Available online at www.sciencedirect.com
ScienceDirect
Expert Systems with Applications 36 (2009) 2155–2160
Expert Systems with Applications
www.elsevier.com/locate/eswa

An artificial neural networks approach on automobile pricing

Ali İşeri [a,*], Bekir Karlık [b]

[a] Department of Industrial Engineering, Fatih University, İstanbul, Turkey
[b] Department of Computer Engineering, Fatih University, İstanbul, Turkey

Abstract

The aim of this study is to find an automobile pricing model using artificial neural networks (ANN). As commonly known, pricing is a difficult matter for both automobile manufacturers and buyers/sellers. Developing a neural networks based on the technical properties of automobiles will allow both groups to price autos with great ease. However, in this study there are two basic assumptions. The first is that supply and demand are in equilibrium and they have no positive or negative effect on pricing. Alfred Marshall [Alfred Marshall. (1920). Principles of economics (Vol. 9). Macmillan] describes how the price and availability of goods and services are related to consumer demand in competitive markets in the Law of supply and demand. The second is that our data set represents the whole market since we will determine market prices of other automobiles according to the network that is trained by this dataset. Proposed novel model estimates prices of automobiles on a stable market from their technical and physical properties.

Keywords: Pricing; Artificial neural networks; Automobile; Stepwise linear regression

1. Introduction

Automobile industry grows worldwide and becomes more competitive each year (Office of National Statistics, 2006). So pricing an automobile right is very important for both manufacturers and customers in this competitive market. For many years people are confused about the price of the automobile they are buying or selling, especially ones that do not have special interest in automobiles and their technical properties. So they get help from automobile magazines, from internet, or from auto-dealers. But this takes a long time and people can also be confused afterwards.

Last decades, computational methods are used to solve pricing and valuation problems. For example Montagna, Nicrosinib, and Morenic (2002) have presented an efficient computational algorithm to price financial derivatives which is based on a path integral formulation of the pricing problem. It is shown how the path integral approach can be worked out in order to obtain fast and accurate predictions for the value of a large class of options, including those with path-dependent and early exercise features. Song and Regan (2005) examined computationally tractable approximation methods for estimating bid values and constructing bids for freight transportation contracts. Feurstein and Natter (2000) derived fast approximations to the stochastic dynamic program for the valuation of flexible manufacturing systems.

Recently, some artificial intelligent (AI) techniques are started to be used for solution of pricing and valuation problems. For example; Wang and Ramsay (1998) proposes a neural-network-based approach to predict system marginal price (SMP) with particular reference to weekend and public holidays for electricity pricing. Teodorovi’c and Edara (2007) proposed a real-time road pricing system in the case of a two-link parallel network using dynamic programming and fuzzy logic. Pao (2007) proposed a new artificial neural network (ANN) with single output node structure by using direct forecasting approach for forecasting long-term electricity market pricing, in order to avoid

* Corresponding author. Tel.: +90 212 8663300x5604.
E-mail address: aliiseri@fatih.edu.tr (A. İşeri).

0957-4174/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2007.12.059

excessive round-off and prediction errors. Wilson, Paris, Ware, and Jenkins (2002) presented an ANN which forecasts future trends within the housing market and is trained using national housing transaction time series data.

This work shows that statistically and economically significant out-of-sample pricing performance is possible using the ANN approach. In this proposed pricing problem, a multi layer perceptron (MLP) structure is trained with back-propagation algorithm which has generalized delta rule learning. In the first section the pricing problem is defined and solution method is proposed. The second section explains how our model is set and works with the given data set.

2. Problem statement and solution approach

There are no rules showing how much an automobile makes that have some certain specifications. This is because the pricing rules are very complex to be explained by simple rules. The solution to this problem can be to find an automobile pricing model using ANN.

