* \le \mathrm{C} \tag{11}$$

The w-parameters may be found by:

$$w = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) x_i \tag{12}$$

Finally, the estimate is found by the expansion of the supporting vectors:

$$f(x) = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*)(x_i{}' x) + \vartheta \tag{13}$$

The parameters ϑ can be obtanied by exploiting the Karush Kun Tuker (KKT) conditions [13] [14], which state that the product between dual variables and constraints vanishes at the optimal solution. Thus, the constraints shown in Equation (14) are added to the problem.

$$\begin{gathered} \alpha_i(\varepsilon + \xi_i - y_i + x_i' w + \vartheta) = 0 \\ \alpha_i{}^*(\varepsilon + \xi_i^* + y_i - x_i' w - \vartheta) = 0 \\ \xi_i(C - \alpha_i) = 0 \\ \xi_i^*(C - \alpha_i{}^*) = 0 \end{gathered} \tag{14}$$

To extend the formulation to nonlinear regression problems, it is required to replace the dot product $(x_i' x)$ with a nonlinear mapping function, known as the kernel function $K(x_i' x)$

$$f(x) = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) K(x_i{}' x) + \vartheta \tag{15}$$

In this study, a polynomial kernel function of degree d is chosen as shown in Equation (16).

$$K(x_i{}' x) = (1 + x_i{}' x_j)^d \tag{16}$$

2.3 Dataset simulations

The simulations of the structural response were carried out using the multimode theory in the frequency domain, more details in [15].The along-wind turbulent spectrum (S_u) on the bridge girder was obtained from a Kaimal-type spectrum [16] given in Equation (17), where the frequency axis is represented by f, the wind field is characterized by its mean and standard deviation $V\ and\ \sigma$, respectively. The subscript u indicates the along-wind turbulent components and the term A represents the spectral quantity related to the turbulence length scale at a reference height z.

$$\frac{S_u f}{\sigma_u^2} = \frac{\mathrm{A_u} f_z}{(1 + 1.5\mathrm{A_u} f_z)^{5/3}}, f_z = \frac{zf}{V} \tag{17}$$

The inputs for the dataset were the first two statistical moments of the random wind field. Therefore, several realizations of the couples (V, σ_u) were generated and introduced in Equation (17) to obtain the wind turbulence spectra. Further, the CPSD of the structure's response for each generation was computed. Finally, by numerical integration, the output features, namely the standard deviation of the lateral, vertical and torsional responses at the midspan $(\sigma_y, \sigma_z, \sigma_\vartheta)$ were obtained.

According to the observations of the Hardanger bridge measurement campaign reported in [1] (Figure 2), most of the data points lay in a central region. This uneven spread of the samples over the variable-space leads to a strong challenge for training the machine learning models. Therefore, the couples of input features in the training dataset were generated from a uniform linearly spaced function (Figure 3) which evenly distribute the samples. On the other hand,

those of the testing set were random numbers generated from two independent normalized functions (Figure 4) which represent a behavior closer to the real observations. The parameters for the numeric generation are reported in Table 1.

