Prediction of Tractor Repair and Maintenance Costs Using RBF Neural Network

Abbas Rohani

College of Agriculture, Shahrood University of Technology, Shahrood, Iran

Abstract: In this article the potential of Radial Basis Function Neural Network (RBFNN) technique has evaluated as an alternative method for the prediction of tractor repair and maintenance costs. The study was conducted using empirical data on 60 two-wheel drive tractors from Astan Ghodse Razavi agro-industry in Iran. In this paper, the performance of Basic Back-propagation (BB) training algorithm was also compared with Back-propagation with Declining Learning Rate Factor algorithm (BDLRF). It was found that BDLRF has a better performance for the prediction of tractor's costs. It has been concluded that RBFNN represents a promising tool for predicting repair and maintenance costs.

Key words: RBFNN • BDLRF algorithm • Basic Back-propagation • Repair and maintenance cost

INTRODUCTION

To understand why it is important to have an accurate method of predicting tractor's repair and maintenance costs, it is necessary to first have an understanding of what these forecast costs can be used for. To properly analyze economical life, one must be armed with detailed knowledge of the elements and behavior of owning and operating costs. Ownership costs are not too difficult to understand and quantify, instead operating costs are complex and highly intensive in data. If the operating costs stream is properly tracked and analyzed, it can be a reliable input into the economic modeling process. The mathematical models proposed in the literature are very simplistic, old and broad in scope. Moreover, these models are seldom, if ever, used in practice. Regression models were firstly employed for prediction of repair and maintenance costs of farm machinery by American Society of Agricultural Engineers (ASAE) [1], since then it has been continued by others [2-4]. Owing to natural uncertainties associated with repair and maintenance costs of different operations of tractor, exact mathematical relationships between these parameters and tractor age are difficult to be derived. Hence, recourse is normally made to the statistical technique of non-linear regression. Despite this, the resulting equations suffer from approximation and unreliability. An attempt is therefore made in this paper to provide an alternative to these conventional statistics-based methods, by adapting Artificial Neural Network (ANN).

The main advantages of using neural networks are learning directly from examples without attempting to estimate the statistical parameters. More generally, there is no need for firm assumptions about the statistical distributions of the inputs and generating any continuous nonlinear function of input (universal approximating). ANNs are highly parallel which makes them especially amenable to high-performance parallel architectures [5, 6]. Because of these unique characteristics, it also can be employed for prediction of repair and maintenance costs of tractor.

The main objective of this study was to develop tractor's repair and maintenance costs prediction models with readily available data that could be easily applied by a farm manager. The specific objectives were: (1) to investigate the effectiveness of RBFNN for predicting repair and maintenance costs for typical conditions using field-specific costs and historic costs data; (2) study the variation of model performance with different RBFNN model parameters; (3) select optimum RBFNN parameters for accurate prediction of repair and maintenance costs.
MATERIALS AND METHODS

Data Recording: First of all, it has to be assumed that the data were completely and accurately collected by the company (Astan Ghodse Razavi). Historical data (1986–2003) of repair and maintenance costs of tractors was obtained from Astan Ghodse Razavi agro-industry Company in Iran. Records of the repair and maintenance costs, including parts, labor, fuel and oil, were available for 60 two-wheel drive (2WD) tractors, over 18 years.

The available data contain: monthly usage, monthly repair costs (including parts and labor), monthly maintenance costs (including fuel, oil, fuel filter and oil filter), year of purchase and tractor make and model. The data were shuffled and split into two subsets: a training set and a test set. The training set is used to estimate model parameters and the test set is used to check the generalization ability of the model. The training set should be a representative of the whole population of input samples. In this study, the training set and the test set includes 130 patterns (60% of total patterns) and 86 patterns (40% of total patterns), respectively. There is no acceptable generalized rule to determine the size of training data for a suitable training; however, the training sample should cover all spectrums of the data available [7]. The training set can be modified if the performance of the model does not meet the expectations [6].

