Contents lists available at Science-Gate

International Journal of Advanced and Applied Sciences

Journal homepage: http://www.science-gate.com/IJAAS.html

Sensitivity analysis and optimisation to input variables using winGamma and ANN: A case study in automated residential property valuation

Nguyen Vo *, Hao Shi, Jukab Szajman

College of Engineering and Science, Victoria University, Melbourne, Australia

ARTICLE INFO

Article history:
Received 27 December 2015
Received in revised form
17 January 2016
Accepted 17 January 2016

Keywords:
ANN
Optimisation
Sensitivity
Rank
Fitness
Java
AVM
Error Threshold
AVM
WinGamma and Encog 3

ABSTRACT

In this research work, an optimal Automated Valuation Model (AVM) using ANNs was developed to evaluate residential property price in Brimbank, Victoria, Australia. Optimisation to ANNs was achieved by determining the best number of the hidden layers, the hidden neurons, and finding the best value of training error threshold. The input variables were analysed in terms of sensitivity using winGamma. The results were displayed as input "masks", i.e. a combination of input variables. The importance of each input variable was determined from the input mask. The top three input masks were tested by ANN models. An optimal ANN in terms of network topology and top input mask was excellent in predicting residential property prices within the accuracy of ±10% error of the actual sale price. It also successful modelled the annual changes in residential property prices for hard to predict periods 2007-2008 during the global financial crisis and 2010-2012 residential boom when the interest rates were on a downwards trend.

1. Introduction

Residential properties in Victoria are re-valued manually every two years by the Department of Sustainability and Environment (DSE), Victoria, Australia with up to ±30% uncertainty of the market values. Municipal councils use the values established by DSE to determine property rates and land tax liabilities. According to rpdata.com (Comparing the quality of property valuation mythologies, 2010), there are currently five types of AVMs used in residential property valuation in Australia: sales comparison approach, cost approach, hedonic, income capitalisation approach and price indexation. The calculation backbone for these AVMs is still based on traditional statistics approach. At the time of writing this research paper, only a handful of researchers in the world have used Artificial Neural Network (ANN) in AVMs to estimate residential property prices. Using ANN in AVM can be considered to be in its infancy and has not been used in the AVMs for residential property valuation in Victoria (Hayles, 2006).

The research work also aims to develop a framework for ANN optimisation for both internal topology (i.e. hidden layers, hidden neurons and training error threshold) and the input set variables. In addition, a neural network performance criteria was also identified and modified from (Vo et al., 2011). Software packages associated with ANN can be a problematic as they do cost dearly. Therefore, some of the open sources of ANN Java library were investigated to use for ANN development.

The paper is structured as follows. In section 2, a brief theory of neural networks is reported. In section 3 the case study is introduced: it is relative to the sample of residential properties sold in Brimbank, Victoria, Australia. In section 4 the ANN model and winGamma are specified and the results are illustrated. In section 5 the conclusions of the work are discussed.

2. Outline of ANNs

An ANN model consists of input, hidden and output layers. It must also include a training type in order for the network to learn. The two training types are supervised and unsupervised. There are other ANN models which have no hidden layers, for example, the SOM neural network.

The MLP network topology shown in Fig. 1 consists of interconnected layers of neurons. The number of neurons in an input layer that do not do any processing but take the inputs to the next layer depends on the number of input variables. The weights of each processing neuron, which is in the hidden layers, are adjusted by using the error, which is calculated at the end of each iteration, by

* Corresponding Author.
Email Address: vo.nguyen@vu.edu.au (N. Vo), hao.shi@vu.edu.au (H. Shi), jakub.szajman@vu.edu.au (J. Szajman)

comparing the estimated output with the ideal output from the training set. A trained neural network could take thousands of iterations to complete. The MLP neural network is known as feed forward back propagation ANN, that is, the signal feeds forwards through the network and the error adjustments are propagated backwards.

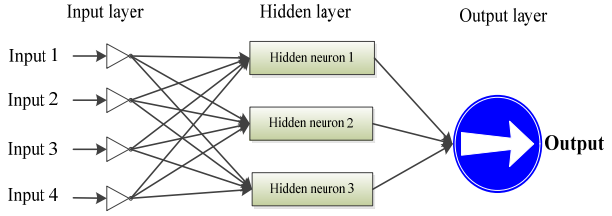

Fig. 1: An example of a MLP (4 input neurons; 3 hidden neurons; 1 output neuron) neural network topology

Despite having many satisfactory characteristics of ANN, building a neural network for a particular problem is still challenging. First, a neural network topology has to be made by determining the number of hidden layers and neurons therein and a bias neuron in a hidden layer if required. Second, other neural network design decisions have to be made including an activation function for the processing neurons, a training type, data normalisation methods, and performance criteria (Zhang and Patuwo, 1998). The main role of a bias neuron is to allow a neural network to learn patterns more effectively (Heaton, 2010). Its function is similar to the hidden neurons. However, unlike any other neurons in a neural network, a bias neuron never receives input from the previous layer because it always outputs a constant value of one. As a result, a neural network can produce an output value of one more effectively when the input is zero for some neural network applications.

