Synthetic Datasets for Numeric Uncertainty Quantification

Hussain Mohammed Dipu Kabir (hussain.kabir@deakin.edu.au)
Deakin University
Moloud Abdar
Deakin University
Abbas Khosravi
Deakin University
Darius Nahavandi
Deakin University
Shady Mohamed
Deakin University
Dipti Srinivasan
National University of Singapore
Saeid Nahavandi
Deakin University
Ponnuthurai Nagaratnam Suganthan
Nanyang Technological University

Research Article

Keywords: synthetic datasets, Numeric Uncertainty Quantification, point prediction, train, validation, test sets, model benchmarking, Equations

Posted Date: September 24th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-909718/v1

Synthetic Datasets for Numeric Uncertainty Quantification

H M Dipu Kabir[1,*], Moloud Abdar[1], Abbas Khosravi[1], Darius Nahavandi[1], Shady Mohamed[1], Dipti Srinivasan[2], Saeid Nahavandi[1], and Ponnuthurai Nagaratnam Suganthan [3]

[1]Institute for Intelligent Systems Research and Innovation (IISRI), Deakin University, Australia.
[2]Department of Electrical & Computer Engineering, National University of Singapore, Singapore.
[3]School of Electrical & Electronic Engineering, Nanyang Technological University, Singapore.
[*]corresponding author(s): H M Dipu Kabir (hussain.kabir@deakin.edu.au)

ABSTRACT

In this paper, we propose ten synthetic datasets for point prediction and numeric uncertainty quantification (UQ). These datasets are split into train, validation, and test sets for model benchmarking. Equations and the description of each dataset are provided in detail. We also present representative shallow neural network (NN) training and Random Vector Functional Link (RVFL) training examples. Both training train models for the point prediction. We perform uncertainty quantification with the consideration of a gaussian and homoscedastic distribution. As the distribution consideration and models are rudimental, much room exists for further explorations and improvements. The dataset and scripts are available at the following link: https://github.com/dipuk0506/UQ-Data

Background & Summary

Datasets with a standard splitting between train and test often become a platform for machine learning (ML) and deep learning (DL) developers to test the effectiveness of novel methods. With the help of such a dataset, researchers can test the effectiveness of their methods with less computation[1]. Such datasets are very common in image vision and natural language processing tasks[2,3]. However, few datasets exist with train and test subsets for applications of machine learning and deep learning methods with UQ in regression[4,5]. UQ is getting vast popularity due to its demand in ML and DL methods[6–9]. In most of the previous studies, researchers in UQ are splitting the dataset randomly[10,11]. The performance of the trained NN varies based on the data splitting[12,13]. Researchers often randomly split data several times and then train NNs for each split and usually get average performance. That process is computationally expensive[14]. Single training on train and validation datasets requires less computation.

Researchers usually test a proposed model on several datasets to show the nature of their model. One image classification method can be good at distinguishing natural images. Another model can be good at distinguishing handwritten digits while another one can perform well on medical images. Similarly, in the regression problem, researchers verify their methods on economic, social, engineering, etc. datasets. Some datasets have higher uncertainty than others[15]. A model can perform outstandingly on a deterministic dataset while another one may perform outstandingly well on a highly uncertain dataset[16,17]. We provide an exact level of uncertainty and relation between input and outputs in a synthetic dataset. Therefore, researchers can observe the performance of their models for different input-output relations and the nature of uncertainties. In this regard, we provide several synthetic datasets for numeric UQ in regression. According to our literature search, this is the first study introducing synthetic datasets for UQ.

Datasets and their Characteristics

Many quantities have partly random and partly deterministic parts[18]. The following equation represents such an uncertain quantity:

$$t_i = y_i + \varepsilon_i, \tag{1}$$

where, t_i is the target, y_i is the true regression mean, and ε_i is the aleatoric uncertainty for the i^{th} sample. $i \in \mathbb{N}$; t, y, $\varepsilon \in \mathbb{R}$. Therefore, we prepare our datasets with deterministic equations and pseudorandom variables.

Datasets with Different Input-Output Relations: Dataset-1 to Dataset-5

Future researchers can investigate the strength of their models in terms of different input-output relations. One model can be good for one input-output relationship, where another model can be good at another input-output relationship. Therefore, we generate the first five datasets with different input-output relations. The following five equations respectively represent the relationship between the input and the output in Dataset-1 to Dataset-5:

$$y = sin(2X1) + M_\varepsilon\{rand[0,1] - 0.5\}, \tag{2}$$

$$y = sin(2X1) + M_\varepsilon\{rand[0,1] sin(2X2 + \frac{\pi}{4})\}, \tag{3}$$

$$y = sin(2X1) + M_\varepsilon\{rand[0,1] sin(2X2 + \frac{\pi}{4})\} + (0) \times X3, \tag{4}$$

$$y = X1(10 - X1) + M_\varepsilon.\{rand[0,1]\}^2 X2(1 - X2)X1(10 - X1) + (0) \times X3, \tag{5}$$

$$y = M_\varepsilon\{rand[0,1]\}^2 [sign\Big(X2 - round(X2)\Big) + 1]/2 + |X1 - round(X1)| + (0) \times X3, \tag{6}$$

where, y in the output, $X1$, $X2$, $X3$ are the first, second, and third inputs, M_ε is the noise magnitude, $rand[0,1]$ is any random value between zero and one inclusive. $|.|$ represents the modulus, $round(.)$ represents the rounding of number, and $sign(.)$ represents the sign of the number. The distribution of the random number generation is uniform over the range [0,1].