Tractors Differences: In this study, we are aiming to provide an effective tool for accurately forecasting repair and maintenance costs of tractors. Repair and maintenance costs, as well as initial purchase price, can differ considerably among different models of tractor. Despite this variation, it is essential to create the accumulative repair and maintenance costs of different models of tractor relatively. The convenient way of comparing the repair and maintenance costs of dissimilar tractors is to index them to their initial price. Subsequently, the repair and maintenance costs of different tractors can be compared by means of cumulative cost index [8]. In this regard, the cumulative cost index (CCI) can be calculated as

\[
CCI = \sum_{k=1}^{t} \frac{(P_k + L_k + O_k)}{PP_0}
\]

where, CCI, is the cumulative cost index at time "t", P, and L, are costs of tractor parts and labor at time "t" respectively; O, is the other miscellaneous maintenance costs including fuel, oil, fuel filter and oil filter at time "t" and PP, is the initial purchase price of tractor. CCI is the output of network and bearing in mind that it should not decrease with increasing tractor age, but it may increase or remain constant with tractor age.

Tractor Age: The tractor age is considered as input of network and may be defined in various terms. These are the calendar age of tractor, tractor age as units of production and tractor age as cumulative hours of usage [8]. The calendar age is conveniently obtained by subtracting the original purchase date from the current date. Because of natural uncertainties associated with tractor repair and maintenance costs, they do not accrue as a result of elapsed calendar time. Tractor age as units of production is the measure of amount of work a tractor has actually accomplished. Defining tractor age as units of production is difficult and may be defined in a number of ways. It could be in terms of working area, working hours, traveling distance, etc. The actual quantification of production units can also be a difficult task. Tractor age as cumulative hours of use is a measure of how many hours the tractor physically operated. It dampens many of the cyclical variations in repair and maintenance costs. Considering the characteristics of three types of tractor age, the cumulative hours of use was chosen. The data for tractor age in cumulative hours of use is not always easy to obtain but it can be available. The company under study follows up oil-changing program. The times of oil-change in machines’ life are usually recorded in terms of a calendar date. Considering the calendar date of the engine oil change and the associated monthly cost data, the cumulative costs for a given number of cumulative hours is determined.

Inflation Effect: The impact of inflation can be a major concern when trying to make a conscious business decision regarding cash flows that takes place over any appreciable length of time. The company under study keeps the tractors for at least twelve years. During this time, the economy could be subjected to any number of twists and turns. If the original purchase year is used as the base year, then the inflation-adjusted cost per month is \( c_i \) which is calculated as

\[
c_{ki} = c_k \times (1 + I_{oa})^{n-k}, \quad k = 1,2,...,n
\]

where, \( c_k \), \( n \) and \( I_{oa} \) are the monthly cost, total tractor life in month and the average yearly inflation rate, respectively. The cumulative cost per month \( (cc_{ki}) \) is then calculated as
\[ cc_{ki} = c_{k-\ell_i} + c_{ki} \]  

Consequently, the cumulative cost index per month (CCI) can be calculated as

\[ CCI_k = \sum_{i=1}^{max} (cc_{ki} \times 100) \]  

Data Preprocessing: Based on these available data, the cumulative hour of usage as a percentage of 100 hours (CHU) was selected as variable input. The cumulative repair cost index (CCI_{repair}), the cumulative oil cost index (CCI_{oil}), the cumulative fuel cost index (CCI_{fuel}) and the cumulative repair and maintenance cost index (CCI_{total}) were selected as variable outputs. Prior to any ANN training process with the trend free data, the data must be normalized over the range of \([0, 1]\). This is necessary for the neurons’ transfer functions, because a sigmoid function is calculated and consequently these can only be performed over a limited range of values. If the data used with an ANN are not scaled to an appropriate range, the network will not converge on training or it will not produce meaningful results. The most commonly employed method of normalization involves mapping the data linearly over a specified range, whereby each value of a variable \(x\) is transformed as follows.

\[ x_n = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \times (r_{\max} - r_{\min}) + r_{\min} \]  

where, \(x\) is the original data, \(x_n\) the normalized input or output values, \(x_{\max}\) and \(x_{\min}\) are the maximum and minimum values of the concerned variable, respectively. \(r_{\max}\) and \(r_{\min}\) correspond to the desired values of the transformed variable range. A range of \(0.1 - 0.9\) is appropriate for the transformation of the variable onto the sensitive range of the sigmoid transfer function.

