Online Supplement for "A Benchmark-Based Method for Evaluating Hyperparameter Optimization Techniques of Neural Networks for Surface Water Quality Prediction"

Xuan Wang[1, 2, 3], Yan Dong[1, 2], Jing Yang[1, 2, 3], Zhipeng Liu[1, 2], Jinsuo Lu[1, 2, 3] *

[1] School of Environmental and Municipal Engineering, Xi'an University of Architecture and Technology, Xi'an 710055, China.

[2] Shaanxi Key Laboratory of Environmental Engineering, Xi'an University of Architecture and Technology, Xi'an 710055, China.

[3] State Key Laboratory of Green Building in West China, Xi'an University of Architecture and Technology, Xi'an 710055, China.

*Corresponding author(s): Tel: +86 13991126672

E-mail address: lujinsuo@xauat.edu.cn

A Details of the Data Sources

A summary of the data sources we use in our study as well as the time series for all variables are shown in the following.

Table S1 Data sources of historical observations of different measurement variables. CNEMC= China National Environmental Monitoring Centre; CNMIC= China National Meteorological Information Center.

	Type	Measurement variable	Unit	Note	Temporal frequency	Data source
1	Water quality	pH	-	Potential of Hydrogen	real-time data every 4 hours	CNEMC

2	data	DO	mg/L	Dissolved Oxygen	real-time data every 4 hours	CNEMC
3		TOC	mg/L	Total Organic Carbon	real-time data every 4 hours	CNEMC
4	Air quality data	AQI	-	Air Quality Index	hourly real-time data	CNEMC
5		CO	mg/m^3	Carbon monoxide concentration	hourly real-time data	CNEMC
6		CO_24h			24-hour moving average	CNEMC
7		NO2	$\mu g/m^3$	Nitrogen dioxide concentration	hourly real-time data	CNEMC
8		NO2_24h			24-hour moving average	CNEMC
9		O3	$\mu g/m^3$	Ozone concentration	hourly real-time data	CNEMC
10		O3_8h			8-hour moving average	CNEMC
11		PM10	$\mu g/m^3$	Particulate matter 10 concentration	hourly real-time data	CNEMC
12		PM10_24h			24-hour moving average	CNEMC
13		PM2.5	$\mu g/m^3$	Particulate matter 2.5 concentration	hourly real-time data	CNEMC
14		PM2.5_24h			24-hour moving average	CNEMC
15		SO2	$\mu g/m^3$	Sulfur dioxide concentration	hourly real-time data	CNEMC
16		SO2_24h			24-hour moving average	CNEMC
17	Meteorological data	EVP	mm	Large evaporation	daily average	CNMIC
18		GST_AVE	℃	Surface air temperature	daily average	CNMIC
19		PRE	mm	Precipitation	daily accumulated value	CNMIC
20		PRS_AVE	hPa	Atmospheric pressure	daily average	CNMIC
21		RHU_AVE	%	Relative humidity	daily average	CNMIC
22		SSD	h	Sunshine duration	daily accumulated value	CNMIC
23		TEM_AVE	℃	Air temperature	daily average	CNMIC
24		WIN_AVE	m/s	Wind speed	daily average	CNMIC

B Brief Definition of NN Development and Hyperparameters

Here is a brief review of the water resources NN models' development process from the perspective of hyperparameters, including input and output selection, model structure selection, model training and validation. For more details regarding the specific methods, please refer to the literature (Wu et al., 2014).

(1) Input and Output Variables

Generally, the variable(s) to be predicted, such as precipitation, water level, water quality, etc., is selected as output(s), which is the target of the model. The input(s) selection is carried out based on a prior knowledge and the availability of data, utilizing linear or non-linear, model-based or model-free techniques (Bowden et al., 2005a, 2005b; Castelletti et al., 2012). Redundant inputs often lead to overfitting and local minimum, while ignoring essential input variables will directly impair the performance

of the model.

(2) Hyperparameters of Model Structure

The NN model consists of an input layer, hidden layer(s) and an output layer. The layers are connected by weights and bias. After the input and output selection process, the input and output layers are determined accordingly. Then the number of hidden layers and neurons are to be selected, which determine the depth and width of the network, also the size of the hypothesis space of the model. The essence of model structure selection is to find a balance between generalization capability and complexity.

Then a specific activation function is to be added to each neuron to introduce non-linear informational transformation to the NNs. The most commonly used activation functions are Sigmoid, Tanh and ReLU.

(3) Hyperparameters of Model Training

Gradient descent algorithms are the most commonly used training methods, including batch-GD (batch Gradient Descent), SGD (Stochastic Gradient Descent), Momentum, Adam (Adaptive Moment Estimation), etc., classified by the number of samples used by each epoch to update parameters and the learning rate decay strategy. Our previous experiments showed the performance of Adam algorithm was significantly better than other algorithms, so it was used as the only training algorithm in this study. Adam is an algorithm for first-order gradient-based optimization of stochastic objective functions, based on adaptive estimates of lower-order moments (Kingma and Ba, 2014). The training process is as follows:

$$v_t = \beta_1 v_{t-1} + (1-\beta_1)\cdot \nabla_\theta J(\theta) \quad (1)$$

$$s_t = \beta_2 s_{t-1} + (1-\beta_2)\cdot (\nabla_\theta J(\theta))^2 \quad (2)$$

Where *J(θ)* is the loss function of the network, and MSE (mean of square error) is commonly used (Eq. (6)).

$\boldsymbol{v_t}$ and $\boldsymbol{s_t}$ are biased estimates of the first-order and second-order moments of the gradient, which need to be corrected:

$$\hat{v}_t = \frac{v_t}{1-\beta_1^t} \quad (3)$$

$$\$_t = \frac{s_t}{1-\beta_2^t} \quad (4)$$

Parameters are updated as follows:

$$\theta = \theta - \frac{\eta}{\sqrt{\hat{s}_t + \int}} \cdot \$_t \quad (5)$$

Where the values for β_1, β_2 and $\in$ were set to 0.9, 0.998 and $1e^{-8}$ respectively via trial and error in the case study. As an adaptive learning rate decay algorithm, the initial learning rate value needs to be set as a hyperparameter.

Then we adopted a mini-batch gradient descent strategy, which has the advantages of batch-GD and SDG, converging faster than batch-GD and more stable than SGD. The hyperparameter "batch size" here represents the number of samples used by each epoch to update parameters.

(4) Model Validation

The purpose of model validation is to ensure the validity of the model. Complete model validation should contain replicative validation, predictive validation and structural validation (Humphrey et al., 2017). Among them, replicative validation ensures the model has captured the underlying relationship of the training data. Then predictive validation is needed to ensure the trained model's generalization capability and to avoid overfitting.

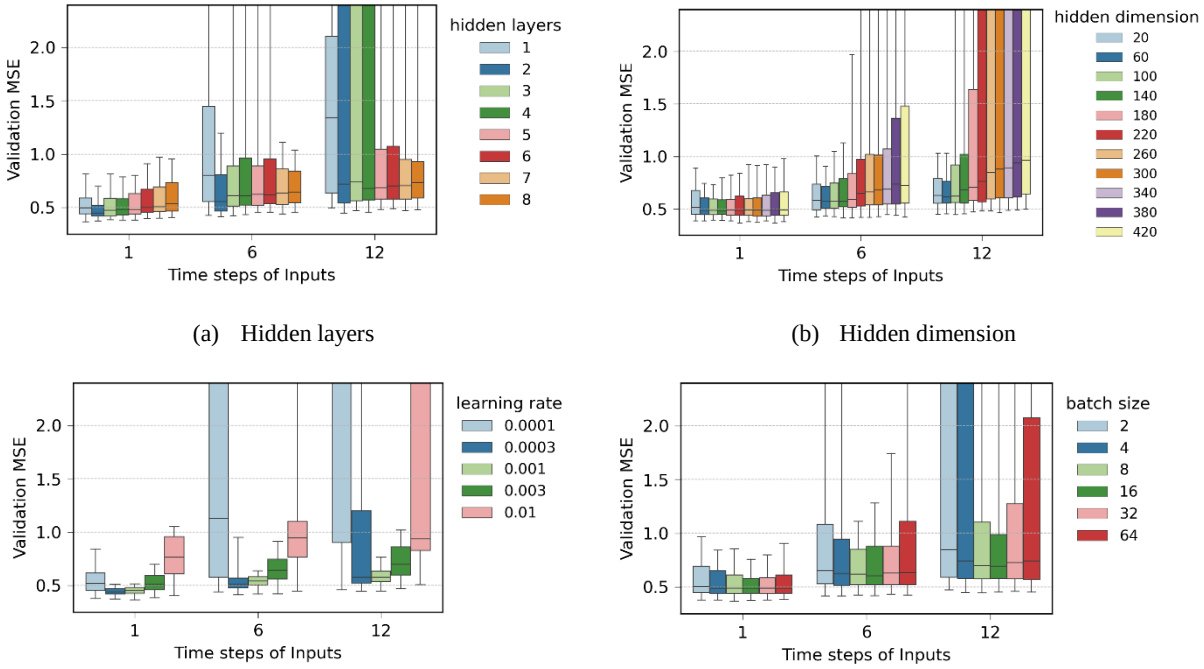
C Additional Results of Numerical Experiments

