Prediction of pilling of polyester–cotton blended woven fabric using artificial neural network models

Qi Xiao¹,², Rui Wang¹, Shujie Zhang¹, Danyang Li¹, Hongyu Sun³ and Limin Wang⁴

Abstract
In this article, an intelligent pilling prediction model using back-propagation neural network model and an optimized model with genetic algorithm is introduced. Genetic algorithm is proposed in consideration of the initial weight and threshold of back-propagation artificial neural network, and further improves training speed and the accuracy for prediction pilling of polyester–cotton blended woven fabrics. The results show that the maximum numbers of training steps of the optimized model by genetic algorithm are reduced from 164 steps to 137 steps compared with that of back-propagation model. The training fitness of optimized model by genetic algorithm is improved from 0.914 to 0.945. The simulation fitness is increased from 0.912 to 0.987. And the root mean square error decreased from 1.0431 to 0.6842. The optimized model by genetic algorithm shows a better agreement between the experimental and predicted values.

Keywords
Genetic algorithm, back propagation, artificial neural network, polyester–cotton blended woven fabric, pilling

Date received: 6 May 2019; accepted: 18 December 2019

Introduction
The pilling of fabrics is a classic problem for the apparel industries, especially for polyester blended fabrics.¹ Polyester–cotton blended woven fabrics which have crisp, wrinkle resistance, dimensional stability and moisture absorption properties have been developed and are more and more popular among people. However, polyester–cotton woven fabrics also have some disadvantages, such as easy pilling, which affect the appearance and handle of fabrics. It is realized that the problem of pilling is one of the biggest quality issues for polyester–cotton blended fabrics. But the problem remains.

Many factors have been identified to affect fabric pilling, which include fiber properties (type,² modulus,³ friction coefficient⁴), yarn properties (spinning method,⁵ twist,⁶ yarn count⁷) and fabric properties (structure,⁸ cover factor⁹), whose synergism determines the pilling propensity. Many researchers have investigated the influence of numerous factors contributing to the pilling of fabrics. Beltran et al.¹⁰ found that the larger the fabric...
cover factor is, the smaller the pilling tendency of the fabric is. The larger the yarn count is, the easier the fabric is prone to pilling. The effect of fabric thickness on pilling is first increased and then decreased. Li and Zhou\textsuperscript{11} found that yarn twist has a great influence on the pilling performance of wool knit fabrics. Lohrasbi et al.\textsuperscript{12} found that the warp and weft density of fabrics has a great influence on the pilling performance of wool fabrics. Kayseri et al.\textsuperscript{13} used nonlinear regression analysis to study the effects of yarn twist, twist ratio, fabric density on pilling performance of fabrics and found that the effect of these factors was particularly significant. It can be seen from many studies that the factors such as fabric twist, twist ratio, warp and weft density, and fabric cover factor have important effects on pilling performance of fabrics. However, they did not consider the complex interactions of the numerous factors, and their synergistic effects on the pilling propensity cannot be fully understood.

The advent of artificial neural networks (ANNs) provides the opportunity to model the interactions among the numerous factors to predict the pilling propensity of fabrics. Back-propagation (BP) ANN which has the advantages of high nonlinearity, self-learning, and mapping is a very important and classical ANN.\textsuperscript{14} It does not need to seek explicit relations and mathematical models between nonlinear sample data and can accurately approximate the best function of characterizing the training sample data to overcome many limitations and difficulties of the traditional methods. Furferi et al.\textsuperscript{15} said that BP neural networks have been widely used in textile field since last decades, such as predicting drapability property,\textsuperscript{16} air permeability,\textsuperscript{17} wrinkle recovery,\textsuperscript{18} dimensional stability,\textsuperscript{19} thermal resistance,\textsuperscript{20} UV (ultraviolet) protection,\textsuperscript{21} and so on. There are also many studies on the application of BP model to the pilling propensity of wool fabrics\textsuperscript{10} and cotton fabrics.\textsuperscript{6} However, there is so far no research on predicting the pilling performance of polyester–cotton blended fabrics by using BP model. In practical applications, BP model has disadvantages, which include slow convergence rate and local minimum value.\textsuperscript{22} Using genetic algorithm (GA) to optimize the initial weight and threshold of BP model can improve the training speed of neural network and overcome local minimum. Zhang et al.\textsuperscript{23} used GA to detect and classify color-textured fabrics defects, and found that it was more feasible and applicable.