An ANN topology consists of three main layers: input, hidden and output layers. The number of neurons in the input layer corresponds to the number of the input variables. The hidden layer can have more than one layer with many neurons therein. The number of neurons in the output layer equals to the number of outputs; in the case of this research work, the output size was one. The neurons in the input and hidden layers have significant effects on ANN design and performance. In general, according to Zhang and Patuwo (Zhang and Patuwo, 1998), the number of neurons in the input layer has a much larger influence than the number of neurons in each hidden layer when building a forecasting model. It is important that the number of input variables must be sufficient in order for a neural network model to produce an output with a high order of accuracy. Too little input variables might not be sufficient for the neural network model to produce a reliable output. Whereas too many input variables can adversely affect the performance of a neural network model as the input variable set may contain redundant variables.

3. The case study

Properties in Brimbank have been selected for this research work because of the availability of data and it is one of the most culturally diverse municipalities in Australia. The Brimbank municipality is the largest in metropolitan Melbourne, Victoria, Australia and its closest point is about 12 km away from the City of Melbourne. Brimbank covers an area of about 123 km^2 with has five districts (Deer Park district, Keilor district, St Albans district, Sunshine district and Sydenham district) and a total of 25 suburbs.

According to the economic theory discussed by Adair, Berry (Adair et al., 1996) as well as Andrew and Meen (Andrew and Meen, 1998), the ANN model would benefit by including new input variables with a significant impact on house prices such as interest rate, geo-location (longitude and latitude), sale type and sale date. These new variables were therefore incorporated in addition to the standard house characteristic input variables such as land area, floor area, number of bedrooms, number of bathrooms, number of stories, number of garages, year built, home type, main construction material and suburb code used by other researchers (Do and Grudnitski, 1992; Ibrahim et al., 2005). All input variables, including inputs and output, were to be normalised accordingly by using Equation 1. Table 1 shows a list of input and output variables.

$$y_i = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}}, \quad (1)$$

Where y_i is the normalised value, x_i is un-normalised value, $x_{\max}$ is maximum un-normalised value and $x_{\min}$ is the un-normalised minimum value.

Fig. 2 shows the number of data records collected for each specified year after pre-processing. The testing set is the same as the validation set, although the terms are sometimes interchanged in the literature. The usage here corresponds with that of Bishop (Bishop, 1995) and of (Stegemann and Buenfeld, 1999).

Table 1: List of ANN inputs and output variables

	Variable name	Input	Output
	Sale price		✓
1	Sale date	✓	
2	Land area	✓	
3	Floor area	✓	
4	Bedroom	✓	
5	Year built	✓	
6	Construction type	✓	
7	Property type	✓	
8	Storey	✓	
9	Longitude	✓	
10	Latitude	✓	
11	Bathroom	✓	
12	Garage	✓	
13	Sale type	✓	
14	Interest rate	✓	
15	Suburb rank	✓	

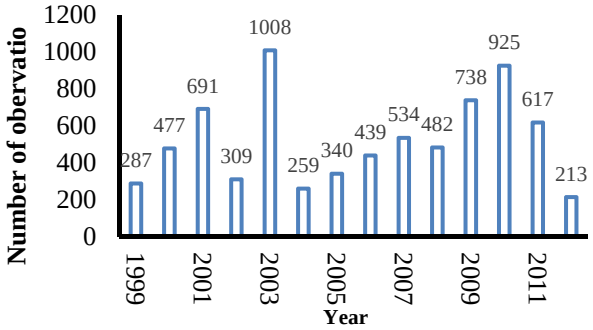

Fig. 2: Sample collection distributions

Training set: A set of experimental data is used to train the neural network.

Testing set: An independent set of data which the neural network has not seen before, which is used to test how well the neural network has learned to generalise.