C1 Results of the benchmark method, GS

Table S2 The benchmark hyperparameter value sets for the four study sites

Site	Input steps	Hidden layers	Hidden dimension	Learning rate	Batch size
Anqing	1	{2, 3}	[220, 420]	[0.0003, 0.001]	{8, 16, 32}
	6	2	[20, 100]	[0.0003, 0.001]	{4, 16, 32}
	12	2	[20, 140]	[0.0003, 0.001]	{16, 32}
Yichang	1	{1, 2}	[100, 420]	[0.0001, 0.001]	{4, 8, 16, 32}
	6	{2, 4, 5}	[20, 300]	[0.0001, 0.001]	{8, 16, 32, 64}
	12	{4, 5, 6}	[20, 220]	[0.0003, 0.001]	{8, 16, 32, 64}
Nanning	1	{1, 2}	[60, 180]	[0.0001, 0.001]	{2, 4, 8}
	6	2	[60, 220]	[0.0001, 0.001]	{2, 4, 16, 32}
	12	2	[60, 220]	[0.0001, 0.001]	{16, 32}
Jiujiang	1	{1, 2}	[60, 420]	[0.0001, 0.001]	{2, 4, 8, 16}
	6	{1, 2}	[20, 220]	[0.0003, 0.001]	{4, 8, 16, 32}

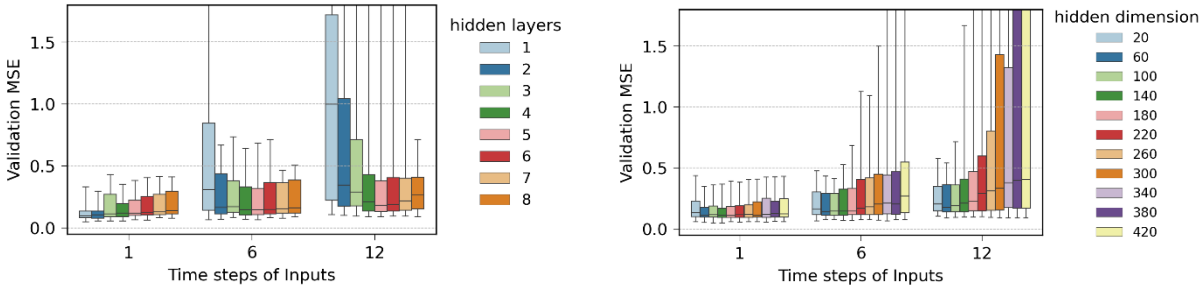
	12	{1, 2}	[20, 140]	[0.0003, 0.001]	{4, 8, 16, 32, 64}
	1	1	[100, 420]	[0.0003, 0.001]	{2, 4, 8, 16}
Luzhou	6	{1, 2}	[20, 140]	[0.0003, 0.001]	{4, 8, 16}
	12	{1, 2}	[20, 220]	[0.0003, 0.001]	{4, 8, 16}



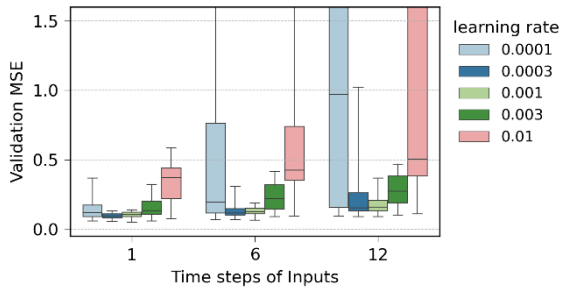
(a) Hidden layers (b) Hidden dimension

(c) Learning rate (d) Batch size

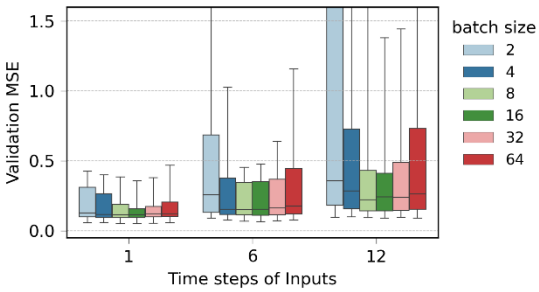
Fig. S1 Boxplot of the NNs' validation MSE with different hyperparameter values, Site Anqing (Caption: lower and upper limit of the whisker refer to 5%-95% of the data)



(a) Hidden layers (b) Hidden dimension

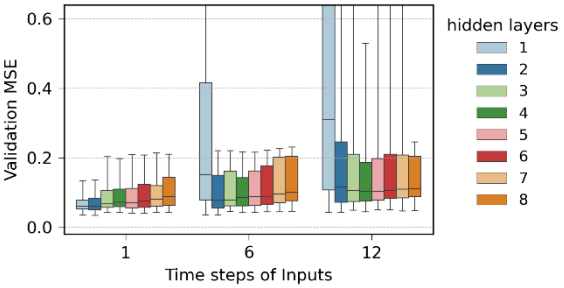


(c) Learning rate

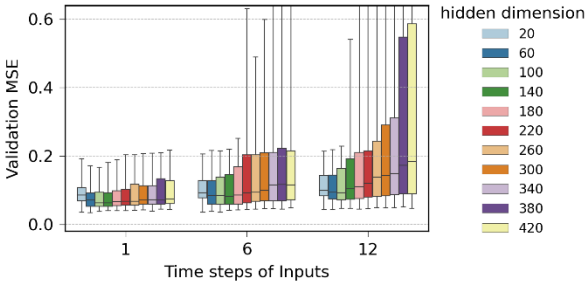


(d) Batch size

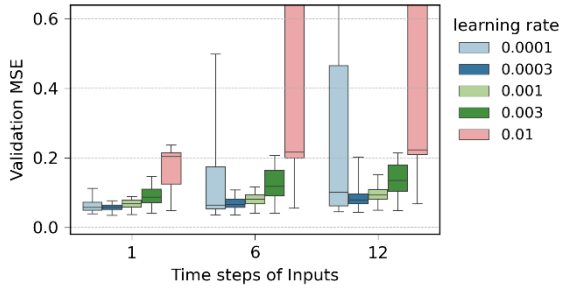
Fig. S2 Boxplot of the NNs' validation MSE with different hyperparameter values, Site Yichang