The Radial Basis Function Neural Network: The RBF network is a two layered network (Fig. 1). The main idea is to divide the input space into subclasses and to assign a prototype vector for every subclass in the center of it. Then the membership of every input vector in each subclass will be measured by a function of its distance from the prototype. The neurons in the hidden layer of network have a Gaussian activity function and their input-output relationship is:

\[ y_m = f_m(x) = \exp \left(-\frac{||x - v_m||^2}{2\sigma^2_m}\right) \]  

where \(v_m\) is the prototype RBF or the center of the \(m^{th}\) subclass and \(\sigma\) is the spread parameter, through which we can control the receptive field of that neuron.

The neurons in the output layer could be sigmoid, linear, or pseudo-linear, we have used the sigmoidal activity function, since it results in less sensitivity to learning parameters, faster convergence and lower recognition error.

\[ z_j = \frac{1}{1 + \beta S_j} \]  

where

\[ S_j = \sum_{m=1}^{l_j} y_{mj}, \quad j = 1, ..., l_3 \]  

To determine the initial values of kernel vectors, many methods have been suggested, among them the most popular is the first samples of the training set. The spread parameter \(\sigma\) can be obtained from \([10]\):

\[ \sigma = \frac{d_{\max}}{\sqrt{2l_2}} \]  

where \(d_{\max}\) is the maximum distance between the chosen centers and \(l_2\) is the number of centers.

To assign initial values to the weights in the output layer, can be selected some random values in the range \([-0.1, +0.1]\).

The common method for RBF network training is the back propagation algorithm. Using the back propagation algorithm for training RBF network has three main drawbacks: overtraining, which weakens the network’s generalization property, slowness at the end of training and inability to learn the last few percent of vector associations. In this study, to improve the performance of the network, employed a modified version of BB algorithm which is back-propagation with declining learning-rate.
factor (BDLRF) algorithm [9, 11]. A computer code was also developed in MATLAB software to implement these ANN models.

**BB Algorithm for the RBF Network:** In this algorithm the total sum-squared error (TSSE) is considered as the cost function. Network learning happens in two phases: forward pass and backward pass. In forward phase an input and the desired output is inserted to the network and the network outputs are computed by proceeding forward through the network, layer by layer. In backward pass the error gradients versus the parameters, i.e. \( \frac{\partial E}{\partial u_{mj}} \) (for \( m=1,...,J \), \( j=1,...,L \)), \( \frac{\partial E}{\partial v_{im}} \) (for \( i=1,...,L \), \( m=1,...,J \)) and \( \frac{\partial E}{\partial \sigma_m^2} \) (for \( m=1,...,J \)) are computed layer by layer starting from the output layer and proceeding backwards. The parameters of different layers are updated using the following equations:

\[
\begin{align*}
\eta_{mj}(n+1) &= \eta_{mj}(n) - \eta_3 \frac{\partial E}{\partial u_{mj}} \\
\eta_{im}(n+1) &= \eta_{im}(n) - \eta_2 \frac{\partial E}{\partial v_{im}} \\
\eta_{m}(n+1) &= \eta_{m}(n) - \eta_1 \frac{\partial E}{\partial \sigma_m^2}
\end{align*}
\]

where \( \eta_1, \eta_2, \eta_3 \) are learning-rate factors in the range \([0,1]\). The details could be seen in [9, 11, 12].