According to Durrant (Durrant, 2001), training is a method used to minimise the total fitting error of a neural network. In ANN world, there are many training types or training algorithms (Rossini, 1998). Some training algorithms require appropriate learning and momentum rates and it can take a lot of time to find. Therefore, a training algorithm used in (Vo et al., 2011) was employed in this research work because it did not require learning and momentum rates. However, the only difference was that the Resilient Propagation (RPROP) training type with iRPROP+ replaced the RPROP+ training type used in (Vo et al., 2011). The iRPROP+ training type was chosen over RPROP+ training type because Heaton (Heaton, 2010) and as well as Riedmiller and Braun (Riedmiller and Braun, 1993) claimed that iRPROP+ training type was the optimum RPROP training type. There are four types of RPROP supported by Encog 3, while previous versions of Encog only support RPROP+. Once a neural network was completely trained it could be used to forecast the prices of residential properties (Table 2).

Table 3: Top five Gamma values and input masks

	Input masks														
Gamma values	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0.00015679	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0.00025850	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0.00098909	1	1	1	1	1	1	1	1	1	1	1	0	1	1	0
0.00155384	1	1	1	1	1	1	1	1	1	1	1	0	0	1	0
0.00203358	1	1	1	1	1	0	1	1	1	1	1	0	0	1	1

3. Optimisation and data analysis to ANN inputs

3.1. WinGamma optimisation to ANN inputs

The efficiency and accuracy of ANN model can be improved by optimising the input variable set. In the past, there were some theoretical attempts to find an optimal ANN topology but finding theoretical optimal input variable set for a neural network is challenging (Vo et al., 2011). In practice, winGamma can quickly optimise and rank input variables set.

This software package can be used to be build models such as ANNs with three different types of training algorithms, including two layer feed-forward back-propagation models. It also comes with a number of training set analysis options including the Gamma test and the M-test, and model identification options including Genetic Algorithms (GA).

WinGamma can also calculate the Gamma test of a given data set. The Gamma test is an estimate variance of the noise on each output (ideal output). This allows estimating the best Root Mean Square Error (RMSE) that an ANN model can achieve for a corresponding output. The Gamma test is useful because it can help to determine if there is sufficient data to form a smooth non-linear model and predict the "goodness" of the model from the data consideration only. WinGamma also includes a number of model identification options. These may be used to assist in choosing a selection of inputs that minimises the asymptotic value of the Gamma statistic and, hence, finds variables of least sensitivity thereby redundant variables can be identified and removed from the inputs set. Model identifications are designed to produce an "embedding" – a selection of inputs chosen from all the inputs, and designated by a string binary mask. The mask "110111" for six inputs indicates that all inputs are to be used except the third. The best inputs combination results in a model which has minimal RMSE when used to predict the output sale price.

Initially, there were 15 variables in the input set as shown in Table 5.11; winGamma might eliminate some least sensitive variables by running a model identification method. Optimisation was required to remove the unwanted variables because an ANN topology that is most efficient is the one with least number of neurons, excluding a bias neuron (Zhang and Patuwo, 1998).

In the first experiment, winGamma has been used to run the Gamma test with all of the input variables available in the data set. The Gamma value has been found to be 0.00236267. Since the output variable (Sale price) range was [0, 1], after normalisation, the Gamma value indicated a small error variance. This means that there was a standard deviation of the prediction error of $\sqrt{0.00236267} = 0.048607$, i.e., 4.86% of the range. Input mask was applied for purpose of model identification with "full embedding" experiment to determine the best selection of inputs. winGamma has performed 32,767 ($2^{15} - 1 = 32{,}767$) experiments for

representing a total number of possible combinations of 15 input variables and has taken about two days to compute on an Intel core i7 computer!

Table 2 shows the top five Gamma values and input masks. The smaller the Gamma values the better for ANN modelling. The winGamma results have indicated that suburb rank variable (input 15) was the least variable sensitivity; therefore it could be safely removed from the input set. This was a good choice, considering location input conveys similar information.

The very top combination of input variables was chosen to test with ANN. The performance was improved by as much as 5.23% (Vo, 2014).

3.2. Sensitivity of input variables

Gamma values were validated using Fitness values obtain from ANN experiments. The top three Gamma values in Table 2 correspond to the Fitness order of Model B (14 inputs), Model A (15 inputs) and Model C (13 inputs) respectively as illustrated in Fig. 3. The Gamma values listed in Table 4 summarises the relative sensitivity of each variable calculated in (Vo, 2014). Smaller Gamma value corresponds to lower sensitivity of the input variable. For example, "Suburb rank" had the lowest sensitivity because it had the smallest Gamma value with input mask "111111111111110". Consequently, the highest sensitivity or the largest input weight variable was found to be the "Sale date". This finding was not surprising since it allows ANN to compensate for movement in house prices due to inflation. Table 4 lists sensitivities of the input variables. It is interesting to note that the latitude and longitude input variables were listed in table next to each other, intuitively expected, because together they formed a single input variable corresponding to the street address.