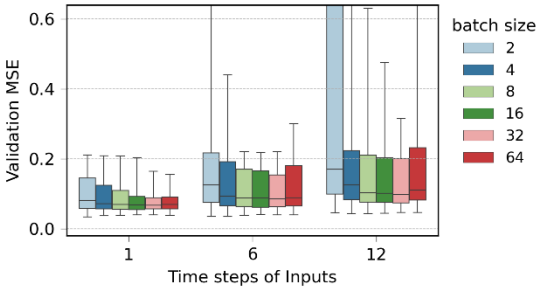
(a) Hidden layers



(b) Hidden dimension

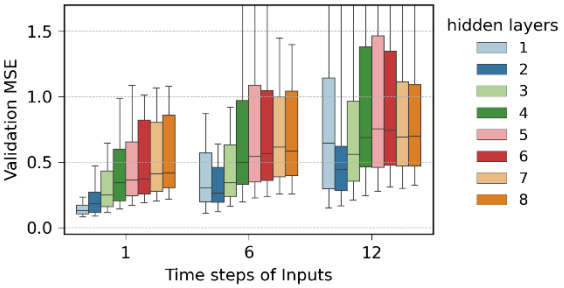


(c) Learning rate

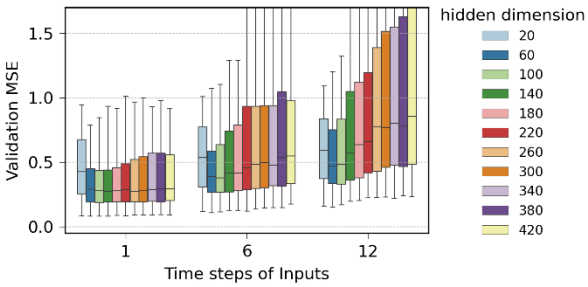


(d) Batch size

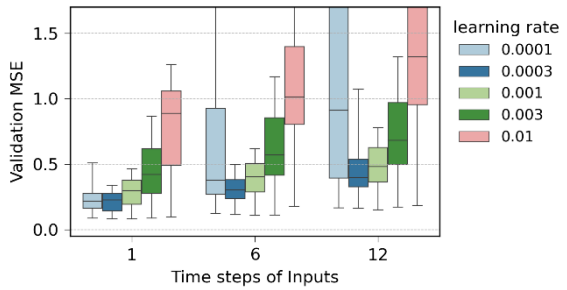
Fig. S3 Boxplot of the NNs' validation MSE with different hyperparameter values, Site Nanning



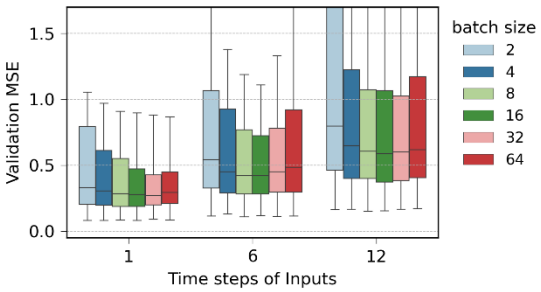
(a) Hidden layers



(b) Hidden dimension

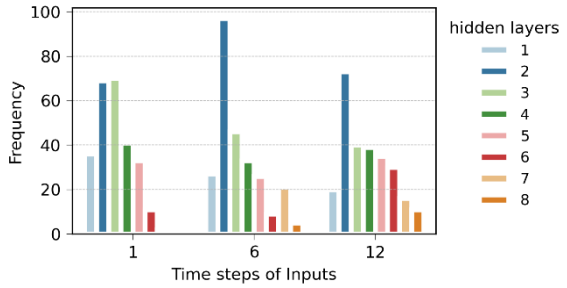


(c) Learning rate

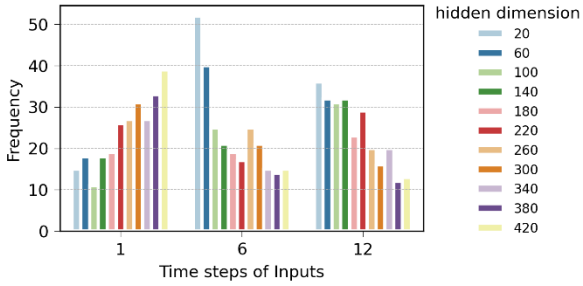


(d) Batch size

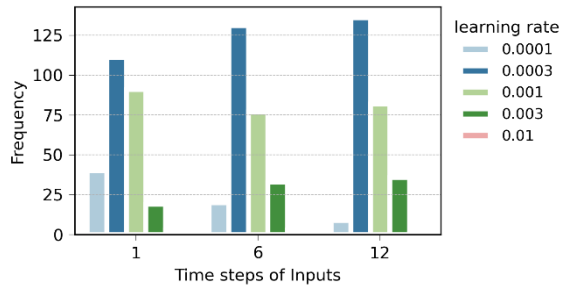
Fig. S4 Boxplot of the NNs' validation MSE with different hyperparameter values, Site Jiujiang



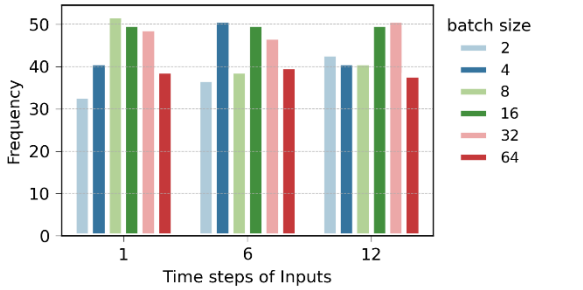
(a) Hidden layers



(b) Hidden dimension

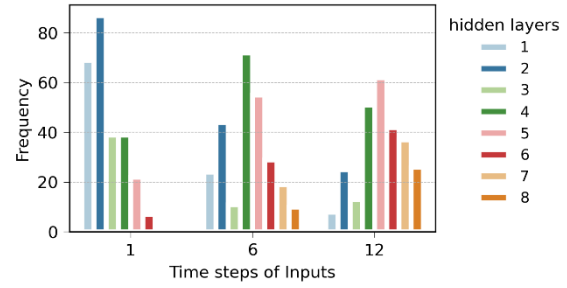


(c) Learning rate

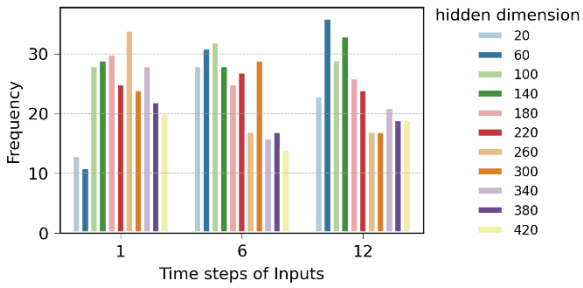


(d) Batch size

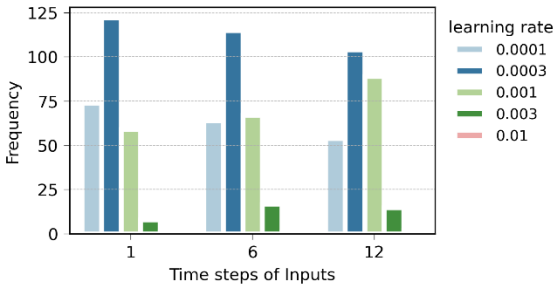
Fig. S5 Hyperparameters' posterior distributions of the top 10% best NNs, Site Anqing



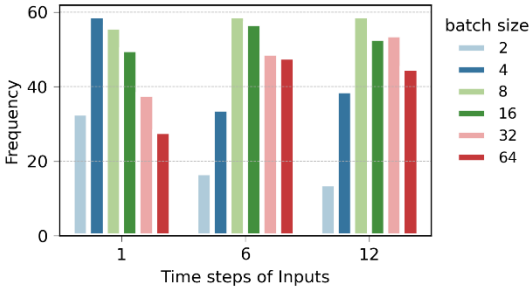
(e) Hidden layers



(f) Hidden dimension

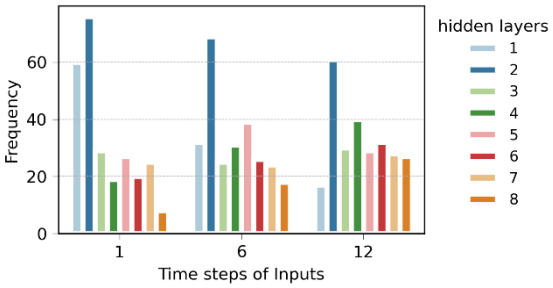


(g) Learning rate

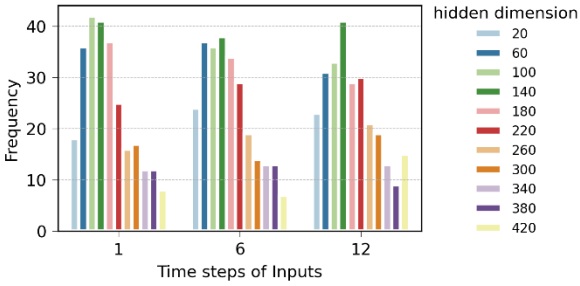


(h) Batch size

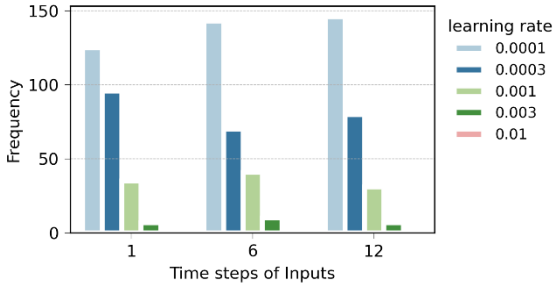
Fig. S6 Hyperparameters' posterior distributions of the top 10% best NNs, Site Yichang