Dataset-1 has single variable input and single variable output with homoscedastic uncertainty. The noise magnitude (M_ε) for this dataset is 0.1 representing lower uncertainty. This dataset has symmetric uncertainty, and there exists a trigonometric relationship between input and output.

Dataset-2 has two inputs and one output. The first variable ($X1$) determines the predictive portion. The second variable ($X2$) determines the direction and magnitude of the noise. The uncertainty is heteroscedastic, changes over $X2$. As the random number generator provides uniform distribution over the range, the uncertainty is symmetric. The noise magnitude (M_ε) for this dataset is 0.5 representing medium uncertainty. There exists a trigonometric relationship between inputs and output. Fig. 1a and Fig. 1b, respectively represents the relationship between target (T) and input ($X1$) on Dataset-1, and Dataset-2.

Dataset-3 has three inputs and one output. The third input is multiplied by zero (0) in Eq. (4) to show no relationship between $X3$ and y. The first variable ($X1$) determines the predictive portion. The second variable ($X2$) determines the direction and magnitude of the noise. The third variable is random to output. The uncertainty is heteroscedastic, changes over $X2$. As the random number generator provides uniform distribution over the range, the uncertainty is symmetric. The noise magnitude (M_ε) for this dataset is 0.5 representing medium uncertainty. There exists a trigonometric relationship between inputs ($X1$, and $X2$) and the output. Fig. 2a represents a sine wave with an uncertainty of magnitude (0.5). In Fig. 2b and for $X2$ close to zero, the target range is [-1, $1 + 1/2\sqrt{2}$], as the uncertainty range is ([0, $0.5 \times \sin(\pi/4)$] = [0, $1/2\sqrt{2}$]) for $X2 = 0$ and the deterministic portion have range of [0, 1] based on the value of $X1$. In Fig. 2c, target is random towards the value of $X3$. However, the target density is higher in the range [-1, 1]. Dataset 3 has more samples near the output range [-1, 1], which is the characteristic of the dataset and that has no relation with the value of X3. Even if we put another related or unrelated input on the X-axis of this plot, there will be no target out of [-1.5, 1.5] range.

Dataset-4 has three inputs and one output. The first variable is responsible for both the prediction and uncertainty. The second variable is also partly responsible for the uncertainty. Fig. 3 presents relationships between targets and input variables. A heteroscedastic and asymmetric uncertainty is observable in Fig. 3a. As the second variable is partly responsible for the uncertainty, some minor changes are observable towards the X-axis of Fig. 3b. As $X3$ is random to output, no observable difference is found in the distribution of targets towards the X-axis of Fig. 3c.

Dataset-5 has three inputs and one output. $M_\varepsilon = 4$ for this dataset. The relationship between inputs and the output is the combination of triangular and rectangular signal equations. Fig. 4 presents the distributions of targets (T) for values of ($X1$, $X2$, and $X3$). In Dataset-5, the magnitude of uncertainty becomes zero and four when $\{X2 - round(X2)\}$ is negative and positive, respectively.

Datasets with Different Distributions: Dataset-6 to Dataset-10

Models trained with statistical data often perform poorly over a specific input range. Such phenomena happen mainly due to non-uniformly distributed data over the input range. Non-uniformly distributed data over the training, the validation, and

the test subsets can also bring a poor validation or test performance. We generate the last five datasets and their subsets with different input-output ranges and distributions. The following equation represents the relationship between the input and the output in Dataset-6 to Dataset-10:

$$y = sin(8X1) + M_\varepsilon\{rand[0,1] - 0.5\} \tag{7}$$

where, all notations contain the same definition as Eq. (2). Dataset-6 to Dataset-10 have single variable input, homoscedastic and symmetric uncertainty, and the trigonometric relationship between the input and the output.

In Dataset-6, the sample density decreases with the increment of $X1$. Fig. 5a presents the training set of Dataset-6. Validation and test subsets have a uniform density over the input range. The noise magnitude (M_ε) is 0.5.

In Dataset-7, the sample density increases with the increment of the absolute value of $X1$. Fig. 5b presents the training set of Dataset-7. Validation and test subsets have a uniform density over the input range. The noise magnitude (M_ε) is 0.5.

The artificial NN may confuse the noise with a high-frequency signal, where human brains can easily recognize the pattern of the signal. The equation for the generation of Dataset-8 is the same as Eq. (7). The noise magnitude (M_ε) in Dataset-8 is 0.2. However, the input range of Dataset-8 is larger than Dataset-6. Fig. 5c indicates the training set of Dataset-8. All subsets of Dataset-8 have a uniform density over the input range.

The noise magnitude (M_ε) in Dataset-9 is 0.2. Fig. 6a, 6b, and 6c presents the training, validation, and test sets of Dataset-9. The training set has no sample near $X1 = 0$.

The noise magnitude (M_ε) in Dataset-10 is 0.2. Fig. 7a, 7b, and 7c presents the training, validation, and test sets of Dataset-10. The test set has a slightly higher range of $X1$ in training and validation sets.

