

The Study of Machine Learning for Wire Rupture Prediction in WEDM

Ping Hsien Chou (✉ flyfox0724@gmail.com)

National Central University <https://orcid.org/0000-0001-9788-2756>

Yean-Ren Hwang

Department of Mechanical Engineering, National Central University

Bling-Hwa Yan

Department of Mechanical Engineering, National Central University

Research Article

Keywords: WEDM, wire rupture, Artificial Neural Network

Posted Date: August 17th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-755393/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Version of Record: A version of this preprint was published at The International Journal of Advanced Manufacturing Technology on November 11th, 2021. See the published version at <https://doi.org/10.1007/s00170-021-08323-5>.

Abstract

During wire electrical discharge machining (WEDM), wire rupture may deteriorate workpieces' machined surfaces and increase the processing time. However, only a few referenced papers focused on wire rupture during past decades because of its complexity. In this research, Machine Learning (ML) technique was applied to analyze the relationship between manufacturing parameters and the chance of wire rupture. Three parameters, including gap voltage (GV), feed rate (FR), and water resistance (WR), were considered as training features, and a total of 298 sets were used to train an Artificial Neural Network (ANN). The results show that the prediction accuracy of wire rupture for 10 seconds in advance is above 85%.

Introduction

Nowadays, WEDM is one of the widely used non-conventional machining processes. It may be applied to cut electrically conductive materials and complicated machining shapes that traditional machining processes cannot produce [1–2]. It also has the potential to cut the polycrystalline silicon in recent years [3–4]. There are some optimization researches for better manufacturing performance is published due to its importance [5–6]. WEDM uses a continuously traveling wire electrode made of copper, brass, or tungsten. The wire is kept in tension using a tensioning device, reducing the inaccurate production. During the process, the material is eroded by heat ahead of the wire. There is no direct contact between the workpiece and the wire, eliminating the mechanical stresses during machining [7]. Therefore, the wire electrode is the key component of the machining process [8]. The risk of wire breakage has undermined the full potential of the WEDM, drastically reducing the efficiency and accuracy of the process. Different factors lead to wire breakages, like high wire tension, electrical discharge impact, thermal load, and high-temperature influence the wire strength and, consequently, wire rupture [9]. Moreover, these phenomena come from the machining parameters, so the relationship between machining parameters and the possibility of wire rupture can be considered a complicated nonlinear problem.

ML, a computer algorithm to extract useful knowledge from experience, has been shown an effective method to improve machining manufacturing [10]. A mathematical model builds in the ML algorithm based on sample data, known as "training data," to make predictions or decisions. And then, "testing data" is used to prove that the accuracy and robustness of the model are suitable for the application. ML is frequently used to solve problems that only can be solved by a lot of experiences or data in recent years. It has been applied to solve many complicated nonlinear problems on machining processes and obtained an outstanding performance [11]. Sharma et al. [12] formed the ANN model of surface roughness in terms of speed, feed, depth of cut, and approaching angle. Karayel [13] derived surface roughness by ANN, close to actual values.

One of the critical steps in Machine learning is data processing. The quality and correctness of the data set affected the training result obviously and unignored. Therefore, the collected data should be preprocessed cautiously by domain knowledge before input into the neural network. In this research, the

machining status can be considered two statuses. First is the normal machining process, which means the wire does not break. The second is wire rupture, which means the machining process stop by wire rupture. Because the label of wire state does not exist in the original data so needs to do data processing. The method of processing of data set will be introduced in the next chapter carefully. This situation is one kind of classification problem by these two kinds of wire states. Luckily, ML can solve the classification problem in lots of ways. Therefore, the ANN was applied to analyze the relationship between manufacturing parameters and the chance of wire rupture for WEDM.

Methodology

2.1 Data Processing

Data processing is a critical component for applying ML to wire rupture prediction. During the WEDM process, CHMER machines will record almost 200 parameters data every second, including the error code of the controller, the state of machining processing, the G-Code of the processing path, the parameters of the operation... etc. However, many parameters do not change during the process or do not affect the manufacturing; hence are neglected for training. Filtering for raw data and making labels are the essential step in the data processing. In Fig. 1, the rupture point is 23290, and the code of "Wirestate" is 2. The 10 seconds (23280–23289) data before the rupture point are "rupture data", and the label of "Wirestate" is 0. The data has 1 in the "Wirestate" column means working data. There is no apparent difference for machining parameters between rupture data and working data in Fig. 1. Therefore, ML be used to train and find the trend of the parameters. There has another label, Rupture Chance, which is 200 on rupture point. The probability of rupture is 100% before rupture point 1 second and decreases 10% per second in order.

This study aims to build a model to predict the wire state and prevent wire rupture. Therefore, the 10 seconds data were used as one data set to predict the wire state in the next second. Furthermore, choosing which manufacturing parameters recorded the machining process as the neural network input is a serious step. The machine tool used in this research already has a monitor to record the manufacturing parameters. Among all recorded data, Gap voltage (GT), Feed rate (FR), and water resistance (WR) were chosen as input data according to the experiment results in previous research documents [6]. A set containing 30 data of 10 seconds of machining process was considered an input data set for ML. Totally, 298 data sets were used for training, validation, and testing. The prediction of wire rupture can be considered a classification problem containing two states, i.e., 'wire rupture' and 'normal working' classes. Therefore, the data set for these two converse states were collected to be the same amount so that the data-unbalanced situation in classification may be avoided.

Three groups of data sets, i.e., 'Training', 'Validation' and 'Testing', were formed by random selection from 298 sets. Among 298 data sets, 214 sets (72%) were used for training the network, 24 sets (8%) were used for validation, and 60 sets (20%) were used for testing. The percentages of these groups are shown in Fig. 2.

2.2 Architecture of Network

In this research, ANN was used to analyze the relationship between the sensors' data and the state of wire rupture. Generally, an ANN is made up of some neurons connected via the link likes Fig. 3. The feed-forward neural network based on back-propagation is the best general-purpose model among various neural network models. In [14], it is mentioned that ANN is the most commonly used network to solve classification problems.

An ANN is built by one input layer, one or several hidden layers, and one output layer. The input layer is a 1D array with specific lengths. Base on the flexibility of ANN, various combinations of the different layer numbers and neurons number were tested in this study to find out the better architecture. First, one hidden layer with different neurons was tested. Some specific numbers of neurons that have high-grade performance are expected to choose for the next experiment. For the same total neuron number, the hidden layers will vary from 1 to 4 to obtain the best network system. For example, if there are N total neurons are used in the network, each layer will be N/L_n , where L_n represents the layer number.

The ANN has two output values and modifies them into 0 to 1 by the softmax function in this research. The classification layer uses the value passed by the softmax layer to classify into two classes, includes 'Wire rupture' and 'Wire no rupture'. The loss function is the cross-entropy loss function, likes Eq. 1, for multi-class classification problems with mutually exclusive classes. The complete model built by MATLAB is shown in Fig. 4.

$$Loss = \frac{-1}{N} \times \sum_1^N y_i \times \log \hat{y}_i + (1 - y_i) \times \log(1 - \hat{y}_i)$$

N is the number of classes, y_i is the real value, \hat{y}_i is the predicted value y_i and \hat{y}_i are between 0 to 1.

Eq 1. The cross-entropy loss function

2.3 Finding the important parameter

During the training process, it is found that the result of trained network may not perform better if too many input parameters included in a data set. By observing the accuracy, valuable parameters may be identified.

Three parameters (GT, FR, WR) were considered in the first experiment. In this part, the different combinations of parameters, shown in Table 1, were trained network and the effect between parameter and prediction accuracy were evaluated.

Result And Discussion

3.1 Comparison of ANN architecture in different neuron numbers

In this research, different neuron numbers were used in the hidden layer to predict the state of wire rupture. Because the network produces the weight randomly in each training time, every network is different even they have the same data source and architecture. Therefore, each architecture was trained five times to check the model's robustness and average performance.