**BDLRF Algorithm:** We have also used a modified version of BB algorithm which is back-propagation with declining learning-rate factor (BDLRF) algorithm [12]. This training algorithm is started with a relatively constant large step size of learning rate \( \eta \) and momentum term \( \alpha \). Before destabilizing the network or when the convergence is slowed down, for every \( T \) epoch (3 \( \leq T \leq 5 \) these values are decreased monotonically by means of arithmetic progression, until they reach to \( x\% \) (equals to 5) of their initial values. \( \eta \) (and similarly \( \alpha \)) was decreased using the following equations:

\[
\begin{align*}
m &= \frac{Q - n_1}{T} \\
\eta_m &= \eta_n + m \eta_1 \frac{x - 1}{m}
\end{align*}
\]

where, \( m, n_1, \eta_n \), and \( \eta_1 \) are the total number of arithmetic progression terms, the start point of BDLRF, the learning rate in \( n^\text{th} \) term of arithmetic progression and the initial learning rate, respectively.

**Performance Evaluation Criteria:** Five criteria were used to evaluate the performance of RBF model. They were mean absolute percentage error (MAPE), root mean-squared error (RMSE), TSSE and the coefficient of determination of the linear regression line between the predicted values from the RBF model and the actual output (\( R^2 \)). They are defined as follows:

\[
RMSE = \sqrt{\frac{\sum_{i=1}^n \sum_{j=1}^m (d_{ji} - p_{ji})^2}{nm}}
\]

\[
R^2 = \frac{\left( \sum_{j=1}^n (d_j - \bar{d})(p_j - \bar{p}) \right)^2}{\sum_{j=1}^n (d_j - \bar{d})^2 \cdot \sum_{j=1}^n (p_j - \bar{p})^2}
\]

\[
TSSE = \sum_{j=1}^n (d_j - p_j)^2
\]

\[
MAPE = \frac{1}{nm} \sum_{j=1}^n \sum_{i=1}^m \left| \frac{d_{ji} - p_{ji}}{d_{ji}} \right| \times 100
\]

where, \( d_{ji} \) is the \( i^\text{th} \) component of the desired (actual) output for the \( j^\text{th} \) pattern; \( p_{ji} \) is the \( i^\text{th} \) component of the predicted (fitted) output produced by the network for the \( j^\text{th} \) pattern; \( \bar{d} \) and \( \bar{p} \) are the average of the desired output and predicted output, respectively; \( n \) and \( m \) are the number of patterns and the number of variable outputs, respectively. A model with the smallest RMSE, TSSE, MAPE and the largest \( R^2 \) is considered to be the best.

**RESULTS AND DISCUSSION**

Individual networks were developed in order to establish the relationships between (i) \( CCI_{rep} \) and \( CHU \); (ii) \( CCI_{m} \) and \( CHU \); (iii) \( CCI_{m} \) and \( CHU \); (iv) \( CCI_{m} \) and \( CHU \). All networks were 3-layered feed forward type, trained using both BB and BDLRF training algorithms.
Table 1: Performance variation of a three-layer BB-RBF with different number of neurons in the hidden layer.