Common Characteristics

Dataset-1 to Dataset-8 have five thousand samples for training, two thousand samples for validation, and two thousand samples for the test sets. Dataset-9 and Dataset-10 have two hundred samples each for train, test, and validation subsets.

Besides all patterns mentioned in dataset sections, the range of inputs and outputs in some datasets is much higher than unity. Algorithm developers need to ensure that their algorithm works properly for different ranges of inputs and outputs.

Performance Parameters

The purpose of developing this dataset is to help regression and uncertainty quantification algorithm researchers. Researchers consider several performance criteria to judge the quality of the model. Popular performance criteria of regression and uncertainty quantification are as follows:

Regression

Common performance parameters in regression are the mean-square error(MSE), the root-mean-square error (RMSE), mean absolute error (MAE), mean-absolute-percentage error (MAPE), sum-square-error (SSE), etc. MSE and RMSE are more popular than other parameters. In the current study, we consider MSE as a convenient performance criterion, as it is commonly used in optimizations. Moreover, as RMSE is the square root of MSE, researchers can easily find RMSE values from the MSE values. The equation of MSE is as follows:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{8}$$

where N is the number of samples, y_i is the target, and $\hat{y}_i$ is the predicted value for i^{th} sample.

Uncertainty Quantification

The probability density or cumulative probability density function can represent the exact uncertain situation. However, it is not possible to express the probability density or cumulative probability density function in a few words or numbers. Therefore, prediction interval (PI) with certain coverage probability is becoming a popular technique to represent the level of uncertainty[19,20].

PIs are formulated with an expected coverage probability. A PI may fail to cover one target due to aleatoric or epistemic uncertainty. No matter, how well the NN training is, PIs fail to cover a certain percentage of targets. Therefore, performance on a single sample cannot represent the quality PI generation algorithm. The statistical performance of PI depends on several factors. Khosravi et al. consider the PI coverage probability (PICP) and PI normalized average width (PINAW) as important determinants for the quality of PI[21]. Definitions of PICP and PINAW are as follows:

$$PICP = \frac{1}{N}\sum_{j=1}^{N} c_j$$

provided that,

$$c_j = \begin{cases} 1, t_j \in [\underline{y}_j, \bar{y}_j] \\ 0, t_j \notin [\underline{y}_j, \bar{y}_j] \end{cases} \tag{9}$$

$$PINAW = \frac{1}{NR} \sum_{j=1}^{N} (\bar{y}_j - \underline{y}_j) \tag{10}$$

where $\underline{y}_j$, $\bar{y}_j$, and t_j are lower bound, upper bound, and target for j^{th} sample respectively. N, and R are the total number of samples and the range of targets respectively, the upper bound is always greater than the lower bound ($\bar{y}_j > \underline{y}$). L. G. Marin et al.[22] consider, PICP, PINAW, and the deviation from the mid interval as important quality criteria of PI. They denote the deviation from the mid interval with $||e||$ notation. The definition $||e||$ can be written as follows:

$$||e|| = \sqrt{\frac{1}{N} \sum_{j=1}^{N} \left| t_j - \frac{\bar{y}_j + \underline{y}_j}{2} \right|^2} \tag{11}$$

where we divide the sum square by N to avoid a large $||e||$ for large N. Also, in their paper, they take deviation from the point prediction and they consider a Gaussian and symmetric uncertainty. Mid-interval and the point predictions are the same for Gaussian and symmetric uncertainty. All notations in Eq. (11) have the same meaning as Eq. (9). According to Eq. (11), the value of $||e||$ is equal to the root mean square deviation between target and the mid interval. Kabir et al.[23] consider PICP, PINAW, and the PI normalized average failure distance (PINAFD) as important quality criteria of PI. The definition of PINAFD is as follows:

$$PINAFD = \frac{\sum_{j=1}^{N} (1 - c_j) \times min(|t_j - \bar{y}_j|, |\underline{y}_j - t_j|)}{R \times \left\{ \sum_{j=1}^{N} (1 - c_j) \right\} + \varepsilon_c} \tag{12}$$

where, the upper bound is always greater than the lower bound ($\bar{y}_j > \underline{y}$), c_j contains the same meaning as of Eq. (9). When $(1 - c_j) = 1$, the PI does not cover the target. At that situation, $min(|t_j - \bar{y}_j|, |\underline{y}_j - t_j|)$ results in the minimum distance of bounds to the target. In other words, the gap between the nearest bound to the target. Total failure distances are normalized by the range R and the total number of PI non coverage($\sum_{j=1}^{n} (1 - c_j)$). We add a small constant number ($\varepsilon_c = 10^{-10}$) with the denominator. It helps the computation machine to avoid divide by zero error when all targets are covered.

Researchers have developed many cost function based on different combinations of these criteria[21–25]. Different cost function works well in different datasets. Therefore, we compute and present these four basic quality criteria. Future researchers can easily compute cost functions from basic criteria.