The aim of this study was to use BP model and optimized model with GA for predicting the pilling properties of polyester–cotton fabrics. Six parameters of fabrics were used as input variables. BP model and optimized model with GA were used to correlate six parameters with the number of pills. In order to validate the two models, the predicted values were then compared with the experimental values.

### Materials and methods

**Experimental materials**

Selecting the appropriate input and output variables is the key to predict the pilling propensity of fabrics. There are many factors affecting the pilling performance of polyester–cotton fabrics. If each of the influencing factors is used as the input variable of the model, the model will become complicated, and the increase of variables will also increase the difficulty of sample data collection. Accurately selecting the factors associated with pilling as input variables will directly affect the accuracy of the predictive model. From previous work, yarn twist, twist ratio, warp density, weft density, fabric tightness, and fabric thickness are the most effective factors. Therefore, these six factors are used as the input variables of the model, and the number of pills on the surface of fabrics after pilling test is taken as the output variable.

**Samples.** The polyester–cotton blended fabrics where the content ratios of polyester fibers and cotton fibers are 65\% and 35\% (T65/C35) are widely used and representative. So this blended ratio is used in the experiment. The diameters of polyester fibers and cotton fibers are 16.5 and 23.4 µm, respectively. The lengths of polyester fibers and cotton fibers are 3.35 and 2.76 cm, respectively. A total of 30 fabric samples are selected randomly from a total of 100 samples that have been chosen for the entire experiment. And these samples come from the company. The sample specifications and fabric structure parameters are shown below in Table 1.

**Measuring pilling properties of fabrics.** According to Martin-dale method of ISO 12945-2:2000, the polyester–cotton fabrics pilling test was conducted by YG (B) 401E Martin-dale wear-resistant instrument which was produced by Dalong Textile Instrument Co., Ltd. The degree of pilling was characterized by the number of pills on the surface of the polyester–cotton woven fabrics after 5000 pilling rubs. The specific results are shown in Table 1.

In order to avoid network paralysis caused by excessive raw data before prediction by using BP algorithm, the original data should be normalized. The original data in Table 1 should be standardized between \([-1, 1]\), so that the data can be smoothed as much as possible to eliminate the prediction result. The standardized data can be used as the training sample of the model.

**Analytical methods**

**The construction of BP model.** BP model belongs to multilayer forward neural network. Increasing the number of network layers can make the result more accurate and reduce the error, but it will make the calculation too large and the training process is difficult to fit. Therefore, the classical
A three-layer structure BP neural network is adopted, which consists of input, hidden, and output layers.

There are training and testing parts in BP model. The 1–25 sets of data in Table 1 are used for training of the model, and 26–30 sets of data are considered as testing of the model. Six physical factors shown in Table 1 are regarded as the training input vector. And the number of pills is considered as the learning target data. Therefore, the number of input nodes of training is 6, the number of nodes of the output layer is 1, and the number of nodes of the hidden layer is calculated according to equation (1)

\[ N = \sqrt{m + n + a} \]  

where \( N \) represents the number of hidden layer nodes, \( m \) represents the number of input nodes, \( n \) represents the number of output nodes, and \( a \) represents a constant between [1, 10]. The best hidden layer nodes are selected from the value range. The same samples with different node numbers should be trained to make the errors minimized. The error is evaluated by using mean absolute percent error (MAPE), root mean square error (RMSE), and mean absolute error (MAE). The specific calculations are as shown in equations (2)–(4)

\[ \text{MAPE} = \frac{1}{k} \sum_{i=1}^{k} \frac{|A(i) - B(i)|}{B(i)} \]  

where \( \text{abs} \) represents the absolute value, \( A(i) \) represents the model output value, \( B(i) \) represents the experimental value, and \( k \) represents the number of samples

\[ \text{RMSE} = \sqrt{\frac{\sum_{i=1}^{k} (A(i) - B(i))^2}{k}} \]  

where \( k \) represents the number of samples, \( A(i) \) represents the output layer prediction value, and \( B(i) \) represents the experimental value

