valori
e valutazioni
teorie ed esperienze

Società Italiana di Estimo e Valutazione

Neural Networks and Linear Models in Real Estate Appraisal: the impact of sets selection procedures

Matteo Galante[1], Silvio Giove[2] e Paolo Rosato[3,*]

[1] Head of Financial and Liquidity Risk, Risk Management Dept. Banca Ifis, Via Terraglio, 63 - 30174 Venezia (VE); matteo.galante@bancaifis.it

[2] Department of Economics, Ca' Foscari University of Venice, Dorsoduro 3246 - 30123 Venezia (VE); sgiove@unive.it

[3] Department of Engineering and Architecture, University of Trieste, Piazzale Europa, 1 - 34128 Trieste; paolo.rosato@dia.units.it

* corresponding author

Keywords

artificial neural networks; real estate appraisals; similarity measures; fuzzy distances

Abstract

The use of Neural Networks in real estate appraisal has been recently subject of renewed interest by the scientific community. Generally, their effective use requires the availability of a large database, otherwise facing the real risk, even with an excellent performance on the «training set», of obtaining unsatisfactory generalisation properties (the so called over fitting effect). The well-known multiple regression models (MRAs), on the other side, require fewer parameters for their optimisation but are unable to capture complex nonlinear relationships. Since large databases are usually difficult to find in the real estate market, MRA models often provide better results than Artificial Neural Networks (ANNs). Furthermore, the latter require considerable effort to be effectively trained, both in finding the best structure and in estimating the characterising parameters. The optimisation process that leads to an efficient neural network requires a long job as well as considerable computational capabilities. This contribution, after outlining the state of the art in the use of ANNs and confirming that the scarcity of real estate market data often turned out to be a serious obstacle in their concrete application, proposed an innovative algorithm for selecting the data used in the training process. Such an algorithm seems to be able to improve predictive performance: networks that seek to take full advantage of the information available for learning seem to have better abilities in generalising the behaviour of the underlying phenomenon than those that are trained with completely randomly selected data, as usually done in practice.

1. Introduction

Until the late 1980s, the construction of quantitative models to support property appraisal used, almost exclusively, an econometric approach known as «Hedonic Housing Price» which is based on multiple linear regression analysis.

The method assumes that properties are aggregates of different characteristics, each of which contributes to a market value (Lancaster, 1966).

This kind of models start from market prices and proceed to valuate the weights of individual characteristics, in order to obtain a value function to be used in an appraisal.

This function is usually linear. The theory on which the hedonimetric method is based, introduced by Rosen (1974), assumes a competitive market and involves the simultaneous estimation of supply and demand functions (Witte et al., 1979). In the particular case of the real estate market, given the rigidity of supply, the model can be simplified and traced back to the neoclassical scheme of consumer demand theory (Diamond and Smith, 1985). The use of the hedonimetric approach in the definition of value functions, requires that properties would be traded in a market substantially transparent on the supply side, and homogeneous and competitive on the demand side. These conditions are often difficult to find in the real estate markets. Hedonimetric approaches have been widely used in the study of the impact of housing characteristics on value but have raised caution in appraisal because of some critical questions: the availability of market data and the choice of independent variables are probably the most relevant issues. In particular, the need for information is greater the more heterogeneous the assets under consideration are and are positively correlated with the number of the variables considered in the model. In addition, special care must be taken in the selection of the independent variables because of multicollinearity, which significantly distort the coefficients associated to the various characteristics (Ozanne and Malpezzi, 1985)[1].

Moving from the operational criticalities of the econometric approach in simulating the real estate market for appraisal purposes, in the early 90s of the last century the neural networks were introduced as a new approach to support real estate appraisal.

A neural network is an artificial intelligence model inspired by the learning mechanisms of the human brain, aimed at understanding and subsequently using the information presented, for the purpose of predictions over time (Alpaydin, 2010; Bishop, 2006; Gareth et al., 2013). As in the human brain there are cells (so-called neurons) which are interconnected with each other via appropriate junctions (so-called synapses), also an artificial neural network is organised into basic processing units (artificial neurons) that receive, process and send signals to other neurons. In artificial neural networks, as in biological ones, learning occurs through the exchange of information between these units.

In artificial neural networks, neurons are distributed in various layers, characterised by different functions: 1) the input layer; 2) the hidden layers; 3) the output layer. The input layer receives the «raw» information to be processed and has a number of neurons equal to the independent variables. The hidden layers process the information received from the input layer. The number of hidden layers can be variable, such as the number of neurons of each layer. The output layer highlights the result of the processing and is composed of a number of neurons equal to the number of response modes of the network. In traditional networks, as the multilayer neural networks (Multi-layer Perceptron - MLP), all the neurons of a certain layer are connected to all the neurons of the adjacent layers.

Figure 1 illustrates the basic structure of a neural network with the three types of layers.

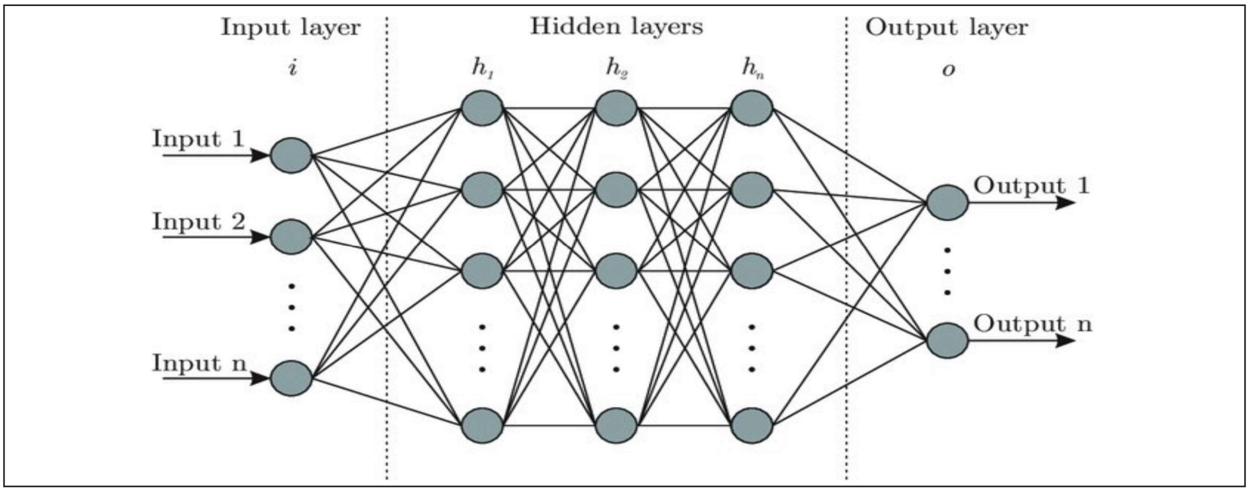

Figure 1. The structure of an artificial neural network MLP.

---

[1] The detailed discussion of econometric approaches goes beyond the objectives of this article, therefore, please refer to the copious scientific literature (Pagourtzi et al., 2003; Lisi, 2019; Khoshnoud et al., 2023; Binoy et al., 2022).

As anticipated, the network is fed with data introduced through the input layer and the information is then propagated to the neurons of the hidden layers. This passage of knowledge is obtained through the so called 'synaptic connections', which assign an appropriate weight to each of the values received as input. So, the neuron in the hidden layers receives a certain number of inputs from the ones in the previous layer and calculate a unique value using an activation function, defined ex-ante linear or non-linear. This mechanism of propagation is repeated for all layers of the network until the information reaches the output one. The last layer (the so called 'output layer') is responsible for returning the final value. With this mechanism, a neural network is able to represent relationships between independent and dependent variables that are much more complex than the multiple regression models.

Unlike traditional econometric approaches, neural networks do not require a priori specification of the relationship between input and output since, using a trial-and-error procedure, they endogenously estimate the relationships between the independent variables and the dependent variable. The valuation of the best network structure, on the other hand, is much more complex compared with the econometric approaches, because the performance achieved depends significantly on the characteristic structure of the network itself, that is, the number of layers, the number of neurons in each layer and the type of activation functions present in each feature. These are actually all topics defined through a trial-and-error procedure.

It should be pointed out that the relationships that link through the neural network inputs to the outputs are particularly complex and it is not possible to highlight them transparently: these types of models are usually preferred when it is acceptable to sacrifice the readability of the results in favour of a greater accuracy in the output.

Before being used, neural networks must be properly trained. Calibration is done using a training set and a validation set obtained from a proper database, which contains a certain number of observations defined as couples of inputs-outputs. Training the network using known examples is a procedure called 'supervised learning'. The most common procedure for training a neural network involves an initial random assignation of values to various synaptic weights, thus obtaining a first estimate of the output. The following step is the comparison of the model outcomes with the actual observed ones. In this way, it is possible to obtain, through the application of a specific error function, a prediction error. If the error is too high, the optimisation algorithm proceeds backward from the output layer through the hidden layer nodes, adjusting the weights and so reducing the prediction error (back propagation error). The process, recursive, continues until the error is less than a predetermined threshold.

The trained network is then tested through an appropriate test set, consisting of cases not in the training set, in order to verify its generalisation ability with an unknown dataset.

As in human learning, the aim of training a machine learning model is not to exactly simulate the relationships between input and output of the training set, but to effectively estimate the output in the presence of new and unknown observations. This is possible only if the model has actually understood the real relationships that govern the phenomenon: an excessive alignment of the network of the data proposed in the training set may lead to a poor interpretation of the test set because the network tends to over align itself to what is proposed during its training. This phenomenon is usually referred to as the 'overfitting effect' and is the same risk that occurs in human learning when, in the presence of little experience, a person tends to believe to have understood a problem, when in fact he has only memorised what happened in the few observed case histories. This risk, for both humans and machines, is greater when the training set is limited to a few observations.

However, it should be remembered that neural networks, at least in the context considered, can be interpreted as semi-parametric inferential procedures that are particularly flexible, and able to simulate very complex nonlinear relationships (Lawrence, 1984), handle uncertain information, and, once trained, are quite computationally efficient.

Starting from the original insight of McCulloch and Pitts (1943) to the present day, there are nearly 280 thousand contributions on artificial neural networks covering almost 30 different subject areas. It should also be noted that nearly 200 thousands of these contributions have been published in the last 10 years (Source: Scopus). Artificial neural networks, therefore, constitute an instrument used in a variety of fields and are characterised by increasing popularity.