Data Records

We generate ten synthetic datasets with the help of the following python script in the Kaggle: https://www.kaggle.com/dipuk0506/toy-dataset-for-regression-and-uq. This notebook also presents the characteristics of each dataset. The *Datasets and their Characteristics* section of this paper explains all ten generated datasets and their characteristics. We also upload datasets to GitHub at the following link: https://github.com/dipuk0506/UQ-Data We also upload datasets to Figshare at the following link: https://figshare.com/articles/dataset/Synthetic_Datasets_for_Numeric_Uncertainty_Quantification/16528650

Technical Validation: Model Training for Initial Performance

An initial result on a new dataset encourages many algorithm developers to develop a better algorithm and report the result[26]. Moreover, the publicly available code of the initial algorithm helps many researchers in understanding the dataset. Therefore, we train a shallow NNs and RVFL networks[27] on the proposed datasets, provide training details with publicly available codes (Links https://www.kaggle.com/dipuk0506/shallow-nn-on-toy-datasets, https://www.kaggle.com/dipuk0506/rvfl-on-synthetic-dataset).

We train shallow NNs and RVFL networks on the train Dataset and validate them in a two hundred epoch interval. Before training these models, we normalize the dataset. According to our observation, the dataset normalization statistically brings

uniform performances over the input range. Fig. 8 visualizes a representative performance of our trained NN models. Fig. 9 visualizes a representative performance of our trained RVFL models. We have drawn similar plots for all datasets in Kaggle notebooks. As all plots are similar, the paper does not present other plots. We present details of NNs and performances in each dataset in Table 1. We present RVFL networks and their performance in Table 2. We train networks for point prediction. Then, we compute the upper and lower bounds with a Gaussian and homoscedastic uncertainty assumption[9,28]. In the Gaussian and homoscedastic uncertainty assumption, the PI is presented as $[\mu - z\sigma, \mu + z\sigma]$. In this representation, μ is the mean or the mean of expected outcomes, σ is the variance, and z=1.96 for 95% PICP.

Point prediction and overall standard deviation values compute uncertainty bounds for 95% coverage probability. Some of our proposed datasets have heteroscedasticity and asymmetry. We consider homoscedastic and symmetric uncertainty. Moreover, we apply simple, and lightweight networks. There exist ample opportunities to improve results with the consideration of heteroscedasticity, asymmetry, deeper NN, etc. Moreover, the developer of novel models may consider these initial results for comparison.

Usage Notes

In this study, we propose ten different synthetic datasets for numeric uncertainty quantification. One can investigate the strength of the model in terms of different input-output relations. Through the release of synthetic datasets, we aim encouraging research groups within the broader machine learning and deep learning community to investigate and quantify uncertainty of their methods using these synthetic datasets and thereafter apply their models on various real datasets.

We have sample python scripts showing how to use data in Kaggle. Also, we save the data in both 'pkl' and 'csv' formats. The user of dataset can load 'csv' files in other programming languages while training models in another language.

Code Availability

The following Kaggle notebook generates datasets along with detailed relationships between inputs and the output:
https://www.kaggle.com/dipuk0506/toy-dataset-for-regression-and-uq

The following Kaggle notebook presents example shallow NN training on datasets:
https://www.kaggle.com/dipuk0506/shallow-nn-on-toy-datasets
Version-N of the notebook applies a shallow NN to Data-N.

The following Kaggle notebook presents example RVFL network training on datasets:
https://www.kaggle.com/dipuk0506/rvfl-on-synthetic-dataset
Version-N of the notebook applies an RVFL network to Data-N.

We also upload datasets and example scripts at the following GitHub repository:
https://github.com/dipuk0506/UQ-Data.

Conclusion

This paper has proposed ten uncertainty and regression related datasets with details of their construction and example performance evaluations. Presented datasets and methodologies may help future researchers developing novel machine learning and deep learning algorithms. The paper has also presented examples of model training with publicly available codes. We also provided scripts of data generation, plotting, and the training of models on the datasets to assist future users of the dataset.

References

1. Coates, A., Ng, A. & Lee, H. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, 215–223 (2011).
2. Deng, J. *et al.* Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255 (IEEE, 2009).
3. Hudson, D. A. & Manning, C. D. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6700–6709 (2019).
4. Lu, S., Li, Z., Qin, Z., Yang, X. & Goh, R. S. M. A hybrid regression technique for house prices prediction. In *2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 319–323 (IEEE, 2017).

5. Morala, P., Cifuentes, J. A., Lillo, R. E. & Ucar, I. Towards a mathematical framework to inform neural network modelling via polynomial regression. *Neural Networks* **142**, 57–72 (2021).

6. Kabir, H. D. *et al.* Uncertainty-aware decisions in cloud computing: Foundations and future directions. *ACM Comput. Surv. (CSUR)* **54**, 1–30 (2021).

7. Gal, Y. & Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, 1050–1059 (PMLR, 2016).

8. Kendall, A. & Cipolla, R. Modelling uncertainty in deep learning for camera relocalization. In *2016 IEEE international conference on Robotics and Automation (ICRA)*, 4762–4769 (IEEE, 2016).

9. Abdar, M. *et al.* A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Inf. Fusion* https://doi.org/10.1016/j.inffus.2021.05.008 (2021).

10. Abdar, M. *et al.* Uncertainty quantification in skin cancer classification using three-way decision-based bayesian deep learning. *Comput. biology medicine* 104418 (2021).

11. Xu, Z. & Saleh, J. H. Machine learning for reliability engineering and safety applications: Review of current status and future opportunities. *Reliab. Eng. & Syst. Saf.* 107530 (2021).

