Lancaster University Management School
Working Paper
2010/023

Advances in forecasting with artificial neural networks

Nikolaos Kourentzes and Sven Crone

The Department of Management Science
Lancaster University Management School
Lancaster LA1 4YX
UK

Working Paper

Advances in forecasting with artificial neural networks

Nikolaos Kourentzes and Sven F. Crone, Lancaster Centre for Forecasting, Lancaster University Department of Management Science
Email: n.kourentzes@lancaster.ac.uk

There is decades long research interest in artificial neural networks (ANNs) that has led to several successful applications. In forecasting, both in theoretical and empirical works, ANNs have shown evidence of good performance, in many cases outperforming established benchmark models. However, our understanding of their inner workings is still limited, which makes it difficult for academicians and practitioners alike to use them. Furthermore, while there is a growing literature supporting their good performance in forecasting, there is also a lot of scepticism whether ANNs are able to provide reliable and robust forecasts. This analysis presents the advances of ANNs in the time series forecasting field, highlighting the current state of the art, which modelling issues have been solved and which are still critical for forecasting with ANNs, indicating future research directions.

Keywords: neural networks; forecasting

Advances in forecasting with artificial neural networks

Nikolaos Kourentzes and Sven F. Crone, Lancaster Centre for Forecasting, Lancaster University Department of Management Science

1. Introduction

It has been almost half a century since the first application of artificial neural networks (ANNs) to regression and forecasting problems. Since then, a lot of research has been invested to improve our knowledge of modelling and using them, which has generated a wide variety of applications in forecasting and several other fields like control, optimisation, classification, pattern recognition, data mining, etc [1, 2]. ANNs are biology inspired models that mimic neural networks in the human brain, which allows them to learn from the available information and generalise [3, 4]. A decade old survey [2] on ANNs identified the following key features that make them useful in forecasting:

1. ANNs are data driven self-adaptive methods with very few a priori assumptions. They learn the underlying data generating process from the training data, without the need to input hard to infer theoretical knowledge. This makes them attractive as it is often easier to have wealth of data for a problem than good understanding of the laws that govern it.
2. They can generalise in the future. Once an ANN has been trained to learn the known sample, they are able to infer the relationship between the inputs and the outputs and simulate well future behaviours, even in the presence of noise. This is a necessary model property for forecasting applications.
3. They are universal function approximators. It has been shown that relatively simple structures of ANNs can approximate any function to an arbitrary degree of accuracy, with the same model form [5, 6]. This inherent flexibility allows them to model observed or unobserved relationships in the data, without assuming a rigid functional form, which is common in statistical models, thus allowing them to model complex real systems that are not always fully understood.
4. They are flexible nonlinear models. In the forecasting literature there are several nonlinear models, however they usually assume a specific type of nonlinearity, which

may not describe well the observed data. ANNs have the advantage that there is no need for apriory knowledge of the nature of the nonlinearity and are entirely data-driven.

The same survey concludes with four important research questions that must be answered to improve of understanding of ANNs and make their use in forecasting accurate and reliable. How do ANNs model time series that allows them to produce better results than conventional methods? How to systematically build an ANN for a given forecasting problem? What is the best training algorithm/method for time series forecasting? What is the effect of sampling and data pre-processing for ANNs and how should they be carried out?

With this study we aim to explore the published forecasting literature since then and try to assess whether the evidence supports the portrayed key advantages of ANN in forecasting, investigate whether the stated key research challenges have been resolved and identify the current important research questions in the field. Since the last extensive review in forecasting with ANNs [2] a wealth of research has been published, but remains largely disconnected, making it difficult to extract conclusions about the application of ANNs in forecasting as a whole. With this study we try to highlight the big picture of ANNs in forecasting. To accomplish this, a literature review of major established management science and forecasting journals is done in order to identify the current trends. We show which are the current modelling methodologies for ANNs and the main application areas, the current advances and how ANNs fare when compared to more traditional forecasting models. Furthermore, we investigate the validity of the published research in the light of the criticism received by the forecasting literature. The study concludes with the current important modelling issues for ANNs and a discussion about future research.

This study is organised as follows. Section 2 provides a brief overview of the literature survey design. Section 3 discusses the findings of the survey while section 4 presents the conclusions of this study.

2. Research methodology

The main bulk of the papers analysed here was collected by performing an online survey using the ISI Web of Knowledge database[1]. The search was focused on influential journals in forecasting, operational research and management science. The journals were selected due to their relevance with forecasting and their ranking in two different systems, the Vienna List[2] [7] and the impact factor as measured at the ISI Web of Knowledge [8]. Table I lists these journals with their respective scores in both ranking systems.

Journals that mostly specialise in ANNs from an engineering perspective were not included due to their limited relevance with economic/business forecasting. This is a limiting factor of this survey, however the aim of this study was to explore extensively the ANN forecasting literature with a special interest to operational research and management science problems; therefore, we follow the criteria set by Adya and Collopy [9] to exclude weather, biological processes and other non-business applications which are numerous in those journals.

Table I: Ranking of Journals in the Literature Survey

Journal	Vienna List		ISI Web of Knowledge		
	New list*	Old list**	Impact Factor	5-Year Factor	Impact
Computers and Operations Research (C&OR)	A	A	1.366	1.789	
Decision Sciences (DS)	A	A	2.318	3.131	
European Journal of Operational Research (EJOR)	A	A	1.627	2.084	
International Journal of Forecasting (IJF)	-	B	1.685	1.596	
Journal of Forecasting (JF)	A	A	0.508	1.018	
Management Science (MS)	A+	A+	2.354	4.065	
Naval Research Logistics (NRL)	A	A	0.735	0.993	
Operations Research (OR)	A+	A	1.463	2.547	

*The new list contains 322 journals ranked A+ (32) and A; ** The old list ranks 1,877 journals classified as A+ (42), A (701), B (735), C (250) and D (142). The numbers in brackets show the number of journals in each category.

The keywords used to perform the search were relatively broad, ensuring that all the articles of interest would be identified[3]. No publication year restrictions were enforced,

[1] http://portal.isiknowledge.com/portal.cgi

[2] Vienna list is compiled by Wirtschafts Universitat Wien and the journals are graded from A+ to D. The journals used in this study are graded from A+ to B.

[3] Those were: "Neural AND Net*" and "Multilayer AND perce*". The results were manually filtered to identify relevant papers to forecasting. These words were selected after experimentation with different combinations to ensure a very wide range of results. "Forecasting" and similar words were not used as keywords in order to find related papers, even if they had no such keywords associated to them.

however most online articles date after 1995. For older papers only their abstracts were available online. The printed articles were retrieved for the highly cited papers published before 1995. This is not a limiting factor of this study, since the majority of older publications are analysed in previous reviews [2]. The total number of relevant papers that were used in this study is 126 and a list of them can be found in table XIII.

To ensure a systematic analysis of the papers we follow the suggestions in the literature on what constitutes a well implemented and valid ANN paper. Adya and Collopy [9] stressed that several of the ANN forecasting papers do not provide reliable or valid conclusions, because of lacking experimental design, evaluation or documentation, or the networks were not implemented well. To measure these, they set some criteria. The ANN models have to be compared with well-accepted benchmarks, use ex-ante comparisons, a reasonable sample of forecasts, adequate training, stability of the performance and generalisation capabilities. Crone and Preβmar [10] go one step further and construct a framework that enables a systematic evaluation to identify heuristics and sound guidelines in ANN modelling by documenting the individual modelling decisions in each paper. They observe that due to the vast degrees of freedom in ANN modelling it is important that all these are analysed. This leads to an important point; it is imperative that the authors try to make their papers as replicable as possible by documenting all modelling decisions. This will allow transparent analysis of their models and eventually better understanding of what makes ANN models perform well or not. Furthermore, in the forecasting literature there are extensive guidelines of what constitutes an effective validation and a good experimental design [11, 12], which as we will discuss in the following sections is often overlooked in the ANN literature. Here, we create an amalgam of the suggestions briefly discussed above, which is implemented in practice by examining each paper across 42 different dimensions of analysis. The main benefit is that it allows a systematic investigation of the papers for contribution, validity of the evaluation and implementation, assess the replicability and extract knowledge on ANN modelling practices. The dimensions of analysis are classified in six major categories; the general information, like year of publication and area of publication, relevant information to the dataset used in the paper, the network architecture, the network training, the evaluation scheme and the conclusions. A detailed breakdown of these categories into the individual dimensions of analysis can be found in table II.