| No. of samples | Yarn twist | Twist ratio | Warp density | Weft density | Fabric tightness | Fabric thickness | No. of pills |
|----------------|------------|-------------|--------------|--------------|-----------------|-----------------|-------------|
| 1              | 571        | 0.8         | 348          | 161          | 115             | 0.58            | 14.5        |
| 2              | 878        | 0.1         | 119          | 56           | 67              | 0.44            | 20.9        |
| 3              | 625        | 0.3         | 383          | 256          | 85              | 0.67            | 3.0         |
| 4              | 867        | 0.9         | 124          | 58           | 69              | 0.44            | 21.1        |
| 5              | 591        | 0.5         | 250          | 198          | 101             | 0.51            | 19.1        |
| 6              | 584        | 0.4         | 172          | 262          | 104             | 0.42            | 26.8        |
| 7              | 1011       | 0.4         | 364          | 173          | 113             | 0.60            | 12.6        |
| 8              | 721        | 1.0         | 162          | 77           | 84              | 0.45            | 22.0        |
| 9              | 1131       | 1.1         | 365          | 174          | 113             | 0.60            | 12.4        |
| 10             | 578        | 1.0         | 172          | 82           | 88              | 0.45            | 22.1        |
| 11             | 842        | 0.6         | 336          | 160          | 114             | 0.57            | 15.3        |
| 12             | 1081       | 0.2         | 256          | 122          | 108             | 0.50            | 20.5        |
| 13             | 554        | 0.7         | 383          | 182          | 112             | 0.62            | 10.5        |
| 14             | 887        | 0.8         | 398          | 190          | 110             | 0.64            | 8.6         |
| 15             | 729        | 1.2         | 321          | 153          | 114             | 0.55            | 16.6        |
| 16             | 730        | 0.7         | 187          | 89           | 92              | 0.46            | 22.1        |
| 17             | 966        | 1.0         | 139          | 66           | 75              | 0.45            | 21.5        |
| 18             | 814        | 0.2         | 286          | 136          | 112             | 0.52            | 19.0        |
| 19             | 740        | 0.3         | 410          | 195          | 108             | 0.65            | 7.2         |
| 20             | 980        | 0.3         | 410          | 195          | 108             | 0.65            | 7.2         |
| 21             | 840        | 0.4         | 153          | 73           | 81              | 0.45            | 21.8        |
| 22             | 1123       | 1.1         | 152          | 72           | 81              | 0.45            | 21.8        |
| 23             | 779        | 0.3         | 125          | 60           | 70              | 0.44            | 21.1        |
| 24             | 1124       | 0.3         | 291          | 139          | 113             | 0.52            | 18.7        |
| 25             | 868        | 0.4         | 265          | 126          | 110             | 0.50            | 20.1        |
| 26             | 1031       | 0.2         | 186          | 88           | 92              | 0.46            | 19.1        |
| 27             | 1147       | 0.2         | 362          | 173          | 114             | 0.59            | 12.7        |
| 28             | 1062       | 0.7         | 375          | 179          | 113             | 0.61            | 11.4        |
| 29             | 1096       | 1.0         | 303          | 144          | 113             | 0.53            | 17.9        |
| 30             | 683        | 0.2         | 105          | 50           | 61              | 0.44            | 20.4        |
MAE = $\frac{1}{k} \sum_{i=1}^{N} \text{abs} \left( \frac{A(i) - B(i)}{k} \right)$  

where abs represents the absolute value, $A(i)$ represents the output layer prediction value, $B(i)$ represents the experimental value, and $k$ represents the number of samples.

The construction of optimized model with GA. GA is an adaptive search procedure derived from the theories of natural population evolution. BP model optimized by GA is to use GA to optimize the initial weight and threshold of BP neural network, which can more accurately predict the output results.

Population initialization. The individual coding method is real number coding. Each individual is a real number string, which is composed of the connection weight of the input layer and the hidden layer, the threshold of the hidden layer, the connection weight of the hidden layer and the output layer, and the threshold of the output layer. The individual contains all the weights and thresholds of the neural network. When the network structure is known, it can form a neural network with a determinate structure, weight, and threshold.

Fitness function. According to the initial weight and threshold of BP model, the training data are used to train BP neural network and to predict the output of the system. The absolute value of errors between the predicted output and the expected output is taken as the individual fitness value. The equation is as follows

$$F = k \left( \sum_{i=1}^{N} \text{abs} \left( y_i - o_i \right) \right)$$

where $n$ is the output node of network, $y_i$ is the expected output of node $i$ of BP neural network, $o_i$ is the predicted output of node $i$, and $k$ is the coefficient.