The purpose of this paper is to propose an application of neural networks in real estate appraisal with particular reference to their effectiveness in respect to multiple regression models, while proposing appropriate optimisation techniques in determining the optimal training set used to train the models. This aspect is particularly interesting since usually the breakdown of data into training and validation sets is generated automatically and in a completely random way. This can lead to dissimilarities between the characteristics of the two sets, resulting in possible loss of network generalisation abilities. In the following, a proposal will be formulated that attempts to overcome this drawback by verifying its best performance in an appropriate case study. The second paragraph discusses the most significant contributions on the performance of neural networks in property valuations, compared with multiple regression models that have appeared in the specialised literature. Paragraph number three reports the main features of the case study used in the simulations. Paragraph four presents an illustration of the methodology and simulations performed with particular reference to the training of networks and proposes an algorithm for the optimal partitioning of data between training and test sets. The fifth summarises a comparative framework of the various approaches tested.

2. The state of the art

The first applications of Artificial Neural Networks (ANNs) in estimating the market value of real estate are dated back to the early 1990s (Do and Grudnitsky, 1992; Evans et al., 1991; Tay and Ho, 1991/1992). These works aimed to highlight the potential improved performance achievable over the traditional methods in use up to that time, which were essentially based on multiple regression models (MRAs). It was in fact believed that the new models would be able to obtain better performance, thanks to their ability to simulate the complex nonlinear relationships present in observed price formation. Since the neural networks are de facto black-box models, the acceptable cost for their use would be a lower readability of the underlying relationships, as evidenced in the early works. These contributions also stressed that the availability of an appropriate dataset for the training phase, appropriate in terms of both quantity and quality, was still conditional for obtaining adequate performance of the models.

However, subsequent analyses obtained less convincing results on the validity of using ANN for real estate valuations.

Among others, Worzala, Lenk and Silva (1995), applied neural networks to a database of 288 houses transactions in Fort Collins, Colorado and compared the results with those of traditional MRA models. They not only raised doubts about the performance obtained with the networks, but also pointed out other drawbacks, including: inconsistency among results when analysing time series of different lengths, inconsistency of results obtained using different software, and very long computation times.

Thanks to impressive advances in calculators, the use of ANNs has become more common and easier, and many other authors have proposed their application to housing appraisals. Peterson and Flanagan (2009) used ANNs to estimate the value of housing during 1999-2005. Exploiting a database composed by 46,467 observations, using ANN they obtained better results compared to those with multiple regression models. Chiarazzo et al. (2014) tested the potential of ANNs for estimating complex phenomena, including elements that are difficult to handle with linear models, such as pollution, landscape and personal tastes. Basing their estimates on a sample of 193 observations referring to the Taranto urban area, and identifying 42 explanatory variables, the authors obtained interesting results, especially by evaluating the impact of each independent variable in explaining the value estimated by the entire network. Temur et al. (2019) recently proposed several models in forecasting the housing market, both linear and nonlinear (LSTM-type ANN), and concluded that the best performances were obtained with the use of hybrid models, which combined linear and nonlinear elements.

It should be noted, however, that in specialised real estate appraisals literature, there isn't currently a clear preference of ANNs over other models. For example, McCluskey et al. (2012) agree that the use of neural networks can theoretically be applied to real estate valuation, but they also point out how other nonlinear regression models exhibit superior predictive ability. They also observe how the use of ANN models has a limited role in the appraisal practice, because they are less transparent than traditional regression models: this characteristic, they say, is a major obstacle to their concrete use since a high clarity is absolutely required on how value judgments are obtained. Nunez Tabales et al. (2013) applied the ANN methodology to the housing market of some medium-sized cities in southern Spain, comparing the pre - and post -crisis data of the Spanish housing market (years 2008-2010). The results seem to confirm better predictive capabilities of the networks compared to linear models. However, they once again emphasise the need for databases of significant size (often unavailable in the housing market), which is a requirement necessary for an adequate estimation of the many parameters structurally present within the networks. Wisniewski (2017) proposed an ANN-based analysis for predicting the prices of clusters of European countries, relying on macroeconomic indicators (inflationary variables, unemployment rate, consumption, population growth rate etc.). In the study, the results of various ANNs are compared with those of linear models, showing that sometimes the performance of the latter is superior to that of ANNs.

In most recent years, many other contributions have proposed advanced machine learning models applied to real estate appraisal. Among others, Moro et al. (2020) proposed, the use of a combination of models (including ANNs) in forecasting real estate sales in the city of São Paulo, Brazil, showing a significant improvement in performance. Rampini and Cecconi (2021) applied different machine learning models for the purpose of predicting transaction prices in two cities in northern Italy, showing better prediction capabilities of ANNs than other models. However, the authors emphasise that the accuracy of such models depends on the abundance of available data, concluding that this is the reason why contradictions in specialised literature are highlighted. The need for a sufficiently large data set recurs in almost all recent contributions.

The research for alternative machine learning models is also found in the contributions of Liu et al. (2006), who propose the use of neuro-fuzzy models, and Shinde et al. (2019) who present deep learning systems. In their works, emphasis is placed on the fact that these approaches do indeed have superior performance, but that in order to achieve an efficient training, such systems require a significant amount of data. Valier and Micelli (2020) report a summary of the state of the art on the topic, concluding how machine learning models are effective in predictive capabilities, but with less satisfactory results regarding inferential properties. According to them, it does not seem possible to rank the different proposed models in terms of accuracy and, they predict finally that «the creation of machine learning models will only be possible for those who hold large information assets with which to train and optimise learning.» Recent work by Root et al. (2023) summarises machine learning models used in real estate appraisals, highlighting their effectiveness in terms of accuracy, and concluding that future research should focus on greater computational simplicity.

For the purposes of this paper, it is also interesting to note the contribution proposed by Kalliola et al. (2021), where it is shown that the results obtained from the use of ANNs are highly dependent on the parameters, architecture and estimation methods (activation functions, optimisation algorithms, etc.).

Summing up, it seems to conclude that:

1. In general, ANNs, given their flexibility, appear to have a greater ability to model the nonlinearities present in price formation mechanisms in real estate markets than linear models;
2. The complexity of ANNs can be an obstacle to training networks when the relationships are essentially linear. In that case, MRA methodologies achieve better results;
3. The effective use of neural networks seems contingent on the availability of large databases;
4. In appraisal practice, there is some caution in the use of ANNs due to computational complexity, difficulty in interpreting the many parameters and lack of transparency, all of which make the process of value judgment formation obscure;
5. The use of ANNs seems to be, at the state of the art, mostly limited to mass-appraisal operations for accounting or tax purposes;
6. The development of ANNs for real estate appraisals is a very rapidly evolving line of research that is likely to reach concrete results in the next future.

3. Case study

The simulation proposed in this work uses a database composed of 219 property transactions that took place in Caorle, a coastal municipality close to Venice. Caorle is a well-known seaside city characterised by a significant national and international tourist flow. It has around 11,000 inhabitants and is 14 km parallel to the Adriatic coast. It is considered to be one of the most characteristically historic seaside towns in Italy.

Caorle records about 4.4 million holiday visitors annually, which places it ninth overall in Italy among tourist destinations. Its economy is almost completly touristic, although there is significant agricultural activity practiced on the vast inland area. It is characterised by a historical centre of medieval origin, still well preserved, surrounded by recent coastal building expansion, the most renowned being Porto S. Margherita and Duna Verde to the west, and the Villaggio dell'Orologio to the north. The presence of a fishing port and two important docks for pleasure boats should be noted.

The transactions used are related to holiday apartments that occurred in the last 10 years. Prices were discounted to 2023 using local real estate price trends that were available from the Internal Revenue Service.

The survey collected information on the characteristics of the properties and the surrounding settlement environment for a total of 50 variables. After careful analysis conducted through a panel of experts in the field, the following ten variables were selected as those considered the most influential on property value:

1. location inside Villaggio dell'Orologio;
2. location inside Porto Santa Margherita;
3. distance from the historical centre;
4. availability of a garden;
5. availability of a swimming pool;
6. availability of parking space;
7. floor level;
8. quality of finish;
9. surface;
10. sea-view.

Transactions involved small and medium-sized holiday apartments with prices ranging from €2,500 to €5,000/sqm. More than half of the apartments are located in the centre of Caorle or very close; 70 are located in Porto Santa Margherita/Duna Verde, a large settlement with an attached marina built in the 1960s, and 25 are located in the Villaggio dell'Orologio, a more recent holiday complex with an attached dock, characterised by good urban quality. Considering that the location within the Villaggio dell'Orologio, the location in Porto Santa Margherita, and the sea-views have a significant impact on the value of the properties considered, it was chosen to include them in the model as variables of Boolean type. These then take the value of 1 in the case that the apartment is located within one or other specific areas or enjoys a sea view (variables 1, 2, and 10) and while taking the value 0 in the opposite case. The model also contains other variables of the Boolean type, as shown in Table 3 below, which likewise takes the value 1 if the property has a garden, swimming pool or parking space (variables 4,5,6).

Most apartments are characterised by good or excellent finishes and have a courtyard and parking space. 42% of them have access to a swimming pool, and 13.7% enjoy a sea-view (Tables 1, 2 and 3)[2].

---

[2] For the purpose of the computations, the level of finishing was coded into a 3-valued cardinal variable, associating the value 1 with «Mediocre» status, 2 with «Normal» status, and 3 with «Good» status

Table 1. The discrete input variables

Characteristics	Frequency	%	% cumulated
Level of finishing			
Mediocre	13	5.9	5.9
Normal	141	64.4	70.3
Good	65	29.7	100
Total	219	100	
Floor level			
Zero	105	47.9	47.9
First	79	36.1	84
Second	31	14.2	98.2
Third	3	1.4	99.5
Fourth	1	0.5	100
Total	219	100	

Table 2. The continuous input and output variables

Characteristics	Min	Max	Mean
Commercial surface (mq)	45	80	63
Distance from the historical centre (mt)	1	1,700	404
Price (€/mq)	2,448	5,160	3,393

Table 3. The boolean Input variables

Characteristics	NO	YES	Total
Garden			
Frequency	50	169	219
%	22.8	77.2	100
Cum. %	22.8	100	
Swimming Pool			
Frequency	126	93	219
%	57.5	42.5	100
Cum. %	57.5	100	
Parking Space			
Frequency	42	177	219
%	19.2	80.8	100
Cum. %	19.2	100	
Sea-View			
Frequency	189	30	219
%	86.3	13.7	100
Cum. %	86.3	100	
Villaggio Dell’orologio			
Frequency	194	25	219
%	88.6	11.4	100
Cum. %	88.6	100	

Follow **Table 3.** The boolean Input variables

Characteristics	NO	YES	Total
	Porto S. Margherita		
Frequency	149	70	219
%	68	32	100
Cum. %	68	100	

Table 4 shows the correlation coefficients of the variables described, with respect to the transaction price per square metre. The correlation coefficients basically respect the a-priori expressed by the market. There is a negative (significant) correlation with the commercial surface of the property, which represents a decreasing marginal appreciation of the size of the property. Lower values are associated with the location into Porto Santa Margherita, while the values of the properties located into the Villaggio dell'Orologio are generally higher.