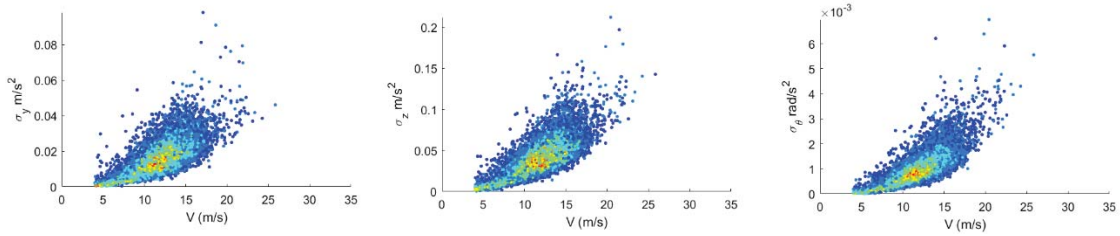

Figure 2: Structural response at midspan

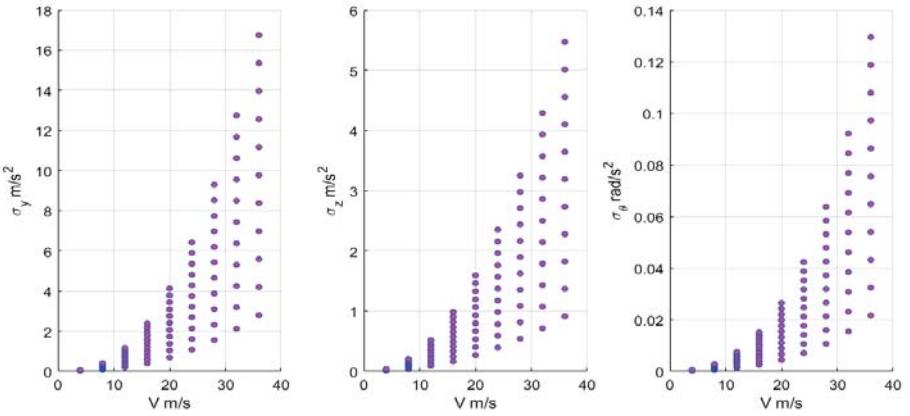

Figure 3: Training dataset

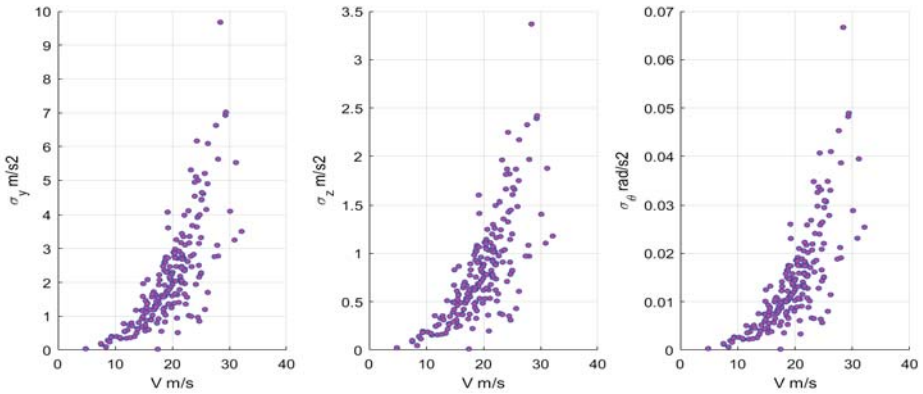

Figure 4: Validation dataset

Set	Function	Mean velocity	Turbulence Intensity	Size
Training set	Uniform Linear Spaced	[4: Δ=3.44:35].	[0.05: Δ=0.27:0.3].	100
Testing set	Rnd. Normal	$\mu = 20\ \sigma = 5$	$\mu = 0.15\ \sigma = 0.05$	200

Table 1: Dataset input generation.

2.4 Noise addition

With the aim of reproducing the challenges found on the measurement campaign, the training procedure was modified by adding a white noise to the output. The mean noise levels were of 2, 15 and 20% added to the structural responses. Nevertheless, since a decrease in the prediction's accuracy is expected with the addition of noise, a strategy for increasing the dataset size was adopted. First, several sets of random numbers were generated from a uniform

distribution according to Equation(18). Then, the different *N-terms* were added independently to the original output and stacked together to create a new larger dataset, Equation(19). The original dataset input was stacked *N-times* to match the size of the noise-output.