In our proposed model, the market of automobiles is in equilibrium. This is not a false assumption since the automobile market is very competitive. In a market like that we estimate various automobile prices by using neural networks. The inputs are the technical and physical properties that can have an effect on the price and output is the price of autos. To solve this problem; first we take a large data set that include technical and physical properties and prices of automobiles. In the data set there are 159 samples which consist of 15 input values and one output value. Totally, 60% of this data is used for training, 20% is used for cross validation and 20% is used for testing. The data for each class are chosen randomly from the used data set. The data set is shown with its definitions about inputs and an output in the following chapter.

3. Model

In this section proposed model with the used data set and artificial neural networks are briefly reviewed.

3.1. Data set

The data set consists of 16 types of entities and contain 159 samples. It is available from The Laboratory of Artificial Intelligence and Computer Science Website (http://www.liacc.up.pt/~ltorgo/Regression/DataSets.html). The first 15 of the entities are the input entities and the last one (price) is the target entity.

1. `Symboling: −3, −2, −1, 0, 1, 2, 3.`
2. `Normalized losses: continuous from 65 to 256.`
3. `Wheel-base: continuous from 86.6 to 120.9 in.`
4. `Length: continuous from 141.1 to 208.1 in.`
5. `Width: continuous from 60.3 to 72.3 in.`
6. `Height: continuous from 47.8 to 59.8.`
7. `Curb-weight: continuous from 1488 to 4066 pounds.`
8. `Engine-size: continuous from 61 to 326 cubic in.`
9. `Bore: continuous from 2.54 to 3.94 in.`
10. `Stroke: continuous from 2.07 to 4.17 in.`
11. `Compression-ratio: continuous from 7 to 23.`
12. `Horsepower: continuous from 48 to 288 hp.`
13. `City mpg: continuous from 13 to 49.`
14. `Highway mpg: continuous from 16 to 54.`
15. `Peak-rpm: continuous from 4150 to 6600 rpm.`
16. `Price: continuous from 5118 to 45400 $.`

where *Symboling* corresponds to the degree to which the auto is more risky than its price indicates. Cars are initially assigned a risk factor symbol associated with its price. Then, if it is more risky (or less), this symbol is adjusted by moving it up (or down) the scale. A value of +3 indicates that the auto is risky, −3 that it is probably pretty safe.

Normalized losses is the relative average loss payment per insured vehicle year. This value is normalized for all autos within a particular size classification and represents the average loss per car per year.

Wheel-base is the distance between the center of the front wheel, and the center of the rear wheel.

Length is the distance from the back of the car to the front of the car.

Width is the distance between two sides of the car.

Height is the distance from the ground to the top of the car.

Curb-weight is the total weight of an automobile with standard equipment, motor oil, coolant and a full tank of fuel and not loaded with either passengers or cargo.

Engine-size is the combined volume of all the cylinders. Bigger engines are more powerful.

Stroke is the distance that piston travels within the cylinder.

Bore is the diameter of a cylinder in a piston engine.

Compression-ratio is a ratio between the volume of a combustion chamber and cylinder, when the piston is at the bottom of its stroke and the volume when the piston is at the top of its stroke. The higher the compression ratio, the more mechanical energy an engine can squeeze from its air–fuel mixture.

Horsepower is a unit of power that is used in the automotive industry for listing the maximum rate of power application of internal-combustion engines.

City mpg is the distance traveled per gallon of fuel used in the city traffic. In this case, the higher the value, the more economic a vehicle is.

Highway mpg is the distance traveled per gallon of fuel in highways.

Peak-rpm represents the maximum number of full rotations the engine makes in one minute.

Price is the value of the automobile.

3.2. Used artificial neural network

In this study, neural networks are used. Neural networks are composed of simple elements operating in parallel. These elements are inspired by biological nervous systems. As in nature, the network function is determined largely by the connections between elements. You can train a neural network to perform a particular function by adjusting the values of the connections (weights) between elements.