The presence of good finishes, the presence of a parking space, and the location on higher floors turn out to be positively (significantly) correlated with the price.

Finally, with regard to the correlations between features, it should be noted that they are always modest in magnitude, and there is a certain consistency among the elements enhancing the value of the property (swimming pool, garden, exclusive parking space).

Table 4. The correlations between independent variables and unit price

Variables	Pearson	p-value
Commercial Surface	-0.5269	0.0000
Garden	-0.0969	0.1530
Level of finishing	0.1858	0.0058
Swimming Pool	0.0868	0.2006
Distance Historical Centre	0.0430	0.5268
Parking Space	0.1791	0.0079
Sea-View	0.0849	0.2107
Floor level	0.3085	0.0000
Villaggio Dell'orologio	0.1243	0.0664
Porto S. Margherita	-0.6506	0.0000

All variables (dependent and independent) were normalised to the range [0,1].

4. Applied Methodology

As widely acknowledged in the literature, and reported above, the performance of neural networks is strictly dependent on the quality and quantity of data available for their training. In particular, the availability of an adequate number of observations is imperative, since neural networks are models characterised by a large number of parameters that need to be estimated. There is a well-known expression in literature, the so-called curse of dimensionality, which expresses this concept: as the dimensionality (i.e. the number of variables to be estimated) increases, the volume of space increases at such a rate that the available real data becomes too sparse, and it is therefore difficult to obtain statistically significant correlations. Without sufficient data, the training generates a model that underfits the data and does not sufficiently generalise the new observations. The expression was introduced by Bellman (1957) 70 years ago in the field of dynamic programming, but it represents a phenomenon easily found in the field of Machine Learning. This explains why a strong comeback has been observed in recent times (currently the expression ‘curse of dimensionality’ obtains about 770,000 results in the Yahoo search engine).

Considering the small size of the available database, software was observed in the Python language in order to extract most of the information in the available data. The coded algorithm aims to search for the combination of observations to be included within the training set and at the same time, as will be better explained later, to optimise the combination of independent variables to be used in the training of the network. The purpose of the present work, experience in developed code, is not so much to identify a network that can outperform the linear model, but rather to test whether a large number of neural networks can, on average, produce improvable results compared to a large number of linear models, partially sharing the proposed approach of Kalliola et al. (2021).

4.1. The partition of data between training and test set

As anticipated above, the process of training an artificial neural network is very similar to the human learning, which is based on a continuous observation of causes and consequences and the consolidation of these into the experience.

Obviously, an unbridgeable gap exists between the natural and artificial learning processes, as artificial networks are extremely simplified structures compared to biological ones. Moreover, the training of a neural network takes place by having a limited amount of information available to it.

These aspects pose some concerns regarding the ability of a neural network to grasp the underlying phenomena in general terms.

The compromise solution, established by practice, is to randomly divide the available observations into two groups. The first is used for training the network, while the second is reserved for later testing. In the present case the first group, which is called the training set, is constituted of 75% of the available observations, or 165 transactions. The second group is accordingly attributed the remaining 54 transactions, i.e. the remaining 25% not considered in the training set. This second group, which is called the test set, is considered as out-of-sample, and used only ex-post for the purpose of performance evaluation. It is important to evidence that it does not intervene in any way in the training process. The data belonging to the first group were divided into another subgroup consisting of 75%, for the actual training set (the so called ‘core training set’), while the remaining 25% is retained as a validation set and used for the purpose of stopping model training. The training of the network is done by the well-known gradient descent technique. Table 5 summarises how the 219 observations are divided between the core training set, validation set and test set.

Table 5. The breakdown of the database in the training phase

ORIGINAL DATABASE			FIRST PARTITION			SECOND PARTITION		
Group	**# record**	**% on total**	**Group**	**# record**	**% on total**	**Group**	**# record**	**% on total Training set**
DataBase	219	100%	*Test set (out of sample)*	54	25%			
			Training set	165	75%	*Validation set*	41	25%
						Training set core	124	75%

Given the limited database available, it has been chosen not to use all available independent variables because, by adopting 10 inputs with 5 neurons in the hidden layer, the learning algorithm would have to optimise an excessive number of parameters, amounting to 11x5+6=61. With only 124 observations available for the core training set, the degrees of freedom would be too small, thus compromising the robustness of the learning, significantly increasing the risk of overfitting.

Therefore, it was decided to select only 5 from the 10 available variables, constructing sub-groups of variables (the inputs of each specific network) so as to cover all possible combinations of 5 variables extracted from the set of 10 available variables. As a result, the number of implemented ANNs is equal to:

$$C(n,k) = {}^{n!}/_{(k!\,(n-k)!)} \qquad (1)$$

where:
n = total database inputs
k = neural network's inputs

Thus, a total of 252 groups of 5 different variables chosen from the 10 originally available were considered.

4.2. The training of Neural Networks

The performance of a neural network, even with the same training set, can vary depending on both the network structure (number of hidden layers, number of neurons within each layer, shape of the activation functions) and as a result of the initial values assigned to different parameters at the start of training. Regarding the structure of the networks, the Multilayer Perceptron (MLP) model, one of the most common in applications, was chosen, with a single hidden state consisting of 5 neurons. The activation functions considered in each of these nodes are tanh (hyperbolic tangent) type, having codomain (-1,1), and sigmoidal, monotone increasing representation, depicted in Figure 2.

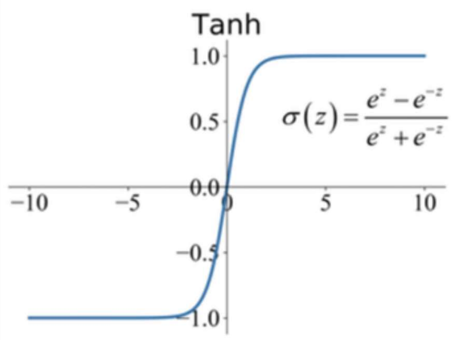

Figure 2. The hyperbolic tangent activation function

In order to reduce the risk of overfitting, a dropout layer, intermediate between the hidden and output layers, was also included (Srivastava et al., 2014).

The learning algorithm is thus faced with optimising 36 parameters, having the networks 5 neurons in the input layer and 5 in the hidden layer, with 124 observations available in the core training set. In conclusion, the networks are characterised by about 3.5 degrees of freedom each.

Figure 3 below graphically represents the structure of the networks.

The performance of the networks may differ not only because of their structure, but also depending on the (random) choice of values initially assigned to the parameters to be estimated. In fact, restarting the training procedure would imply that the optimisation algorithm would almost certainly break down having found optimised parameters different from each other in each iteration. This phenomenon occurs because the purpose of the optimisation algorithm is to minimise the objective function, represented by the difference between the estimated and observed values. The computational time required to reach the absolute minimum of the error can be prohibitive and is therefore normally stopped when a predetermined level deemed acceptable is reached. Since this is also a nonlinear model, there is no guarantee that the search method, often based on gradient descent or derivatives, will converge to an absolute minimum. Therefore, it is common practice to repeat the training several times

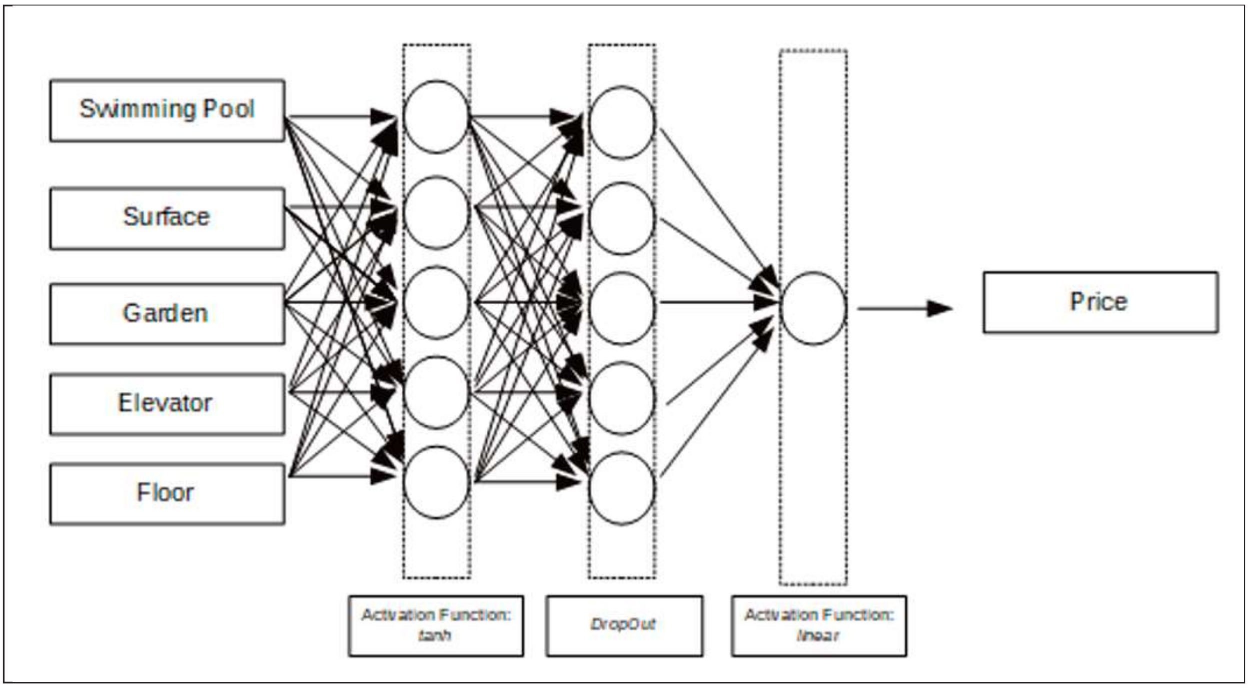

Figure 3. The structure of ANNs

in order to explore the parameter space as exhaustively as possible. Such a technique is called Random Search (Jang et al. 1997).

Applying this procedure to the case study, the algorithm divides the available data between training and test sets. Next, the algorithm selects a group of 5 inputs and trains the network based on the core training set data that are randomly extracted from the training set produced at the beginning. Once the training is completed and the performance is recorded, the algorithm proceeds to a new selection of the core training set, re-initiating the training. This re-initiation is repeated 25 times in total. Having then generated 25 networks on the same input cluster, the algorithm restarts, using a new set of input variables, constructing another 25 networks. The algorithm reaches its conclusion when it finishes the 252 input clusters, for each of which 25 networks are generated, differing only in the values of the synaptic weights optimised in the training phase.

At the end of the procedure, it has obtained 6300 networks, grouped into 252 clusters each of which, as mentioned, use a set of 5 inputs chosen from the 10 available. The algorithm is summarised in Figure 4.