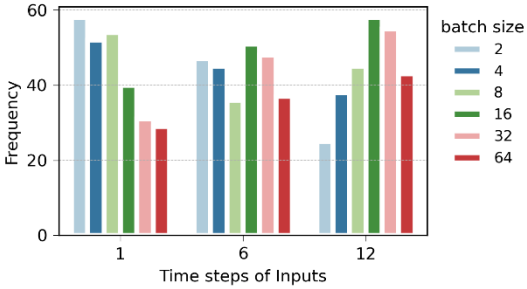
(i) Hidden layers



(j) Hidden dimension

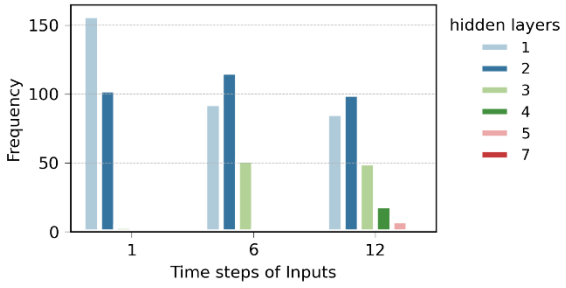


(k) Learning rate

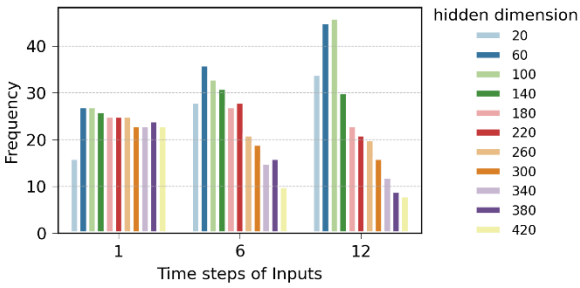


(l) Batch size

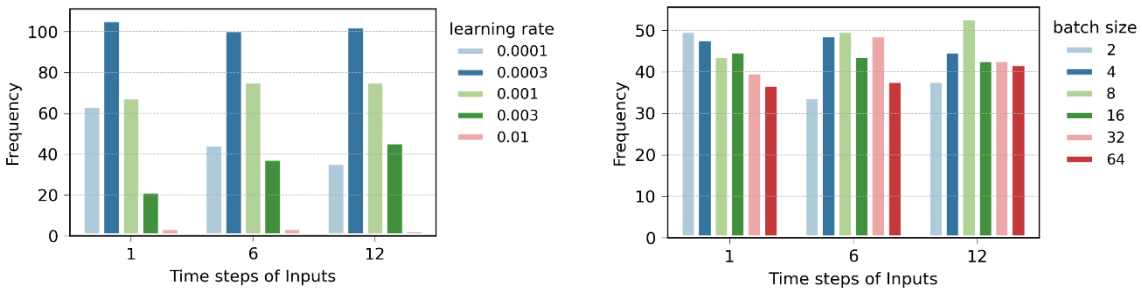
Fig. S7 Hyperparameters' posterior distributions of the top 10% best NNs, Site Nanning



(m) Hidden layers



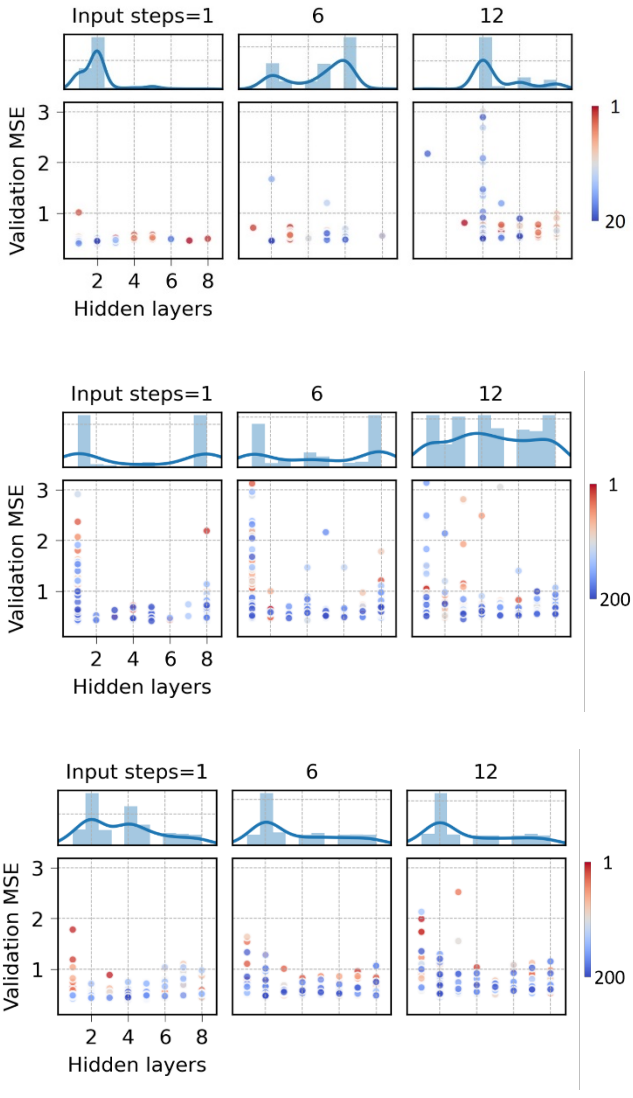
(n) Hidden dimension



(o) Learning rate (p) Batch size

Fig. S8 Hyperparameters' posterior distributions of the top 10% best NNs, Site Jiujiang

C2 Optimization orientation



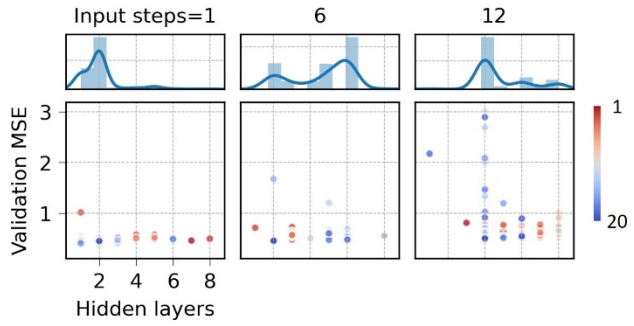
(a) GA

(b) GP

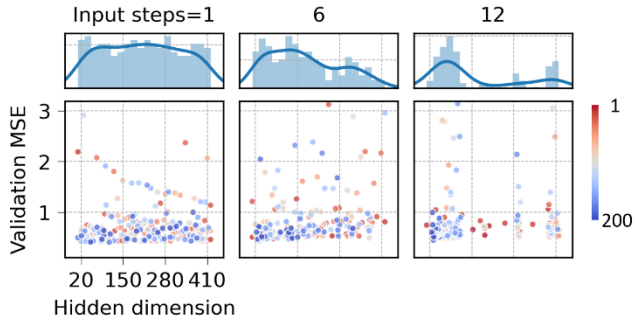
(c) TPE

Fig. S9 Distribution of the sampling values and map of the optimization orientation for hidden layers by the three

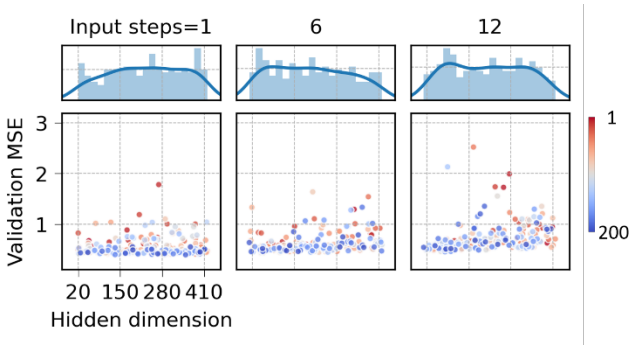
HPO methods, Site Anqing. (Caption: the points' color changes from red to blue represents the iteration number increased from one to the max, the same below.)