12. Posch, K. & Pilz, J. Correlated parameters to accurately measure uncertainty in deep neural networks. *IEEE transactions on neural networks learning systems* **32**, 1037–1051 (2020).

13. Qazani, M. R. C. *et al.* Adaptive motion cueing algorithm based on fuzzy logic using online dexterity and direction monitoring. *IEEE Syst. J.* https://doi.org/10.1109/JSYST.2021.3059285 (2021).

14. Dong, H., Gong, Q. & Zhu, M. Optimal energy management of automated grids considering the social and technical objectives with electric vehicles. *Int. J. Electr. Power & Energy Syst.* **130**, 106910 (2021).

15. Chen, X. *et al.* A stochastic sensitivity-based multi-objective optimization method for short-term wind speed interval prediction. *Int. J. Mach. Learn. Cybern.* 1–12 (2021).

16. Guo, R., Nelson, S. & Menon, R. Needle-based deep-neural-network camera. *Appl. Opt.* **60**, B135–B140 (2021).

17. Abbas, Z., Tayara, H. & to Chong, K. Spinenet-6ma: A novel deep learning tool for predicting dna n6-methyladenine sites in genomes. *IEEE Access* **8**, 201450–201457 (2020).

18. Kabir, H. D., Khosravi, A., Hosen, M. A. & Nahavandi, S. Neural network-based uncertainty quantification: A survey of methodologies and applications. *IEEE Access* 36218–36234 (2018).

19. Shamsi, A. *et al.* An uncertainty-aware transfer learning-based framework for covid-19 diagnosis. *IEEE Transactions on Neural Networks Learn. Syst.* https://doi.org/10.1109/TNNLS.2021.3054306 (2021).

20. Cartagena, O., Parra, S., Muñoz-Carpintero, D., Marín, L. G. & Sáez, D. Review on fuzzy and neural prediction interval modelling for nonlinear dynamical systems. *IEEE Access* **9**, 23357–23384 (2021).

21. Khosravi, A., Nahavandi, S., Creighton, D. & Atiya, A. F. Lower upper bound estimation method for construction of neural network-based prediction intervals. *IEEE transactions on neural networks* **22**, 337–346 (2010).

22. Marín, L. G., Valencia, F. & Sáez, D. Prediction interval based on type-2 fuzzy systems for wind power generation and loads in microgrid control design. In *Fuzzy Systems (FUZZ-IEEE), 2016 IEEE International Conference on*, 328–335 (IEEE, 2016).

23. Kabir, H. D., Khosravi, A., Kavousi-Fard, A., Nahavandi, S. & Srinivasan, D. Optimal uncertainty-guided neural network training. *Appl. Soft Comput.* **99**, 106878 (2021).

24. Zhang, G. *et al.* An advanced approach for construction of optimal wind power prediction intervals. *IEEE transactions on power systems* **30**, 2706–2715 (2014).

25. Cruz, N., Marín, L. G. & Sáez, D. Prediction intervals with lstm networks trained by joint supervision. In *2019 International Joint Conference on Neural Networks (IJCNN)*, 1–8 (IEEE, 2019).

26. Cohen, G., Afshar, S., Tapson, J. & Van Schaik, A. Emnist: Extending mnist to handwritten letters. In *2017 International Joint Conference on Neural Networks (IJCNN)*, 2921–2926 (IEEE, 2017).

27. Shi, Q., Katuwal, R., Suganthan, P. & Tanveer, M. Random vector functional link neural network based ensemble deep learning. *Pattern Recognit.* **117**, 107978 (2021).

28. Davidian, M. & Carroll, R. J. Variance function estimation. *J. Am. statistical association* **82**, 1079–1091 (1987).

Acknowledgements

This research was funded by the Australian Research Council through Discovery Projects funding scheme (DP190102181).

Author contributions statement

H M Dipu Kabir conceived the experiment(s). H M Dipu Kabir conducted the experiment(s). H M Dipu Kabir analyzed results. H M Dipu Kabir and Moloud Abdar wrote the paper. Ponnuthurai Nagaratnam Suganthan, Abbas Khosravi supervised the process. Abbas Khosravi, Darius Nahavandi, Shady Mohamed, Ponnuthurai Nagaratnam Suganthan, Dipti Srinivasan, and Saeid Nahavandi reviewed the manuscript.

Competing Interests

The authors declare no competing interests.

Figures & Tables

Data	Size of NN		MSE	PICP (%)	PINAW	$\|\|e\|\|$	PINAFD
	No. of layer	Neurons per layer					
Data-1	2	300	1.039e-3	96.85	5.984e-2	3.223e-2	2.964e-3
Data-2	2	500	1.072e-2	94.65	1.433e-1	1.035e-1	3.018e-3
Data-3	2	500	1.785e-2	95.05	1.766e-1	1.336e-1	4.851e-3
Data-4	2	500	9.368e0	93.95	2.752e-1	3.060e0	1.384e-2
Data-5	2	500	1.030e0	92.75	8.972e-1	1.015e0	1.181e-1
Data-6	2	500	2.164e-2	99.95	2.307e-1	1.471e-1	0
Data-7	2	700	2.816e-2	94.95	2.436e-1	1.678e-1	7.563e-3
Data-8	2	1200	4.656e-1	100	1.219e0	6.823e-1	0
Data-9	2	1000	2.803e-2	96.00	3.037e-1	1.674e-1	1.045e-3
Data-10	2	400	1.912e-2	86.50	1.608e-1	1.383e-1	2.950e-2

Table 1. Performance of trained shallow neural networks (NN) models. Models are trained on the training subset, validated in the validation dataset in each 200 epoch, and then tested on the test dataset.