$$\begin{gathered} seed_i = Rand\{magnitude: [-0.5 \div 0.5] | size: OutputSet_{original}\} \\ output_{noise}(seed_i) = OutputSet_{original} \otimes \left[1 + \frac{Noise\ level}{25} seed_i\right] \\ Noise\ level \cong mean(output_{noise} - output_{original}) \\ where\ \otimes\ stands\ for\ point\ wise\ multiplication \end{gathered} \tag{18}$$

$$DataSet_{new} = \begin{bmatrix} InputSet_{original} & OutputSet_{original} + output_{noise}(seed_1) \\ InputSet_{original} & OutputSet_{original} + output_{noise}(seed_2) \\ \vdots & \vdots \\ InputSet_{original} & OutputSet_{original} + output_{noise}(seed_n) \end{bmatrix} \tag{19}$$

2.5 Models

Two ANN models were trained and validated: A Single Hidden Layer (SHL) and a Multi-layer Perceptron (MLP). The MLP model is as introduced before and the SHL perceptron model is obtained when the maximum layer index, k, in Equation (1) is equal to 2. The main challenge facing ANN training is the so-called hyperparameter setting. Equation (5) can only find the optimal weights and thresholds (parameters of the problem) as functions of the given network settings. Therefore, the size of the networks, the number of nodes, the learning rate and the selection of the networks functions (activation and loss) were tuned in advance and reported in Table 2.

Setting	SHL	MLP
Activation function	Sigmoid	Relu
Cost function	Mean Squared error	Mean Squared error
Optimization	Bayesian	Bayesian-Adaptative moment
Layers	1	2
Size	64	32-32
Learning rate	-	Min=1e-7
Number of cycles	-	1000

Table 2: Networks settings.

The main challenge for training SVR models is that Equation (15) must be solved separately for each output feature. Therefore, three different models, one for each component of the structural response, were trained with the settings presented in Table 3. The numerical values reported correspond to the hyperparameter optimization result.

Setting	SVR-σ_y	SVR-σ_z	SVR-σ_ϑ
Box constrained	0.1689	0.5511	1.1781
Epsilon	0.0010	0.0016	0.0013
Kernel Function	Polynomial degree:4 scale:2.1438		
Solver	Sequential Minimal Optimization SMO		

Table 3: SVR settings.

3 RESULTS

For each sample point in the testing set, a prediction was obtained. A comparison plot of the standard deviation of the lateral response component σ_y is shown in Figure 5 where the stars represent the model predictions while the dots show the simulated data. The global performance of the models in terms of their NRMSE are as reported in Table 4. The results indicate that the SVR and MLP models showed better performance than the SHL model.

Error	σ_y	σ_z	σ_ϑ
SVR	4,528e-05	4,101e-05	8,682e-04
SHL	5,686e-05	4,487e-05	2,738e-04
MLP	6,481e-06	1,235e-05	2,035e-04

Table 4: NRMSE.