| Parameters   | Criterion | 2   | 3   | 4   | 5   | 6   | nt/8 | nt/4 | nt/2 | nt |
|--------------|-----------|-----|-----|-----|-----|-----|------|------|------|----|-----|
| $CCI_{rep}$  | MAPE(%)   | 28.92 | 4.53 | 4.30 | 1.52 | 1.33 | 1.27 | 0.68 | 0.65 | 0.45 |      |
| RMSE         |           | 35.86 | 5.17 | 4.96 | 1.89 | 1.64 | 1.58 | 0.95 | 0.89 | 0.59 |      |
| TSSE         |           | 7.114 | 0.148 | 0.136 | 0.019 | 0.014 | 0.013 | 0.005 | 0.004 | 0.001 |      |
| $R^2$        |           | 0.627 | 0.983 | 0.984 | 0.997 | 0.998 | 0.999 | 0.999 | 0.999 | 0.999 |      |
| $CCI_{oil}$  | MAPE(%)   | 4.56  | 2.50 | 1.98 | 1.11 | 0.62 | 0.21 | 0.13 | 0.09 | 0.07 |      |
| RMSE         |           | 5.61  | 3.10 | 2.75 | 1.31 | 0.79 | 0.26 | 0.17 | 0.12 | 0.09 |      |
| TSSE         |           | 6.632 | 2.025 | 1.594 | 0.359 | 0.130 | 0.014 | 0.005 | 0.003 | 0.001 |      |
| $R^2$        |           | 0.500 | 0.750 | 0.809 | 0.954 | 0.983 | 0.998 | 0.999 | 0.999 | 0.999 |      |
| $CCI_{fuel}$ | MAPE(%)   | 5.73  | 2.87 | 2.50 | 2.32 | 2.76 | 1.19 | 0.27 | 0.15 | 0.09 |      |
| RMSE         |           | 7.15  | 3.58 | 3.44 | 3.09 | 3.77 | 1.77 | 0.44 | 0.19 | 0.12 |      |
| TSSE         |           | 5.420 | 1.360 | 1.257 | 1.012 | 1.507 | 0.333 | 0.020 | 0.003 | 0.001 |      |
| $R^2$        |           | 0.337 | 0.838 | 0.856 | 0.881 | 0.827 | 0.959 | 0.997 | 0.999 | 0.999 |      |
| $CCI_{rm}$   | MAPE(%)   | 46.93 | 6.19 | 6.18 | 6.18 | 1.94 | 1.76 | 1.04 | 0.76 | 0.60 |      |
| RMSE         |           | 53.73 | 7.56 | 7.17 | 7.25 | 2.40 | 2.20 | 1.50 | 1.03 | 0.78 |      |
| TSSE         |           | 8.261 | 0.163 | 0.147 | 0.014 | 0.017 | 0.013 | 0.006 | 0.003 | 0.001 |      |
| $R^2$        |           | 0.164 | 0.980 | 0.982 | 0.998 | 0.997 | 0.998 | 0.999 | 0.999 | 0.999 |      |

$^a$TSSE is estimated in the training phase
$^b$CCI$_{rep}=$CCI$_{rep}+$CCI$_{oil}+$CCI$_{fuel}$ c nt=number of training phase

Table 2: Optimum parameters of neural network (BB-RBF)

<table>
<thead>
<tr>
<th>Parameters of neural network</th>
<th>$C_{1}$</th>
<th>$C_{2}$</th>
<th>$C_{3}$</th>
<th>Epoch</th>
<th>Topology</th>
</tr>
</thead>
<tbody>
<tr>
<td>$CCI_{rep}$</td>
<td>1*10$^{-8}$</td>
<td>0.002</td>
<td>0.4</td>
<td>1000</td>
<td>2-nt-1</td>
</tr>
<tr>
<td>$CCI_{oil}$</td>
<td>1*10$^{-8}$</td>
<td>0.003</td>
<td>0.6</td>
<td>500</td>
<td>2-nt-1</td>
</tr>
<tr>
<td>$CCI_{fuel}$</td>
<td>1*10$^{-8}$</td>
<td>0.005</td>
<td>0.6</td>
<td>200</td>
<td>2-nt-1</td>
</tr>
<tr>
<td>$CCI_{rm}$</td>
<td>1*10$^{-10}$</td>
<td>0.005</td>
<td>0.6</td>
<td>1000</td>
<td>2-nt-1</td>
</tr>
</tbody>
</table>

Table 3: Optimum parameters of neural network (BDLRF-RBF).

<table>
<thead>
<tr>
<th>Parameters of neural network</th>
<th>$C_{1}$</th>
<th>$C_{2}$</th>
<th>$C_{3}$</th>
<th>$C_{4}$</th>
<th>Epoch</th>
<th>Topology</th>
</tr>
</thead>
<tbody>
<tr>
<td>$CCI_{rep}$</td>
<td>500</td>
<td>1*10$^{-4}$</td>
<td>0.003</td>
<td>0.4</td>
<td>1.5*10$^{4}$</td>
<td>0.04</td>
</tr>
<tr>
<td>$CCI_{oil}$</td>
<td>250</td>
<td>1*10$^{-4}$</td>
<td>0.005</td>
<td>0.8</td>
<td>2.5*10$^{4}$</td>
<td>0.04</td>
</tr>
<tr>
<td>$CCI_{fuel}$</td>
<td>180</td>
<td>1*10$^{-4}$</td>
<td>0.006</td>
<td>0.8</td>
<td>2.9*10$^{4}$</td>
<td>0.04</td>
</tr>
<tr>
<td>$CCI_{rm}$</td>
<td>200</td>
<td>1*10$^{-10}$</td>
<td>0.007</td>
<td>0.7</td>
<td>3.5*10$^{4}$</td>
<td>0.035</td>
</tr>
</tbody>
</table>