Data	Neurons in RVFL	MSE	PICP (%)	PINAW	$\|\|e\|\|$	PINAFD
Data-1	1000	1.255e-3	95.85	6.633e-2	3.543e-2	9.827e-4
Data-2	1000	2.094e-2	93.50	2.009e-1	1.447e-1	2.009e-3
Data-3	1000	2.19e-1	97.10	6.187e-1	4.680e-1	2.462e-2
Data-4	1000	1.418e1	88.95	2.731e-1	3.766e0	2.006e-2
Data-5	2000	1.114e0	92.65	9.316e-1	1.055e0	1.235e-1
Data-6	1000	2.159e-2	100	2.305e-1	1.469e-1	0
Data-7	1000	3.416e-2	95.45	2.930-1	1.848e-1	2.475e-2
Data-8	1000	4.478e-1	100	1.196e0	6.692e-1	0
Data-9	1000	1.943e-2	92.00	2.529e-1	1.393e-1	3.832e-2
Data-10	1000	4.485e-3	96.50	1.211e-1	6.197e-2	3.537e-3

Table 2. Performance of trained shallow Random Vector Functional Link (RVFL) models. Models are trained on the training subset, validated in the validation dataset in each 200 epoch, and then tested on the test dataset.

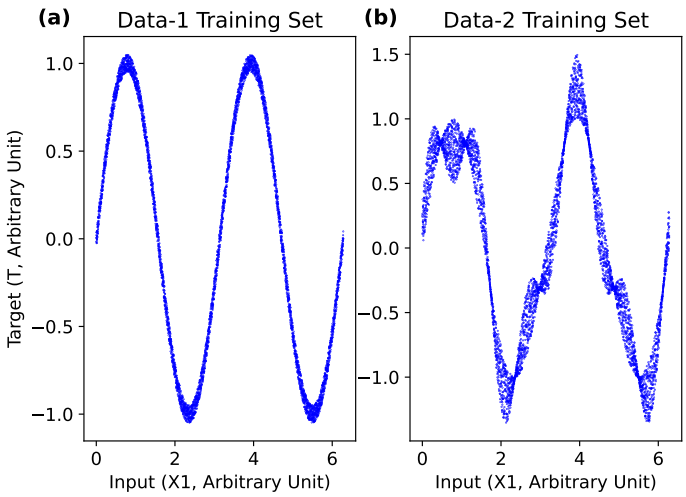

Figure 1. Relationship between target (T) and input ($X1$) on (a) Dataset-1, and (b) Dataset-2.

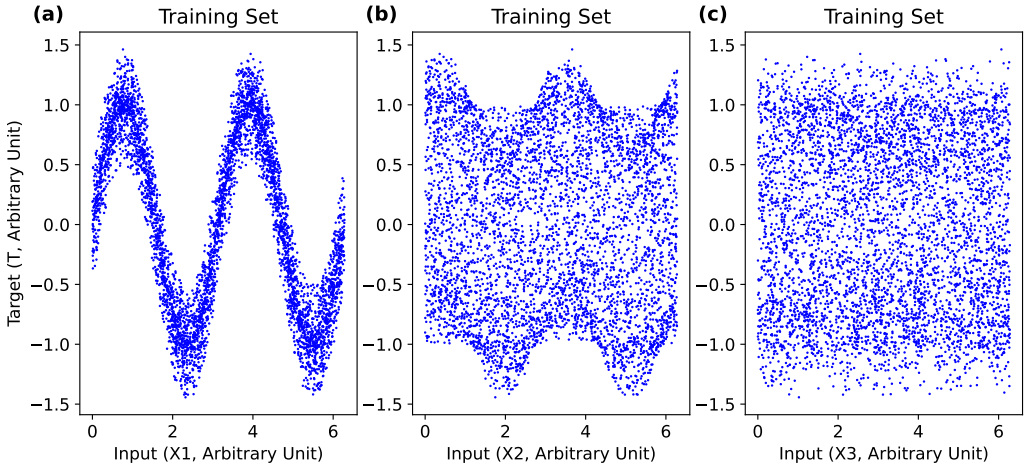

Figure 2. Relationship between target (T) and inputs (X1, X2, and X3) on Dataset-3. The first input (X1) highly influences the point prediction. The second input (X2) highly influences the level of uncertainty. The target is not related to the third input (X3).