Table 0-II: Categories and dimensions of the literature survey

General			
1	Author	3	Journal
2	Year	4	Area of application
Time Series			
5	Uni/Multivariate time series	10	Pre-processing
6	Time series type	11	Scaling
7	Real/Synthetic time series	12	Train/Valid/Test set sizes
8	Sample size	13	No. of time series used
9	Time series granularity		
Architecture			
14	ANN type	21	Number of output nodes
15	Method to model the ANN	22	Forecat horizon
16	Number of input nodes	23	Transfer function
17	Method to identify input nodes	24	Output function
18	Number of hidden layers	25	Shortcut connections
19	Number of hidden nodes	26	Pruning
20	Method to identify hidden layer/nodes	27	Iterative/Multiple step-ahead forecast
Training			
28	Training method	32	Learning rate
29	Epochs/Iterations	33	Momentum rate
30	Error function	34	Initialisations
31	Early stopping		
Evaluation			
35	Error Metric	39	Comparison with other models
36	In-sample evaluation	40	Which models
37	Ex-ante evaluation	41	Generalisability of the results
38	Fixed/Rolling origin evaluation		
Evaluation			
42	ANN found better?	43	Additional info/notes

It was impossible to fill all the dimensions of analysis for each paper, since most of this information is either not documented or too vague. Furthermore, there is a strong lack of standardisation in the ANN nomenclature that makes the correct classification challenging. Once all the articles were analysed then the collected information was grouped to allow inference of meaningful information. The results are presented by category in the following section.

3. Survey findings

3.1. Publication trends

Initially, we explore the publication trends. Figure 1 presents the number of papers per year and journal since 1992. Note that the 2009 data includes only papers published in the first 7 months of the year. Over the years there is an increasing number of publications

that use ANNs in forecasting, demonstrating that it is an active research topic. There seems to be a cycle of 4 to 5 years that the number of publications peaks. More than 75% of the papers are published in three journals, the Journal of Forecasting, the International Journal of Forecasting and the European Journal of Operational Research, in order of percentage. Note that there are no forecasting related papers with ANNs in the Naval Research Logistic and Operations Research journals.

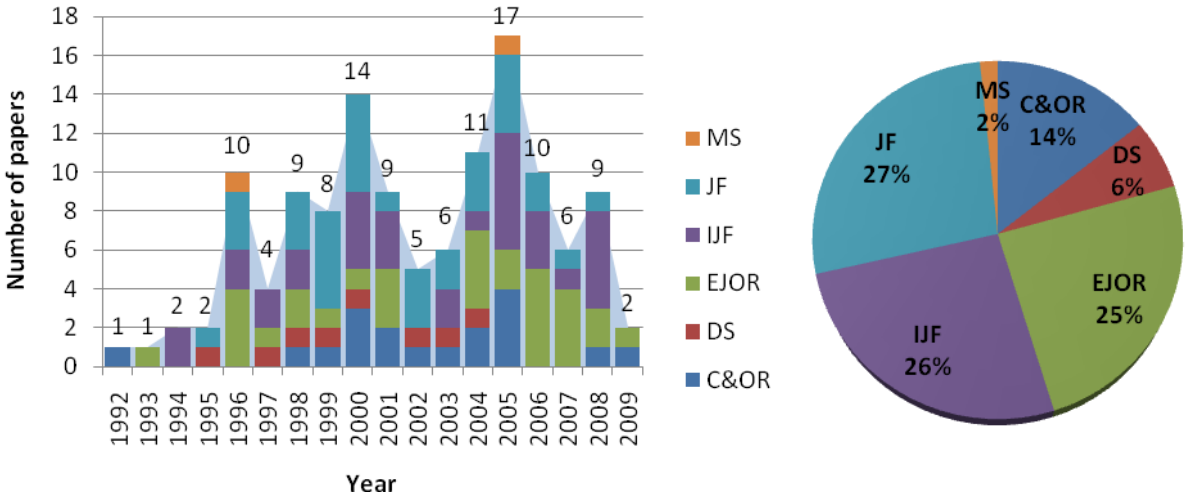

Fig. 1: Publications per year and journal. Note that the 2009 figure includes only the first 7 months.

Comparing the number of ANN forecasting related papers with the total number of ANN papers, in the same journals, there is a similar trend. There is an increasing volume of papers that peaks every 4-5 years. The total number of ANN papers for the same period is 449, which makes the 126 forecasting papers account for 28% of the total published research in the selected eight journals.

In figure 2 the areas of application or the broader topic of the papers are presented. The majority of the papers discuss ANN modelling issues, followed by finance and macroeconomic applications and electricity demand/load forecasting. Under the category "other" all different smaller categories with only one paper are included. A few examples of the varied applications of ANNs include crime forecasting [13], success rates of countries in the Olympic games [14], ozone concentration forecast [15], television viewership [16] and call centre forecasting [17]. More numerous are the applications on traffic volume forecasting [18-20], retail demand forecasting [21-23] and marketing applications where the

utility or the brand choice of consumers is forecasted [24-31]. It is apparent that there is a wide interest in ANN applications in forecasting.

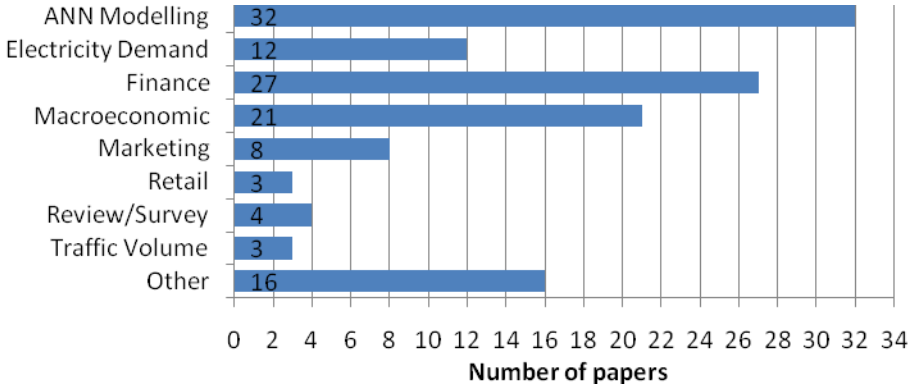

Fig. 2: Areas of application / broad topics of the papers.

3.2. Dataset properties

Here we explore the dimensions related to the dataset that is used in the publications. Note that as some papers are not empirical or do not include experiments the total figures presented hereafter maybe less than the total of 126 papers. First we investigate the form of the dataset, i.e. if the papers use univariate data, multivariate data or both in their experiments. The majority of the articles address multivariate problems, as can be seen in table III. About 40% of the papers discuss univariate time series forecasting problems and only 7 papers (6.8%) examine both possible forms. Regarding the type of time series, i.e. if it is a real dataset or a synthetic, nearly all papers (92%) use real time series. Again 7 papers use both real and synthetic time series in their experiments. Although real time series have apparent practical importance, synthetic time series allows the researcher to control the properties of the dataset and get a better understanding of the modelling process. Therefore, the literature is lacking in that sense, since in many cases the authors of the papers conclude that it is unclear why the ANNs forecast or fail to do so accurately, because the true properties of the time series are unknown.

Table III: Dataset form and type.