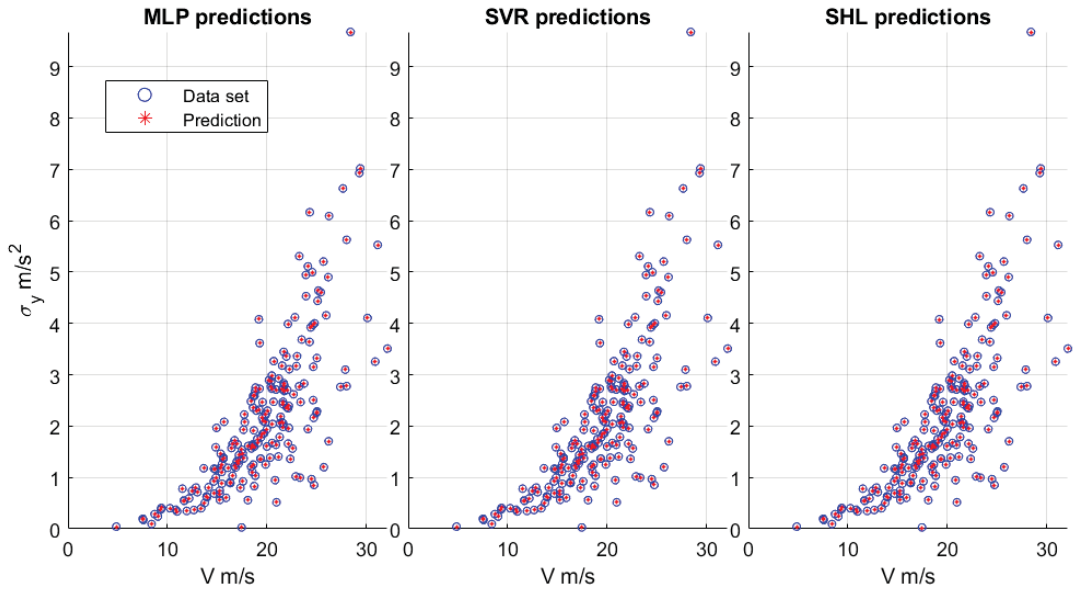

Figure 5: Prediction Scatter plot for σ_y

The performance comparison was extended to the experiment with the noise addition. The dataset creation followed Equation (19). Table 5 reports the Mean Average Percentual Error (MAPE) for the σ_y response component estimated with SVR and MLP models and with noise levels ranging from 2 to 20%. Further, Figure 6 shows a graphical comparison of the target and the estimation of the quantities for noise levels of (a) 2% and (b) 20 %. The MAPE was used for the accuracy metrics because it has comparable units with the added noise level.

Noise	*SVR*	*MLP*
2	1,02	1,12
15	10,95	6,86
20	26,97	6,91

Table 5: σ_y MAPE for different noise levels.

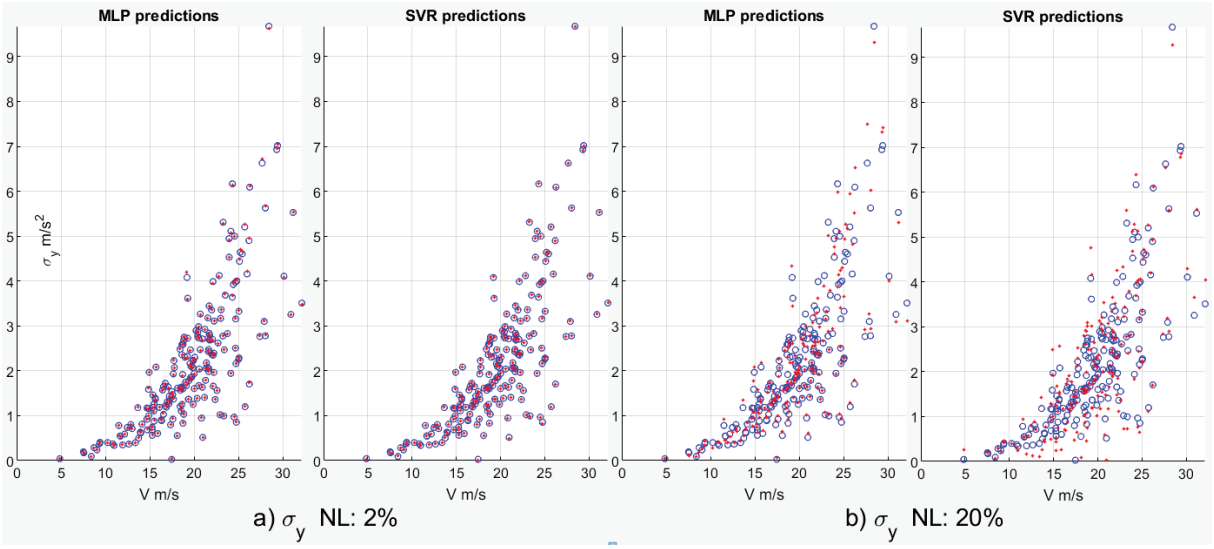

Figure 6: σ_y for: a) NL Noise level 2% b) Noise level 20%

The 27% error reported for a noise level of 20% in the case of the SVR model is particularly high. Nevertheless, after applying the dataset size-increasing strategy (Figure 7) the accuracy improves significantly. Table 6 reports the evolution of the MAPE-σ_y for the different noise levels as the dataset size is increased.