Selecting. The GA selection has many methods such as roulette method and tournament method. The roulette method is adopted in this study, which is the selection strategy based on the fitness ratio. The selection probability $p_i$ of each individual $i$ is shown in equations (6) and (7)

$$f_i = \frac{k}{F_i}$$

$$p_i = \frac{f_i}{\sum_{j=1}^{N} f_j}$$

Crossover. Since real number coding is used for individual, the real number intersection method is adopted for crossover. The crossover method of chromosome $a_j$ and chromosome $a_j$ in the $j$ position is shown in equations (8) and (9)

$$a_{ij} = a_{ij} (1-b) + a_{ij} b$$

$$a_{ij} = a_{ij} (1-b) + a_{ij} b$$

where $b$ is a random number between [0, 1].

Mutation. The gene $a_{ij}$ of individual $i$ is selected for mutation. The mutation method is shown in equations (10) and (11)

$$a_{ij} = \begin{cases} a_{ij} + (a_{ij} - a_{\text{min}}) f(g), & r > 0.5 \\ a_{ij} + (a_{\text{max}} - a_{ij}) f(g), & r \leq 0.5 \end{cases}$$

$$f(g) = r_2 \left( \frac{1-g}{G_{\text{max}}} \right)^2$$

where $a_{\text{max}}$ is the upper bound of the gene $a_{ij}$, $a_{\text{min}}$ is the lower bound of the gene $a_{ij}$, $r_2$ is a random number, $g$ is the current number of iterations, $G_{\text{max}}$ is the maximum number of evolution, $r$ is a random number between [0,1].

Results and discussion

Analysis of BP model

Determination of the number of hidden layer nodes. In this study, the Neural Network Toolbox of MATLAB (R2016a) mathematical software was adopted. A three-layer BP neural network is selected, which includes an input layer, a hidden layer, and an output layer. According to equation (1), the value range of the hidden layer node in this study is \([4, 13]\).

As a result, the errors with different number of hidden layer nodes are shown in Table 2. It is seen from Table 2 that the best number of hidden layer nodes is 9 when the error is determined to be the smallest. This is mainly because if the number of hidden layer nodes is too small, it is not enough to reflect the law of data. The error value will fluctuate. Just like for nodes 7 and 8, the error values rise. However, if the number of nodes is too large, the network learning time will increase. And an over-fitting phenomenon may occur, which may also cause a large error. Therefore, the selection of the number of hidden layer nodes should be moderate.
Determination of input and output layer transfer function. The transfer functions of the input and output layers are chosen to make the predictions accurate. It denoted that MAPE, RMSE, and MAE must be minimized.

In the case where the network structure and the weights and thresholds are the same, the relationships between MAPE, RMSE, and MAE and the transfer function of the hidden layer and the output layer are shown in Table 3.

It can be seen from Table 3 that the selection of transfer function of hidden and output layer has a great influence on the prediction accuracy of BP model. The transfer functions of the hidden layer and the output layer, in which errors are the smallest, are logsig and logsig, respectively: the logsig transfer function is an S type of logarithmic function, which is consistent with the law of pilling, and the tansig transfer function is an S type of tangent function. However, purelin transfer function is a linear function, which does not conform to the law of pilling.

Determination of training parameters. According to the above-selected parameters, the settings of the relevant training parameters are shown in Table 4.

According to the above training parameters, the BP network is trained. The performance function that comes with the neural network toolbox is the mean square error function. So the Y-axis of the error curve of the training is mean squared error, which is shown in Figure 1. As can be seen from Figure 1, the training step of BP neural network is 164. It means the network reaches the target value at this step.

Analysis of optimized model with GA

A global search of the weights and thresholds of neural network is performed by using GA to determine its optimal solution. The local optimal solution is searched by using BP neural network. Then, the neural network is trained. The specific algorithm flowchart is shown in Figure 2.

Table 2. Errors of different hidden layer nodes.

| Hidden layer node | MAPE | RMSE | MAE |
|-------------------|------|------|-----|
| 4                 | 0.019| 1.633| 1.613|
| 5                 | 0.018| 1.707| 1.678|
| 6                 | 0.018| 1.489| 1.448|
| 7                 | 0.040| 6.365| 3.436|
| 8                 | 0.035| 4.404| 2.934|
| 9                 | 0.010| 0.511| 0.864|
| 10                | 0.016| 1.165| 1.292|
| 11                | 0.018| 1.430| 1.466|
| 12                | 0.021| 1.856| 1.850|
| 13                | 0.023| 3.327| 2.257|

MAPE: mean absolute percent error; RMSE: root mean square error; MAE: mean absolute error.