The sensitivities of input variables as shown in Table 4 were tested by Encog 3 in order to determine a relationship between Fitness and Gamma values. The same experiment procedures outlined in (Vo, 2014) were closely followed to determine the optimal neural network topologies. These experiments were performed systematically by removing one input variable at a time.

The variable was then restored and another removed. The process was repeated for all of the input masks shown in Table 5. A total of 14 new experiments were carried out and the last one already been done, i.e., Model B in Fig. 3. Each experiment had a different input variable set, for example, experiment EXP01 would have all input variables except the "Sale date".

The experiments, including optimisation, ran continuously for one month on VU cloud with four AMD CPUs. The results were displayed in Table 5. The order of variable sensitivity based on Fitness turned out to be very similar to winGamma predictions, except the ranking of "Floor area" and "Land area" was swapped.

Table 4: Gamma values and input masks

	Input masks														
Gamma values	Suburb rank	Sale type	Construction type	Garage	Floor area	Land area	Bathroom	Interest rate	Latitude	Longitude	Year built	Bedroom	Property type	Storey	Sale date
0.00015679*	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0.0002585	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0.00232797	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1
0.00233209	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1
0.00235707	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1
0.00235984	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1
0.00237185	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1
0.00237342	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1
0.00239281	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1
0.0024102	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1
0.00241707	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1
0.00249642	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1
0.00251414	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1
0.00254677	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1
0.00258495	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1
0.00733932	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0

*the optimal input variable set

Table 5: Weighting of input variables

Gamma values	Variables	Rank
0.00733932	Sale date	1
0.00258495	Storey	2
0.00254677	Property type	3
0.00251414	Bedroom	4
0.00249642	Year built	5
0.00241707	Longitude	6
0.00241020	Latitude	7
0.00239281	Interest rate	8
0.00237342	Bathroom	9
0.00237185	Land area	10
0.00235984	Floor area	11
0.00235707	Garage	12
0.00233209	Construction type	13
0.00232797	Sale type	14
0.00015679	Suburb rank	15

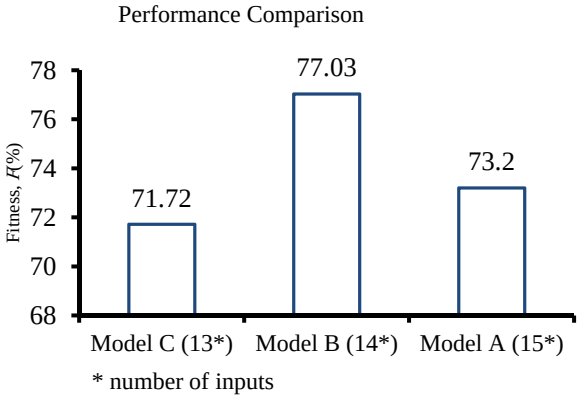

Fig. 3: Performance comparison of Models A, B and C

There was only a small margin of 0.11% between the two input variables. ANN's predictions that the "Floor area" in Brimbank was more important than "Land area" agreed with Hansen (Hansen, 2009) ranking for the whole of Victoria. Overall, winGamma was proved to be useful in finding the variable sensitivity ranking and requires an order of magnitude less computer time than Fitness based Encog 3 calculations. The experimental results also highlighted that the "Sale date" variable was the most important variable in residential property evaluation. Training without "Sales date" was stagnated. The experiments established a functional relationship between Gamma and Fitness as shown Fig. 4.