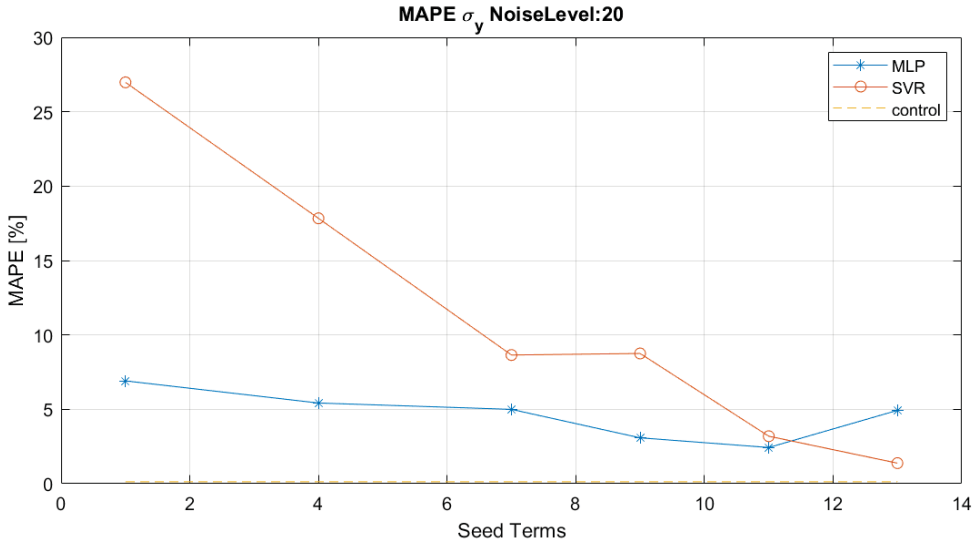

Figure 7: σ_y MAPE evolution with data size increase for 20% noise level

Random Terms	Dataset Size	Noise Level 2%		Noise Level 15%		Noise Level 20%	
		SVR	*MLP*	*SVR*	*MLP*	*SVR*	*MLP*
1	100	1.0246	1.1211	10.9553	6.8677	26.9745	6.9103
4	400	0.7315	1.6877	8.4082	3.5602	17.8352	5.4231
7	700	2.4348	1.7832	3.6389	3.8424	8.6487	4.9967
11	1100	3.2335	0.8224	3.4141	6.2085	3.1915	2.4334
13	1300	0.7284	1.5060	3.4606	9.8955	1.3829	4.9280

Table 6: σ_y MAPE evolution for different Random terms.

4 DISCUSION

The scatter plots from Figure 5 show that the methods can describe the behavior of the structural response in the testing set with reasonable accuracy. This is supported by the NRMSE reports in Table 4. The predictions of the Multilayer perceptron model gave the lowest error for all three response components; however, the other two methods also described the behavior reasonably well.

With the addition of noise to the response, an increase in the MAPE of the predictions were observed. In particular Table 5 reported that for a 20% noise level the error in the SVR model was 27% while that of the MLP model was around 7%. Therefore, the MLP model seems to be more resilient to noise addition than SVR. However, by increasing the size of the dataset the SVR model performance becomes better than the MLP model. According to Table 6, after adding 13 random terms to the dataset with 20% noise level the SVR model showed 25% reduction in the MAPE compared to almost 1% for the MLP. It is also remarkable that for the low noise level i.e. 2%, no appreciable improvement was obtained by increasing the data size. Additionally, the MAPE was almost half of magnitude of the added noise. It is worth noting that the MAPE does not necessarily descreases with increasing data size. For the MLP model an increase in the error was observed for all the noise levels. This behavior implies overtraining phenomena in this type of model.