Form	# of papers	Type	# of papers
Multivariate	60	Synthetic	8
Univariate	42	Real	92
Both*	7	Both*	7

*Included in the above forms/types

The next dimension of analysis is the sample size of the time series. Table IV provides descriptive statistics of the different sample sizes used in the literature and figure 3 represents this visually with a boxplot. ANNs have been used in both short and long time series. The effect of the sample size is systematically analysed by Markham and Rakes [32] who find that at large sample sizes ANNs outperform linear regression, whereas the opposite is true for short samples. Therefore, they conclude that ANNs perform better when long samples are available. Hu et al. [33] model daily exchange rate time series and conclude that ANNs perform well with large sample sizes. Zhang [34] and Zhang et al. [35] find that sample size is not an important determinant for ANN accuracy. However they note that more data are found helpful to overcome overfitting problems.

Table IV: Sample size statistics

Percentile	Min	18.0
	10%	68.1
	20%	111.2
	30%	130.0
	40%	153.6
	50%	234.0
	60%	385.8
	70%	720.1
	80%	1637.8
	90%	8866.2
	Max	105024.0

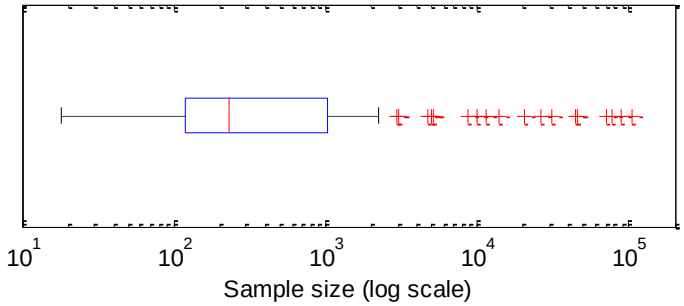

Fig. 3: Sample size used in the ANN literature

The sample size is connected to the time series granularity. In the literature twelve different granularities are used, the shortest being observations every 20 seconds for road traffic data [18] and the longest being annual time series covering a variety of different data types. Although counting all the individual granularities has limited interest, it is important to distinguish between low and high frequency applications. There is no formal definition of what constitutes high frequency data, since the characterisation changes with the available techniques, computational resources and what is the most common time series granularity [36]. For this analysis we use the daily time series granularity as the boundary between high and low frequency time series. Any time series of daily or shorter intervals will be counted as high frequency. Granger [37] has observed that conventional statistical methods can have problems in interpreting high frequency information. Taylor et al. [38] suggest that conventional statistical methods need to be modified to forecast high frequency time series.

In their analysis they use a modification of the exponential smoothing and ARIMA models to forecast hourly electricity load data. Therefore, it is interesting to investigate whether ANNs are able to forecast both low and high frequency time series and if there is need for special modifications of the models. Table V shows the number of papers that use each time series granularity that is identified in the literature. The number of papers is provided for all area of applications and separately the three major ones, as shown in figure 2. Both high and low frequency problems are strongly represented in the literature. However, if the finance and electricity demand forecasting applications, which are inherently high frequency problems, are excluded then the majority of the applications is for low frequency problems. It is unclear whether this preference to low frequency applications is due to data availability or modelling problems. Figure 4 presents visually the number of papers per time granularity for all areas of ANN applications.

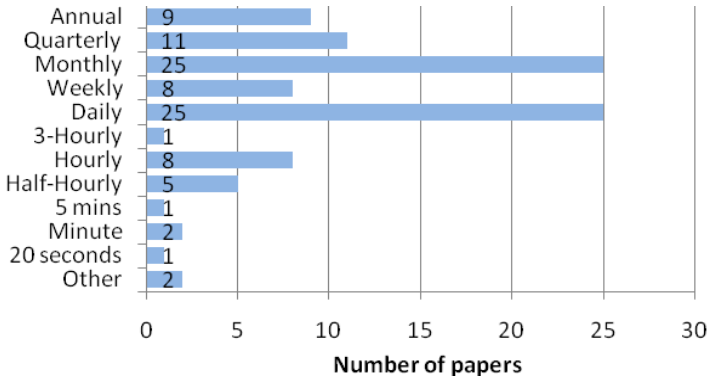

Fig. 4: Number of papers per time series granularity

Table V: Number of papers per time granularity

Time granularity		Area of application			
		All areas	Finance	Electricity	Macroeconomics
High frequency	20 seconds	1			
	Minute	2	1		
	5 mins	1			
	Half-Hourly	5		4	
	Hourly	8		6	
	3-Hourly	1			
	Daily	25	11	2	5
	Total	43	12	12	5
Low frequency	Weekly	8	1	2	
	Monthly	25	4		8
	Quarterly	11	2	1	4
	Annual	9	3		1
	Other*	2			
	Total	55	10	3	13

*In these cases the time granularity is not defined due to the dataset characteristics

There is only one paper that uses both low and high frequency data [39]. In this study the authors use polynomial neural networks and common multilayer perceptrons to forecast the monthly airline passenger time series, a daily Dow Jones industrial index series and an hourly electricity load time series. They compare the ANNs with statistical benchmarks in order to establish whether the network models are better and if the proposed polynomial neural network outperforms multilayer perceptrons. The findings are mixed and it is difficult to assess whether ANNs are applicable to several different time series frequencies without modifications or different modelling practices. Note that this is not the main research question of this study, so the authors have not designed their experiment likewise. Hippert et al. [40] and Hahn et al. [41] discuss the application of ANNs in electricity load forecasting, a typically high frequency problem. Both conclude that ANNs have been successfully applied in this type of problem, outperforming established forecasting benchmarks. The first paper concludes that large overparametrised ANNs perform very well for electricity load forecasting problems and note that this may be due to the dataset properties, since such networks are typically avoided in other ANN forecasting applications. This provides some evidence that high frequency time series is a special case for ANN models, but there is no extensive research on the effects of the data frequency to the performance of the networks. Therefore, it is important that more research is invested on understanding the effects of the data frequency on ANN forecasting performance, especially since high frequency time series are becoming more common [36].

Another issue that is connected with the dataset is the type of pre-processing of the data, if any, and the scaling that is applied to the inputs. 80.2% and 78.6% of the papers do not provide these figures respectively. Regarding the pre-processing of the time series 52% of the papers that report it (13 papers) transform the inputs by removing the trend and/or the seasonality of the time series. This is connected to an ongoing debate on how to best model time series with trend and season components. Hill et al. [42] use time series from the M1 competition and deseasonalise them. They fit ANNs models and find that they outperform standard statistical models. Nelson et al. [43] repeat the experiment without deseasonalising the time series and find that the performance gets significantly worse,

concluding that deseasonalising is a necessary step in time series forecasting with ANNs. They argue that by removing the seasonal component the network can learn better the trend and the cyclical components in the time series. Lachtermacher and Fuller [44] propose first and seasonal differencing as a pre-processing step, based on the ARIMA modelling procedure. The authors aim to model time series in their stationary form as it would be required by the Box-Jenkins model. In addition to that they consider Box-Cox transformation as an additional pre-processing step. When applied, the authors find significant improvement in the training time and the forecasting accuracy, however for the accuracy the exact magnitude of the improvement is not documented. Furthermore, it is unclear why this transformation is beneficial for such nonlinear models. They also do not provide evidence that using differenced inputs is better than modelling the time series in the original domain. Conversely, Balkin and Ord [45] quote that differencing is an unnecessary step, but they do not explore its effect. Zhang and Qi [46] investigate the effect on forecasting accuracy of different ways to remove trend and seasonality from time series for forecasting with ANNs. They conclude that removing both trend and season is beneficial for the accuracy of the forecasts and that the best way to do this is through 1st and seasonal differencing. They argue that the detrended and deseasonalised time series do not contain long dynamic autocorrelations that make it difficult to choose an appropriate input vector. Curry [47] address the issue from a theoretical perspective suggesting that for ANNs to model seasonality the input vector should be long enough to adequately capture the seasonal effects and that it is not a matter of pre-processing, implying that Zhang and Qi results can potentially hide input misspecification errors. Crone and Dhawan [48] demonstrate this, by modelling monthly seasonal patterns using only an adequate number lags of the time series and no deseasonalising. Zhang and Kline [49] verify their previous findings by using quarterly time series to model ANNs. They find that deseasonalising improves accuracy and the best results are achieved through seasonal differencing. They argue that coding seasonality with dummy variables does not allow the ANNs to capture the dynamic structure of the real time series, however they do not distinguish between deterministic and stochastic seasonality in their dataset, which conventionally requires a different modelling approach [50].