4.3. Use of random algorithms for random selection of the training set

The performances of each of the 6,300 networks are measured by the well-known coefficient of determination, R^2. The same metric is also used to evaluate the performance of the corresponding linear models, which are trained on the same training set core. The performances, both of the linear model and the ANN, are evaluated using the test set, which of course is kept out of the sample so far, not affecting the training of the models in any way.

The results of the first 10 networks, compared with those of the corresponding linear models, are shown in Table 6.

Adopting a random selection of the training set, the performance of the best 10 networks in terms of R^2 is on average similar to that of the MRA models, but still slightly lower. The networks that manage to perform better than the corresponding linear models are only 53 out of 6,300 (0.84%). It should be noted that not only are the R^2 of the networks lower than those of the linear models, but they also have greater variability and sometimes negative values.

This dispersion affects, inevitably, the average performance in each cluster: there are clusters where networks perform well but others that, due to particularly poor performance, register distinctly modest

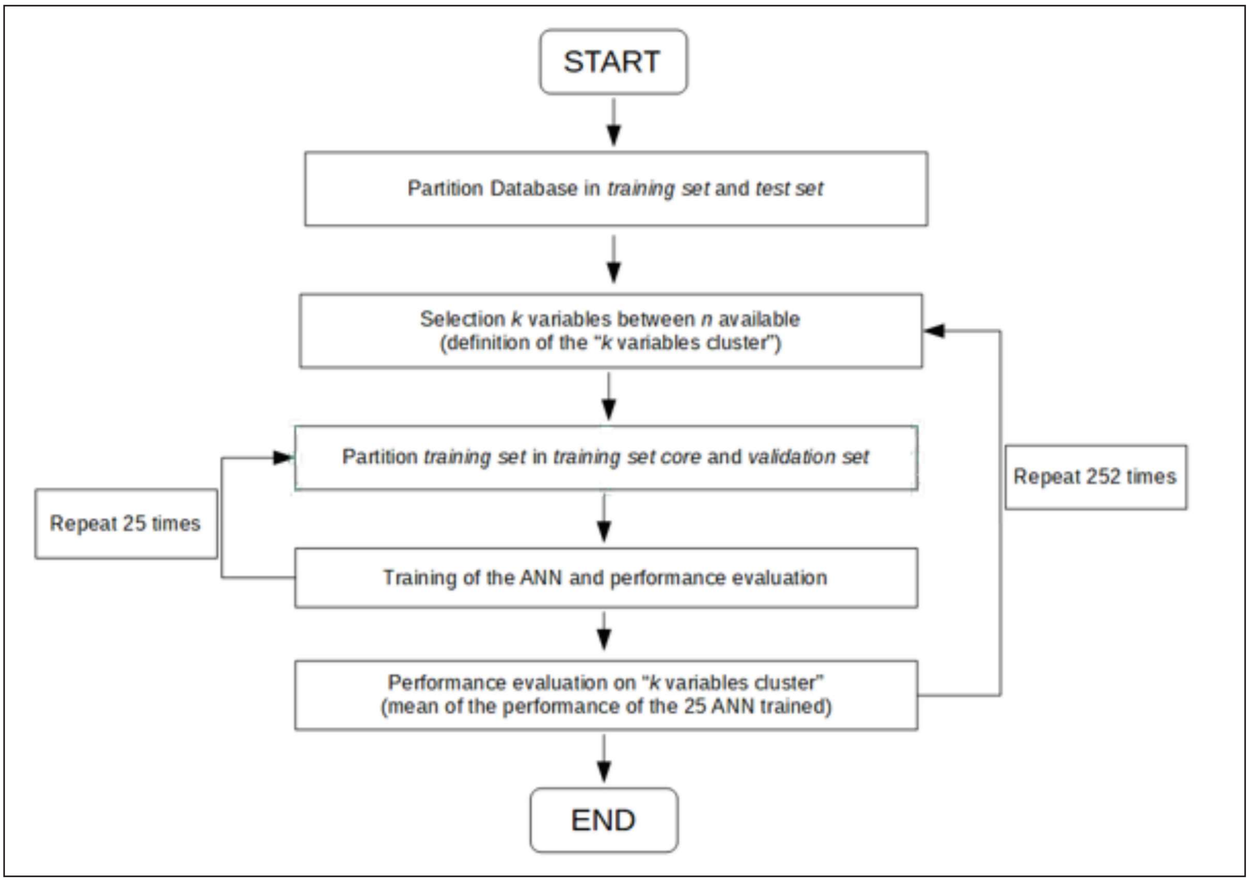

Figure 4. The algorithm for preparing the networks attributed to each cluster

Table 6. Comparison of the top 10 ANNs with corresponding MRA models

ANN	Inputs	R^2	
		ANN	MRA
4207	Commercial Surface, Finishing, Parking space, Sea-view, S_Margherita	0.5201	0.5502
4209	Commercial Surface, Finishing, Parking space, Sea-view, S_Margherita	0.5141	0.5332
4277	Commercial Surface, Finishing,, Distance historical centre, Orologio, S_Margherita	0.5117	0.5742
4622	Commercial Surface, Finishing, Swimming pool, Sea-view, S_Margherita	0.5076	0.5626
4142	Commercial Surface, Finishing, Parking space, Orologio, S_Margherita	0.5075	0.5213
4571	Commercial Surface, Finishing, Swimming pool, Floor, S_Margherita	0.4998	0.5345
4684	Commercial Surface, Finishing, Swimming pool, Parking space, S_Margherita	0.4986	0.5288
4690	Commercial Surface, Finishing, Swimming pool, Parking space, S_Margherita	0.4900	0.5411
4552	Commercial Surface, Finishing, Swimming pool, Floor, S_Margherita	0.4886	0.4962
5780	Commercial Surface, Garden, Finishing, Orologio, S_Margherita	0.4861	0.5186

average R^2. Since the goal of the work is to find ANN models with a higher than average performance, from now on the analysis will focus no longer on individual networks but on aggregated clusters.

Figure 5 presents the frequency, for each of the 252 clusters, of the R^2 of the networks and linear models.

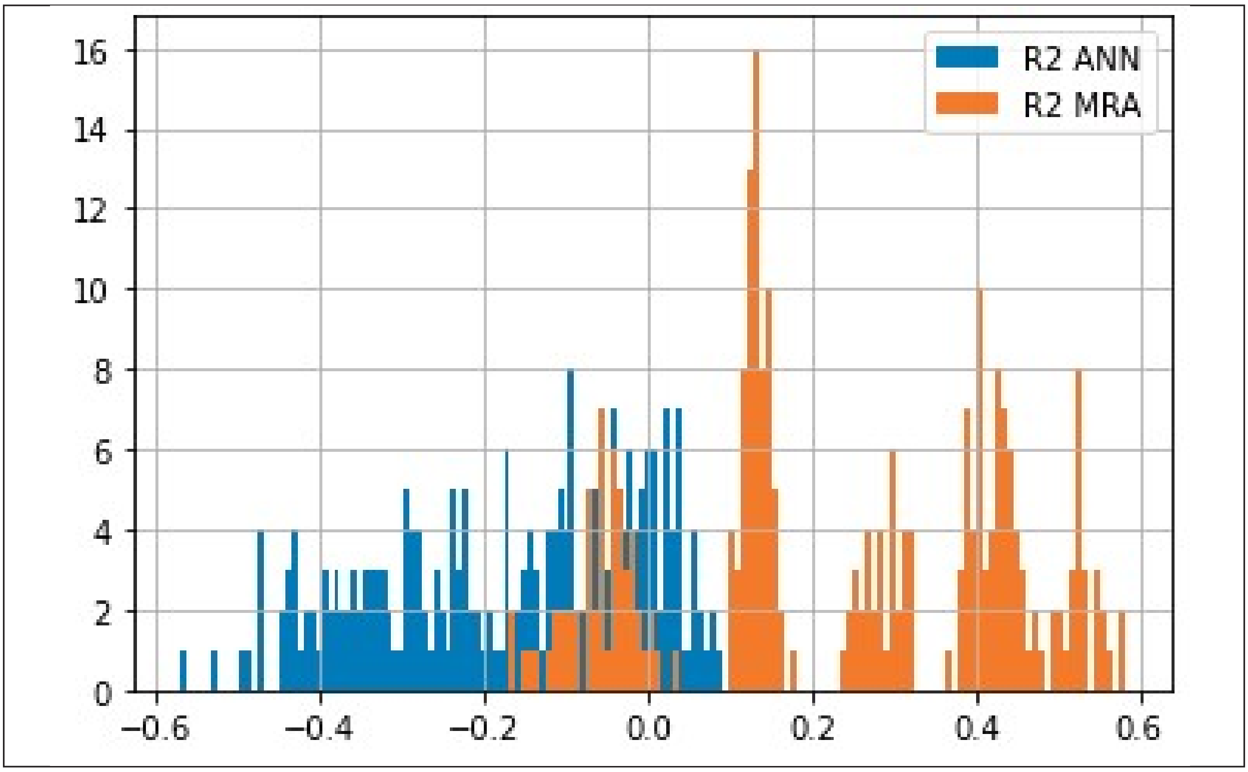

Figure 5. Frequencies of the observed R^2 coefficients of determination for ANN and MRA

The performance of the top 10 clusters is shown in Table 7.

Table 7. The R^2 coefficients of determination of the top 10 clusters of ANNs and MRAs

INPUTS	R^2	
	ANN	MRA
Commercial Surface, Finishes, Swimming Pool, Floor, S_Margherita	0.3265	0.5833
Commercial Surface, Finishes, Floor, Orologio, S_Margherita	0.3098	0.6086
Commercial Surface, Distance Historical Center, Sea-View, Floor, S_Margherita	0.2878	0.6451
Commercial Surface, Finishes, Distance Historical Centre, Floor, S_Margherita	0.2856	0.7066
Commercial Surface, Sea-View, Floor, Orologio, S_Margherita	0.2835	0.5476
Commercial Surface, Finishes, Parking Space, Orologio, S_Margherita	0.2833	0.6425
Commercial Surface, Finishes, Parking Space, Floor, S_Margherita	0.2814	0.6394
Commercial Surface, Swimming Pool, Distance Historical Centre, Floor, S_Margherita	0.2798	0.6105
Commercial Surface, Swimming Pool, Floor, Orologio, S_Margherita	0.2777	0.5261
Commercial Surface, Finishes, Sea-View, Orologio, S_Margherita	0.2743	0.5655

The difference between the average performance of the networks and MRA models is very high: the R^2 index of the ANNs is about half that of the linear regression models.