Table 3. Errors of different transfer functions.

| Hidden layer function | Output layer function | MAPE | RMSE | MAE |
|-----------------------|-----------------------|------|------|-----|
| logsig                | tansig                | 0.014| 1.009| 1.086|
| logsig                | purelin               | 0.014| 0.809| 1.136|
| logsig                | logsig                | 0.010| 0.511| 0.864|
| tansig                | tansig                | 0.015| 0.911| 1.180|
| tansig                | logsig                | 0.013| 0.822| 1.131|
| tansig                | purelin               | 0.017| 1.272| 1.506|
| purelin               | logsig                | 0.015| 0.959| 1.276|
| purelin               | tansig                | 0.015| 0.877| 1.163|
| purelin               | purelin               | 0.016| 1.182| 1.429|

MAPE: mean absolute percent error; RMSE: root mean square error; MAE: mean absolute error.

Table 4. Settings of training parameters.

| Training parameters                 | Settings                                      |
|-------------------------------------|-----------------------------------------------|
| Transfer function of hidden layer   | logsig                                        |
| Transfer function of output layer   | logsig                                        |
| Training function                   | traingdx                                      |
| Show interval time                  | net.trainParam.show = 200                     |
| Epochs                             | net.trainParam.epochs = 1000                  |
| Goal                               | net.trainParam.goal = 1e-8                    |
| Learning rate                      | net.trainParam.min_lr = 0.01                  |
| Input node                         | 6                                             |
| Hidden node                        | 9                                             |
| Output node                        | 1                                             |

Figure 1. The training error curve of BP neural network.

According to the above-mentioned population initialization and genetic methods, the main parameters of optimized model with GA are shown in Table 5.
According to the training parameters of the above-optimized model, the network is trained. The performance function that comes with the neural network toolbox is the mean square error function. So the Y-axis of the error curve of the training is mean squared error, which is shown in Figure 3. It can be seen from Figure 3 that the training step of optimized model is 137. The comparison between Figures 1 and 3 demonstrates that the optimized model can reach the convergence more quickly than the BP model during the training process. This is because GA can overcome the local minimum of BP model.

When neural work is used for prediction, the closeness between the real measured value and the predicted value of the neural network is generally characterized by the degree of fitness. The BP model and the optimized model with GA are trained by using the same data. The results of the two networks are shown in Figures 4 and 5. It can be seen from Figure 4 that the fitness of BP model is 0.914. It can be seen from Figure 5 that the fitness of optimized model with GA is 0.945. The above analysis shows that the optimized model is more consistent with the predicted value and actual value of pills of polyester–cotton blended woven fabrics.

The simulation results of the two models are analyzed. Figure 6 is the compared results between the measured values and the predicted values of the number of pills. It can be seen from Figure 6 that the predicted values of two models are basically consistent with the measured values. Moreover, the predicted value of the optimized model with GA is closer to the experimental value.
The correlation analysis of the simulation results of the two models are shown in Figures 7 and 8. It can be seen from Figure 7 that the fitness of BP model is 0.912. It can be seen from Figure 8 that the fitness of the optimized model with GA is 0.987. The analysis results indicate that the optimized model with GA is more effective in predicting the pilling property of polyester–cotton blended woven fabrics.

In order to verify the superiority of optimized model with GA, the prediction accuracy of the two models is compared by using three characteristic indexes, which include MAPE, RMSE, and MAE. The results of statistical indicators are shown in Table 6. It can be seen from Table 4 that the prediction error of optimized model with GA is the smallest, that is, the prediction accuracy is the highest. So the predicted effect is better than BP model.

### Conclusion

In this work, the pilling propensity of polyester–cotton woven fabrics was investigated. The BP model and the optimized model with GA were used to correlate the
pilling property with the six parameters. To validate the two models in the training steps, training precision, and simulation precision, 30 samples were tested, and the predicted pilling property was obtained. The predicted values were then compared with the experimental values. The results show that the predicted values are very close to the experimental values, and the values predicted by optimized model with GA are closer than the values predicted by BP model to the experimental values. This shows that optimized model with GA can predict the pilling propensity of polyester–cotton woven fabrics with acceptable accuracy.