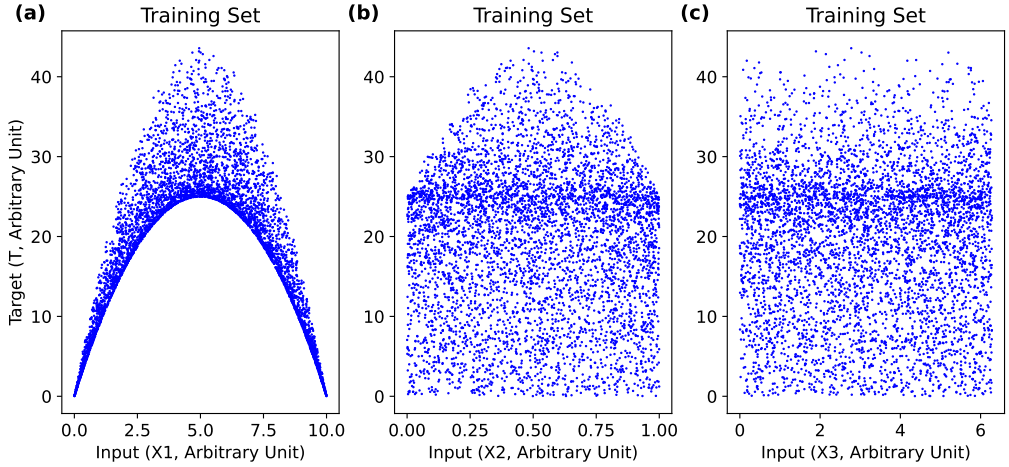

Figure 3. Relationship between target (T) and inputs (X1, X2, and X3) on Dataset-4. The first input (X1) highly influences the point prediction. The second input (X2) highly influences the level of uncertainty. The target is not related to the third input (X3).

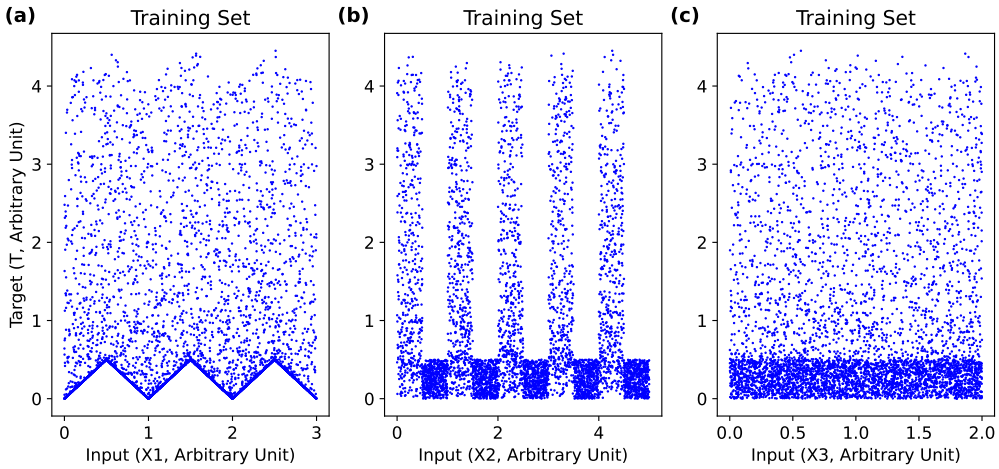

Figure 4. Relationship between target (T) and inputs (X1, X2, and X3) on Dataset-5. The first input (X1) highly influences the point prediction. The second input (X2) highly influences the level of uncertainty. The target is not related to the third input (X3).

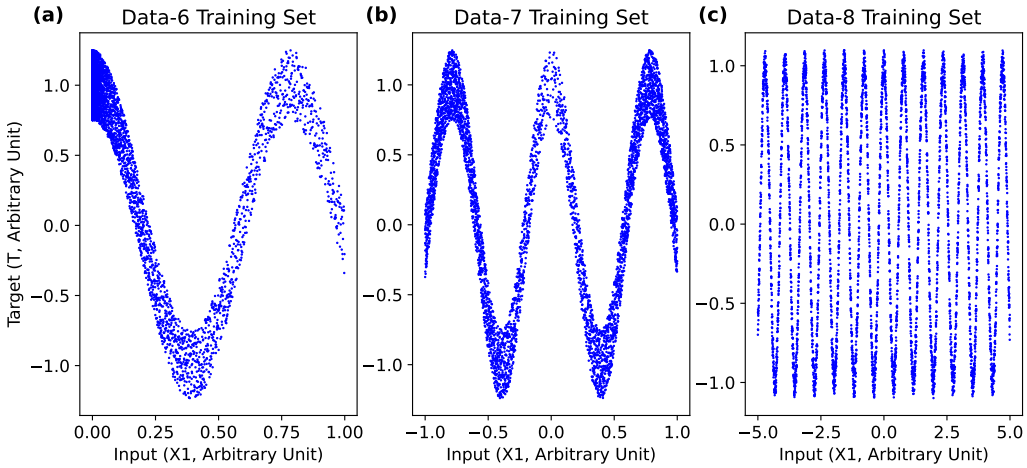

Figure 5. Relationship between target (T) and input (X1) on (a) Dataset-6, (a) Dataset-7, and (c) Dataset-8.