Figure 5 shows the training result (blue line) and testing result (red line) of all architecture. Each architecture result includes the highest, the lowest, and the median accuracy in five experiments. In addition, the training result and the testing result are shown in Fig. 6 and Fig. 7, respectively. The accuracy becomes stable and similar when the neuron number is varying from 100 to 300. That means it does not need more than 300 neurons to simulate this nonlinear model. However, the accuracy decreases by increasing neurons when neurons number over 300. The reason is that when low-dimensional data is extended to too many neurons, some useless values and noise will be generated. Therefore, using too many neurons in the network is unsuitable in this research. Table 2 lists some results for the 0.001 learning rate.

Moreover, the 0.001 learning rate applied on prior training, and the result of another learning rate (0.0001) is shown in Fig. 8. The accuracy does not drop down in the high neurons number, but the accuracy is not high as the previous training.

3.2 Comparison of ANN architecture in different hidden layer numbers

In order to find a better ANN architecture, one to four hidden layers were used for comparison. The total amount of neurons were kept the same for all structures. For example, if 300 neurons are used in the structure, each layer contains 150 neurons for 2-hidden-layers structure, 100 neurons for 3-hidden-layers, etc. During the experiment, the total number of neurons varied from 100 to 300, and the learning rate is 0.001. As shown in Fig. 11, the result shows that the accuracy becomes better for more neuron numbers and more hidden layers used in the structure. However, when the neuron number increases to 400, there is no apparent improvement trend in Fig. 12.

After applying the basic artificial neural network to predict the wire state, the application of the ML tool on the wire rupture problem of WEDM is proved. The development of machine learning is quick and efficient that many powerful networks are developed already. Maybe the wire rupture problem can be solved by other functional networks.

3.3 Comparison of different input parameter

After designing ANN architecture, an applicable ANN network, which is one hidden layer with 100 to 300 neuron numbers, was defined to predict the state of the wire. In this experiment, different combinations of three parameters were trained and try to find the critical parameter. For example, case 2 uses the gap voltage and the feed rate as the input parameters, and the input size of the network is 20. In addition, case 1 was the control group with one hidden layer and a 0.001 learning rate. All combination results show the part of steady accuracy in which neuron numbers from 100 to 300. Each architecture in the

case in Table 2 also was trained five times, likes Fig. 11 to Fig. 17. The medium testing accuracy is shown in Table 3.

Table 3 shows that the accuracy closes 70% by using Feed Rate (FR) and over 70 % by using gap voltage (GT) as input. However, the water resist (WR) shows low relevance with wire rupture. Compare to the result of case 1 and case 2, the participation of water resists even lower the prediction accuracy.

Based on this result, the author supposes a control strategy. If the machining parameters make the wire rupture in our model, adjust the manufacturing parameter to affect the gap voltage or feed rate. Therefore, a new controller can be designed into the original controller in the WEDM machining tool to increase efficiency. When the controller gets the 10-second data from the sensors, it can predict the wire's state in the next second and modify the machining parameters by the predicted result.

Table 3 shows that the accuracy gets over 80% by using Feed Rate (FR) and over 70 % by using gap voltage (GT) as input. However, the water resist (WR) shows low relevance with wire rupture. Compare to the result of case 1, the participation of water resists even lower the prediction accuracy.

Based on this result, the author supposes a control strategy. If the machining parameters make the wire rupture in our model, adjust the manufacturing parameter to affect the gap voltage or feed rate. Therefore, a new controller can be designed into the original controller in the WEDM machining tool to increase efficiency. When the controller gets the 10-second data from the sensors, it can predict the wire's state in the next second and modify the machining parameters by the predicted result.

Conclusion

This paper shows that the ML algorithm can be used to predict wire rupture for WEDM. Although the basic artificial neural network, which maybe is not the best one was used in this research, it still has a good performance to point out the potential of the application on the ML. This research uses 130 neuron numbers in the single hidden layer to get 85% accuracy. The author expects that applying other functional networks to solve the wire rupture more efficiently in the future.

Moreover, the ML learning tool and other powerful references study help the researcher determine the critical machining parameter for wire rupture. Finding the critical parameter makes the prediction network more efficient and correct one the actual control application. In the future, the author supposes combining with other algorithms that can find out the suitable machining parameters combination to avoid wire breakage. It is an automatic strategy that does not need experience from the operator. When the predicted model meets the wire rupture result, other algorithms automatically adjust the machining factors to avoid the situation.

Declarations

Funding (Not applicable)

Conflicts of interest/Competing interests (Not applicable)

Availability of data and material The industrial processing data of CHMER WEDM machines are collected for training ANN.

Code availability This research used MATLAB to build the neural network architecture.

Authors' contributions The main contribution of this paper is to predict the wire rupture in advance using ANN techniques.

Ethics approval (Not applicable)

Consent to participate (Not applicable)

Consent for publication (Not applicable)

References

1. Maher I., Sarhan A.A.D, Marashi H. (2017) Wire Rupture Optimization in Wire Electrical Discharge Machining using Taguchi Approach. MATEC Web Conf 95. <https://doi.org/10.1051/mateconf/20179507014>
2. Yeh C.-C, Wu K.-L., Lee j.-W., Yan B.-H. (2013). Processing Characteristics Using Phosphorous Dielectric on Wire Electrical Discharge Machining of Polycrystalline Silicon. The International Journal of Advanced Manufacturing Technology 29:146-152. <https://doi.org/10.1080/10426914.2013.852203>
3. Lin J.-L., Wang K.-S., Yan B.-H, Tarng Y.-S. (2000) Optimization of the electrical discharge machining process based on the Taguchi method with fuzzy logics. Journal of Materials Processing Technology 102:48-55. [https://doi.org/10.1016/S0924-0136\(00\)00438-6](https://doi.org/10.1016/S0924-0136(00)00438-6)
4. Yeh C.-C., Wu K.-L., Lee J.-W., Yan B.-H. (2013). Study on surface characteristics using phosphorous dielectric on wire electrical discharge machining of polycrystalline silicon. The International Journal of Advanced Manufacturing Technology 69:71–80. <https://doi.org/10.1007/s00170-013-4995-9>
5. Tzeng H.-J., Yan B.-H., Hsu R.-T., Chow H.-M. (2007). Finishing effect of abrasive flow machining on micro slit fabricated by wire-EDM. The International Journal of Advanced Manufacturing Technology 34:649-656. <https://doi.org/10.1007/s00170-006-0632-1>
6. Ishfaq K., Anwar S., Ali M.A. et al (2020) Optimization of WEDM for precise machining of novel developed Al6061-7.5% SiC squeeze-casted composite. The International Journal of Advanced Manufacturing Technology 111: 2031–2049. <https://doi.org/10.1007/s00170-020-06218-5>
7. Ho K.H., Newman S.T., Rahimifard S., Allen R.D. (2004) State of the art in wire electrical discharge machining (WEDM), International Journal of Machine Tools and Manufacture, Volume 44:1247-1259. <https://doi.org/10.1016/j.ijmachtools.2004.04.017>

8. Maher I., Sarhan A.A.D., Hamdi M. (2015) Review of improvements in wire electrode properties for longer working time and utilization in wire EDM machining. *Int J Adv Manuf Technol* 76: 329–351. <https://doi.org/10.1007/s00170-014-6243-3>
9. Luo Y.F., (1999) Rupture failure and mechanical strength of the electrode wire used in wire EDM, *Journal of Materials Processing Technology* 94:208-215. [https://doi.org/10.1016/S0924-0136\(99\)00107-7](https://doi.org/10.1016/S0924-0136(99)00107-7)
10. Mia M., Dhar N.R. (2016) Prediction of surface roughness in hard turning under high pressure coolant using Artificial Neural Network, *Measurement* 92:464-474. <https://doi.org/10.1016/j.measurement.2016.06.048>
11. Han J.-H., Chi S.-Y. (2016) Consideration of manufacturing data to apply machine learning methods for predictive manufacturing, *Eighth International Conference on Ubiquitous and Future Networks (ICUFN)*. 109-113, <http://dx.doi.org/10.1109/ICUFN.2016.7536995>
12. Sharma V.S., Dhiman S., Sehgal R., Sharma S.K. (2008) Estimation of cutting forces and surface roughness for hard turning using neural networks, *Journal of Intelligent Manufacturing* 19:473-483. <https://doi.org/10.1007/s10845-008-0097-1>
13. Karayel D. (2009) Prediction and control of surface roughness in CNC lathe using artificial neural network, *Journal of Materials Processing Technology* 209:3125-3137. <https://doi.org/10.1016/j.jmatprotec.2008.07.023>
14. Graupe D. (2013) *Principles of Artificial Neural Networks*, 3rd Ed, Advanced Series in Circuits and Systems, World Scientific Publishing Co. Pte. Ltd., Singapore.