However, there are some other aspects that deserve further investigation, including:

1 the effect of the values initially assigned to the various parameters of the network to be trained. As already mentioned, in order to search for better networks a procedure is usually applied that tends to repeat the training process, each time assigning different initial values to the numerous parameters present within the network, and interrupting the training near local minimum

points of the error function which are always different from each other. In the case of linear models, however, by repeating the optimisation process, but showing the same training data, the results would not change;

2 the random choice of observations used as the training set, both for neural networks and linear models. It should be remembered that the construction of the training set is carried out in each of the 25 iterations in a completely random way, attributing 75% of the available data to the training set core, and the remaining 25% to the validation set. This randomness can cause the training set to contain observations that are significantly similar to those of the test set. While this certainly represents an advantage for linear models, for neural networks, which try to generalise the underlying phenomenon, the final effect appears to be more uncertain. In fact, it is conceivable that there is a negative correlation between the similarity of the data present in the training set and the values of the coefficients of determination (R^2), as it seems reasonable to expect that the more the data shown for training is different from each other, the more the model is in a position to «learn better», managing to generalise the behaviour of the underlying phenomenon, and therefore leading to better performance. On the other hand, training carried out with similar data could result in only a partial understanding of the dynamics of the phenomenon, penalising the results obtained in the presence of new and different observations compared to those present in the training set.

The effect, due to the similarity/diversity between the sets used for training the networks on their performance deserves further investigation, using similarity measures between the sets themselves.

The measure of similarity between the training and test data can be calculated using two metrics well known in the literature, namely the Euclidean distance and the Czekanowski distance.

Euclidean distance is probably the best-known metric and considers the distance between two points in an n-dimensional space, $P = (p_1, p_2, \dots, p_n)$ and $Q = (q_1, q_2, \dots, q_n)$ as:

$$\sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2} = \sqrt{\sum_{k=1}^{n} (p_k - q_k)} \tag{2}$$

The Czekanowski distance is instead defined as:

$$\frac{|p_1 - q_1| + |p_2 - q_2| + \cdots + |p_n - q_n|}{n} = \frac{\sum_{k=1}^{n} |p_k - q_k|}{n} \tag{3}$$

and provides values in the range [0,1]. In particular, values close to zero imply high similarity, while values close to 1 imply low similarity.

It can be observed that as the distance increases, the difference between the R^2 of the networks and the R^2 of the linear models tends to reduce considerably, confirming the expectations formulated a priori: the similarity between training and test data is certainly an advantage for linear models, while a dissimilarity between them implies a more robust training of the ANNs, with better performances compared to the corresponding linear models.

The graph in figure 6 highlights how the performance differences of the two models tend to concentrate in some clusters also present in figure 5; the densification is more evident in the case of the Euclidean distance which presents more marked variations. The apparently curious phenomenon is due to the effect of the introduction of the Boolean independent variables. The insertion or removal of these variables significantly affects the distance measurement, generating the clusters observed in figure 6.

4.4. Use of alternative algorithms for non-random selection of the training set

The results obtained with the random selection of the sets, with which to train the networks, have highlighted the opportunity to explore alternative procedures, in particular for the

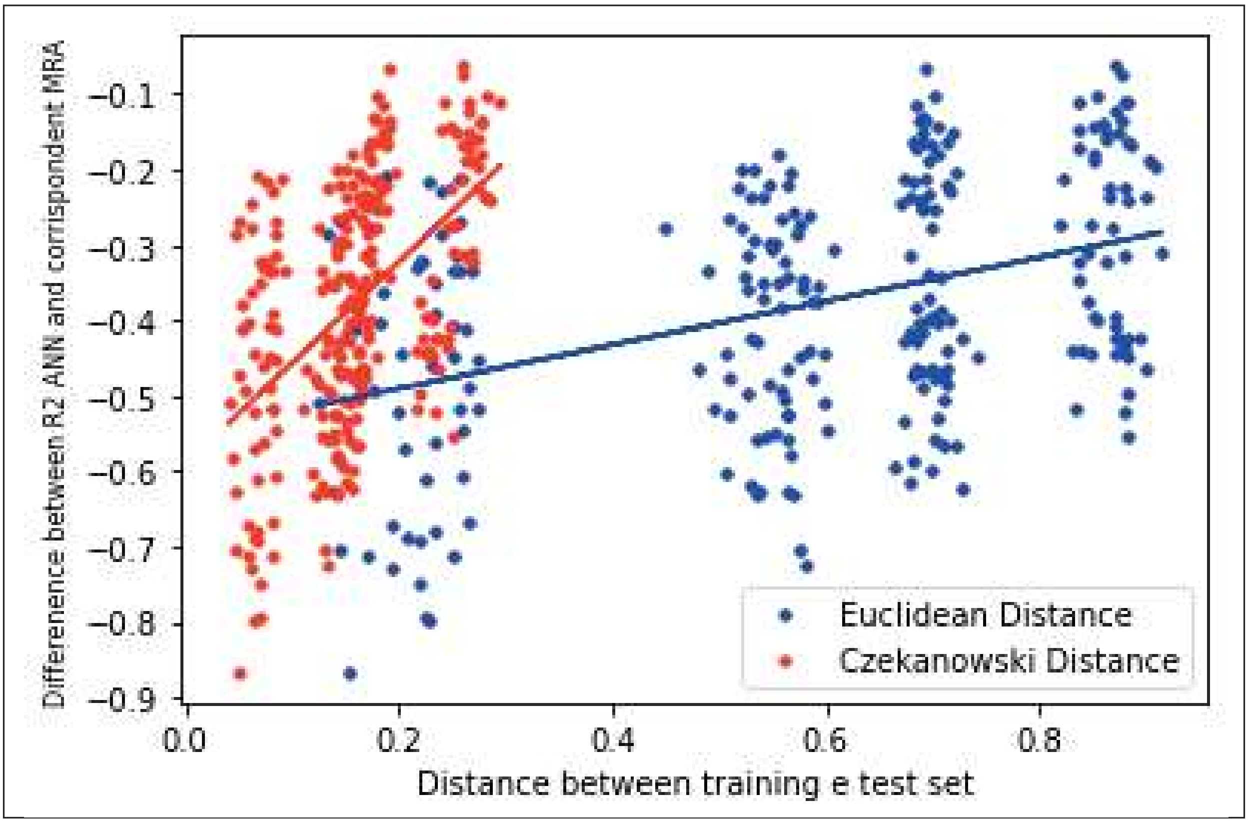

Figure 6. R^2 differences between ANN and MRA in relation to the distance of the training set from the test set.

construction of the training set core, moving from a totally random selection to an ad hoc defined selection.

The alternative algorithm we are going to propose initially considers only a limited part of the original training set to be included in the training set core, while most of the data is used in the validation set. In the following examples, 5% has been assigned to the core training set, while the remaining 95% is kept in the validation set. This first distribution occurs randomly.

Subsequently, the algorithm calculates a distance measured between each of the observations present in the validation set and the average of the core training set values. The data characterised by the maximum distance is removed from the validation set and added to the core training set. This procedure is repeated until the size of the core training set reaches 75% of the original training set, thus leaving 25% of the data available in the validation set. Unlike the approach that is usually used in practice, 75% of the observations of the training set are randomly selected only in the initial phase and constitute a marginal fraction of the training set core (just for the first 5%) which is thus made up of observations with very different characteristics between them. The non-random selection algorithm is shown in Figure 7.

A crucial role in the procedure is played by the metric used to calculate the distance between the mean of the core training set and the observations of the validation set. The proposed algorithm involves calculating the distances between the training set core and the validation set, using the following distance measures from time to time[3]:

1. Euclidean distance, always please refer to Formula 2 described previously;
2. Manhattan distance, calculated using the expression:

$$\sum_{k=1}^{n} |p_k - q_k| \tag{4}$$

---

[3] To calculate the distance measurements used, the Scipy Python library was used, which automatically provided the calculation of each measurement. In the Scipy manual it is possible to find some bibliographical references regarding their operation.

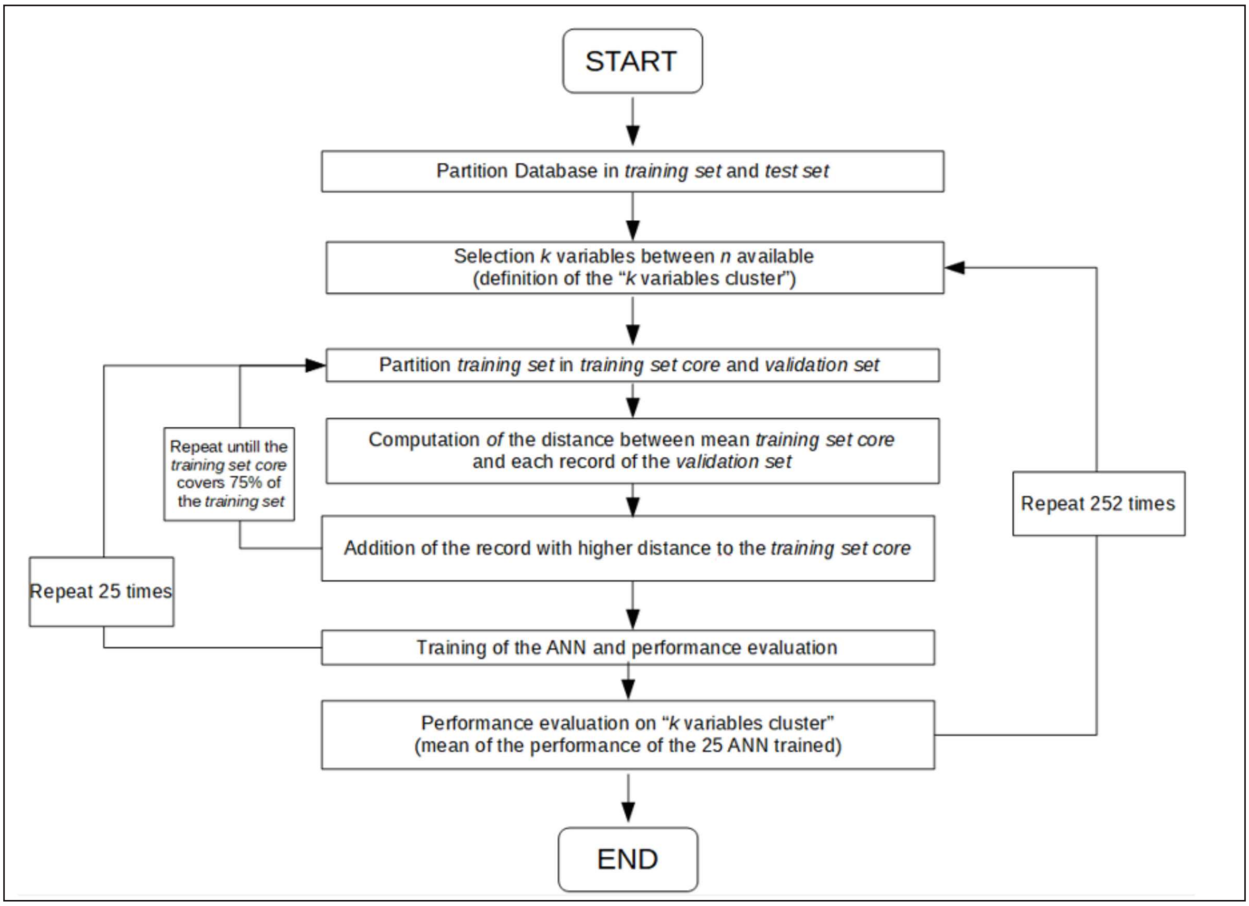

Figure 7. Cluster construction algorithm with non-random selection of data in the training set