Commonly neural networks are adjusted, or trained, so that a particular input leads to a specific target output. Typically many such input/target pairs are needed to train a network. Neural networks have been trained to perform complex functions in various fields, including pattern recognition, identification, classification, speech, vision, and control systems. Today neural networks can be trained to solve problems that are difficult for conventional computers or human beings.

The Multi-layer perceptron is the most widely used type of neural network. It is both simple and based on solid mathematical grounds. Input quantities are processed through successive layers of "neurons". There is always an input layer, with a number of neurons equal to the number of variables of the problem; and an output layer, where the perceptron response is made available, with a number of neurons equal to the desired number of quantities computed from the inputs. The layers in between are called "hidden" layers. With no hidden layer, the perceptron can only perform linear tasks. All problems which can be solved by a perceptron can be solved with only one hidden layer, but it is sometimes more efficient to use two hidden layers.

In this study the network is trained by using Matlab Neural Networks Module (nftool). 60% of the available data is used for training, 20% is used for cross validation and 20% is used for testing. The data for each class are chosen randomly from the data set.

In the network, we used Levenberg–Marquardt backpropagation function. This function is a network training function that updates weight and bias values according to Levenberg–Marquardt optimization method.

Function (net, Pd, Tl, Ai, Q, TS, VV, TV) takes these inputs:

- Net: Neural network.
- Pd: Delayed input vectors.
- Tl: Layer target vectors.
- Ai: Initial input delay conditions.
- Q: Batch size.
- TS: Time steps.
- VV: Either an empty matrix [] or a structure of validation vectors.
- TV: Either an empty matrix [] or a structure of test vectors.

and returns:

- netTrained: Network.
- TR: Training record of various values over each epoch.
- Epoch number.
- Training performance.
- Validation performance.
- Test performance.
- Adaptive μ value.

Training occurs according to *trainlm*'s training parameters, shown here with their values:

1000	maximum number of epochs to train
0	performance goal
5	maximum validation failures
1	factor to use for memory/speed tradeoff
$1e^{-10}$	minimum performance
0.001	Initial μ
0.1	μ decrease factor
10	μ increase factor
$1e^{10}$	Maximum μ
25	Epochs between displays
Inf	Maximum time to train in seconds

Validation vectors are used to stop training early if the network performance on the validation vectors fails to improve or remains the same for max fail epochs in a

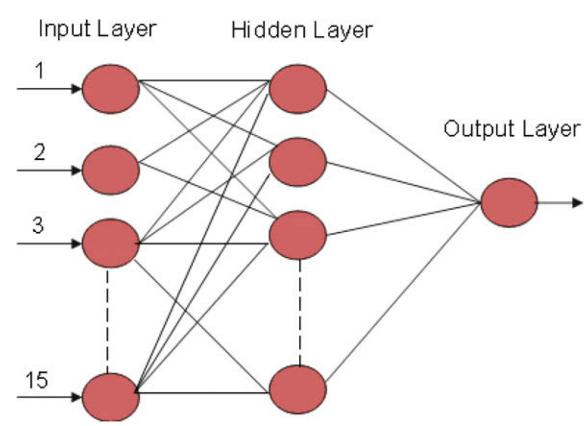
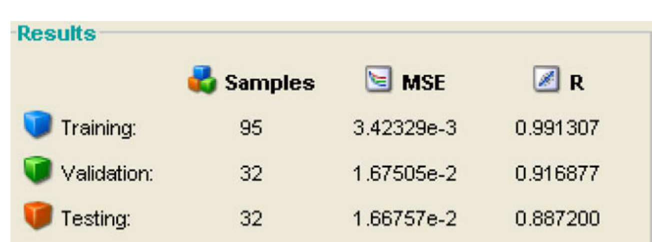


Fig. 1. Used MLP Structure of ANN.


Results

	Samples	MSE	R
Training:	95	3.42329e-3	0.991307
Validation:	32	1.67505e-2	0.916877
Testing:	32	1.66757e-2	0.887200

Fig. 2. Total MSE error according to 900 epochs.

row. Test vectors are used as a further check that the network is generalizing well, but do not have any effect on training Neural Network Toolbox (xxxx).