5 CONCLUSION

The paper showed that it is possible to train machine learning models with the simulated wind induced vibrations of the Hardanger bridge dataset and obtain accurate results. The testing procedure showed a good match between the predictions and the simulations, confirmed by the low NRMSE values. The Multilayer perceptron algorithm implemented in the study gave better results mainly because of the tuning of the hyper parameters. However, to apply the methods to real datasets further analysis must be performed by adding noise to the training samples and checking the accuracy of the predictions.

The MLP model is more resistant to noise for reduced data sizes. Nevertheless, the SVR model improves with increasing data size. Additionally, the MLP model shows a tendency to overtraining as the data size increases. Therefore, extending this experience to the measurement campaigns, it is recommended to use ANN models, specifically MLP, for small datasets with high noise level and SVR models for larger datasets.

REFERENCES

[1] A. Fenerci, O. Øiseth and A. Rønnquist, "Long-term monitoring of wind field characteristics and dynamic response of a long-span suspension bridge in complex terrain," *Engineering Structures,* vol. 147, pp. 269-284, 2017.

[2] A. Fenerci and O. Øiseth, "Site-specific data-driven probabilistic wind field modelling for wind-induced response predeiction of cable-supported bridges," *Journal of Wind Engineering and Industrial Aerodynamics,* vol. 181, pp. 161-179, 2018.

[3] X. Yuwang, O. Øiseth, T. Moan and A. Naess, "Predition of long-term extreme load effects due to wave and wind actions for cable-supported birdges with floating pylons," *Engineering Structures,* pp. 321-333, 2018.

[4] Ø. W. Petersen, O. Øiseth and E.-M. Lourens, "Estimation of the dynamic response of a slender suspension bridge using measured acceleration data," in *X International conference oon Structural Dynamics, EURODYN 2017*, Rome, 2017.

[5] B. Xu, Z. S. Wu and K. Yokoyama, "Neural Networks for decentralized Control fo Cable-Stayed Bridge," *Journal of Bridge Engineering ASCE,* vol. 8, no. 4, pp. 229-236, 2003.

[6] S. Chang, K. Dookie and C. Chang, "Active response control of an offshore strucutre under wave loads using a modified probabilistic neural network," *Journal of Marine Science and Technology Springer,* vol. 14, pp. 240-247, 2009.

[7] X. Yuwang, O. Øiseth and T. Moan, *Artificial Neural Netwroks and support vector machines in the analysis of the lon-term extreme load effects of cable-supported bridges,* Trondheim: Doctoral theses at NTNU 2018:229, 2018.

[8] F. Rosenblatt, "The perceptron: a probabilisitc model for information storage and organization in the brain," *Psychological Review,* vol. 65, no. 6, pp. 386-408, 1958.

[9] C. M. Bishop, Neural Networks for Patter Recognition, New York: Oxford University Press, Inc., 1994.

[10] D. E. Rumelhart, G. E. Hinton and R. J. Williams, "Learning representations by back-propagating errors," *Letters to nature,* vol. 323, no. 9, pp. 533-536, 1986.

[11] A. J. Smola and B. Scholkopf, "A tutorial on support vector regression," NeuroCOLT2 , Berlin, 1998.

[12] V. N. Vapnik, The Nature of Statistical Learning Theory, New York: Springer, 1995.

[13] K. William, *MInima of several variables with inequalities as side conditions,* Chicago: Thesis (S.M.) University of Chicago, 1939.

[14] H. Kuhn and A. Tuker, "Nonlinear Programing," in *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, Berkeley, 1951.

[15] O. Øiseth, A. Ronnquist and R. Sigbjornsson, "Simplified prediction of wind-induced response and stability," *Journal of Wind Engineering,* vol. 98, pp. 730-741, 2010.

[16] J. C. Kaimal, J. C. Wyngaard, Y. Izumi and O. R. Cote, "Spectral characteristics of surface-layer turbulence," *Quaterly Journal of Royal Meteorological Society,* vol. 98, no. 417, pp. 563-589, 1972.