Tables

Table 1. All cases for combinations of four parameters

Case	1	2	3	4	5	6	7
Combinations	GT	GT	GT	FR	GT	FR	WR
	FR	FR	WR	WR			
	WR						

GT(gap voltage) □ FR(feed rate) □ WR(water resist)

Table 2. Testing accuracies of all cases

Case	1	2	3	4	5	6	7	8	9	10	11
Hidden layers number	1	1	1	1	1	1	1	1	1	1	1
Neuron number	5	10	20	30	40	50	60	70	80	90	100
Testing accuracy (%)	74.3	79.3	79.3	77.3	77.3	84.6	85.3	79.6	83.6	80	79.3
Case	14	21	31	41	51	61	71	81	91	101	
Hidden layers number	1	1	1	1	1	1	1	1	1	1	
Neuron number	130	200	300	400	500	600	700	800	900	1000	
Testing accuracy (%)	85.6	82	80.6	79.6	82.3	78.3	75.6	80.6	78	78.3	

Table 3. Accuracies of all cases for combinations of three parameters

Case	1	2	3	4	5	6	7
Combinations	GT	GT	GT	FR	GT	FR	WR
	FR	FR	WR	WR			
	WR						
Medium Testing Accuracy	79.71%	81.76%	76.84%	70.63%	74.38%	68.57%	57.37%

Figures

	1	2	3	4	5	6	7
	DATA_TIME	GAP	FEED	WATER_RESIST	CUT_TM	WireState	RuptureChance
23262	1.5434e+09	70.2794	0.6012	51	656698	1	0
23263	1.5434e+09	70.2794	0.6410	51	656699	1	0
23264	1.5434e+09	70.2794	0.6154	51	656700	1	0
23265	1.5434e+09	70.2794	0.6344	51	656701	1	0
23266	1.5434e+09	70.1996	0.6437	51	656702	1	0
23267	1.5434e+09	70.1996	0.6164	51	656703	1	0
23268	1.5434e+09	70.1617	0.6198	51	656704	1	0
23269	1.5434e+09	70.1617	0.6142	51	656705	1	0
23270	1.5434e+09	70.3194	0.6142	51	656706	1	0
23271	1.5434e+09	70.3194	0.6298	51	656707	1	0
23272	1.5434e+09	70.4870	0.6275	51	656708	1	0
23273	1.5434e+09	70.4870	0.6436	51	656710	1	0
23274	1.5434e+09	70.2475	0.6527	49	656711	1	0
23275	1.5434e+09	70.4910	0.6486	49	656712	1	0
23276	1.5434e+09	70.4910	0.6449	49	656713	1	0
23277	1.5434e+09	70.3693	0.6367	49	656714	1	0
23278	1.5434e+09	70.3693	0.6310	49	656715	1	0
23279	1.5434e+09	70.2695	0.6125	49	656716	1	0
23280	1.5434e+09	70.2695	0.6204	49	656717	0	10
23281	1.5434e+09	70.4571	0.6360	49	656718	0	20
23282	1.5434e+09	70.4571	0.6235	49	656720	0	30
23283	1.5434e+09	70.4571	0.6358	49	656721	0	40
23284	1.5434e+09	70.4571	0.6230	50	656722	0	50
23285	1.5434e+09	70.2994	0.6518	50	656723	0	60
23286	1.5434e+09	70.2994	0.6149	50	656724	0	70
23287	1.5434e+09	70.2475	0.6461	50	656725	0	80
23288	1.5434e+09	70.5010	0.6782	50	656726	0	90
23289	1.5434e+09	70.5010	0.6586	51	656727	0	100
23290	1.5434e+09	55.1337	-0.5653	51	656728	2	200
23291	1.5434e+09	55.1337	0	51	656728	0	0
23292	1.5434e+09	0	0	51	656728	0	0