Acknowledgements

The authors would like to express thanks to Binzhou Huafang Engineering Technology Research Institute for supplying the samples.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

ORCID iD

Qi Xiao  https://orcid.org/0000-0002-5515-1514

References

1. Chandrasekaran R, Periyasamy S and Shabaridharan K. Statistical analysis of the pilling behaviour of polyester/cotton and polyester/viscose blended woven fabrics. Fibres Text East Eur 2018; 26(3): 66–70.
2. Wan AL, Jiang GM, Yu WD, et al. Fuzzing mechanism and fibre fatigue of wool knit. Indian J Fibre Text 2014; 39(3): 238–243.
3. Hajilari M, Esfandiar AH, Dabiryan H, et al. Investigation of effect of fibres modulus on pilling of acrylic fabrics. Text Res J 2009; 100(2): 135–140.
4. Wan A, Yu W and Jiang G. Pilling properties of wool single jersey made of compact and conventional ring yarns after anti-felting treatment. Text Res J 2014; 84(7): 673–683.
5. Can Y. Pilling performance and abrasion characteristics of plain-weave fabrics made from open-end and ring spun yarns. Fibres Text East Eur 2008; 16(1): 81–84.
6. Kayseri GO and Kirtay E. Part II. Predicting the pilling tendency of the cotton interlock knitted fabrics by artificial neural network. J Eng Fiber Fabr 2015; 10(4): 62–71.
7. Rejali M, Hasani H, Ajeli S, et al. Optimization and prediction of the pilling performance of weft knitted fabrics produced from wool/acrylic blended yarns. Indian J Fibre Text 2014; 39(1): 83–88.
8. Azeem M, Ahmad Z, Wiener J, et al. Influence of weave design and yarn types on mechanical and surface properties of woven fabric. Fibres Text East Eur 2018; 26(1): 42–45.
9. Beltran R, Wang L and Wang X. Measuring the influence of fibre-to-fabric properties on the pilling of wool fabrics. Text Res J 2006; 97(3): 197–204.
10. Beltran R, Wang LJ and Wang XG. Predicting the pilling propensity of fabrics through artificial neural network modeling. Text Res J 2005; 75(7): 557–561.
11. Li L and Zhou W. Analysis on the pilling factors of cashmere knitted fabric. Fiber Polym 2006; 7(2): 213–216.
12. Lohrasbi F, Ghahi JM and Yazdanshenas ME. Influence of weave type and weft density on worsted fabric pilling. Fibres Text East Eur 2011; 19(5): 64–69.
13. Kayseri GO and Kirtay E. Part I. Predicting the pilling tendency of the cotton interlock knitted fabrics by regression analysis. J Eng Fiber Fabr 2015; 10(3): 110–120.
14. Yun SY, Kim S and Park CK. Development of an objective fabric pilling evaluation method. II. Fabric pilling grading using artificial neural network. Fiber Polym 2013; 14(12): 2157–2162.
15. Furfari R, Governi L and Volpe Y. Machine vision-based pilling assessment: a review. J Eng Fiber Fabr 2015; 10(3): 79–93.
16. Taieb AH, Mshali S and Sakli F. Predicting fabric drapability property by using an artificial neural network. J Eng Fiber Fabr 2018; 13(3): 87–93.
17. Baghdadi R, Alibi H, Fayala F, et al. Investigation on air permeability of finished stretch plain knitted fabrics. Fiber Polym 2016; 17(12): 2105–2115.
18. Hussain T, Malik ZA, Arshad Z, et al. Comparison of artificial neural network and adaptive neuro-fuzzy inference system for predicting the wrinkle recovery of woven fabrics. Text Res J 2015; 106(9): 934–938.
19. Kalkanci M, Sinecen M and Kurumer G. Prediction of dimensional change in finished fabric through artificial neural networks. Tekst Konfeksiyon 2018; 28(1): 43–51.
20. Kanat ZE and Ozdil N. Application of artificial neural network (ANN) for the prediction of thermal resistance of knitted fabrics at different moisture content. Text Res J 2018; 109(9): 1247–1253.
21. Majumdar A, Das A, Hatua P, et al. Optimization of woven fabric parameters for ultraviolet radiation protection and comfort using artificial neural network and genetic algorithm. Neural Comput Appl 2016; 27(8): 2567–2576.
22. Khandual A, Luximon A and Pattanayak AK. Optimization of acid cellulose enzyme concentration to reduce pilling of bamboo fabric: an objective assessment approach. Fiber Polym 2011; 12(6): 816–820.
23. Zhang YH, Yuen CWM, Wong WK, et al. An intelligent model for detecting and classifying color-textured fabric defects using genetic algorithms and the Elman neural network. Text Res J 2011; 81(17): 1772–1787.