(a) GA

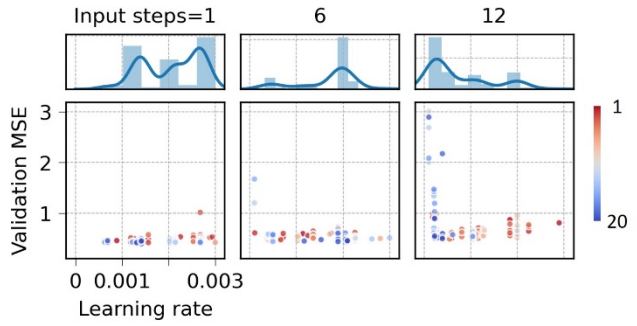


(b) GP

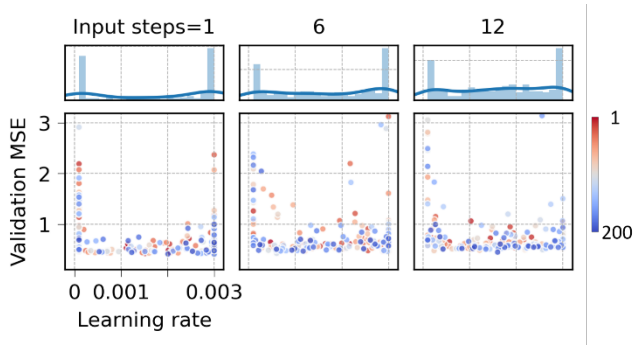


(c) TPE

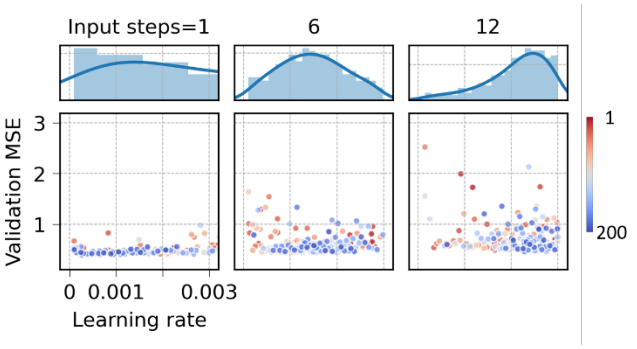
Fig. S10 Distribution of the sampling values and map of the optimization orientation for hidden dimension by the three HPO methods, Site Anqing



(a) GA

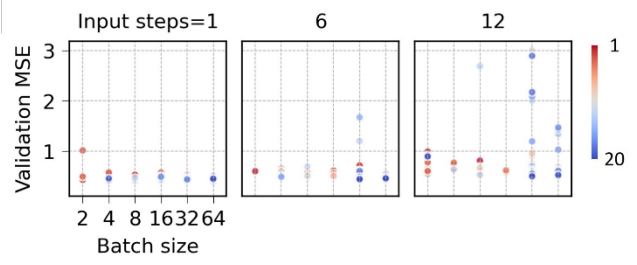


(b) GP

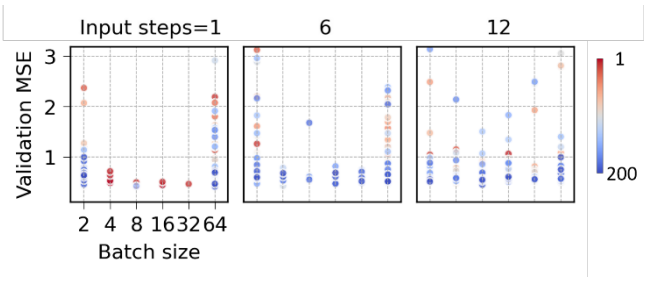


(c) TPE

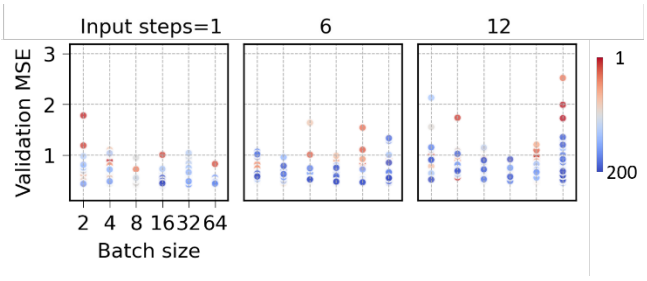
Fig. S11 Distribution of the sampling values and map of the optimization orientation for learning rate by the three HPO methods, Site Anqing



(a) GA

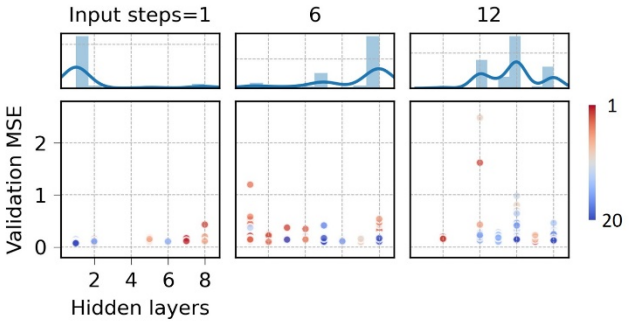


(b) GP

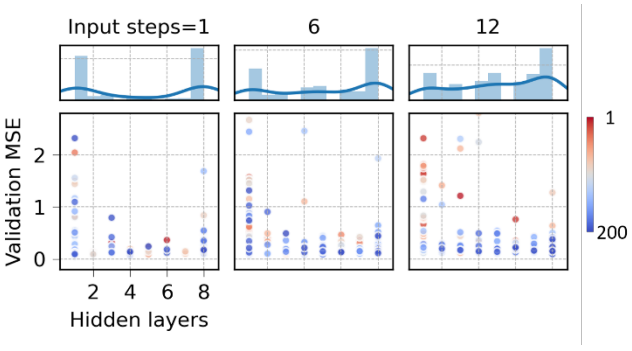


(c) TPE

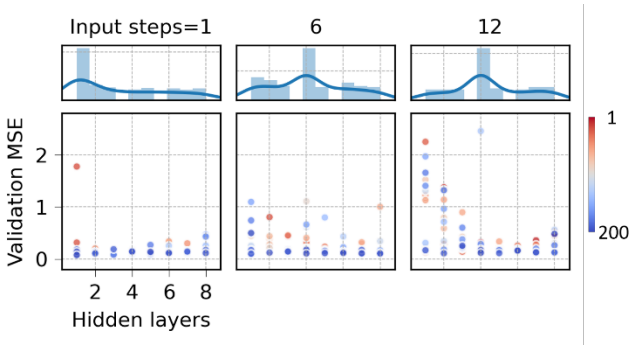
Fig. S12 Distribution of the sampling values and map of the optimization orientation for batch size by the three HPO methods, Site Anqing



(a) GA

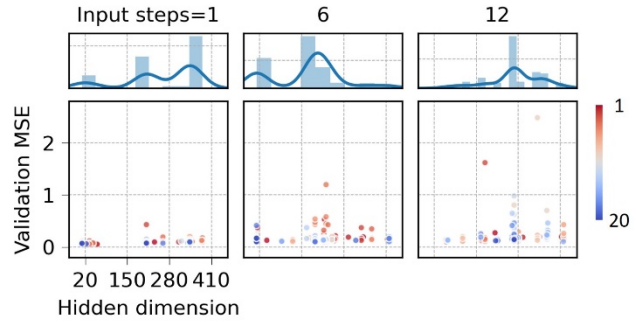


(b) GP

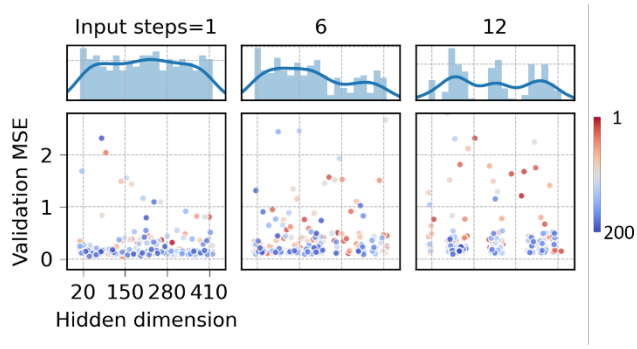


(c) TPE

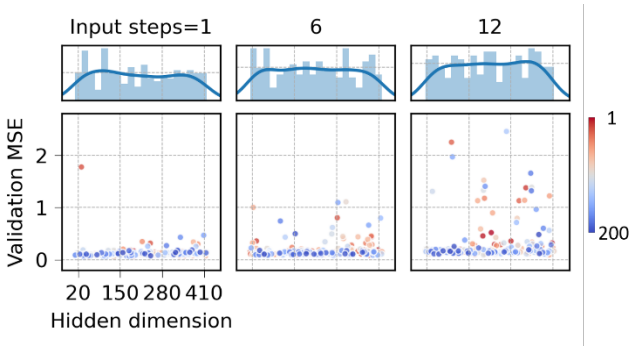
Fig. S13 Distribution of the sampling values and map of the optimization orientation for hidden layers by the three HPO methods, Site Yichang