Used MLP structure is 15:100:1, which are 15 neurons of input layer, 100 neurons of one hidden layer and only one neuron of the output layer. Optimum neurons of hidden layer was found as 100 after using different number of neurons (see Fig. 1).

The network was trained for 900 epochs with Levenberg–Marquardt back-propagation function and showed the following error results as it can be seen in Fig. 2. The mean square error of training phase is $3.42329e^{-3}$, cross validation phase is $1.67505e^{-2}$ and testing phase is $1.66757e^{-2}$.

In Fig. 3, the dashed line is the perfect fit line where outputs and targets are equal to each other. The circles are the data points and colored line represents the best fit between outputs and targets. Here it is important to note that circles gather across the dashed line, so our outputs are not far from their targets. According to these results we can say that used MLP structure of ANN is very well to solve automobile pricing problem.

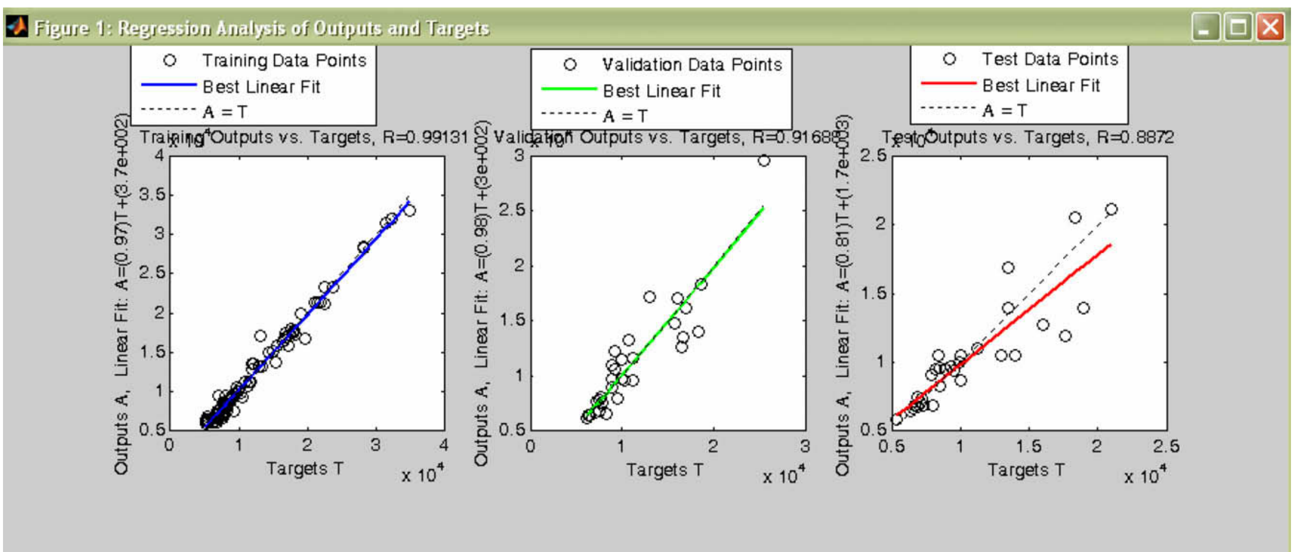


Fig. 3. Regression analyses of outputs and targets.