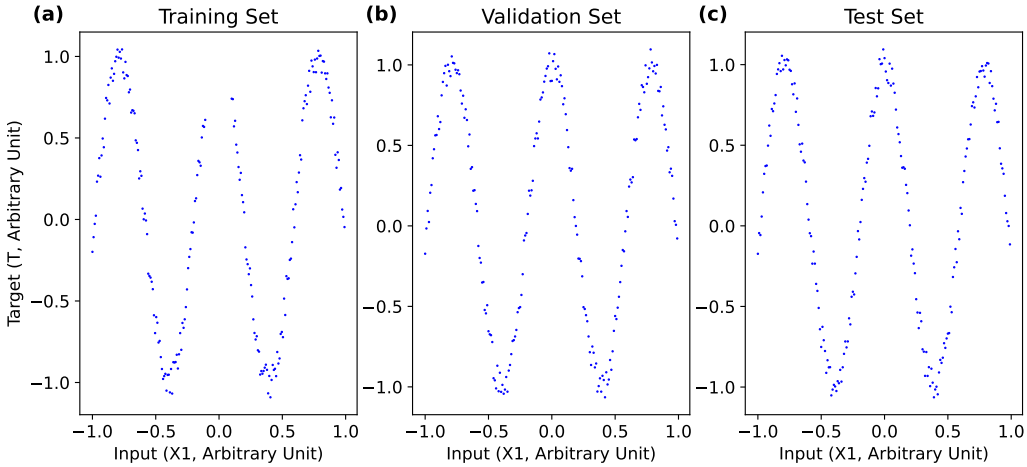

Figure 6. Relationship between the target (T), and input (X1) on Dataset-9. Subplots (a), (b), and (c) represent relationships on the training, the validation, and the test set, respectively.

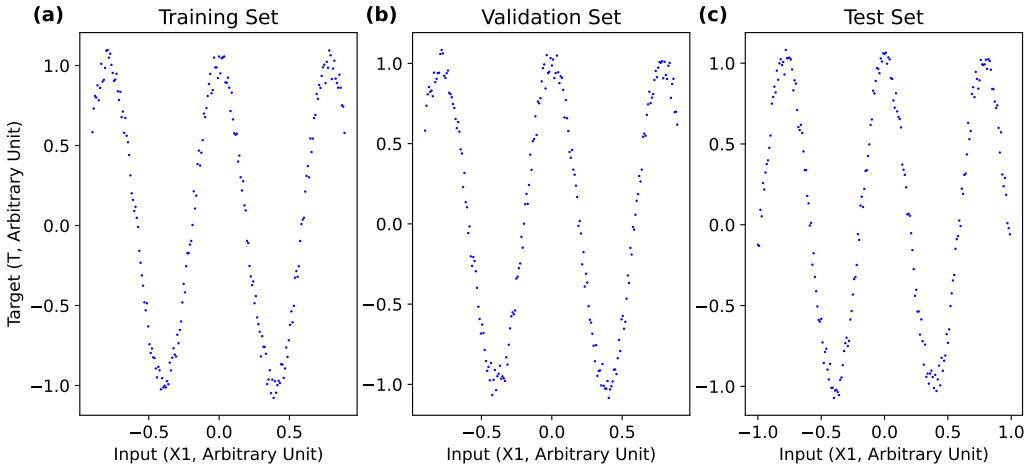

Figure 7. Relationship between the target (T), and input (X1) on Dataset-10. Subplots (a), (b), and (c) represent relationships on the training, the validation, and the test set, respectively.

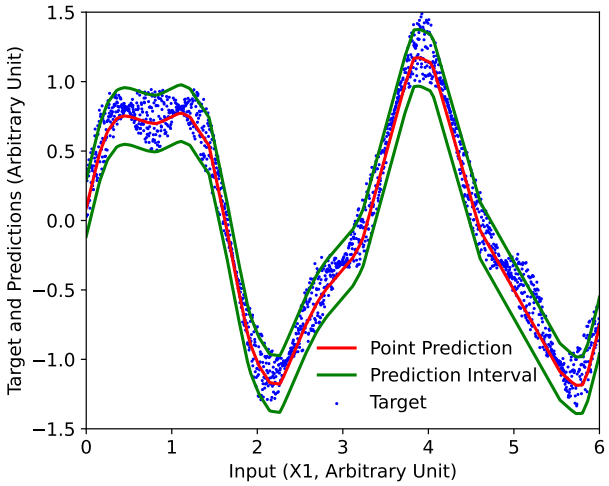

Figure 8. Targets, point predictions, and prediction intervals on the test subset of Dataset-2 and for the shallow NN.

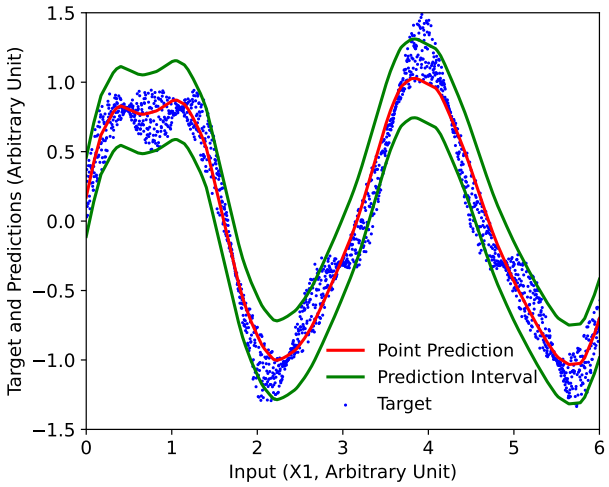

Figure 9. Targets, point predictions, and prediction intervals on the test subset of Dataset-2 and for the RVFL network.