In the literature there is support that both pre-processing and no pre-processing are necessary for ANNs in order to maximise forecasting accuracy, without specifying the

conditions that each would be preferable. This inconsistency complicates ANN modelling. However several aspects of the issue have been overlooked by the ANN literature, like the nature of the trend and the seasonality, i.e. if it deterministic or stochastic, what happens when multiple overlying seasonalities are present, as is common in high frequency time series, etc. Researching these special topics will provide additional understanding of ANNs and thus help to lift the current confusion. The remaining papers that use some form of pre-processing refer to either transformation of the raw data to more useful formats (like taking the percentage difference of the raw time series) and is always connected to domain knowledge or calculate the logarithms of the time series before modelling it with the ANNs. The argument behind the use of logarithmic transformation is outlined by Balkin and Ord [45]. During their training ANNs usually minimise some sort of squared error. Efficient estimates result in least square optimisation when the error terms are independent and have equal variances. The logarithm does exactly that. However, there are no comparative studies that demonstrate a clear benefit of using the log transform of the time series with ANNs and therefore its use is rather limited.

ANNs require the inputs to be scaled to specific bounds that are defined by the transfer function of the hidden neurons [2, 44]. It is a necessary step to produce forecasts with ANNs and it can be safely assumed that most researchers in their papers use some sort of scaling. However, only 21.4% of the papers report the scaling that is used. This renders most of the published work impossible to replicate and also does not offer any evidence on the effect of the scaling on the accuracy of ANNs. In the literature there are no large scale studies concerning its effect on the accuracy and most focus on the effect on the ANN training, for which it is unclear whether it is beneficial or not and how it should be done [2]. Lachtermacher and Fuller [44] argue that scaling should be able to accommodate unobserved future values that are out of the bounds of the historic values. Therefore, scaling should result in values tighter than that required by the transfer function, in order to have room for values outside the range of the original training data. Wood and Dasgupta [51] quote that scaling is one way of reducing the impact of noise to the ANNs, but they do not provide the evidence to demonstrate this. Church and Curram [4] argue that the transfer function becomes increasingly nonlinear at its extremes, so by scaling the input data to tighter ranges overcomes this problem. Furthermore, they also argue that this way ANNs are robust to future unobserved values. Torres et al. [52] mention that scaling the

inputs to tighter ranges helps to avoid the saturation problem of the transfer functions. In the above papers the choice of the new tighter bounds is arbitrary, with the exception of Lachtermacher and Fuller who suggest scaling the time series by a factor of two times the initially intended range. However, it is not discussed why a factor of two is adequate. In the literature it is unclear which of the available scaling methodologies is better (for a discussion of the alternatives see Zhang et al. [2]). Although there are arguments in favour of tighter scaling bounds than those required by the transfer function, there is no rigorous evaluation. Furthermore, there is an open question regarding how one should set these new bounds.

Another dimension of this study related to the dataset is how to split it into training, validation and test sets. ANNs in order to train and avoid overfitting typically require the use of a validation set. Part of the original time series is used during the training of the ANNs to validate that the model has approximated the underlying data generating process and has not been overfitted to the training set, which is used for estimate the network's weights. Therefore, the size of the validation set limits the available sample size for the training of the ANNs. Deciding the size of the validation set is similar to setting the size of the test, which is used for the ex-ante evaluation, and is usually application specific. Therefore, we will not list in detail all the different ways that the time series are split in the literature, but I will refer only to the special cases. Bodyanskiy and Popov [53] use online training to fit their ANNs, which means that the network adapts continuously as new information becomes available. This makes the need for validation set obsolete, therefore none is used. Note that this is a different form of training and forecasting and does not discredit the common offline training of the ANNs that all the data are available and a validation subset can be created. Corcoran et al. [13] use a special scheme to avoid using a validation set. They use the M-test, which is essentially a gamma test applied incrementally to an increasing sample size, to identify the number of training observation that minimises the effect of noise and therefore overfitting. Once this value is identified the appropriate training set is used and the rest of the data is used as test set. However, in their paper they do not provide the evidence that this gives better forecasting accuracy compared to the common use of the validation subset. Note that 29.4% of the accessed papers in this review do not provide information on how the available data are split in training, validation and test subsets. This limits the validity of those papers, as it is unclear how the ANNs are build, on what sample they are trained and how their evaluation is done. Furthermore, these experiments are not replicable.

Table VI provides the descriptive statistics for the number of time series that are used in the literature. Figure 5 provides a visual representation of the same information as a boxplot. More than 70% of the papers use under 5 time series. There are 12 papers that use from 10 to 100 time series and only 8 than use more than 100 time series, up to the maximum of 367. In this classification the M3 competition [54], which has an ANN model submission that was evaluated on 3003 time series, among several other forecasting models, is not included. The relatively small number of time series that is used in most studies implies that it is hard to generalise from their conclusions and the statistical validity of the evaluation framework is questionable. This in conjunction with the limited use of rolling origin evaluation scheme, which is discussed in a following section, limits severely the papers that can be used to assess the performance of ANN models against benchmarks. It is imperative that more large scale studies are conducted in order to provide statistically valid evidence of the ANNs' forecasting performance and best modelling practices.

Table VI: Number of time series

Percentile	Min	1.0
	10%	1.0
	20%	1.0
	30%	1.0
	40%	1.0
	50%	2.0
	60%	3.0
	70%	4.2
	80%	8.0
	90%	45.4
	Max	367.0

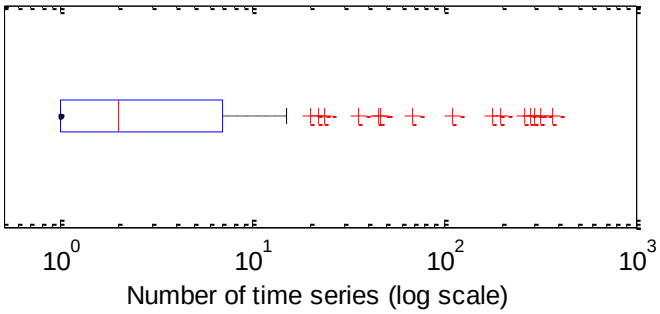

Fig. 5: Number of time series in ANN papers

3.3. ANN architecture

Here we discuss all the dimensions of analysis that are related with the ANNs' architecture that are found in the literature. The questions that are discussed here include what are the types of ANN used, how the models are specified, the input variables and the size of the hidden layers specifically, whether a single or multiple outputs are used, what transfer functions are employed and other special considerations like pruning and shortcut connections.

First we present the most common types of ANNs that are used in the forecasting literature. Figure 6 shows the percentages of papers that use Multilayer Perceptrons (MLP), Recurrent Neural Networks (RNN), Generalised Regression Neural Networks (GRNN), Radial

Basis Function networks (RBF), Probabilistic Neural Networks (PNN) and all the other network types that are represented by only one paper in this review.

The majority of the papers (75%) use MLPs. The second most common type is the RNNs with only 6% of the papers using it. RBF networks follow with 5%. GRNNs are used by 4% of the papers and 1% uses PNNs. The remaining 9% of the papers use different types of ANNs that appear only once in this review and in most cases are variations of the MLP, like the DAN2 which captures the linear and the nonlinear part of the time series in separate neurons (for more information refer to Ghiassi et al. [55]). The dominance of MLPs seems to be unaltered since the last major review of ANNs in forecasting [2], however it does not mean that they are better suited for forecasting. For instance if we consider the papers that discuss RNNs they routinely report outperforming MLPs. Note the validity of several comparative evaluations is questionable, as is discussed in the following sections in more detail.