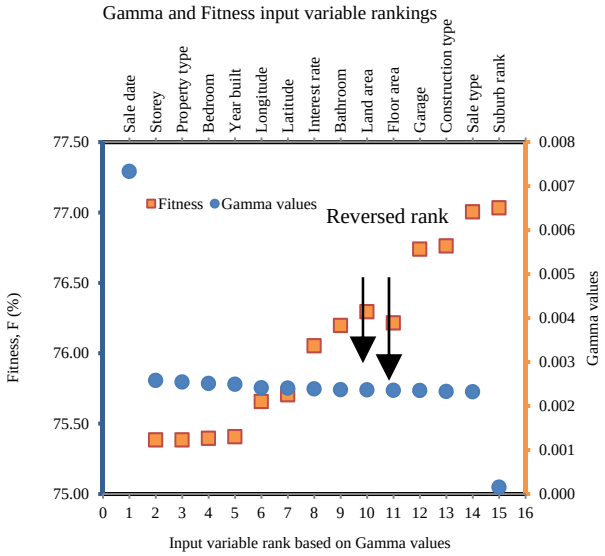

Fig. 4: Gamma values and Fitness vs input variable rankings

Table 6: Gamma values and Fitness

Experiments	Excluded input variables	Fitness, *F* (%)	Gamma values	Rank
EXP01	Sale date	Stagnated	0.00733932	1
EXP02	Storey	75.38414217	0.00258495	2
EXP03	Property type	75.38414217	0.00254677	3
EXP04	Bedroom	75.39553429	0.00251414	4
EXP05	Year built	75.40692641	0.00249642	5
EXP06	Longitude	75.65755297	0.00241707	6
EXP07	Latitude	75.70539986	0.00241020	7
EXP08	Interest rate	76.05302217	0.00239281	8
EXP09	Bathroom	76.19753930	0.00237342	9
EXP10	Land area*	76.29551151	0.00237185	10
EXP11	Floor area*	76.21576669	0.00235984	11
EXP12	Garage	76.73980406	0.00235707	12
EXP13	Construction type	76.76258829	0.00233209	13
EXP14	Sale type	77.00548872	0.00232797	14
EXP15	Suburb rank	77.03349282	0.00015679	15

* inconsistent with Fitness order

4. Conclusion

In this research work an ANN model has been proposed, which was able to forecast house prices by using an MLP with 14 inputs, 1 hidden layer with 7 neurons and a bias neuron, and 1 output neuron neural network topology with iRPROP+ training algorithm. Other training algorithms were considered but iRPROP+ training algorithm was quicker and more efficient as stated by Riedmiller and Braun (Riedmiller and Braun, 1993) and Heaton (Heaton, 2010). Input variables set; hidden neurons and hidden layers were optimised.

WinGamma has been successfully applied to input variables optimisation and rank. With the help of winGamma the whole process of ranking input variables has been reduced dramatically down to days rather than weeks if ANNs were used alone (Vo, 2014).

The improvements observed when new input variables, such as interest rates, property type and sold type, were added to the input variable set suggested that it was both the current input variable set and the addition of new input variables that were important. Increasing the number of input variables for an ANN model might improve the forecast performance, but it could also adversely affect its prediction capability. However, if any new input variable was to be added to the original set, winGamma must be employed to identify possible candidates for inclusion. Sensitivity analysis must then be applied next for determination of input variable set.

References

Adair AS, Berry JN and McGreal WS (1996). Hedonic modelling, housing submarkets and residential valuation. Journal of property Research, 13(1): 67-83.

Andrew M (1998). Modelling regional house prices: A review of the literature. Report Prepared for the Department of the Environment, Transport and the Regions, Centre for Spatial and Real Estate Economics, University of Reading.

Bishop CM (1995). Neural networks for pattern recognition. Oxford university press.

Do AQ and Grudnitski G (1992). A neural network approach to residential property appraisal. The Real Estate Appraiser, 58(3): 38-45.

Durrant PJ (2001). winGamma TM: a non-linear data analysis and modeling tool with applications to flood prediction (Doctoral dissertation, Department of Computer Science, Cardiff University).

García N, Gámez M and Alfaro E (2008). ANN+ GIS: An automated system for property valuation. Neurocomputing, 71(4): 733-742.

Hansen J (2009). Australian House Prices: A Comparison of Hedonic and Repeat-Sales Measures*. Economic Record, 85(269): 132-145.

Hayles K (2006). The use of GIS and cluster analysis to enhance property valuation modelling in Rural Victoria. Journal of spatial science, 51(2): 19-31.

Jeff H (2011). Programming Neural Networks with Encog3 in Java.

Faishal Ibrahim M, Jam Cheng F and How Eng K (2005). Automated valuation model: an application to the public housing resale market in Singapore. Property Management, 23(5): 357-373.

Riedmiller M and Braun H (1993). A direct adaptive method for faster backpropagation learning: The RPROP algorithm. In Neural Networks, 1993., IEEE International Conference on (pp. 586-591). IEEE.

Rossini P (1998). Improving the results of artificial neural network models for residential valuation. In Fourth Annual Pacific-Rim Real Estate Society Conference, Perth, Western Australia.

Stegemann JA and Buenfeld NR (1999). A glossary of basic neural network terminology for regression problems. Neural computing and applications, 8(4): 290-296.

Vo N (2014). A New Conceptual Automated Property Valuation Model for Residential Housing Market (Doctoral dissertation, Victoria University).

Vo N, Shi H and Szajman J (2011). Artificial neural network optimization in automated property valuation models with Encog 2.

Zhang G, Patuwo BE & Hu MY (1998). Forecasting with artificial neural networks: The state of the art. International journal of forecasting, 14(1): 35-62.