3. Correlation distance, defined as follows:

$$1-\frac{\sum_{k=1}^{n}(p_k-\bar{p})-(q_k-\bar{q})}{\sqrt{\sum_{k=1}^{n}(p_k-\bar{p})^2}\sqrt{\sum_{k=1}^{n}(q_k-\bar{q})^2}} \tag{5}$$

4. Cosine distance, calculated according to the expression:

$$1-\frac{\sum_{k=1}^{n}(p_k q_k)}{\sqrt{\sum_{k=1}^{n}(p_k)^2}\sqrt{\sum_{k=1}^{n}(q_k)^2}} \tag{6}$$

5. Simple average of previously defined distance measurements.

Figure 8 shows the comparison between the performance of the networks, calculated as the difference of the R^2 of the same networks, compared to the corresponding linear model, and the Euclid and Czekanowski distances for all 252 clusters with the training set core constructed with respect to the average of the measurement distance just remembered[4].

---

[4] The performances obtained with respect to the other distance measurements are essentially similar to those reported in figure 8.

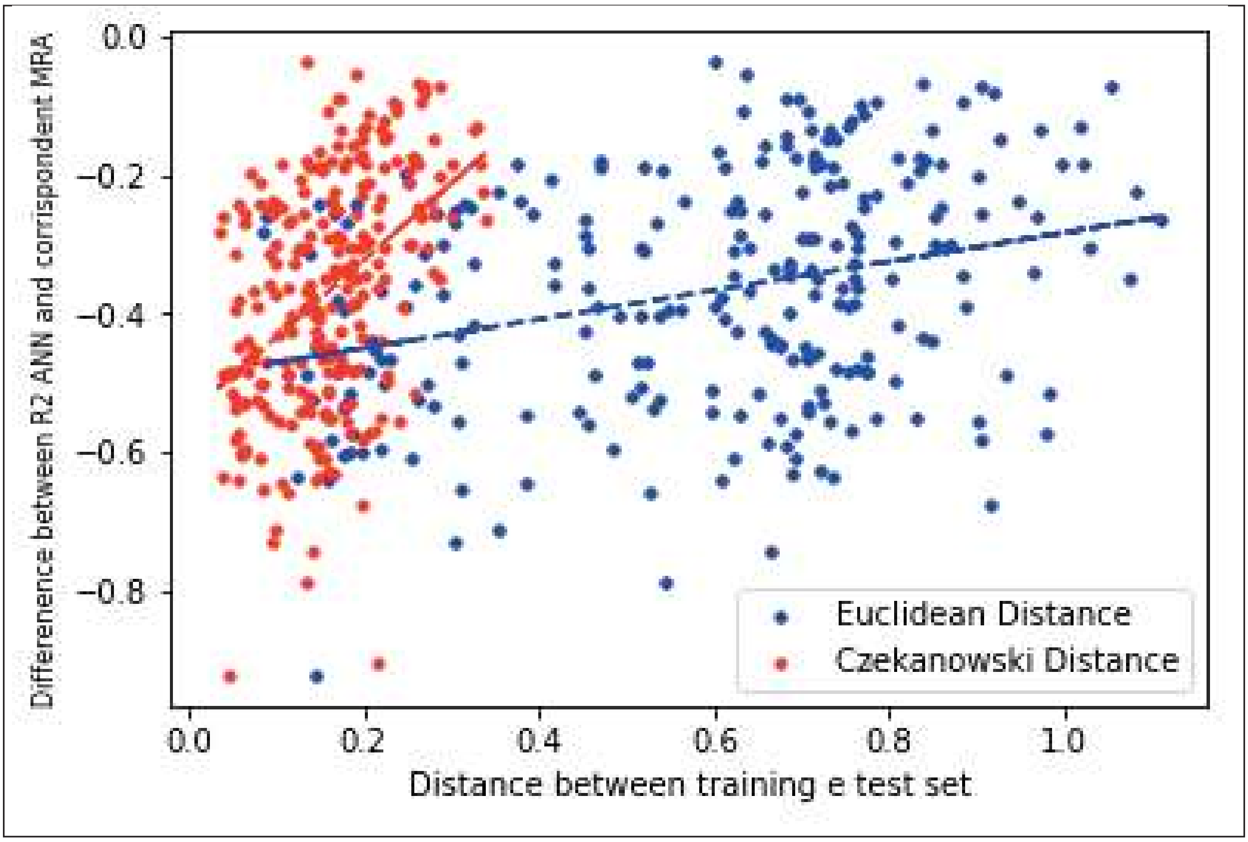

Figure 8. Performance of the 'simple average' model in relation to the distance between training set and test set

The intuition anticipated in the previous paragraph is therefore confirmed: the quality of the result depends on the distance between the training set core and the test set (always compared with the performances of the relative linear model): the more the new data used for the tests differs from those used in the training phase, the more the networks improve their performance compared to linear models.

4.5. Use of a fuzzy algorithm for non-random selection of the training set

The selection of training set data is therefore of great importance for the overall performance of non-linear models, acquiring even more relevance in the presence of databases limited.

In the previous paragraph, an algorithm was proposed that is capable of producing a non-random choice of the values with which to train the model, based on a specific distance measure, chosen ex-ante. A more extensive and general approach is now proposed, capable of obtaining more robust results. The model proposed here allows a ranking of distances, using an inferential system based on fuzzy logic (Fuzzy Inference System, FIS) (Zadeh, 1965 and 1968).

It should be noted that some concepts relating to fuzzy inferential systems, which are nowadays widespread, will be taken for granted, for which reference is made, for example, to Jang-Sun-Mizutani (1997).

The various distance measures proposed previously are used as input variables of the model, namely the Euclidean distance, the Manhattan, the Correlation and the Cosine. Each of these measurements is first normalised and brought within the interval [0,1].

On the domain of each of these measures, 3 Membership Functions (MF), $\mu(x)$, are subsequently constructed, which provide the degree of truthfulness for each value of $x \in$ [«low», «medium», «high»]. Basically, for each distance measure of x the functions return a value that can be read as the degree of truth of the statement «the distance measure x is X».

The fuzzy set A will therefore have the expression:

$$A: \{ \mu A(x) : X \rightarrow (0,1) \} \quad (7)$$

For instance, the «low» membership function will return a value equal to 1 if the normalised distance is equal to 0. If the normalised distance, however, were included in the interval (0.5,1) the «low» MF would return a value of 0, while the two other MFs «medium» and «high» would have a value greater than 0. With a normalised distance equal to 1, the «low» and «medium» MFs would have a value of 0, while the «high» would be the only one activated and would return a value of 1.

Since no particular information is available on the optimal type of MFs to use, nor specific indications on whether it is better to consider MFs not equally distributed in the domain of the various inputs, it was decided to generalise the approach as much as possible and to consider for each distance and appropriate triangular membership functions, equidistributed in the domain.

The result, applied as an example to the Euclidean distance variable, but will also be completely equivalent for the other types of distance measurement used, is a distribution like the one proposed in Figure 9.

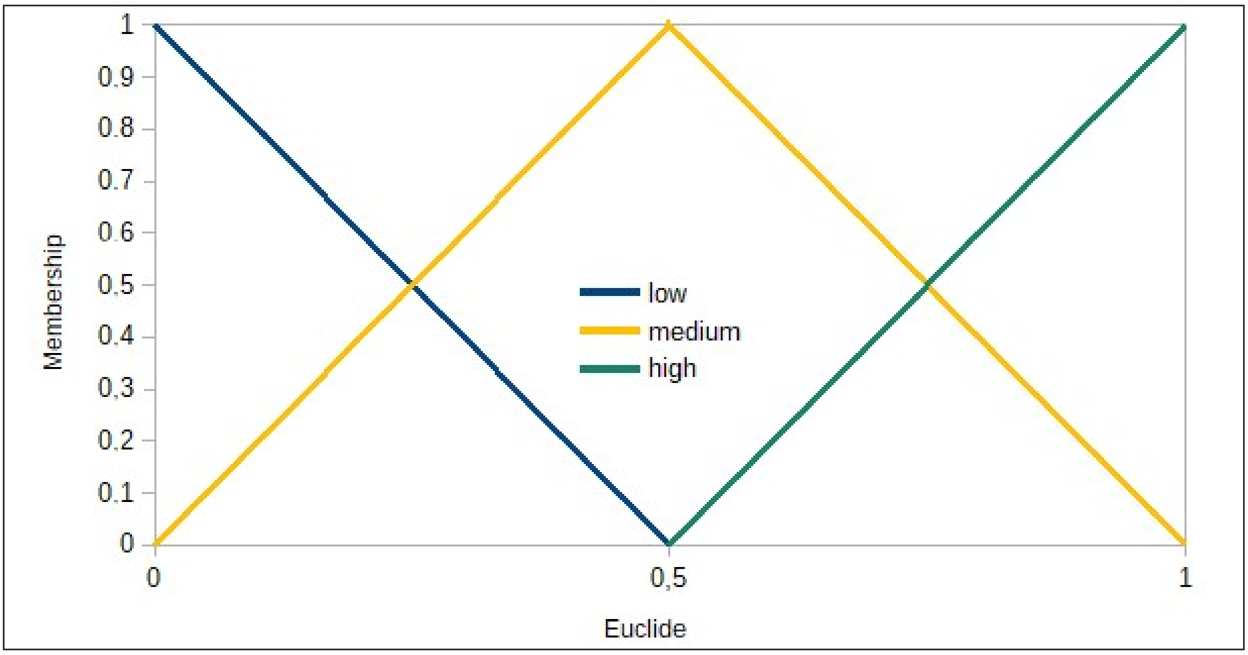

Figure 9. Membership functions for the description of Euclidean-type distance

The inferential which relates the input and output variables, is described using the following fuzzy rules:

Table 8. The fuzzy rules considered in the FIS

RULE 1
IF
Euclidean =[low] AND Cosine=[low] AND Correlation=[low] AND Manhattan[low]
THEN
Distance=[low]

Follow **Table 8.** The fuzzy rules considered in the FIS

RULE 2
IF
Euclidean =[medium] OR Cosine=[medium] OR Correlation=[medium] OR Manhattan=[medium]
THEN
Distance=[medium]

RULE 3
IF
Euclidean =[high] AND Cosine=[high] AND Correlation=[high] AND Manhattan=[high]
THEN
Distance=[high]

The inferential system used can therefore be graphically summarised as in Figure 10.