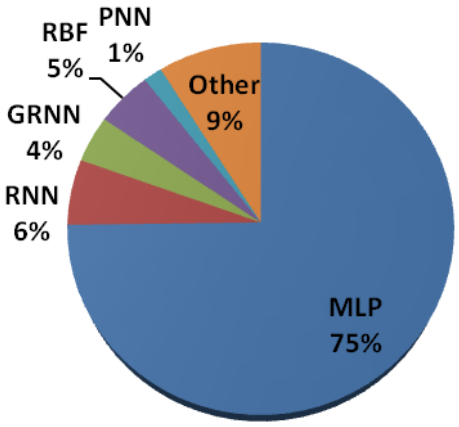

Fig. 6: Type of ANN used

From this point on, only for the papers that use MLPs and RNNs, which are the most common implementations, are discussed. The reason for this is the special nature of the GRNNs, RBFs, PNNs and other types of networks that require completely different architecture, design, modelling considerations and their use in forecasting represents less than 19% of all papers.

Next, we investigate how many papers present a complete methodology to model the ANNs architecture, including selection of inputs, number of hidden layers and nodes, connections and transfer functions. Only 16 papers suggest a unified methodology to specify systematically the inputs and the hidden layer. No papers provide guidelines for selecting

the transfer function. The same is true for shortcut connections, i.e. direct connections between the layers that bypass one or all the hidden layers. Both seem to be set according to the preferences of the modeller. In addition to these 16 papers there are a number of papers that address the selection of solely the input variables of the ANNs or the hidden layer. These papers are discussed together with the ones that offer a complete methodology to specify both. There are in total 25 papers that specify automatically the input variables of ANNs. These can be classified in seven major categories, as it can be seen in table VII. All methodologies based on regression analysis are classified under the category "*Regression"*. Methodologies that use autocorrelation analysis (ACF), partial autocorrelation analysis (PACF), mutual information (MI) or any other similar metric, individually or in combinations, are categorised as "*ACF & PACF or similar"*. Any methodology that makes use of heuristics or rule-based analysis or information criteria is under the category "*Heuristic & rule based"*. All papers than use pruning algorithms to identify the input variables belong to category "*Pruning"*. Methodologies that are based on genetic algorithms and other evolutionary algorithms are under "*Genetic algorithms"* and finally the single paper that identifies the input variables by means of sensitivity analysis is on a separate category named "*Sensitivity analysis"*. The remaining papers, which is the majority (71.3%) do not present or use a systematic way to choose the input variables for the ANNs they use. In most cases the selection methodology is done using a trial and error approach or arbitrarily that limits significantly the input search space and can easily lead to suboptimal and myopic selections. However, there is a lot of evidence in the literature that the input variable selection is the most important modelling variable for ANNs in forecasting. Zhang et al. [2] observed in their review that there are very few systematic input variable selection methodologies available, although the inputs of the ANNs are very important for their forecasting accuracy. Anders and Korn [56] identify the same problem in the ANN literature and in addition they point out that there is no widely accepted or used methodology either. Zhang [34] and Zhang et al. [35] explore the ability of ANNs to model linear and nonlinear time series respectively and conclude that the selection of the input variables is the leading determinant of accuracy, followed by the specification of the hidden layer. There are numerous empirical studies that highlight the importance of the input variable selection for ANNs application (for example Darbellay and Slama [3] stress this issue in electricity load

forecasting problems). Since then there are several publications focused on how to specify the input variables for ANNs for forecasting problems, as it can be seen in table VII.

Table VII: Papers that use input variable selection methodologies

Regression	Heuristic & rule based	Hypothesis testing
Balkin and Ord [45] Church and Curram [4] Dahl and Hylleberg [57] Prybutok et al. [15] Qi and Madalla [58] Swanson and White [59]	Corcoran et al. [13] Liao and Fildes [60] Moreno and Olmeda [61] Qi and Zhang [62]	Anders et al. [63] Medeiros et al. [64] Refenes and Zapranis [65]
ACF & PACF or similar	**Pruning**	**Genetic algorithms**
da Silva et al. [66] Darbellay and Slama [3] Kajitani et al. [67] Lachtermacher and Fuller [44] Moshiri and Brown [68]	Kaashoek and Van Dijk [69] Setiono and Thong [70] Terasvirta et al. [71]	Kim et al. [27] Motiwalla and Wahab [72] Nag and Mitra [73]
		Sensitivity analysis
		Dougherty and Cobbett [19]

However, the number of the different categories of methodologies that has been published illustrates that there is still no consensus on how to specify the input variables of ANNs. Another important observation is that most of these papers use a filter approach to specify the inputs, with the exception of Liao and Fildes [60] who provide a wrapper framework that essentially iterates among a large number of possible candidates and da Silva et al. [66] who use as a possible input variable selection methodology a wrapper that tries several different combinations of inputs automatically. They briefly discuss the distinction between wrappers and filters and identify as the key distinction the higher computation cost of the first. To illustrate the advances in the topic, the different methodologies are discussed by category in chronological order.

The most common specification methodology is based on variants of regression analysis. Church and Curram [4] compare MLPs with econometric ordinary least squares regression models. They suggest modelling the ANN using the same inputs that they identified through the regression analysis. This offers a systematic framework to select the input variables for MLPs. However, the identification of the inputs for a nonlinear model, like the MLPs, is based on linear regression; hence, there is the risk of missing useful nonlinear information. Swanson and White [59] simplify the procedure by using a forward stepwise linear regression to identify the significant input variables. Regressors are added one at a time until the Schwarz Information Criterion (SIC) cannot be improved more.

Although this methodology fails to identify nonlinear information like the previous one, it offers a more automated approach to input variable selection, minimising the required intervention from an expert modeller. However, the use of SIC is criticised by Qi and Zhang [62] as inappropriate. They evaluated its use, along with AIC, as a mean to identify the appropriate number of lags for MLPs and concluded that there is no connection between these information criteria and the forecasting performance of networks. Qi and Maddala [58] identify the inputs for their MLP model through means of linear regression. Initially they build a linear regression and use the significant variables of the regression as inputs to the ANN. These variables, like in the previous cases, can be lagged. The weaknesses of this methodology are similar. The linear regression does not capture nonlinear information, therefore may miss some important nonlinear inputs for the ANN. Furthermore, in this implementation the regression modelling is not automated and a human expert is required. Balkin and Ord [45] propose a hybrid heuristic-regression approach. First, they consider the problem of the maximum lag of the time series that should be evaluated with the regression model. To solve this, which is unanswered by the previous papers, they use a heuristic rule. Depending on the frequency of the time series they provide a maximum number of lags that should be evaluated; for annual time series this is 4 lags, for quarterly 6, for monthly 15 and for any other frequency they propose 6 lags. The possible lags are then evaluated using a forward linear regression. From all the different regressions that are built by combining these lags, those that have an F-statistic greater than 4 are selected. From the selected ones the least parsimonious is chosen to identify the inputs for the ANN. This methodology is fully automatic; however it has a series of problems. First of all, it is calibrated only for low frequency time series, since the heuristic would not be able to provide a reasonable maximum lag for time series of higher than monthly frequency. On the other hand, it is the only attempt to address the issue of maximum lag length in the literature. Secondly, like the previous methodologies it is restricted to identifying linear information. Prybutok and Mitchell [15] chose the input in their study using stepwise linear regression. They deal with a multivariate problem and they do not consider lagged variables, however their methodology can be easily extended to include such. The main weakness is that the identification of the inputs is done considering only linear information. Dahl and Hylleberg [57] try to overcome this by using a nonlinear regression model. They choose to use the random field regression, proposed initially by Hamilton [74]. This model allows identifying

separately linear and nonlinear explanatory variables, thus overcoming the main weakness of the previously mentioned methodologies. In their implementation they use forward regression with AIC and BIC optimisation to build the nonlinear regression model and then use the significant variables as inputs to the ANN. Although this is the only regression based methodology that tries to capture nonlinear information in the inputs of the ANN it can be criticised for using AIC and BIC optimisation for identifying the appropriate number of inputs, which is discouraged in the literature [62]. In addition, this methodology is very computationally expensive due to the estimation of the random field regression models. Interestingly, in the literature only the stepwise and the forward regression models have been considered. Backward regression has not been used.

