Interview task report

Applicant：Bufan Shi

Birthday: 03.02.1995

Graduated School: Tongji University

E-mail: 1731609@tongji.edu.cn

1. Tools & Language

1.1 IDE

Matlab2017b

1.2 Language

Matlab

1.3 Tools

Vector Fitting (Matlab toolbox)
Neural Network Fitting (Matlab toolbox)
Libsvm-3.23 (installed Matlab toolbox)

1.4 Preparation

Before building model with Matlab, the library for Support Vector Machines(libsvm-3.23) should be installed successfully.

Training data and testing data are all placed in the same folder named multiparameter modeling.

2. Modeling & Predicting

2.1 Vector Fitting

Under different candidate designs, the Vector fitting tool, rationalfit(freq, data), is used to fit the frequency input and responses and then acquire the actual orders of transfer function. The array order_TF is used to the order of TF for corresponding candidate design.

2.2 Classification

According to the orders of TF, the test data is divided into 4 categories, namely order = 6, 7, 8, 10.

Then candidate data and responses data are divided into 4 groups, which will be used for

ANN training.

2.3 ANN Training

Neural Network Fitting toolbox is used to create and train the feed-forward backpropagation network (BPNN). And newff() and train() are selected to create and train BPNN respectively.

In order to improve the effectiveness of the training process, the training data set and testing data set are normalized first.

The input node number is 4, representing frequency and candidate. And the output node number is 2, representing the real and imaginary part of the response.

Category 1 (order = 6)

For category 1 (order = 6), the node number of the hidden layer is 300, the number of training epochs is 50, the goal is 1e-8, learning rate is 0.1.

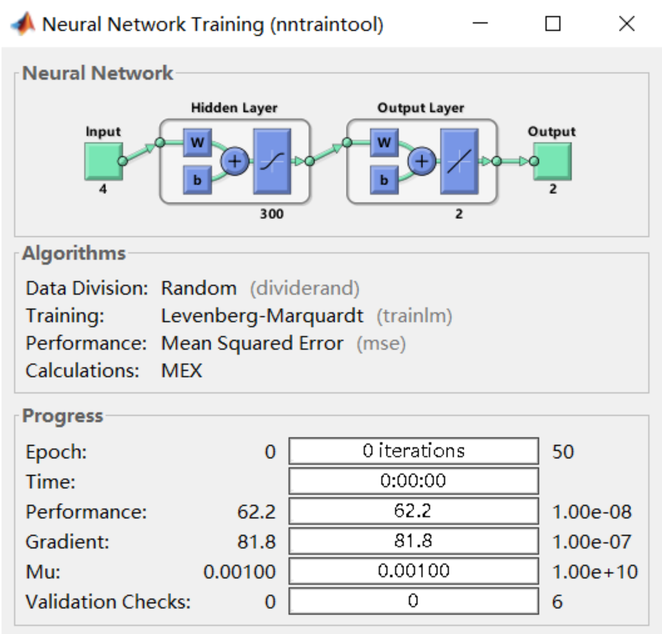

Fig.1. BPNN Training for Category 1

Category 2 (order = 7)

For category 2 (order = 6), the node number of the hidden layer is 300, the number of training epochs is 50, the goal is 1e-8, learning rate is 0.1.

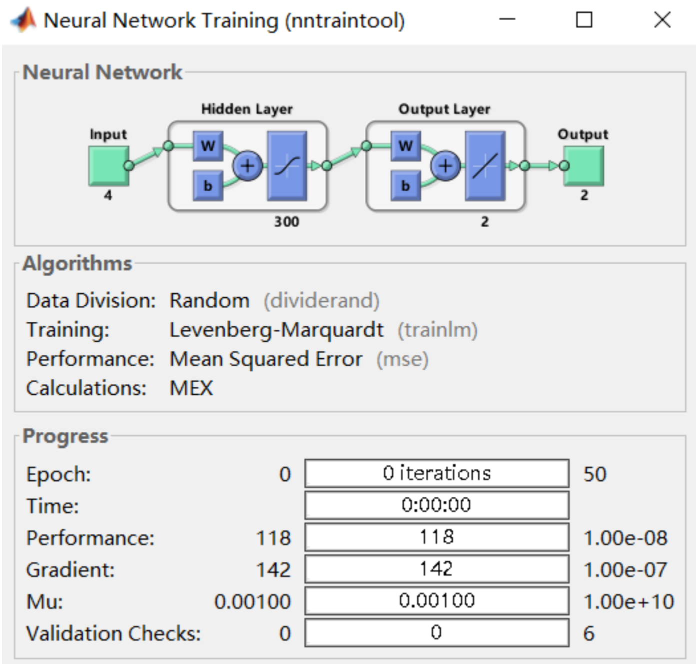

Fig.2. BPNN Training for Category 2

Category 3 (order = 8)

For category 3 (order = 8), the node number of the hidden layer is 250, the number of training epochs is 100, the goal is 1e-8, learning rate is 0.05.