The values obtained from the three inferential rules described above are aggregated into three new MFs whose functionality is equivalent to those used to qualify the original distance measures, but which now express the degree of truthfulness of the linguistic expression «the overall distance (between training set core and the new observation) is X» with X taking a value in the finite set of term-sets [«low», «medium», «high»].

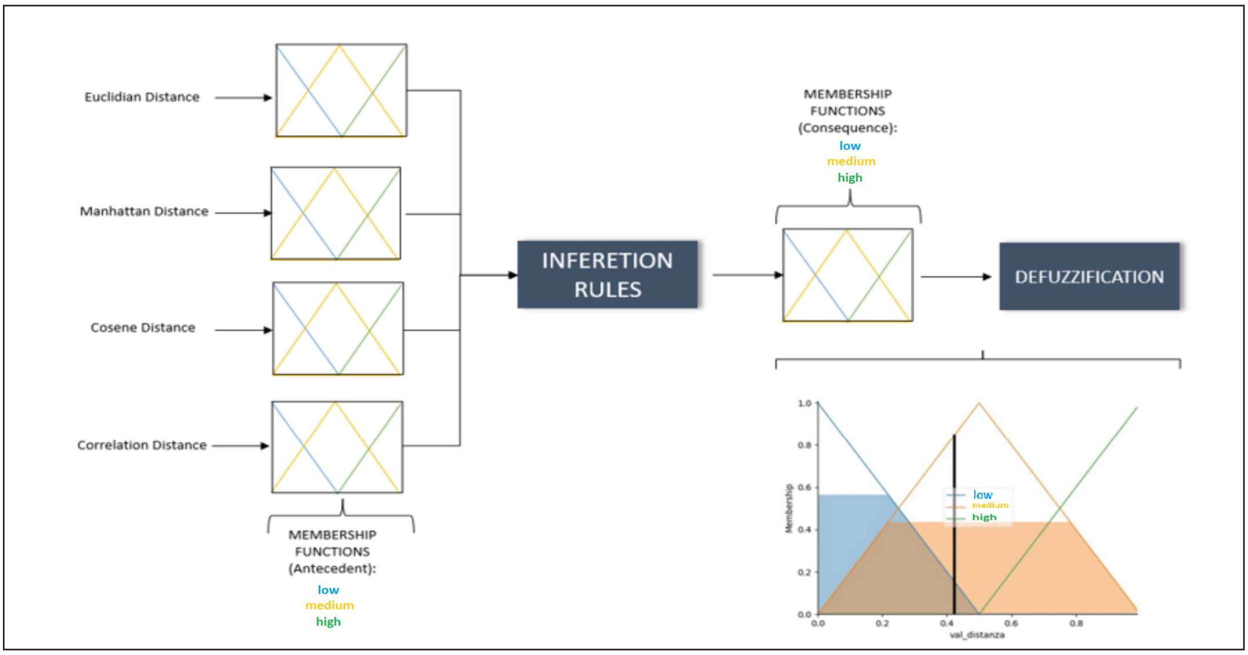

Figure 10. The structure of the Fuzzy Model.

The final step is represented by the «defuzzification» phase, which allows the fuzzy values present in the model to be transformed into the final value to be attributed to the ranking of the new observation. The defuzzification method applied is the «Centre of gravity», expressed through the following Formula 8:

$$Center\ of\ Gravity = \frac{\int_Y y\mu(y)dy}{\int_Y \mu(y)dy} \tag{8}$$

where Y is the domain of the MFs, i.e. the interval [0,1], and μ(y) represents the set of MFs that characterise the ranking.

The procedure proposed above is repeated for each observation present in the validation set, allowing a precise level of distance to be calculated with respect to the training set core. The observation that will present the higher distance is then removed from the validation set and inserted into the training set core. The procedure is repeated recursively until a specific training set core, characterised by the desired dimensions is obtained (in the case under examination the procedure was repeated until reaching 124 observations, i.e. 75% of the training set).

The following Figure 11 shows a comparison between the results obtained with the proposed fuzzy model compared to those obtained with traditional ANN models. More precisely, the difference between the averages of the R^2 observed in each of the 252 clusters is calculated from the approach with a selection of the training set obtained with the fuzzy approach compared to that obtained with ANN with a random selection of the training set. Consistently, the results of the two methodologies are always obtained by comparing the performances of the networks with those of the respective linear models.

In summary, positive values show how the proposed approach allows the obtaining, for each cluster, of an improvement in R^2 compared to the traditional approach of a random selection of the training set.

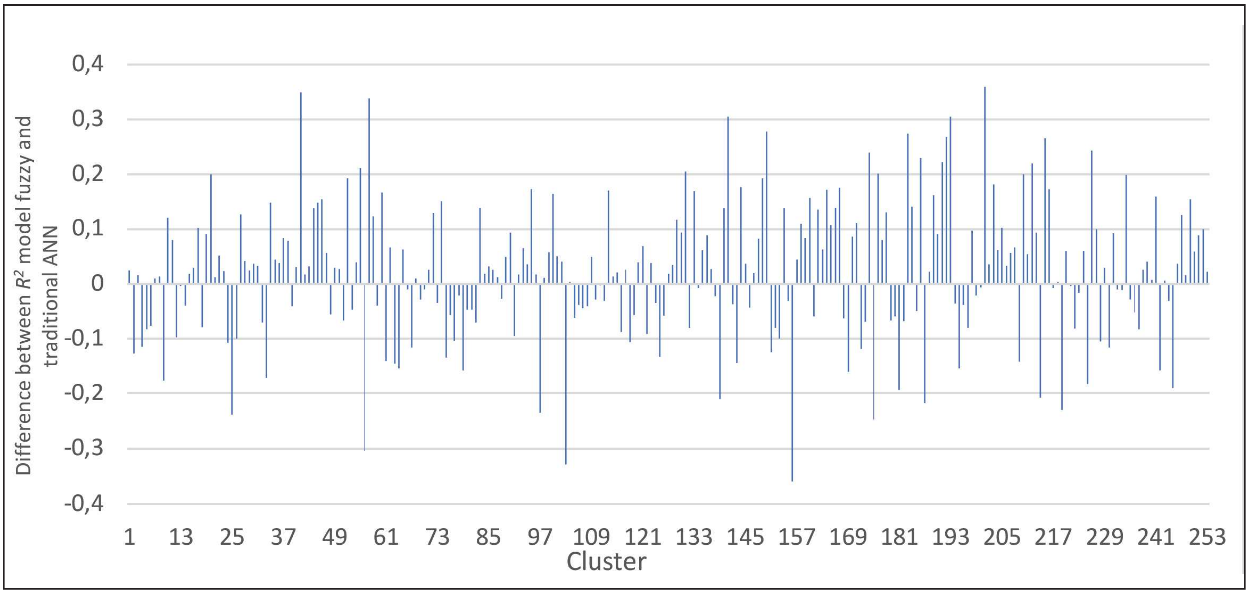

Figure 11. Comparison between results of fuzzy models and traditional ANN models

5. Results

The performances calculated for the models proposed in the previous paragraphs are presented below. Please remember that the performance measures are not related to individual neural networks but are calculated as averages of the R^2 of the clusters obtained by grouping the input variables into groups of 5, so as to fill the entire space of possible input combinations. In this way, more robust results can be obtained since they depend less on the initial parameters of the network training.

The differences between the R^2 of the individual model compared to the R^2 of its linear model always show the better ability of the linear models than the networks for each of the 252 clusters.

The intuition of using a model that considers a combination of multiple distance metrics therefore seems to lead to more convincing results than using a model based on a single distance metric. In particular, the suggested fuzzy approach actually seems to be a valid system for selecting the core training set, allowing for an improvement in the predictive performance of the networks.

Figure 12 shows a comparison between the linear models and the related networks using box plot[5]. It can be observed how the proposed fuzzy model offers not only a lower error but is also characterised by less dispersion.

---

[5] For further information, please refer to a https://en.wikipedia.org/wiki/Box_plot

Table 9. The results of the ANN models compared to the corresponding MRA models

MODEL	Difference between R² Models Vs correspondent MRA
Classic ANN (Crisp)	-0.3784
Model fuzzy	-0.3564
Model EuclideAN	-0.3809
Model Manhattan	-0.3782
Model Correlation	-0.3803
Model Cosine	-0.3839
Model Mean	-0.3691

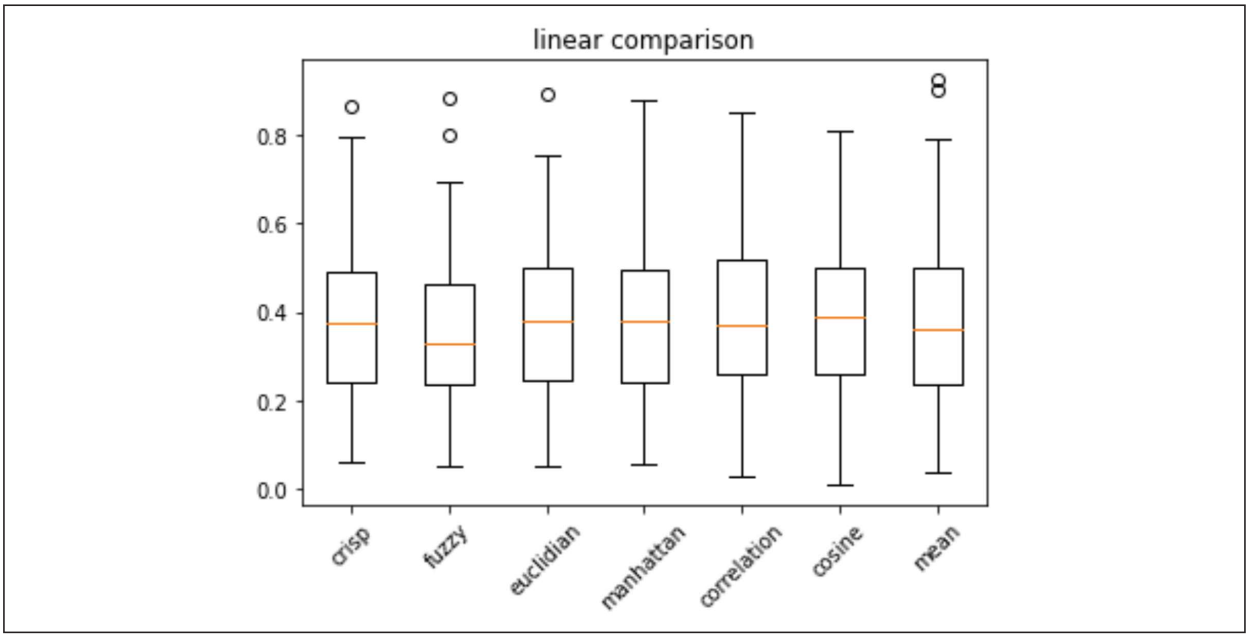

Figure 12. The comparison between MRA models and related ANNs

To also test the statistical significance of these results, a one-tailed t-test was finally applied to these results, with null hypothesis that the R^2 parameter of the individual models is equal to that of the ANN trained on randomly selected data, versus an alternative hypothesis that this is higher.