(a) GA

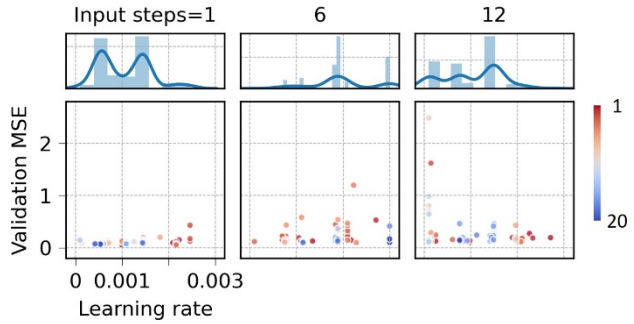


(b) GP

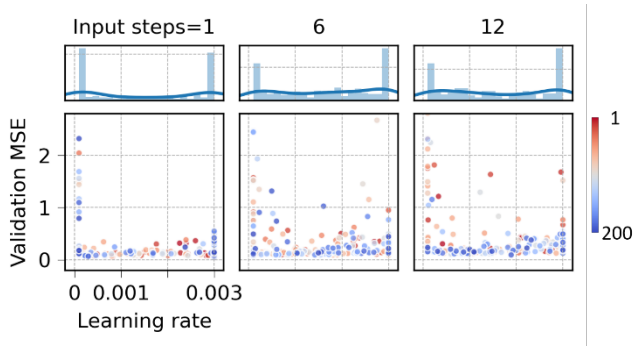


(c) TPE

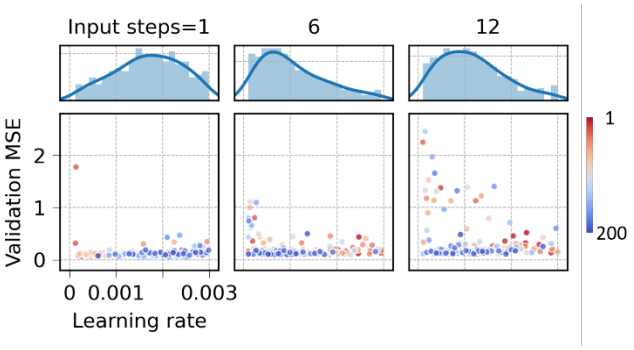
Fig. S14 Distribution of the sampling values and map of the optimization orientation for hidden dimension by the three HPO methods, Site Yichang



(a) GA

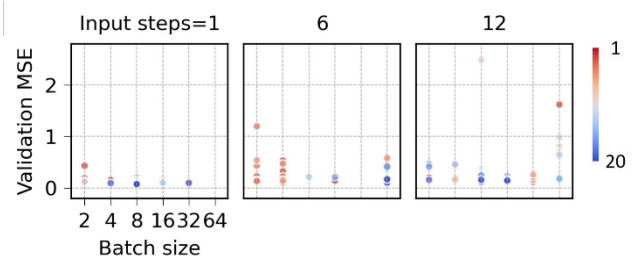


(b) GP

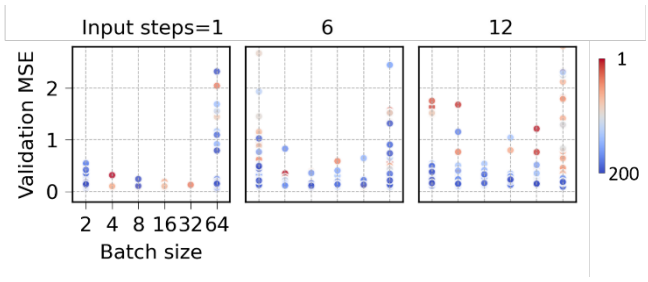


(c) TPE

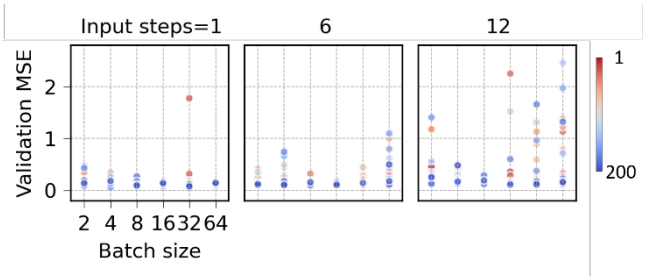
Fig. S15 Distribution of the sampling values and map of the optimization orientation for learning rate by the three HPO methods, Site Yichang



(a) GA

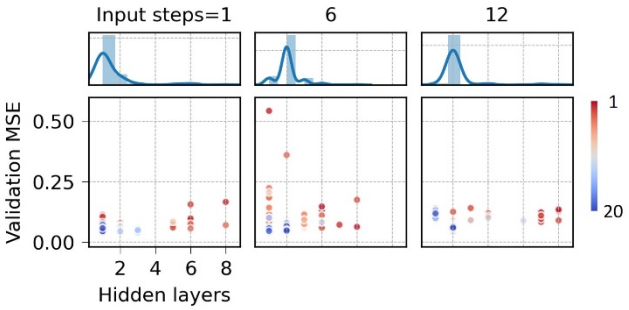


(b) GP

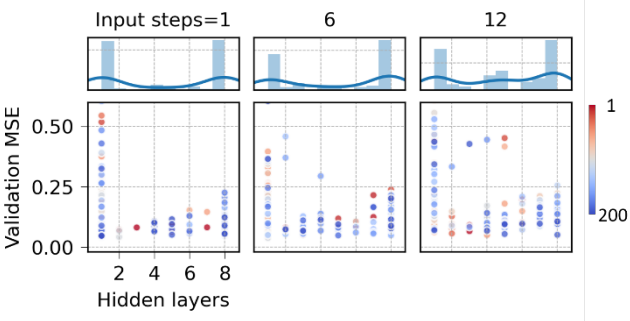


(c) TPE

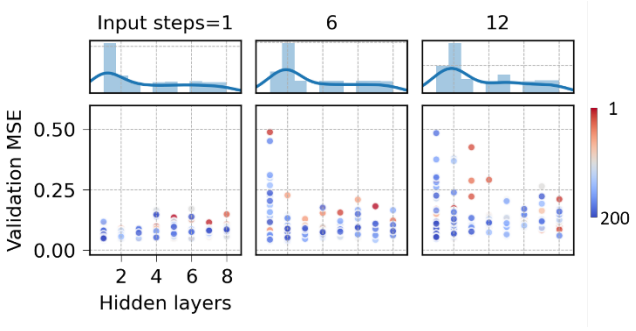
Fig. S16 Distribution of the sampling values and map of the optimization orientation for batch size by the three HPO methods, Site Yichang