The second most common category of methodologies is based on analysing the ACF or PACF of the time series, or similar metrics like mutual information criterion. Lachtermacher and Fuller [44] propose a methodology to model ANNs similar to the ARIMA modelling methodology. ANNs are autoregressive models and naturally make use of the autoregressive structure of the time series, which is captured in the PACF. Therefore, they suggest that identifying the autoregressive structure of the time series in a similar way to what Box and Jenkins describe [75] can help identifying the input variables for an ANN. They also suggest using the autocorrelation information in an attempt to capture the additional nonlinear information that is not identified by the linear PACF. Note, that following the ARIMA methodology the lagged observations of the time series may need to be differenced. This methodology fails to provide evidence why the inclusion of the ACF is beneficial and like most of the previously mentioned methodologies, is based on linear identification tools, which may be a limiting factor for ANNs. Darbellay and Slama [3] try to overcome this by using the nonlinear autocorrelation function. This is defined as the mutual information scaled between 0 and 1. This metric is able to capture nonlinear dependencies and therefore provide a more complete set of inputs to the ANN. The authors identify the significant lagged inputs of the time series using a similar approach to the normal ACF analysis, arguing that all the extra identified significant lags, compared to ACF analysis, contain the nonlinear information. However, this is not entirely true as the ACF and the scaled MI have different bounds and are not directly comparable. Moshiri and Brown [68] use only the PACF information to identify significant lags that should be included as inputs to the ANNs. In contrast to the previous methodology, using only PACF information will

restrict the nature of the identified interactions to linear. Furthermore, as Lachtermacher and Fuller [44] quote, to correctly identify the structure of the autoregressive information it may be necessary to include differenced observations of the time series, which is not considered in this case. Kajitani et al. [67] opt to use the ACF to identify significant lags that should be used as inputs for ANNs. In theory MLPs, which are used in their paper, are autoregressive models and therefore PACF should be preferred, in contrast to RNNs that can capture both autoregressive and moving average processes. Considering that in this study MLPs outperform the benchmarks, it should be explored why this is so, which is not discussed in detail by the authors. Again, this methodology tries to identify inputs for the nonlinear ANNs using a linear filter. Da Silva et al. [66] consider several alternative to specifying the ANN input variables. They consider both filters and wrappers. As a filter they use the interdependence redundancy, which is a normalised mutual information measure. Before applying this filter they first difference the time series for trend and seasonality in order to achieve stationarity. They also consider a Bayesian wrapper which essentially iterates among a large combination of alterative inputs until the best model is identified. This is computationally expensive and the authors first preselect heuristically a set of inputs to consider. The authors propose methodologies that can capture the nonlinear structure of the time series, at additional computational cost, which is side-stepped by using heuristics to preselect a set of possible inputs. The heuristics are not described in the paper, but it is possible that restricting the search space can have negative effects on accuracy. Furthermore, differencing of the time series is used to remove the trend and season components. However, differencing is not established as a necessary step for ANN modelling and furthermore it may lead to model misspecification if the trend or season components are deterministic.

Another set of methodologies makes use of heuristics and rules to identify the appropriate inputs for ANNs. In this category methodologies that minimise some form of information criteria are also included. Qi and Zhang [62] investigate if the use of in-sample model selection criteria is a reliable guide for out-of-sample performance. They use the Akaike information criterion (AIC), the Bayesian information criterion (BIC) and their common variants to investigate if they are useful indicators in selecting the inputs for ANNs and the size of the hidden layer. They conclude that there is no apparent connection between the values of the information criteria and the forecasting performance of the

ANNs. This finding has significant implications for several papers that use some variant of the either the AIC or BIC to choose the ANN topology. A limitation of the paper is that they consider a relatively limited number of lags and hidden nodes (up to 5 for both cases). Moreno and Olmeda [61] use AIC to identify the correct number of inputs to model MLPs and compare them against linear models. They extend the search space to 10 lags, but fail to find MLP models that clearly outperform the benchmarks, providing evidence in agreement with the previous study. Corcoran et al. [13] propose a heuristic based on the Gamma statistic. The statistic is calculated for incremental lag lengths until the minimum Gamma statistic is identified. All lags up to this point are used as input for ANNs. In principle, this methodology is similar to the previous heuristic approaches. All of them force all lags up to a specific order to be included in the input vector, in contrast to the methodologies that are based on regression and ACF/PACF analysis that create sparse input vectors. It has not been explored which method is more appropriate for the ANNs. Furthermore, depending on the dataset properties and especially its frequency, the nonsparse specification of the inputs may lead to very long input vectors that affect negatively the training of the ANNs. Liao and Fildes [60] discuss the difficulty to parameterise ANN models and propose a heuristic framework that allows a systematic search for inputs, number of hidden nodes and learning parameters that will provide the best model for the dataset. Essentially, they suggest a wrapper with heuristics that help to standardise the search. They also suggest using as an additional input a time series constructed by the median of all the past values up to each historical observation. This was found to provide more robust results for their dataset. The main problem of this methodology is its computational cost and that it is time series specific, since it is based on a wrapper [66], which can make it impractical for large scale implementations. In their study they show that their proposed methodology worked well on a dataset of 261 telecommunication time series.

Another approach to the problem of specifying the input variables is to start with an arbitrarily large vector of inputs and prune it to a smaller size of significant inputs. Kaashoek and Van Dijk [69] propose a methodology that the modeller sets the maximum number of inputs and then calculates the incremental contribution of each input in terms of R^2 by removing one input at a time. The residuals that are calculated after removing each input are stored as vectors which are analysed by means of principal components analysis. The

relevant components of the first principal component are used as additional indicators of the significance of the inputs. The inputs with minimal incremental contribution and the smallest components are pruned. The elimination continues until all insignificant inputs are removed. The authors identify that a limitation of this methodology is how to identify what is a low or minimal contribution and an insignificant component. Furthermore, this method is computational intensive, since the ANN model has to be re-estimated several times. Another weakness is that it is hard to know what an adequate starting number of possible inputs is. This is especially important when dealing with time series of different frequencies. Setiono and Thong [70] use pruning to identify the inputs, however the criterion used to decide which input to prune is the ANN accuracy. If removing an input does not harm the accuracy of the network then that input is removed. This is again a top-down pruning approach, i.e. it is necessary to start with a large number of inputs, which may be difficult to specify in advance. Terasvirta et al. [71] uses the methodology described in Medeiros et al. [64] with the addition of pruning to get parsimonious networks. Note that in all these papers, pruning is used to identify the number of hidden nodes as well. In the literature there are arguments that pruning may not always be desirable, especially in the cases of high frequency data [40] or seasonal time series [47], where a large network can provide the flexibility for a better fit.

In an attempt to increase our understanding of ANNs, there are methodologies that are based entirely on statistical hypothesis testing. Anders et al. [63] propose a complete framework to specify both the number of hidden nodes and inputs. Once the number of hidden nodes is identified the ANN is trained with all inputs. Each single input connection (and not the whole input node) is evaluated using the Wald test. The connection with the most insignificant p-value is dropped and the network is retrained. The process is repeated until only significant connections remain. The limitations of this methodology are similar to the pruning ones that are described before. It involves high computational cost and it is difficult to specify in advance the starting set of all the inputs, especially in temporal modelling. Refenes and Zapranis [65] propose a similar top-down approach which is based on different statistical test. They suggest starting with a model that includes all possible inputs and calculate the MFS value [76] for each input. The least significant input (below a set threshold) is dropped from the model. Another difference with the previous methodology is that in this one the number of hidden nodes is reidentified in each iteration

and the next input is evaluated with the "best" number of hidden nodes. The weaknesses of this methodology are similar, but with much higher computational cost, since now the hidden layer is respecified in each iteration. Medeiros et al. [64] try to address the problem of high computational cost by proposing a bottom up approach. For the selection of the input vector a methodology proposed by Rech et al. [77] is used. This methodology is based on the idea of approximating a stationary nonlinear time series by a polynomial of sufficiently high order. Combination of variables (or lags) are included in the polynomial and a model selection criterion (AIC or BIC) is calculated. The polynomial with the lowest selection criterion is selected and indicates which inputs should be used in the ANN. Once the input vector is set the methodology addresses the hidden layer. This methodology uses indirectly AIC or BIC to specify the input variables of the ANN. It is not clear in this case if the findings of Qi and Zhang [62] that such criteria are inappropriate to specify the inputs of ANNs hold and it should be evaluated if this methodology overcomes this problem.