Figure 1

A part of the processed data

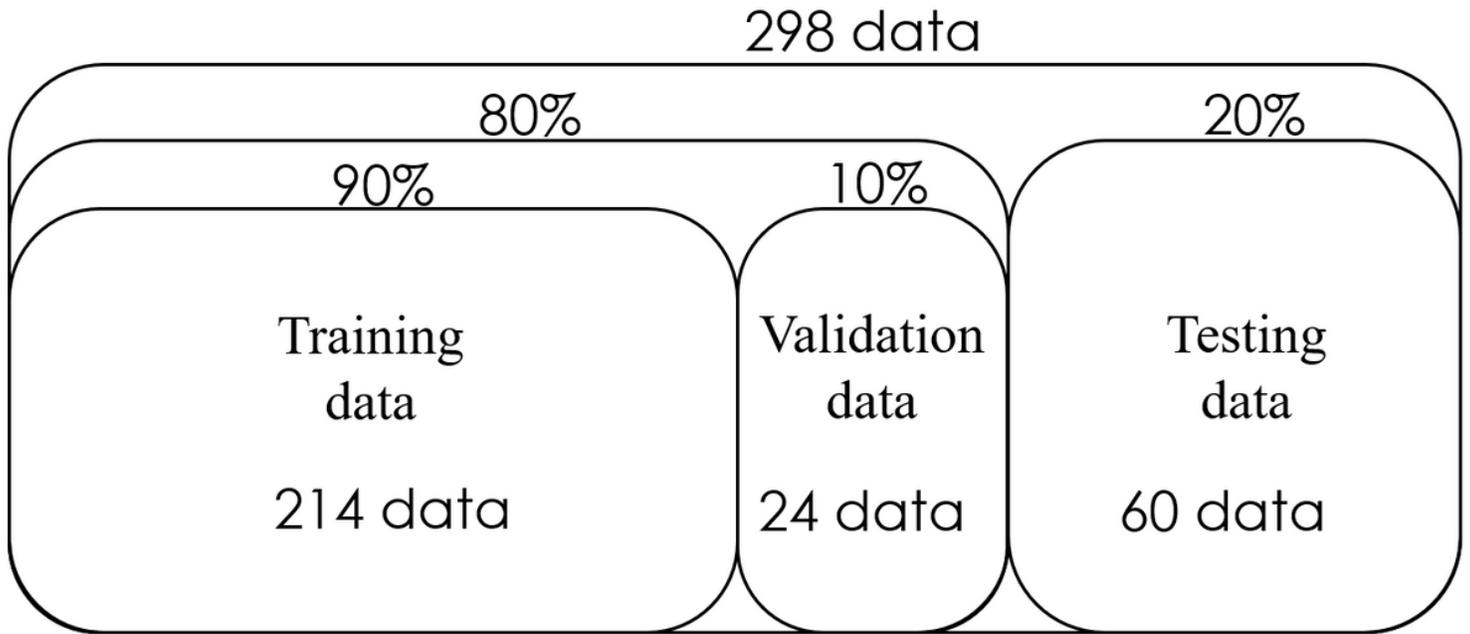


Figure 2

Splitting of the data

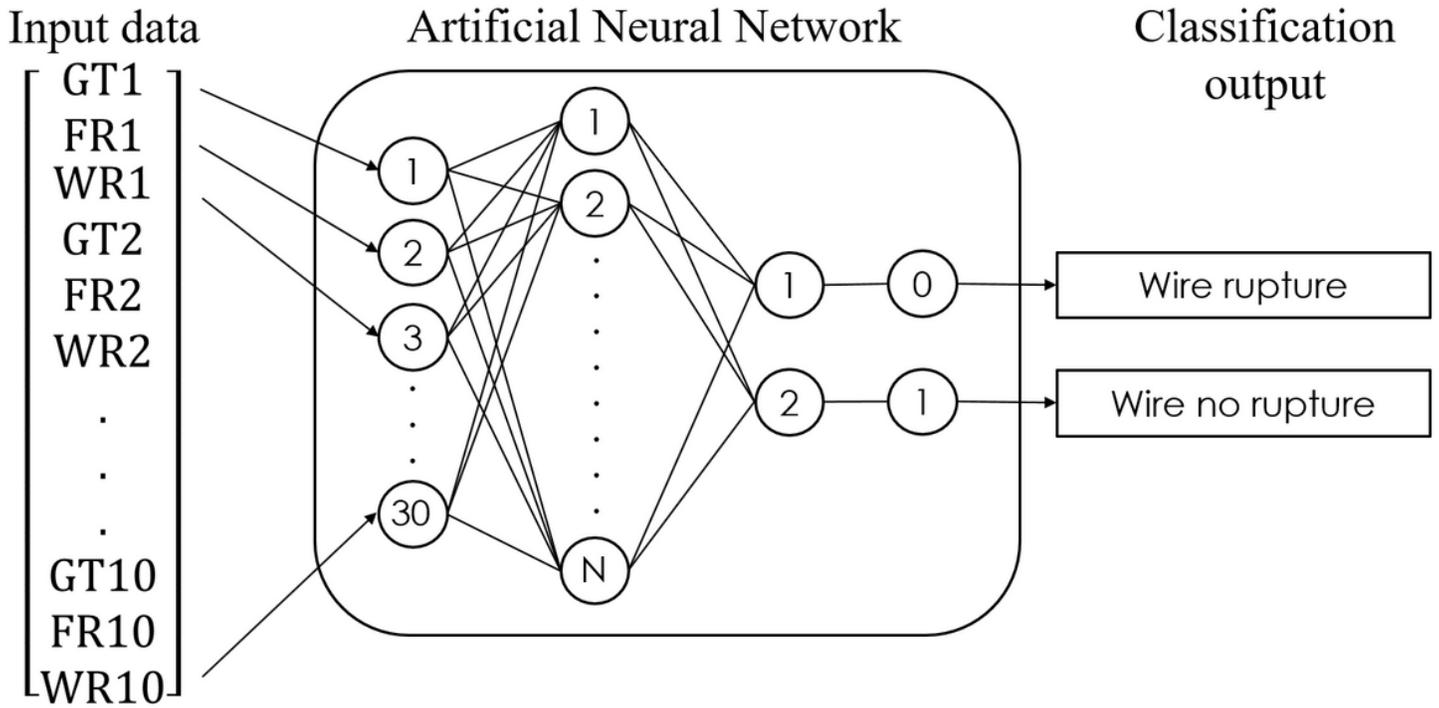


Figure 3

The ANN for classification

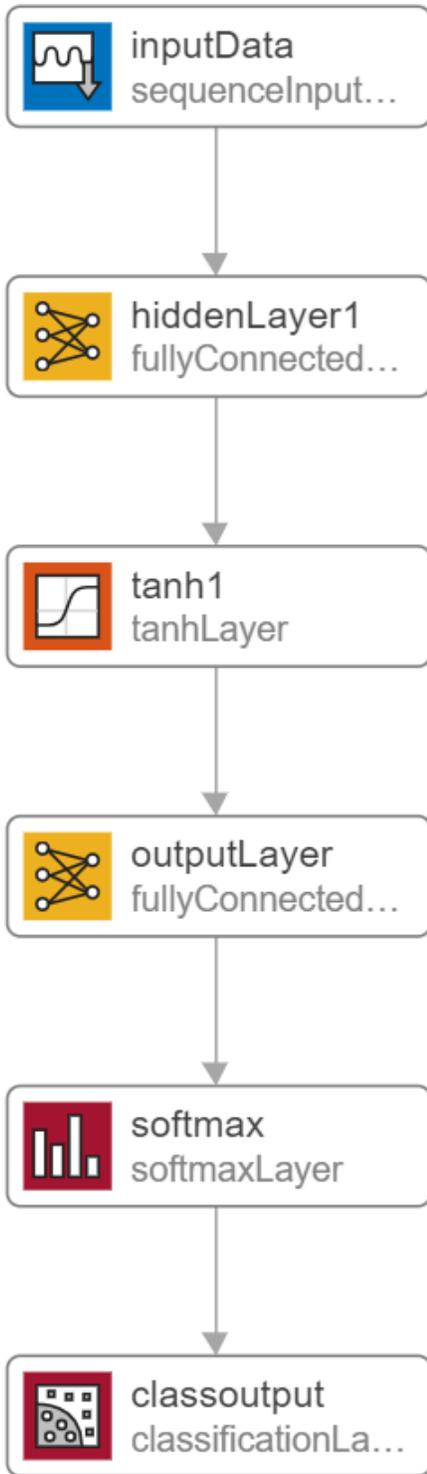


Figure 4

The ANN for classification in MATLAB

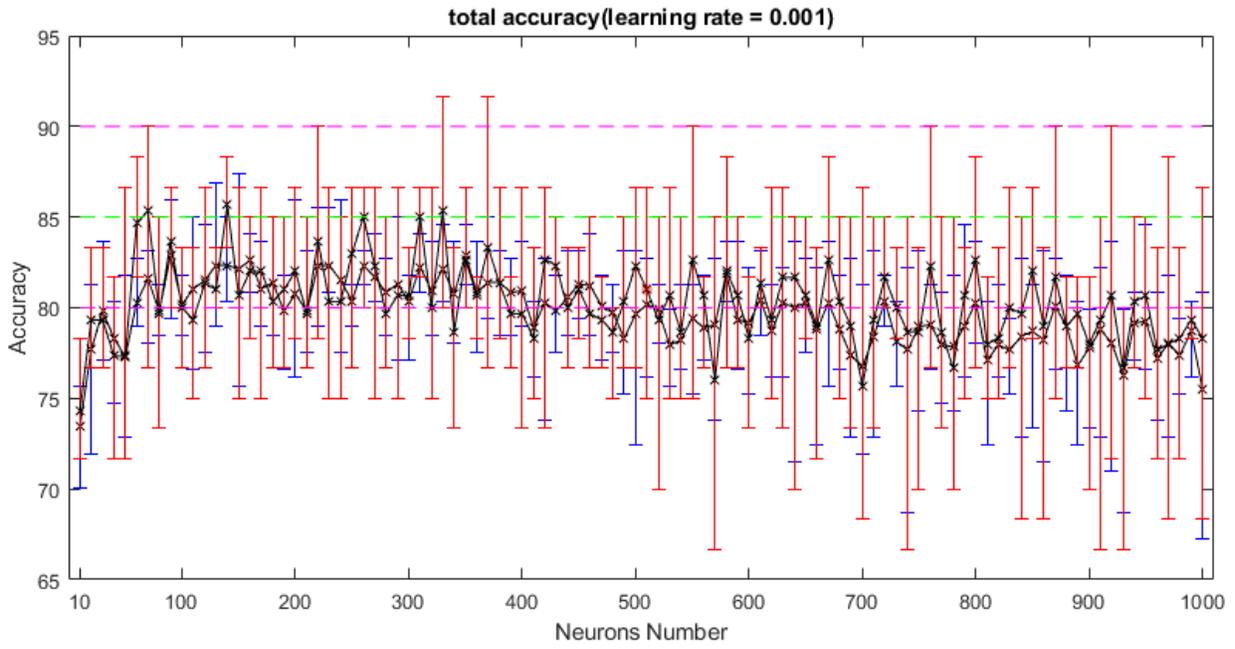


Figure 5

The comparison for training and testing accuracies (learning rate = 10^{-3})