Settings and Initializations: In this study, the optimal parameters of RBF were selected using a trial-and-error method. The process was repeated several times, one for each set of data. Table 1 shows the effect of number of neurons in the hidden layer on the performance of BB-RBF model. It is observed that the performance of BB-RBF is improved as the number of hidden neurons increased. Considering Table 1, a BB-RBF model with nt (number of training phase) neurons in the hidden layer seems to be appropriate for modeling $CCI_{rep}$, $CCI_{oil}$, $CCI_{fuel}$ and $CCI_{rm}$. These topologies can be more versatile for future applications of repair and maintenance costs prediction. Tables 2 and 3 show the optimum parameters associated with BB-RBF and BDLRF-RBF.

Statistical Analysis

Training Phase: During training phase the network used the training set. Training was continued until a steady state was reached. The BB and BDLRF algorithms were utilized for model training. Some statistical properties of the sample data used for training process and the prediction values associated with different training...
algorithms are shown in Table 4. Considering the average values of standard deviation and variance, it can be deduced that the values and the distribution of real and predicted data are analogous. However, the difference of minimum value is remarkable. This is probably due to the fact that the extreme value were not well represented in the training data set, because these were only one point. Accordingly, the neural networks have been learned the training set very well, hence the training phase has been completed.

**Test Phase:** In test phase, we used the selected topology with the previously adjusted weights. The objective of this step was to test the network generalization property and to evaluate the competence of the trained network. Therefore, the network was evaluated by data, outside the training set. Table 5 shows some statistical properties of the data used in test phase and the corresponding prediction values associated with different training algorithms. It can be seen that the differences of statistical values between the desired and predicted data is less than 0.8% and 0.6% for BB and BDLRF, respectively. While in training phase these values were less than 0.01% for both of training algorithms (Table 4). This fact can be justified since these data are completely new for the MLP. On the other hand, the kurtosis, sum and the average values are similar, hence it can be deduced that both series are similar. The predicted values were very close to the desired values and were evenly distributed throughout the entire range. Although the results of training phase were generally better than the test phase, the latter reveals the capability of neural network to predict the repair and maintenance costs with new data.

From statistical point of view, both desired and predicted test data have been analyzed to determine whether there are statistically significant differences between them. The null hypothesis assumes that statistical parameters of both series are equal. \( P \) value was used to check each hypothesis. Its threshold value was 0.05. If \( P \) value is greater than the threshold, the null
hypothesis is then fulfilled. To check the differences between the data series, different tests were performed and $p$ value was calculated for each case. The results are shown in Table 6. The so called t-test was used to compare the means of both series. It was also assumed that the variance of both samples could be considered equal. The obtained $p$ values were greater than the threshold, hence the null hypothesis cannot be rejected in all cases ($p>0.99$). The variance was analyzed using the F-test. Here, a normal distribution of samples was assumed. Again, the $p$ values confirm the null hypothesis in all cases ($p>0.98$). Finally, the Kolmogorov–Smirnov test also confirmed the null hypothesis. From statistical point of view, both desired and predicted test data have a similar distribution for both of training algorithms ($p=1.000$).