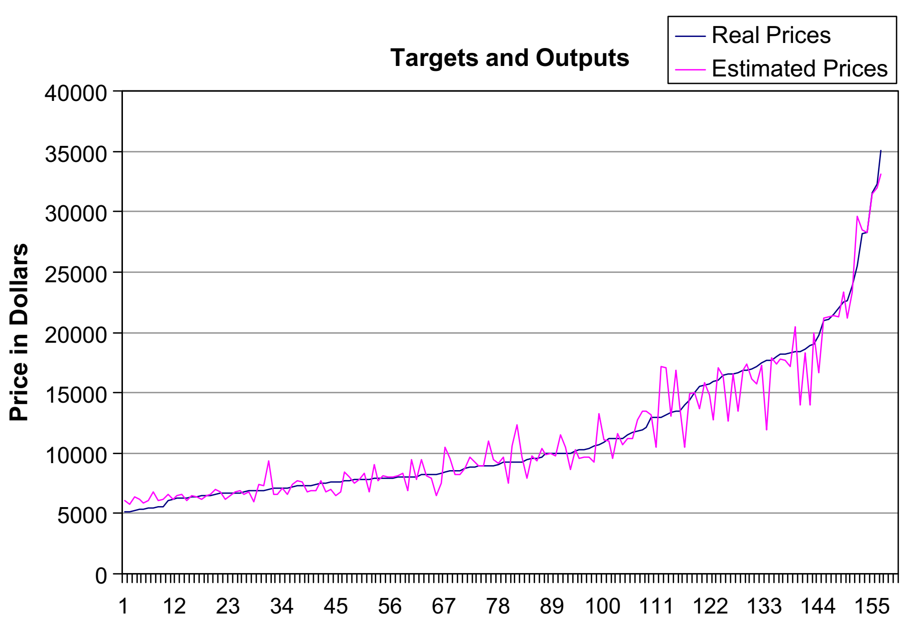


Fig. 4. Comparison between real and estimated prices.

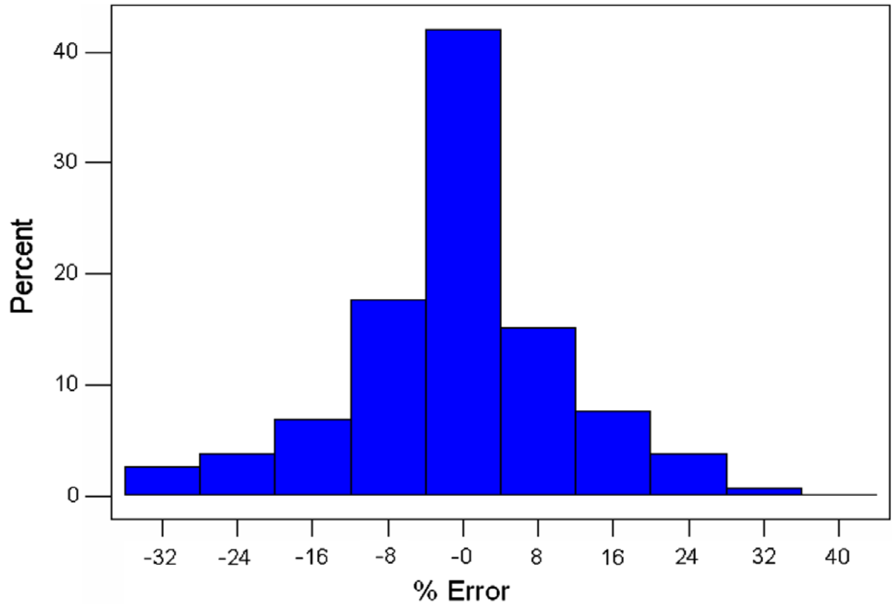


Fig. 5. Histogram of errors.