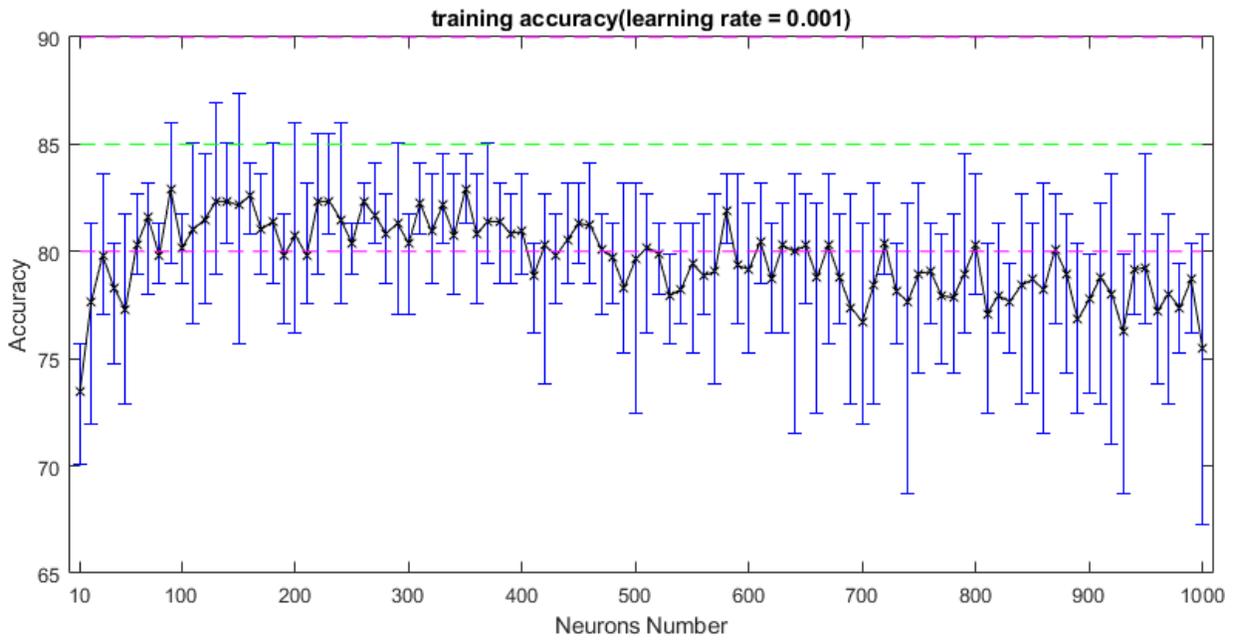


Figure 6

The training accuracy of all architecture (learning rate = 10^{-3})

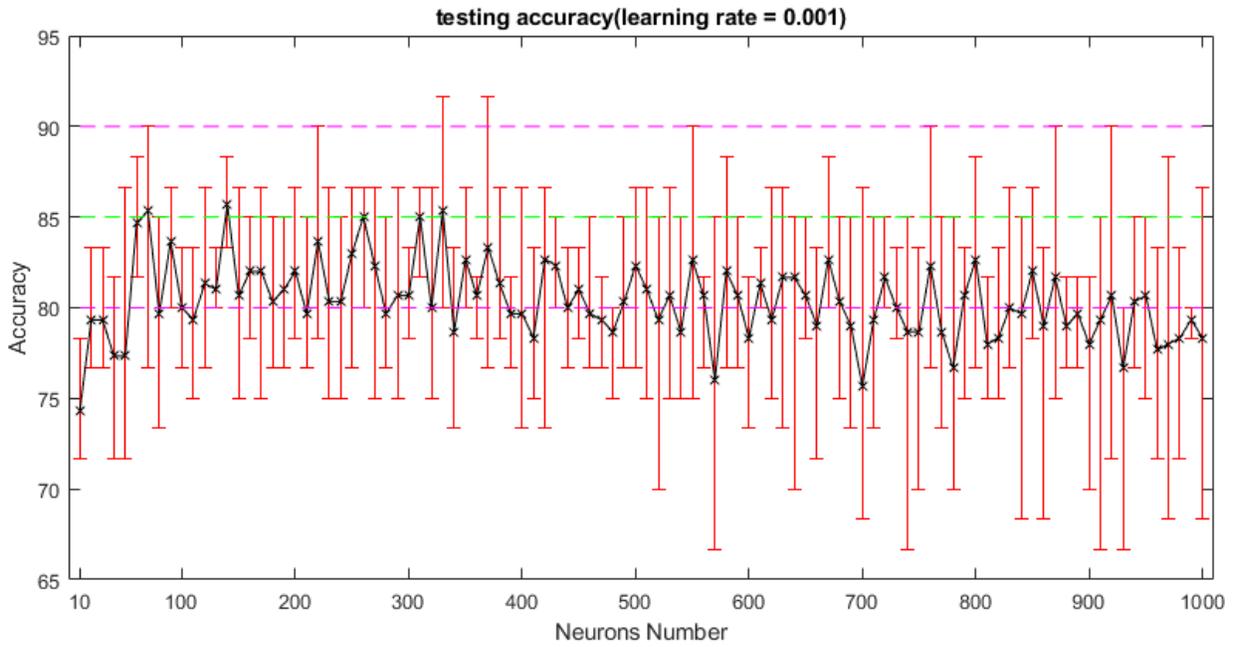


Figure 7

The testing accuracy of all cases (learning rate = 10^{-3})

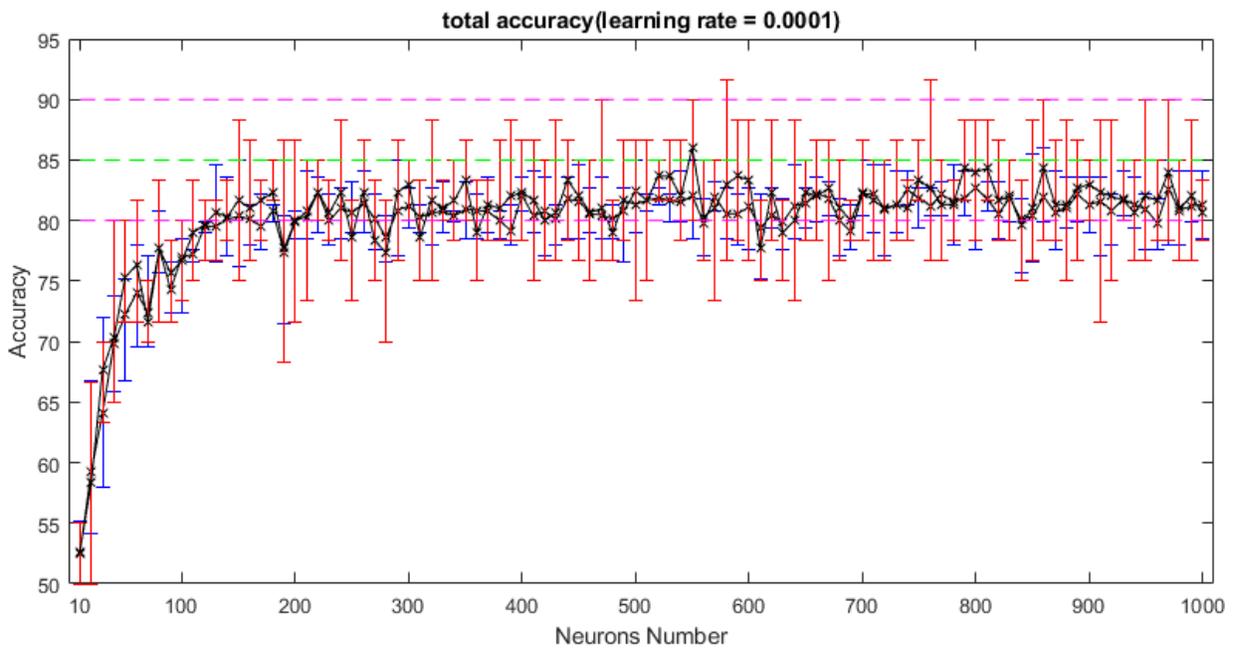


Figure 8

The comparison for training and testing accuracies (learning rate = 10^{-4})