Figure 2 to 5 show the actual cumulative cost indices versus the predicted ones. It is clear that the regression coefficients of determination between actual and predicted data ($R^2=0.999$) are high for the test data sets. Since excellent estimation performances were obtained using the trained network, it demonstrates that the trained network was reliable, accurate and hence could be employed for tractor repair and maintenance costs prediction. These figures reveal that the cumulative cost indices predictions from BB training algorithm were not as good as fit to actual cumulative cost indices in comparison to BDLRF cumulative cost indices prediction. Comparisons of actual versus predicted cumulative cost indices for BB training algorithm resulted in a least squares linear regression lines with slopes equal to BDLRF, while the BDLRF training algorithm resulted in lines with y-intercepts lower than BB.

**Comparison of Training Algorithms:** For prediction of each component, several networks with different settings and training algorithms were trained. The performances of the two training algorithm are shown in Table 7. For this specific case study, the comparison of results reveals that both algorithms are capable of generating accurate estimates within the preset range. However, it was noticed that BDLRF algorithm had a higher decrease of MAPE, RMSE and TSSE for training phase and test phase in
Table 7: Performances of two training algorithm in prediction of tractor repair and maintenance costs indices.

<table>
<thead>
<tr>
<th>Parameters of cost</th>
<th>Training algorithm</th>
<th>Performance criterion</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Training phase</td>
<td>Test phase</td>
</tr>
<tr>
<td></td>
<td>MAPE</td>
<td>RMSE</td>
</tr>
<tr>
<td>$CCI_{repair}$</td>
<td>BB</td>
<td>1.27</td>
</tr>
<tr>
<td></td>
<td>BDLRF</td>
<td>1.08</td>
</tr>
<tr>
<td>$CCI_{oil}$</td>
<td>BB</td>
<td>1.19</td>
</tr>
<tr>
<td></td>
<td>BDLRF</td>
<td>1.08</td>
</tr>
<tr>
<td>$CCI_{fuel}$</td>
<td>BB</td>
<td>1.13</td>
</tr>
<tr>
<td></td>
<td>BDLRF</td>
<td>0.82</td>
</tr>
<tr>
<td>$CCI_{rm}$</td>
<td>BB</td>
<td>1.38</td>
</tr>
<tr>
<td></td>
<td>BDLRF</td>
<td>1.10</td>
</tr>
</tbody>
</table>

Fig. 4: Predicted values of artificial neural network versus actual values of $CCI_{repair}$ for BB and BDLRF training algorithms.

Fig. 5: Predicted values of artificial neural network versus actual values of $CCI_{fuel}$ for BB and BDLRF training algorithms.

comparison to BB algorithm. It was quite clear that the BDLRF training algorithm achieved a much better performance than the BB training algorithm. Bearing all the results obtained by this study in mind, the advantages of the BDLRF training algorithm over BB are: faster convergence, lower training time and also it eases the process of parameter adjusting by decreasing the sensitivity to the parameters’ values. The results also conforms the findings of [3, 11].

CONCLUSIONS

This article focused on the application of RBFNN to predict tractor repair and maintenance costs. To show the applicability and superiority of the proposed approach, the actual data of tractor repair and maintenance costs from Astan Ghodse Razavi agro-industry (in the north east of Iran) were used. To improve the output, the data were first preprocessed. RBF network was used and applied with the past 18 years tractor repair and maintenance costs as variable inputs. The network trained by both BB and BDLRF learning algorithms. Statistical comparisons of desired and predicted test data were applied to the selected ANN. From statistical analysis, it was found that at 95% confidence level (with $p$-values greater than 0.9) both actual and predicted test data are similar. The results also revealed that, using BDLRF algorithm yields a better performance than BB algorithm. Because the ANN does not assume any fixed form of dependency in between the output and input values, unlike the regression methods, it seems to be more successful in the application under consideration. It could be said that the neural network provides a practical solution to the problem of estimating repair and maintenance costs in a fast, inexpensive, yet accurate and objective way. It is hoped that the analysis conducted in this article can provide reference for the choice of RBFNN in such area.

ACKNOWLEDGEMENTS

The authors would like to thank Astan Ghodse Razavi agro-industry in Iran for providing the data and other support to this study.
REFERENCES