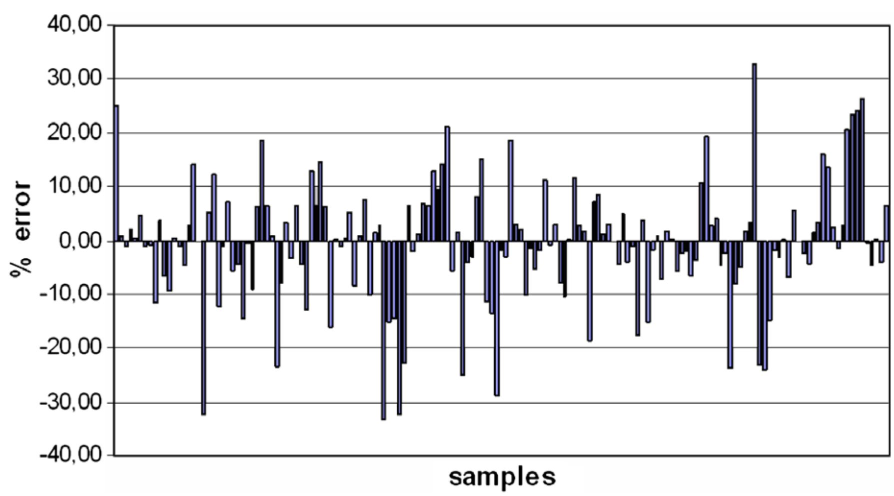


Fig. 6. Bar chart of individual errors.

4. Analyses of errors

Absolute error is the absolute value of the difference between target values and outputs. Mean absolute error of outputs is 8.0%. This is a very good result according to stepwise regression method which produces 14.4% mean absolute error.

Although it seems there is a 8.0% mean absolute error, the actual case is smaller than this value since there are seven extreme errors in the data set greater than 25%. These are the ones that are priced too expensively or too cheap by their manufacturers according to their properties. After these extremes are excluded the mean absolute error is 7.0%.

The comparison between real (target) and estimated (output) prices are shown in the Fig. 4.

The histogram for the errors are shown in Fig. 5. As it is seen, over 40% of the errors accumulate across 0. That shows our network works very well.

The individual percentage errors are shown in a bar chart in Fig. 6. It is seen that most of the errors are between 10% and −10%, most of them smaller than 5%.

5. Conclusions

This paper presented the application of neural networks to pricing systems. It demonstrated how neural networks have been used to test the Efficient Market Hypothesis and how they outperform statistical and regression techniques in forecasting share or compare prices such as cheap or expensive. If the actual price is lower than the proposed price, it can be considered as cheap. And if the actual price is higher than the proposed price, it can be considered as expensive. The work has shown that neural networks are a good method for automobile pricing problems. So one can use neural networks to estimate the value of an automobile very quickly by entering its technical and physical properties to the network even if he/she does not know much about automobile pricing strategies. This will eliminate confusions about prices. But this study is a case study for American first hand market. To develop an ANN for other markets, data set must be changed according to requirements of the new market.

References

Alfred, Marshall (1920). *Principles of economics* (Vol. 9). Macmillan.

Feurstein, Markus, & Natter, Martin (2000). Fast high precision decision rules for valuing manufacturing flexibility. *European Journal of Operational Research, 120*, 108–117.

Montagna, Guido, Nicrosinib, Oreste, & Morenic, Nicola (2002). A path integral way to option pricing. *Physica A, 310*, 450–466.

Neural Network Toolbox for Use with MATLAB, Howard Demuth, Mark Beale, Martin Hagan.

Office of National Statistics: Annual Business Inquiry, published December, 2006, www.statistics.gov.uk/about/data/guides/Labour-Market/sources/business/businquiry.asp.

Pao, Hsiao-Tien (2007). Forecasting electricity market pricing using artificial neural networks. *Energy Conversion and Management, 48*, 907–912.

Song, Jiongjiong, & Regan, Amelia (2005). Approximation algorithms for the bid construction problem in combinatorial auctions for the procurement of freight transportation contracts. *Transportation Research Part B, 39*, 914–933.

Teodorovi'c, Dušan, & Edara, Praveen (2007). A real-time road pricing system: The case of a two-link parallel network. *Computers & Operations Research, 34*, 2–27.

Wang, A. J., & Ramsay, B. (1998). A neural network based estimator for electricity spot-pricing with particular reference to weekend and public holidays. *Neurocomputing, 23*, 47–57.

Wilson, I. D., Paris, S. D., Ware, J. A., & Jenkins, D. H. (2002). Residential property price time series forecasting with neural networks. *Knowledge Based Systems, 15*, 335–341.