Table 10. Student's t test between non-random and random RNA models

T-TEST RESULTS		
Model	**Statistical value**	***p*-value**
Model fuzzy	2.8471	0.0022
Model Euclidean	-0.2460	0.5971
Model Manhattan	-0.2539	0.6001
Model Correlation	-0.3201	0.6255
Model CosINE	-0.3809	0.6483
Model Mean	0.6399	0.2612

The results show that, in the present case, only the model using a fuzzy distance measure produces results that are significantly different from, and still superior to, those obtainable with random set selection models.

6. Conclusions

The literature on the use of quantitative models for property valuations has focused in recent decades first on econometric approaches, and only more recently on artificial neural networks. The results obtained in numerous studies are, in both cases, satisfactory, with some prevalence of neural networks in the most recent studies (Mankad, 2022; Štubňová et al., 2020; Torres-Pruñonosa et al., 2021).

However, there remains among estimators some distrust in the practical use of ANNs due to lack of transparency, high computation time, high data requirements, and, not least, frequent misalignment from the a priori market (Valier, 2020).

An additional reason for caution lies in the repeatability of the results generated by the networks. In fact, use of different network models (and different software) can produce very different results.

In this paper, the performance of well-established MRA models were tested with reference to real market data, against those obtainable with various artificial neural networks in a database of modest size.

Specifically, a procedure capable of testing various networks obtained by recombining the ten independent variables deemed significant, either from the market or from correlation analysis with unit prices, was developed. A total of 252 different networks were obtained by recombining five independent variables chosen from the ten available that were tested. Each of these networks were trained 25 times in order to minimise the effect of random selection of the training set and test set, resulting in 6,200 networks grouped into clusters, each identified by the five independent variables used. The average performance of each cluster was then compared with that obtained with the corresponding MRA models in the respective test sets. In the present case, MRA models produced on average superior results to ANN models. This is due both to the limited database available and to the selection of independent variables made by statistical approaches, as well as to the difficulty of neural networks to approximate linear functions and, last but not least, to the intrinsic structure of the phenomenon under consideration.

The performance of any quantitative approach also depends on the characteristic training set and test set considered from time to time. In particular it depends on the homogeneity of the training set data and the degree of similarity between training set and test set. A particularly homogeneous training set will generate a model with poor performance on a very heterogeneous test set. Likewise, a network trained on a training set that is very different from the test set will provide a similarly poor performance. Ultimately, training requires diverse experience and performance evaluation similarities across the sets.

To this end this paper investigated the effect of the characteristics of different sets on the performance of ANNs in databases of limited size, as frequently observed in studies of housing markets.

In summary, the following was found:

1 The similarity between the training and test data is certainly an advantage for MRA models, while dissimilarity produces more robust training of ANNs; this result is also confirmed by considering the similarity between the core training set and the test set;

2 The effect of similarity in ANN performance varies with the metric used, and the one most discriminating ANN performance against MRA models is the fuzzy distance measure.

In conclusion, it can be said that the use of ANNs does not necessarily produce better results than MRA models, as confirmed by some studies conducted by real estate market scholars (Torres-Pruñonosa et al., 2021). In particular, with limited and homogeneous databases, MRA models seem competitive if not superior. In contrast, with large (big data) and heterogeneous databases, ANNs are a viable alternative.

It follows that, at the state of the art, the use of ANNs seems limited to automatic valuation of property masses. However, given the rapid evolution of ANNs, it is foreseeable that in the near future they can be effectively employed in point assessments.

Authors' contribution

The work should be attributed equally to the authors.

Disclaimer

This paper should not be reported as representing the views of Banca Ifis. The views expressed are those of the authors and do not necessarily reflect those of Banca Ifis.

Bibliography

Alpaydin, E. (2020). Introduction to machine learning. MIT press.

Bellman, R. (1957). Dynamic programming. Princeton University Press, Princeton, NJ, USA, 1, 3-25.

Binoy, B. V., Naseer, M. A., Anil Kumar, P. P., & Lazar, N. (2022). A bibliometric analysis of property valuation research. International Journal of Housing Markets and Analysis, 15(1), 35-54.

Bishop, C. M. (2006). Pattern recognition and machine learning. Springer google schola, 2, 1122-1128.

Chiarazzo, V., Caggiani, L., Marinelli, M., & Ottomanelli, M. (2014). A neural network based model for real estate price estimation considering environmental quality of property location. Transportation Research Procedia, 3, 810-817.

Diamond Jr, D. B., & Smith, B. A. (1985). Simultaneity in the market for housing characteristics. Journal of Urban Economics, 17(3), 280-292.

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

Evans, A., James, H., & Collins, A. (1992). Artificial Neural Networks: An Application to Residential Valuation in the UK. University of Portsmouth, Department of Economics.

Gareth, J., Daniela, W., Trevor, H., & Robert, T. (2013). An introduction to statistical learning: with applications in R. Spinger.

Jang, J. S. R., Sun, C. T., & Mizutani, E. (1997). Neuro-fuzzy and soft computing-a computational approach to learning and machine intelligence [Book Review]. IEEE Transactions on automatic control, 42(10), 1482-1484.

Kalliola, J., Kapočiūtė-Dzikienė, J., & Damaševičius, R. (2021). Neural network hyperparameter optimization for prediction of real estate prices in Helsinki. PeerJ computer science, 7, e444.

Khoshnoud, M., Sirmans, G. S., & Zietz, E. N. (2023). The Evolution of Hedonic Pricing Models. Journal of Real Estate Literature, 31(1), 1-47.

Lancaster, K. J. (1966). A new approach to consumer theory. Journal of political economy, 74(2), 132-157.

Lawrance, J. (1994). Introduction to neural networks. Design theory and applications. Nevada City: California Scientific Software.

Lisi, G. (2019). Property valuation: the hedonic pricing model–location and housing submarkets. Journal of Property Investment & Finance, 37(6), 589-596.

Liu, J. G., Zhang, X. L., & Wu, W. P. (2006, May). Application of fuzzy neural network for real estate prediction. In International symposium on neural networks (pp. 1187-1191). Berlin, Heidelberg: Springer Berlin Heidelberg.

Mankad, M. D. (2022). Comparing OLS based hedonic model and ANN in house price estimation using relative location. Spatial Information Research, 30(1), 107-116.

McCluskey, W., Davis, P., Haran, M., McCord, M., & McIlhatton, D. (2012). The potential of artificial neural networks in mass appraisal: the case revisited. Journal of Financial Management of Property and Construction, 17(3), 274-292.

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics, 5, 115-133.

Moro, M. F., Weise, A. D., & Bornia, A. C. (2020). Model hybrid for sales forecast for the housing market of São Paulo. Real Estate Management and Valuation, 28(3), 45-64.

Núñez Tabales, J. M. N., Caridad, J. M., & Carmona, F. J. R. (2013). Artificial neural networks for predicting real estate price. Revista de Métodos Cuantitativos para la Economía y la Empresa, 15, 29-44.

Ozanne, L., & Malpezzi, S. (1985). The efficacy of hedonic estimation with the annual housing survey.

Evidence from the demand experiment. Journal of Economic and Social Measurement, 13(2), 153-172.

Pagourtzi, E., Assimakopoulos, V., Hatzichristos, T., & French, N. (2003). Real estate appraisal: a review of valuation methods. Journal of property investment & finance, 21(4), 383-401.

Peterson, S., & Flanagan, A. (2009). Neural network hedonic pricing models in mass real estate appraisal. Journal of real estate research, 31(2), 147-164.

Rampini, L., & Re Cecconi, F. (2022). Artificial intelligence algorithms to predict Italian real estate market prices. Journal of Property Investment & Finance, 40(6), 588-611.

Root, T. H., Strader, T. J., & Huang, Y. H. J. (2023). A Review of Machine Learning Approaches for Real Estate Valuation. Journal of the Midwest Association for Information Systems (JMWAIS), 2023(2), 2.

Rosen, S. (1974). Hedonic prices and implicit markets: product differentiation in pure competition. Journal of political economy, 82(1), 34-55.

Shinde, A., Dange, N., Patane, N., Ghopal, S., & Beera, V. (2019). Real Estate Properties Assessment using Deep Neural Network.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1), 1929-1958.

Štubňová, M., Urbaníková, M., Hudáková, J., & Papcunová, V. (2020). Estimation of residential property market price: comparison of artificial neural networks and hedonic pricing model. Emerging Science Journal, 4(6), 530-538.

Tay, D. P., & Ho, D. K. (1992). Artificial Intelligence and the Mass Appraisal of Residential Apartments. Journal of Property Valuation and Investment, 10(2), 525-540.

Temur, A. S., Akgün, M., & Temur, G. (2019). Predicting housing sales in Turkey using ARIMA, LSTM and hybrid models.

Torres-Pruñonosa, J., Garcia-Estevez, P., & Prado-Roman, C. (2021). Artificial neural network, quantile and semi-log regression modelling of mass appraisal in housing. Mathematics, 9(7), 783.

Valier, A. (2020). Who performs better? AVMs vs hedonic models. Journal of Property Investment & Finance, 38(3), 213-225.

Valier, A., & Micelli, E. (2020). Automated models for value prediction: A critical review of the debate. Valori e Valutazioni, (24).

Wiśniewski, R. (2017). Modeling of residential property prices index using committees of artificial neural networks for PIGS, the European-G8, and Poland.

Witte, A. D., Sumka, H. J., & Erekson, H. (1979). An estimate of a structural hedonic price model of the housing market: an application of Rosen's theory of implicit markets. Econometrica: Journal of the Econometric Society, 1151-1173.

Worzala, E., Lenk, M., & Silva, A. (1995). An exploration of neural networks and its application to real estate valuation. Journal of Real Estate Research, 10(2), 185-201.

Zadeh, L. A. (1965). Fuzzy sets. Information and control, 8(3), 338-353.

Zadeh, L. A. (1968)., Fuzzy algorithms - Information and Control, 12(2), pp. 94-102