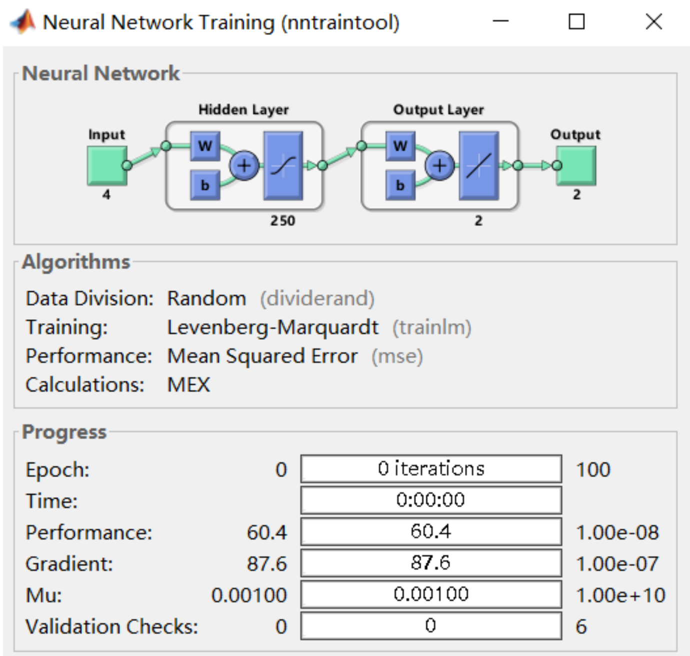

Fig.3. BPNN Training for Category 3

Category 4 (order = 10)

For category 4 (order = 10), the node number of the hidden layer is 200, the number of training epochs is 100, the goal is 1e-8, learning rate is 0.05.

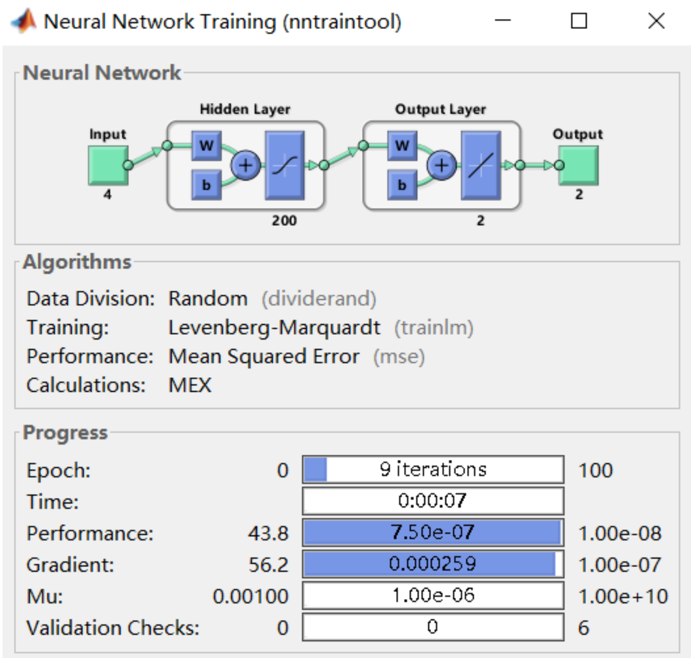

Fig.4. BPNN Training for Category 4

When training is finished, the ANN models are saved.

2.4 SVM Training & Predicting

SVM is used for classification. Libsvm-3.23 toolbox is installed and used for modeling in Matlab. svmtrain() and svmpredict() are used for modeling and predicting respectively.

Grid search and cross-validation method are used to find the optimal parameters for training SVM model. The optimal c is 2^26 and the optimal g is 2^-28. Besides, RBF kernel function is selected.

After training, the classification accuracy of the trained SVM for training data and testing data are both 100%.