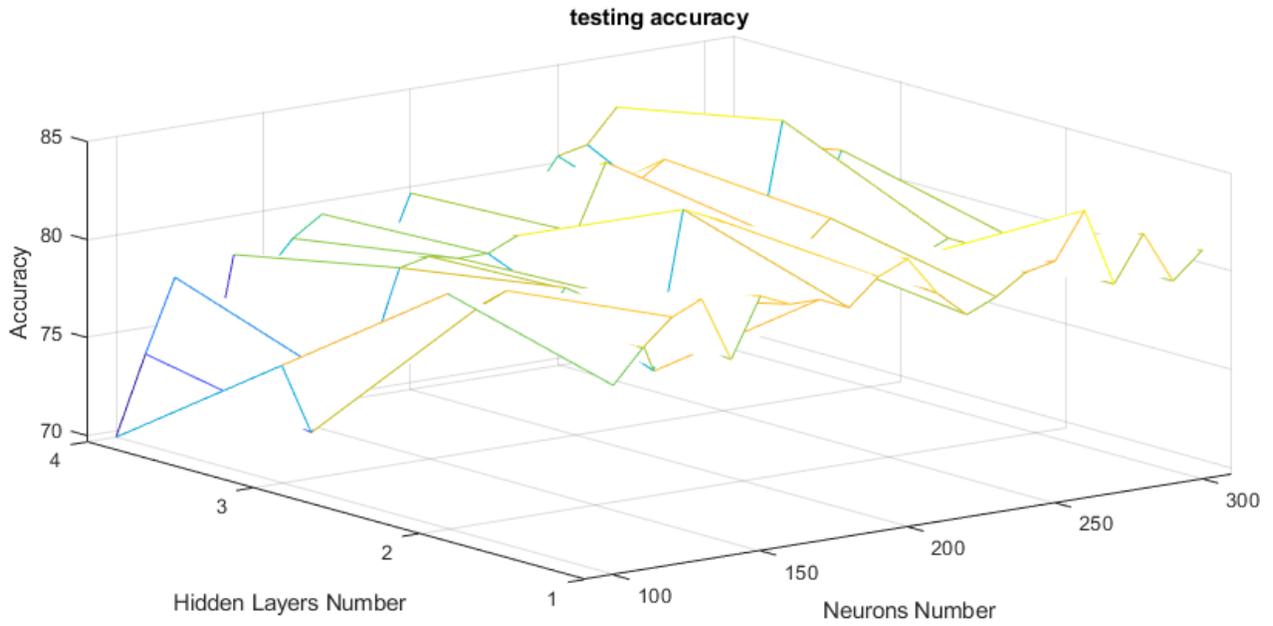


Figure 9

The combination with 1 to 4 hidden layers and 100 to 300 neuron numbers.

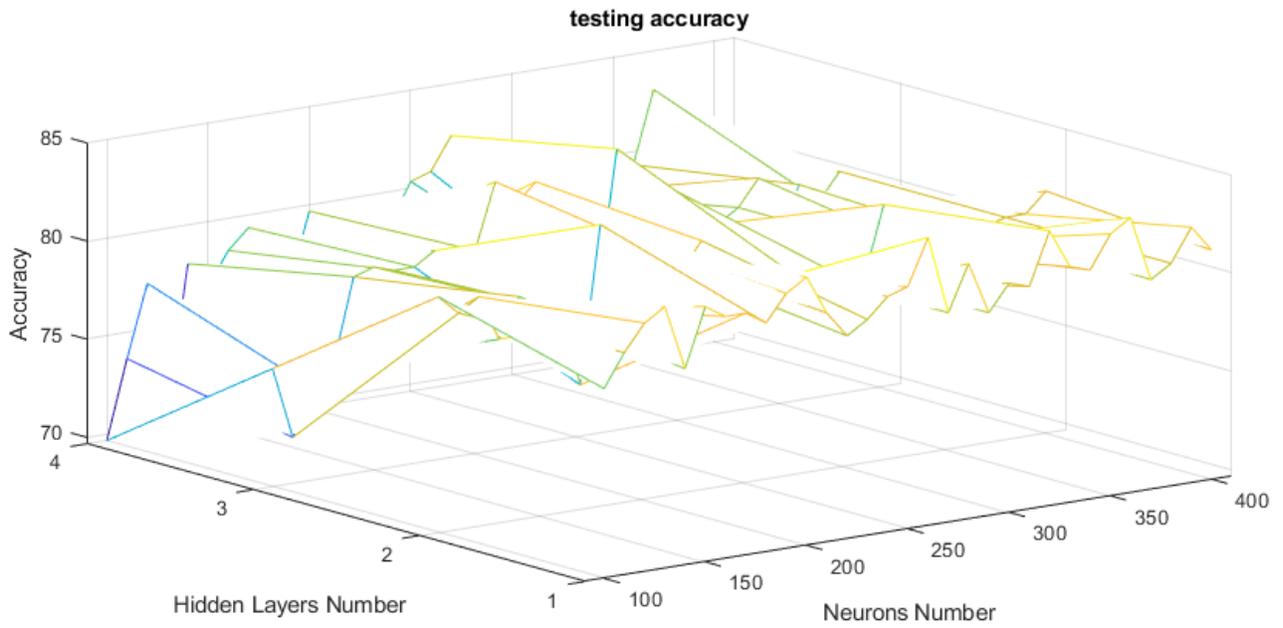


Figure 10

The combination with 1 to 4 hidden layers and 100 to 400 neuron number

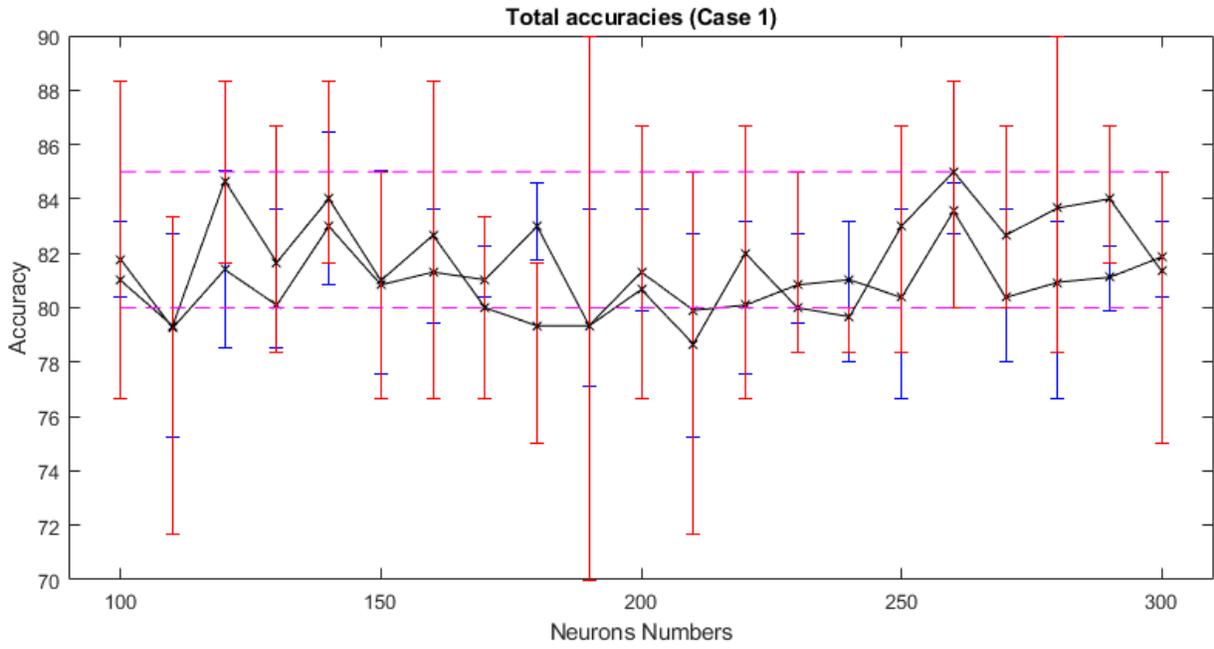


Figure 11

The comparison for training and testing accuracies in case 1 (GT+FR+WR)