(a) GA

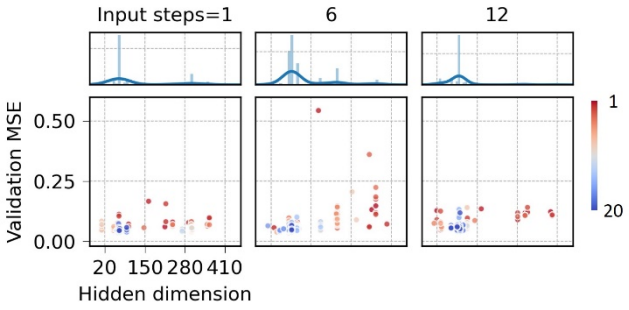


(b) GP

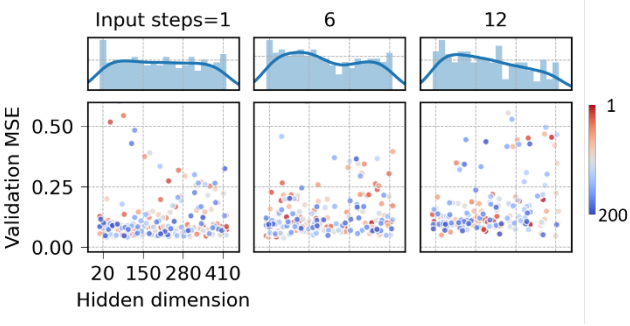


(c) TPE

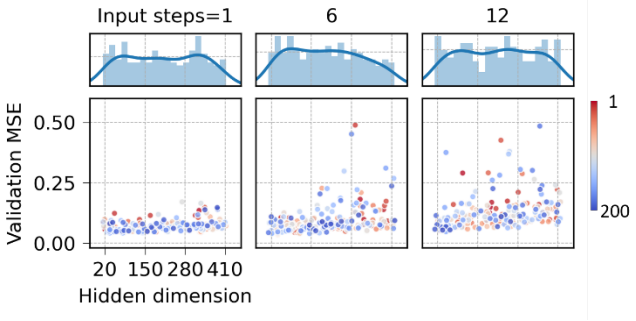
Fig. S17 Distribution of the sampling values and map of the optimization orientation for hidden layers by the three HPO methods, Site Nanning



(a) GA

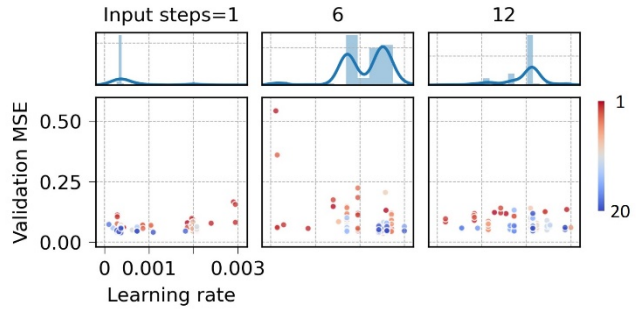


(b) GP

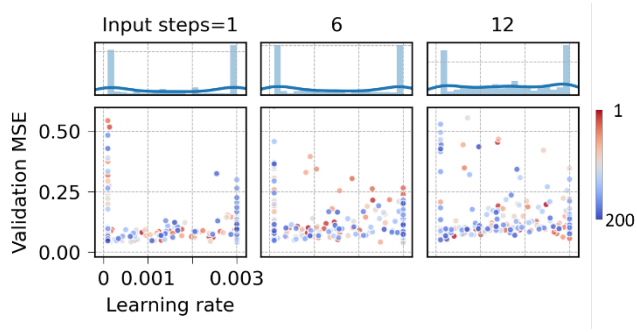


(c) TPE

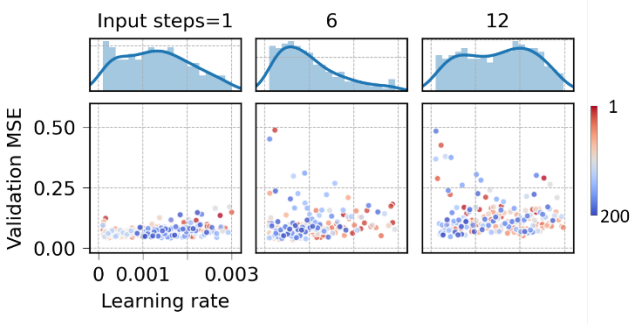
Fig. S18 Distribution of the sampling values and map of the optimization orientation for hidden dimension by the three HPO methods, Site Nanning



(a) GA

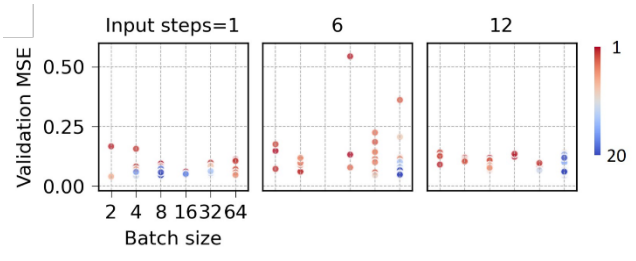


(b) GP

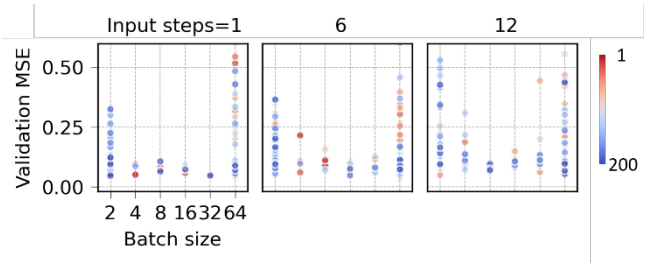


(c) TPE

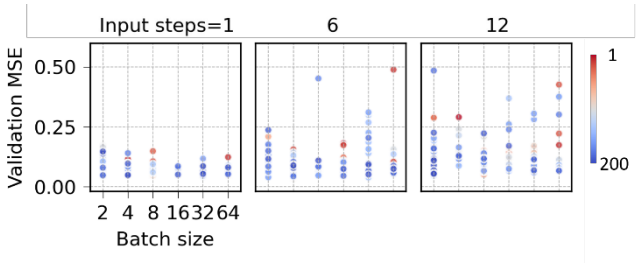
Fig. S19 Distribution of the sampling values and map of the optimization orientation for learning rate by the three HPO methods, Site Nanning



(a) GA

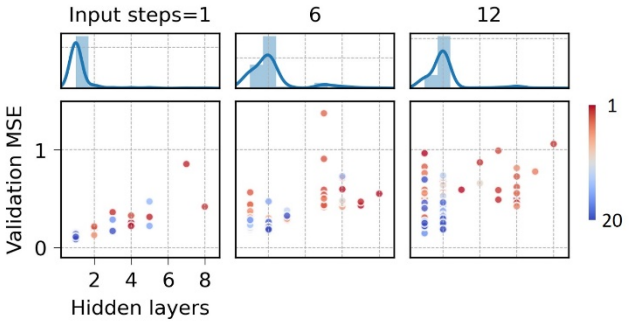


(b) GP

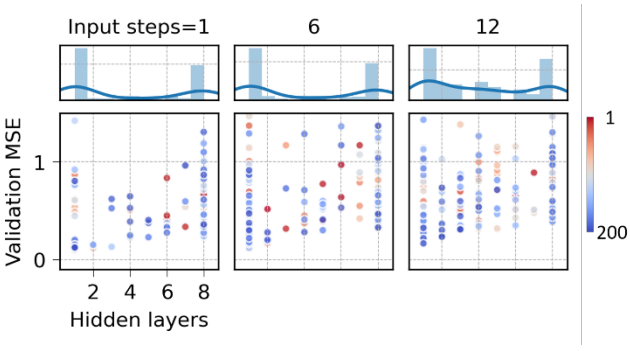


(c) TPE

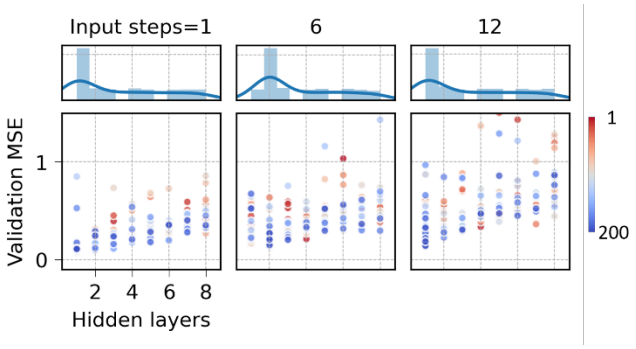
Fig. S20 Distribution of the sampling values and map of the optimization orientation for batch size by the three HPO methods, Site Nanning