Another group of papers propose to identify the input variables for ANNs using genetic algorithms. Motiwalla and Wahab [72], Nag and Mitra [73] and Kim et al. [27] propose different variations of genetic algorithms to identify the best set of inputs. The principal idea is that an initial set of networks is created, trained and evaluated. The best performing networks are then used as "genetic material" for the next generation of networks. The process continues until the best solution is reached. Although these methodologies are not identical they share common points of criticism. All these methods are very computationally intensive, as they require to train and evaluate a very large number of ANN for each time series, which is highlighted by the authors as well. Furthermore, these methodologies will not select every time the same inputs, due to the stochastic nature of the genetic algorithms.

The last methodology is related to sensitivity analysis. Dougherty and Cobett [19] suggest training a ANN with all the inputs and then change the values of one input variable by a small percentage at a time. By measuring the effect of these changes in the accuracy of the ANN it is possible to identify strong positive or negative relationship of inputs to the output of the ANN and relatively neutral inputs. The authors suggest keeping only the inputs that have strong effects on ANN's outputs. Although this methodology overcomes the problem of identifying which inputs capture useful nonlinear information for ANNs, it is limited in the sense that it cannot evaluate synergies between input variables.

A wide variety of input variable selection methodologies have been proposed in the literature, which are classified in this study in six main categories. Methodologies under each category share common limitations, which are usually overcome in other categories. However, there is no identified best methodology. These alternative methodologies have not been compared to each other, even when they belong to the same category. This increases the confusion of what is a good way to specify the input vector. Given the significance of the input vector for the forecasting accuracy of ANNs it is necessary to evaluate the proposed methodologies against each other. This will provide insights why some methodologies work or fail and how ANNs are best modelled.

The specification of the hidden layers and the number of hidden nodes is less researched. A major influence has been the proof that single hidden layer MLPs are universal approximators [5, 6]. Based on this theorem most of the literature uses a single hidden layer and the problem is reduced to identifying the number of hidden nodes in this hidden layer. Zhang [34] and Zhang et al. [35] in their study conclude that the number of hidden nodes is of lesser importance in comparison to the input variables of the ANN and find that a small number of hidden nodes is adequate for most cases. Hippert et al. [40] reach a different conclusion. For electricity load forecasting large ANNs prove to be more flexible in capturing the complex dynamics of the time series and therefore should be preferred to small networks. Levelt [78] observes that the universal approximation theorem requires an infinitely large number of hidden nodes and does not necessarily hold for a small number of hidden nodes, suggesting that more complex architectures might be preferable. Curry et al. [25] argue that with finite data points and finite number of hidden nodes more hidden layers can produce more accurate networks in comparison to single hidden layer ANNs. Nikolopoulos et al. [16] suggest that two hidden layers perform better in television viewership datasets than a single hidden layer. From the accessed papers that use either MLPs or RNNs only 8 articles (less than 10%) use more than a single hidden layer. None provides a systematic way to identify the required number of hidden layers and resort to using the suggestions of previous studies or iterative trial and error approaches.

Table VIII: Hidden nodes selection methodologies

Heuristic & rule based		Hypothesis testing
Balkin and Ord [45]	Prybutok et al. [15]	Anders et al. [63]
Church and Curram [4]	Refenes and Zapranis [65]	Medeiros et al. [64]

Dahl and Hylleberg [57]	Qi and Zhang [62]	Terasvirta et al. [71]
Lachtermacher and Fuller [44]	Sahin et al. [81]	
Leung et al. [79]	Sexton et al. [82]	Pruning
Moshiri and Brown [68]	Swanson and White [59]	Kaashoek and Van Dijk [69]
Motiwalla and Wahab [72]	Swanson and Zeng [83]	Setiono and Thong [70]
Olson and Mossman [80]	Genetic algorithms	
	Nag and Mitra [73]	

The number of hidden nodes in most studies is identified through a trial and error approach or it is arbitrarily preset to a specific number. A minority of papers (24%) provide methodologies that can be used to select the number of hidden nodes. These can be classified in four categories, as it can be seen in table VIII, those that are based on heuristics and rule based decisions, on pruning, on hypothesis testing and those that use genetic algorithms.

The heuristic approaches can be subdivided in three categories. The first category sets the number of hidden nodes (on a single hidden layer) as a function of the number of inputs and/or outputs or training samples of the ANN. Lachtermacher and Fuller [44] suggest to use a number of hidden nodes that will make the total weights of the network be between 1.1 to 3 times more than the number of training samples divided by ten. The rationale behind this selection is that it will offer good generalisation properties. Leung et al. [79] use 75% of the number of inputs as a guideline to identify the number of hidden nodes. Prybutok et al. [15] initially calculate the number of hidden nodes by dividing the number of training cases by 5 times the sum of the number of inputs and outputs. Then they evaluate neighbouring values as well and choose the one that performs best. Olson and Mossman [80] set the number of hidden nodes by rounding up the average number of inputs and outputs. These approaches have been used to provide guidelines to restrict the search space for identifying the best number of hidden nodes, rather than strict definitions of the number of neurons.

Church and Curram [4] argue that too few hidden nodes will not allow the network to capture the structure of the time series, while too many will cause overfitting. Therefore, this can be used to identify the number of hidden nodes. In the proposed methodology the validation error is monitored during the training of the network. If the validation error does not get continuously worse it means that the network does not have enough nodes to overfit the data. In this case the training is stopped and more hidden nodes are added to the

MLP, since the current number will be unable to capture fully the underlying structure. Motiwalla and Wahab [72] employ a heuristic called cascade learning. In contrast to the previous papers this heuristic allows several hidden layers and creates shortcut connections to the inputs as well as the previous hidden layers. The principal idea of cascade learning is that the ANN starts with a small number of nodes. New nodes are added one or more at a time until performance cannot be further improved. Sahin et al. [81] start with 2 hidden nodes and incrementally increase the size of the hidden layer as long as the residuals decrease. All the last three papers use bottom-up construction approaches, starting from a small number of hidden nodes and increase until some error metric cannot be improved further. It is important to note that in their description none of these methodologies would overcome possible local minima of the performance criteria and the search would stop there.

The remaining methodologies follow a similar bottom-up approach but instead of the errors they employ information criteria that penalise for the number of parameters. Swanson and White [59] and Swanson and Zeng [83] use BIC. Balkin and Ord [45] prefer to use the GCV metric, which allows parametric cost for the additional model parameters. Dahl and Hylleberg [57] consider both the AIC and BIC metrics. They add hidden units in a single hidden layer until the performance criterion cannot be improved or the number of hidden nodes has reached 5. Moshiri and Brown [68] consider only the AIC. Qi and Zhang [62], similarly to their analysis for the input variable specification, investigate the usefulness of AIC and BIC in selecting the number of hidden nodes. Their finding is that there is no relationship between the information criteria and ANNs' performance. They conclude that different specification strategies are needed. Refenes and Zapranis [65] use the prediction risk instead. They propose an iterative heuristic that calculates the predictions risk for different number of hidden nodes, up to a specified maximum, and select the one that minimises it. The prediction risk essentially measures the error adjusted for the complexity of the model. The authors note that any other similar metric could be used in the current framework. By replacing the prediction risk with AIC or BIC the proposed heuristic becomes very similar to the methodologies proposed by the previous authors.

The hidden layer specification methodologies that are based on hypothesis testing follow a bottom-up approach, starting from small or linear models and testing the relevance of the nonlinear hidden nodes. Anders and Korn [56, 63], Terasvirta et al. [71] and Medeiros

et al. [64] employ the LM-test [84, 85] to compare between models with H and H+1 number of hidden nodes, until iteratively the optimum number is identified.