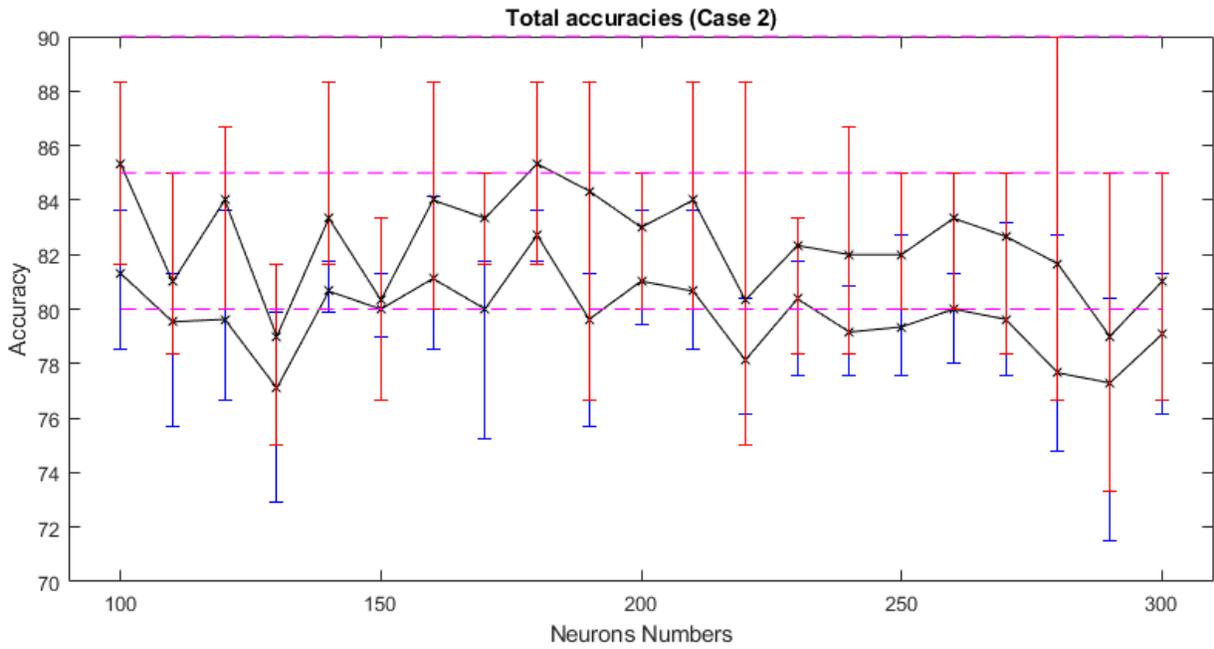


Figure 12

The comparison for training and testing accuracies in case 2 (GT+FR)

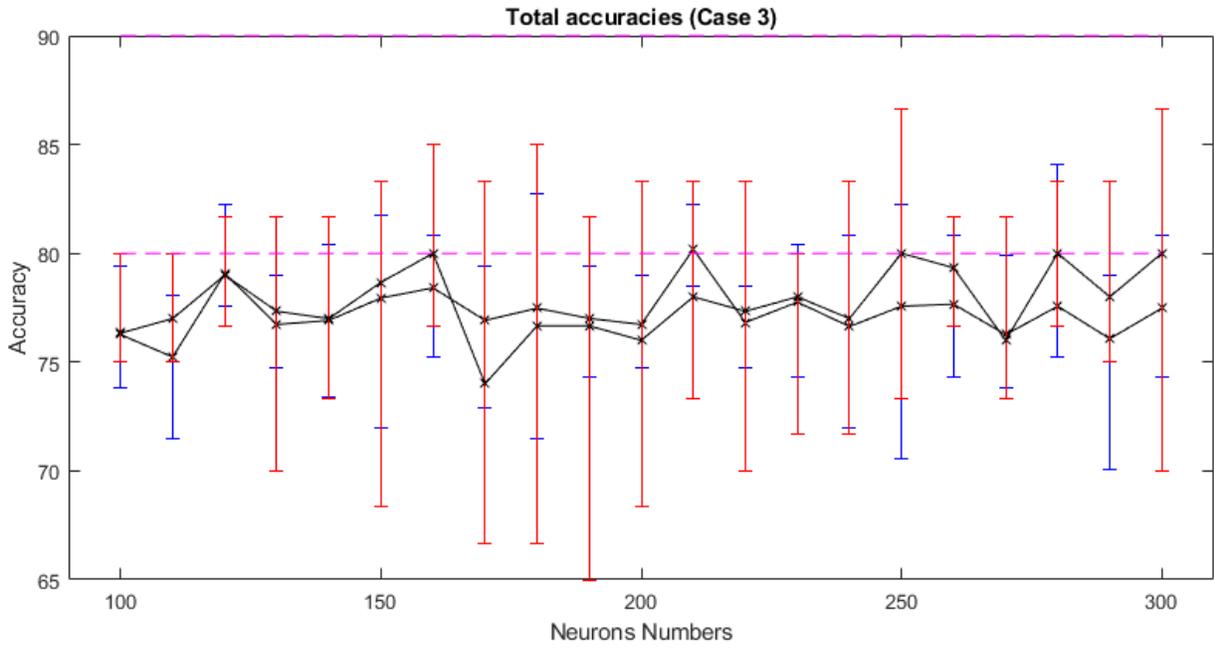


Figure 13

The comparison for training and testing accuracies in case 3 (GT+WR)

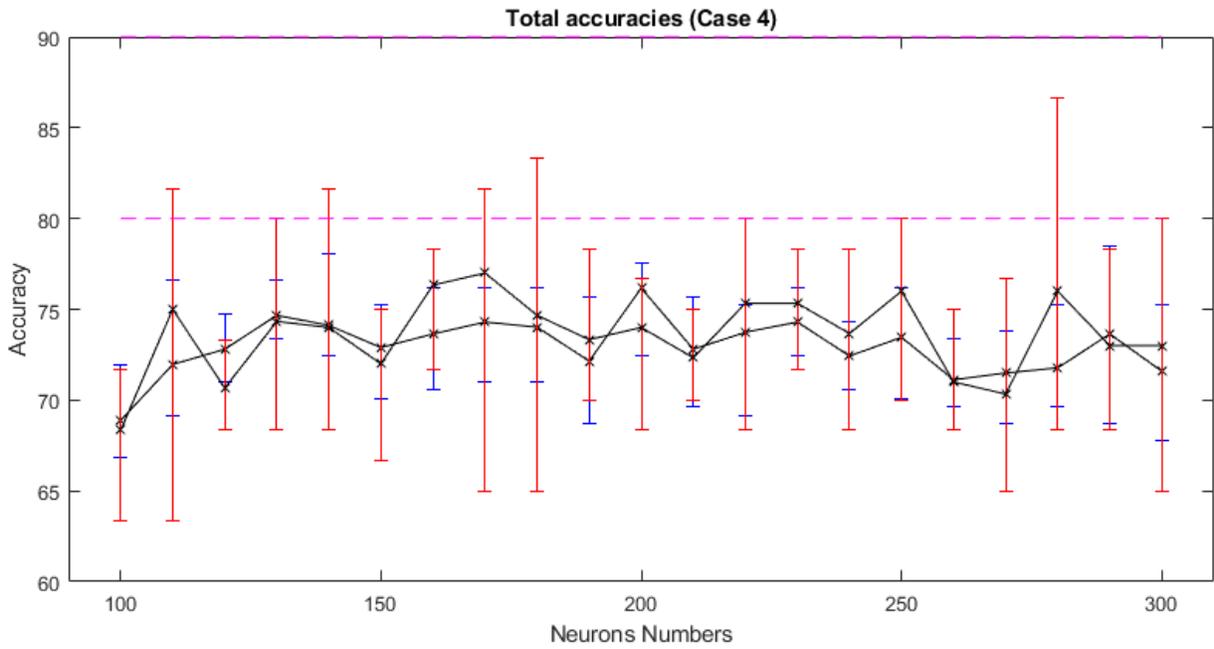


Figure 14

The comparison for training and testing accuracies in case 4 (FR+WR)

