Algorithm 1: Determining the Parameters of Original BPNN

Set the initial value of sum of the determining factor $R_{\text{sum}}^2 = 0$; and sum of the relative error $\varepsilon_{\text{sum}} = 0$.

For a loop running with 10^3 iterations:

Clear all the remaining values of the last iteration;

1: Importing and operating the data.

2: Generate the network:

Change the specific parameters:

i. Set the neuron numbers $\xi \rightarrow$ change from 1 to 30.

ii. Set the training epochs $= 10^5$.

iii. Set the training goals $= 10^{-3}$.

iv. Set the learning rate $\alpha \rightarrow$ change from 10^{-1} to 10^{-5}.

Training the data with given parameters and creating simulation.

3: Calculate the results:

The sum of determining factor $R_{\text{sum}}^2 = R_{\text{sum}}^2 + R^2$, the sum of the relative error $\varepsilon_{\text{sum}} = \varepsilon_{\text{sum}} + \varepsilon$.

The mean value of R^2 and ε is calculated by $\frac{R_{\text{sum}}^2}{\text{Iterations}}, \frac{\varepsilon_{\text{sum}}}{\text{Iterations}}$.

Calculate the CPU running time t_{run}

End the loop.

Algorithm 2: Obtain the BPNN model

For a loop running with 10^5 iterations:

Clear all the remaining values of the last iteration;

1: Operating the data:

Import the water discharge experimental data to define the input and output data, and set the training and testing sets.

Normalize the input dataset.

2: Generate the network:

Set the given parameters obtained from Algorithm 1.

Training the data with given parameters and creating simulation.

3: Calculate the results:

Obtaining errors and deciding parameter R^2;

Outputs the results as comparing the testing data and simulation results, and calculate the relative error ε.

4: Generate a condition to break the loop:

If a value of R^2 that beyond 0.95 is detected, then plot the errors distribution and break the loop.

End the loop.