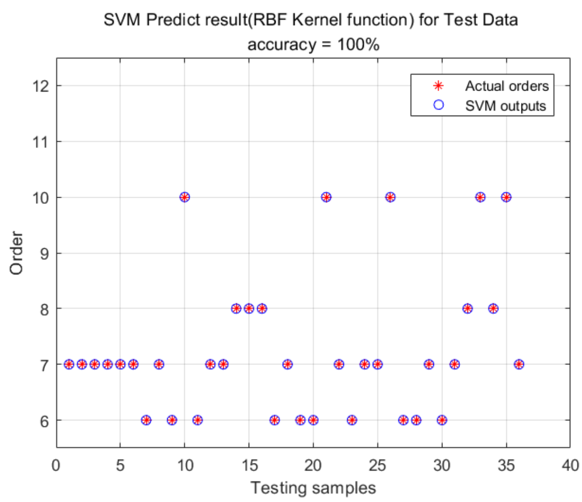

Fig.5. Classifying results of S-parameter from SVM

2.5 ANN Prediction

After using trained SVM to classify the test data, the trained ANN models are used to predict the responses under different candidates and frequency.

The mean absolute percentage error (MAPE) and the mean percentage error (MPE) are used for performance evaluation and comparison followed

$$MAPE = \frac{1}{N}\sum_{n=1}^{N}\left|\frac{y_n - \hat{y}_n}{y_n}\right| \times 100\%$$

$$MPE = \frac{1}{N}\sum_{n=1}^{N}\sqrt{\left(\frac{R_n - \hat{R}_n}{R_n}\right)^2 + \left(\frac{I_n - \hat{I}_n}{I_n}\right)^2} \times 100\%$$

where y_n and $\hat{y}_n$ denote the observed and forecasted absolute value of response, respectively, of the nth datum, R_n and $\hat{R}_n$ denote the observed and forecasted real part of response, respectively, of the nth datum, I_n and $\hat{I}_n$ denote the observed and forecasted imaginary values, respectively, of the nth datum, N is the total number of the data.

After predicting process, the testing MAPE is 0.041206% and the testing MPE is 0.52267%.

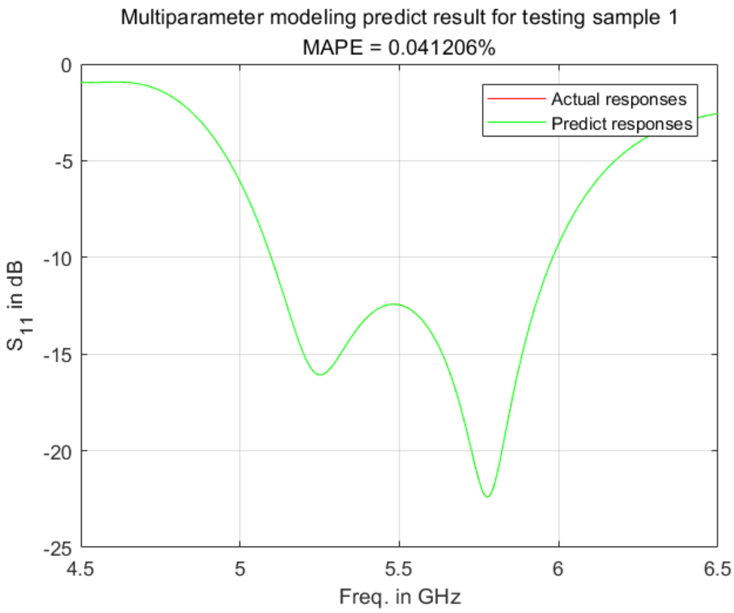

Fig.6. Comparison between actual responses and predict responses for testing sample 1

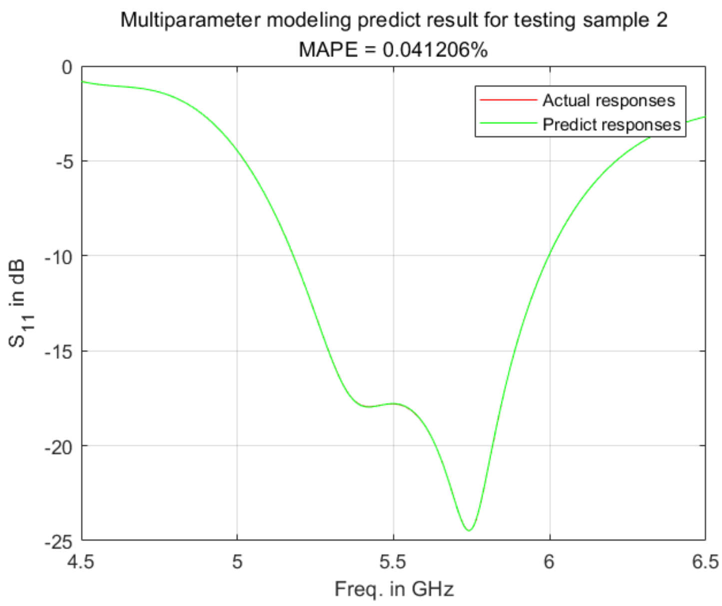

Fig.7. Comparison between actual responses and predict responses for testing sample 2