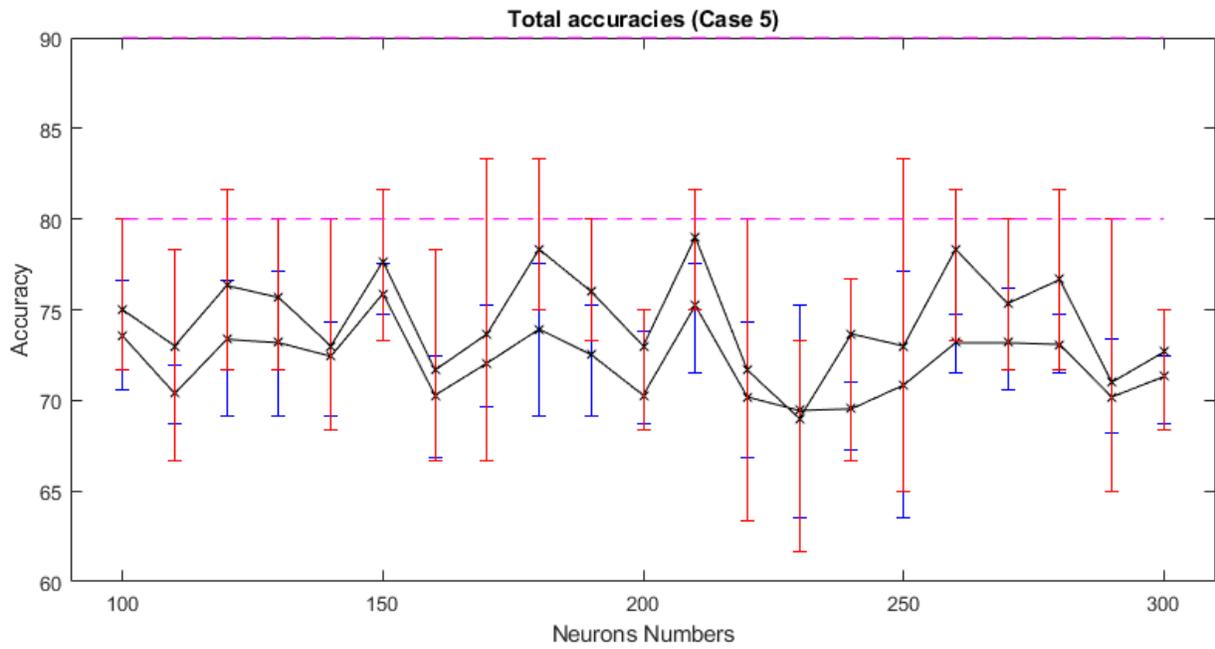


Figure 15

The comparison for training and testing accuracies in case 5 (GT)

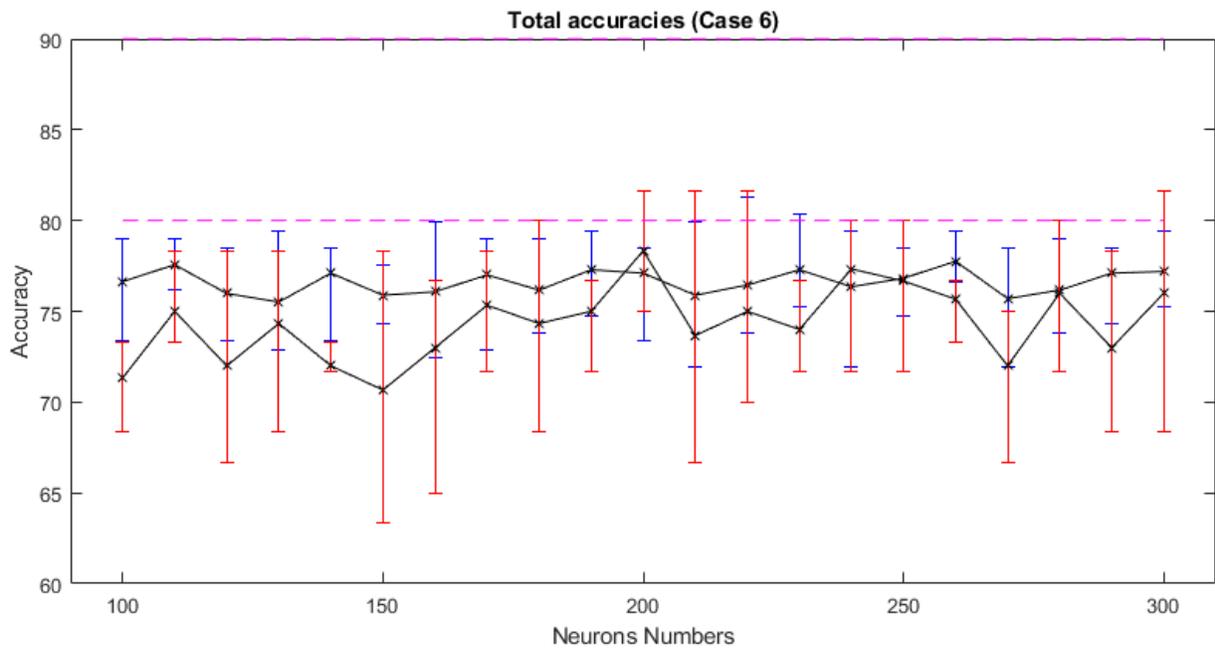


Figure 16

The comparison for training and testing accuracies in case 6 (FR)

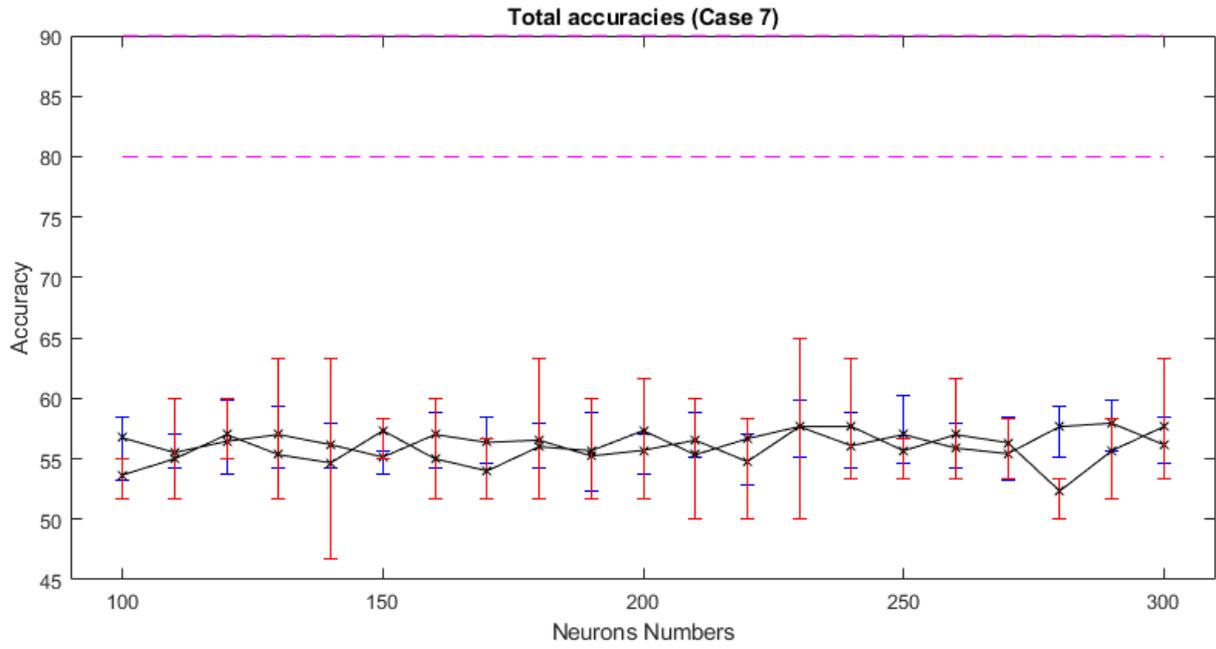


Figure 17

The comparison for training and testing accuracies in case 7 (WR)