(a) GA

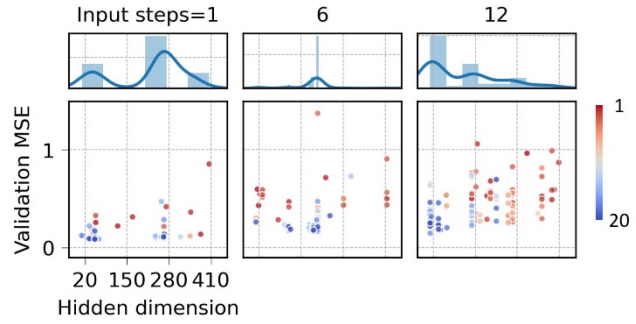


(b) GP

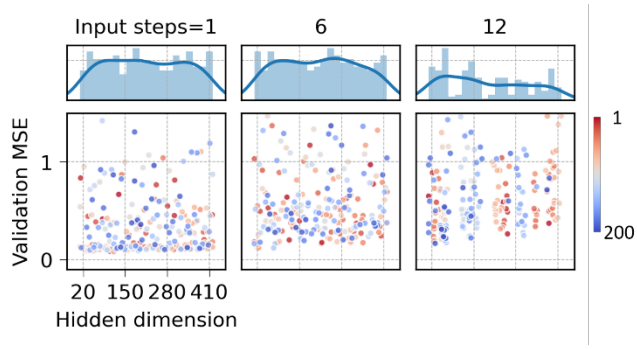


(c) TPE

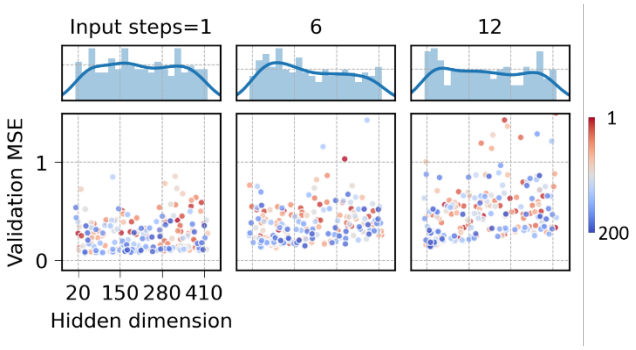
Fig. S21 Distribution of the sampling values and map of the optimization orientation for hidden layers by the three HPO methods, Site Jiujiang



(a) GA

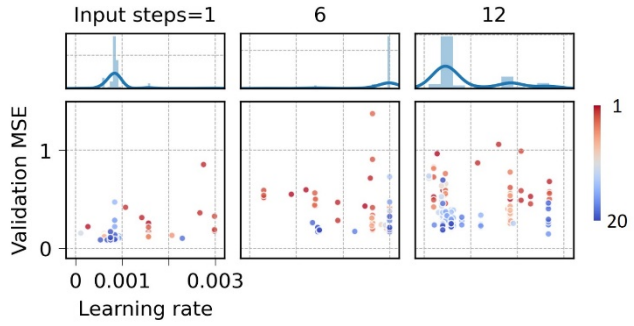


(b) GP

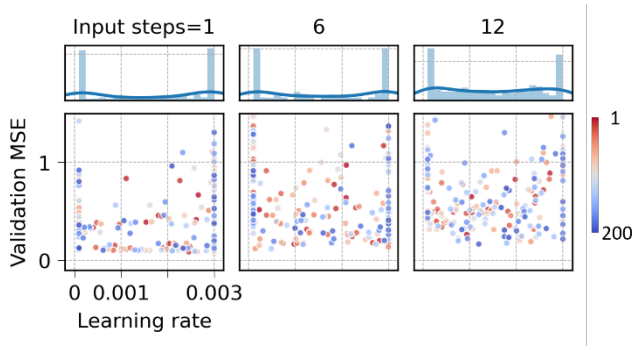


(c) TPE

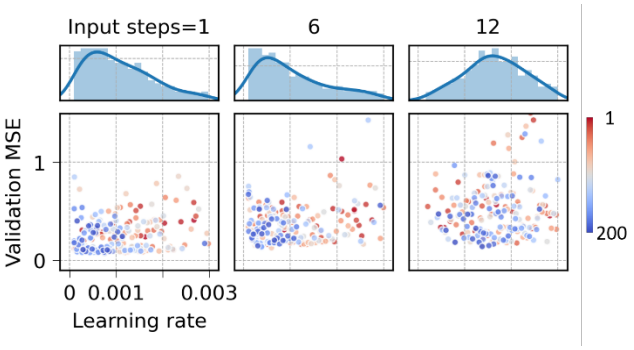
Fig. S22 Distribution of the sampling values and map of the optimization orientation for hidden dimension by the three HPO methods, Site Jiujiang



(a) GA

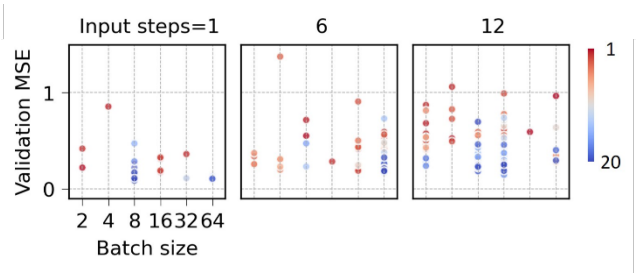


(b) GP

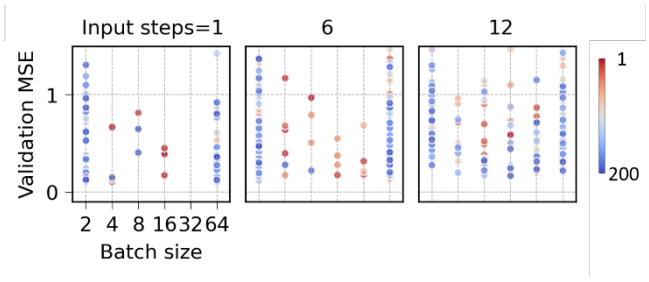


(c) TPE

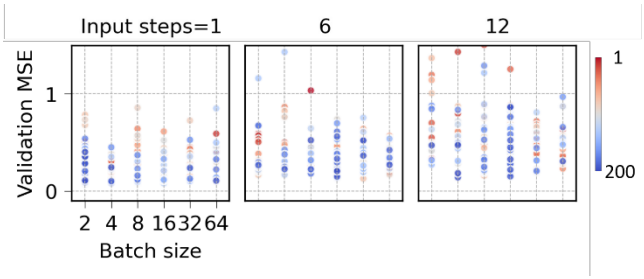
Fig. S23 Distribution of the sampling values and map of the optimization orientation for learning rate by the three HPO methods, Site Jiujiang



(a) GA



(b) GP



(c) TPE

Fig. S24 Distribution of the sampling values and map of the optimization orientation for batch size by the three HPO methods, Site Jiujiang

References

Bowden, G.J., Dandy, G.C., Maier, H.R., 2005a. Input determination for neural network models in water resources applications. Part 1—background and methodology. J. Hydrol. 301, 75–92. https://doi.org/10.1016/J.JHYDROL.2004.06.021

Bowden, G.J., Maier, H.R., Dandy, G.C., 2005b. Input determination for neural network models in water resources applications. Part 2. Case study: forecasting salinity in a river. J. Hydrol. 301, 93–107. https://doi.org/10.1016/J.JHYDROL.2004.06.020

Castelletti, A., Galelli, S., Restelli, M., Soncini-Sessa, R., 2012. Data-driven dynamic emulation modelling for the optimal management of environmental systems. Environ. Model. Softw. 34, 30–43. https://doi.org/10.1016/J.ENVSOFT.2011.09.003

Humphrey, G.B., Maier, H.R., Wu, W., Mount, N.J., Dandy, G.C., Abrahart, R.J., Dawson, C.W., 2017. Improved validation framework and R-package for artificial neural network models. Environ. Model. Softw. 92, 82–106. https://doi.org/10.1016/j.envsoft.2017.01.023

Kingma, D.P., Ba, J., 2014. Adam: A Method for Stochastic Optimization.

Wu, W., Dandy, G.C., Maier, H.R., 2014. Protocol for developing ANN models and its application to the assessment of the quality of the ANN model development process in drinking water quality modelling. Environ. Model. Softw. 54, 108–127. https://doi.org/10.1016/j.envsoft.2013.12.016