Nag and Mitra [73] employ genetic algorithms to identify the number of hidden nodes and layers. They restrict the search space to a maximum of 16 nodes per layer and the maximum number of layers to 2. Similarly, Kaashoek and Van Dijk [69] and Setiono and Thong [70] use the same pruning methodology that they employ to select inputs in order to choose the number of hidden nodes for a single hidden layer. The weaknesses of genetic algorithm specification methodologies are similar to those discussed for the input variable selection.

It is clear that there are numerous alternatives how to specify the hidden layer. Although most authors prefer to use some heuristic or optimisation scheme based on information criteria that penalises for complexity, their performance is not proven. Similarly to methodologies for the selection of the input variables, there is no rigorous comparative evaluation that demonstrates which of these methodologies, or family of methodologies, is better. Furthermore, these methodologies have to be assessed against the simplest approach of selecting the number of hidden nodes arbitrarily or randomly. In order to justify the extra computational cost involved they have to be proven better. Due to our limited understanding of the interaction of the inputs with the hidden layer most of this methodologies resolve to iterative refinement of the hidden layer, which requires retraining the network in each step and do not provide an explanation why the selected number of hidden nodes is adequate.

In addition, it is unclear how the selection of the transfer function interacts with number of hidden nodes. There is no guidance in the literature on how to choose the transfer function of the hidden layer. Figure 7 shows the types and the usage of the hidden layer transfer functions in the literature. Logistic sigmoid is the most common type. It is followed by the hyperbolic tangent (*tanh*) and lastly two papers use linear transfer function. The transfer function defines the bounds that the inputs should be scaled to. However, in the literature there are papers that report good results with neural networks that use different scaling outside these bounds; for instance Wood and Dasgupta [51] use logistic transfer function that is bounded between 0 and 1, but scale the inputs between -0.5 and 0.5. The interaction of the transfer function with the hidden layer, the inputs, the pre-processing and scaling of the inputs is not adequately researched. The literature [34, 35]

suggests that the input variables and the specification of the hidden layers are the most important determinants of ANNs accuracy, however there is no evidence that the choice of the transfer function is of lesser importance. It is imperative that the effect of the transfer function selection is researched more thoroughly in order to evaluate its significance for ANN accuracy and provide guideline on how to select it.

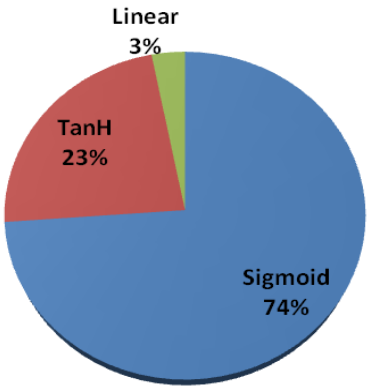

Fig. 7: Percentage of hidden layer transfer functions in the literature.

Selecting the size of the output layer is connected with the forecasting application of the ANNs. Each output node produces a forecast for a single lead time. The modeller can produce a forecast of lead time *t+n* by training directly the network to output forecasts of this lead time, or to produce forecasts with lead time *t+1*, which will be used to produce forecasts of lead time *t+2* until iteratively forecasts of lead time *t+n* are produced. Similarly if the modeller is interested in several lead times, the ANN can be modelled to produce these directly through several output nodes or iteratively through single node. Similarly, an ANN can be trained to output forecasts of several variables simultaneously through multiple output nodes. Table IX summarises the number of output nodes used in the literature.

Table IX: Number of output nodes

Output nodes	Number of papers
1	69
2	2
3	3
4	2
24	1

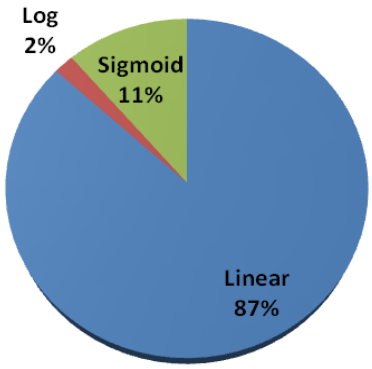

Fig. 8: Output layer transfer function and percentage of ANN papers

Most of the papers (89.6%) use a single output node and only 8 papers use multiple nodes, while 10 papers do not record this information. There has been limited consideration in the literature for directly forecasting simultaneously several lead times or even a single one, but with a longer than *t+1* forecast horizon, through the appropriate selection of the output nodes, even though there is evidence of accuracy advantages [40].

Typically, the output node uses a linear transfer function; however this is not always the case, as it can be seen in figure 8. There are 6 papers that use a logistic sigmoid function instead of linear. A single paper uses logarithm [86]. These papers allow the ANN to capture additional nonlinear behaviour in the output layer. This is not equivalent to an additional hidden layer, since the latter would still use a linear output layer for summing and scaling the intermediate information from the hidden layers. Again, the relative advantage of using nonlinear transfer functions in the output node, instead of additional hidden layers or a simple linear function is unclear and it has not been evaluated. Note that 28 papers do not report the choice of the transfer function of the output node.

Another aspect of the network architecture is related with the connecting weights. The modellers can use ordinary fully connected ANNs, pruned networks, which do not have all nodes fully connected, or opt for shortcut connections, which are connections that bypass intermediate layers, usually connecting the inputs directly to the output node. Only two papers use input to output layer shortcut connections [57, 59]. Both these papers use linear transfer function for the output layer and argue that this allows the ANN to model nonlinear information through the hidden layer and linear information directly through the shortcut connections. However, linear behaviour can be approximated by ANN without shortcut connections as it has been shown empirically [34]. It has not been evaluated

whether the shortcut connections benefit the forecasting accuracy or the training of the network by separating the information flow across the network's layers. Pruned networks, are not fully connected and the rationale behind this decision is keeping only the important connections in order to aid the training of the ANN. Pruned networks are typically created by starting from a fully connected network and removing the least significant connections. This approach was described as an input and hidden layer specification methodology. The modeller can achieve a similar result by establishing only the important connections between the neurons iteratively, instead of starting from a fully connected network. An example of this is Swanson and White [59] who use BIC to decide which connections are important to add to a network. Algorithmically these approaches are different, but the end effect of both is a partially connected network. A critique to the partially connected networks is that in most cases (this is true for all 9 papers identified in this review that use partially connected ANNs) the resulting ANN is constructed following a greedy algorithm, i.e. the decision of cutting or creating a connection is not revaluated once more connections are altered.

The architecture of the ANNs contains some of the most important decisions that the modeller must make in order to use them for forecasting. The different variety of approaches to solve the modelling issues that are presented above, illustrate that there is no generally accepted methodology how to systematically construct neural networks. In many cases different modelling alternatives are not comparatively evaluated, making it difficult to assess if a particular setup is beneficial to forecasting accuracy or not. The literature has been focused in proposing several different methodologies to solve common problems, like the selection of the input variables, and has largely ignored to reconcile the accumulated knowledge, by assessing what works better and thereafter building on that. This has resulted in several publications arguing that the exact opposite is good modelling practice. A good example of this is the use of information criteria like AIC and BIC to select the appropriate inputs and specify the hidden layer for ANNs. Another significant weakness of the literature, which is connected to the architecture, is that important modelling decisions are documented vaguely or not at all. Several papers do not provide a selection methodology for input and hidden nodes and chose them either arbitrarily or by using a trial and error approach. To their support, this is an unsolved problem and there is no best practice. On the other hand, there are papers that do not document other important

architecture information, like the nature of the transfer functions, which makes it impossible to assess the validity of the implementation and replicate the experiments. This calls for stricter evaluation of the ANN literature.

3.4. ANN training

Once the architecture of the ANN is established the modeller has to decide the training algorithm and parameters. This involves a variety of decisions, some of which are directly connected to the training algorithm, like the learning rate, and some which are connected to the modellers approach to training, like the early stopping criterion. In this section we will discuss the findings from the literature that are associated with the ANN training.

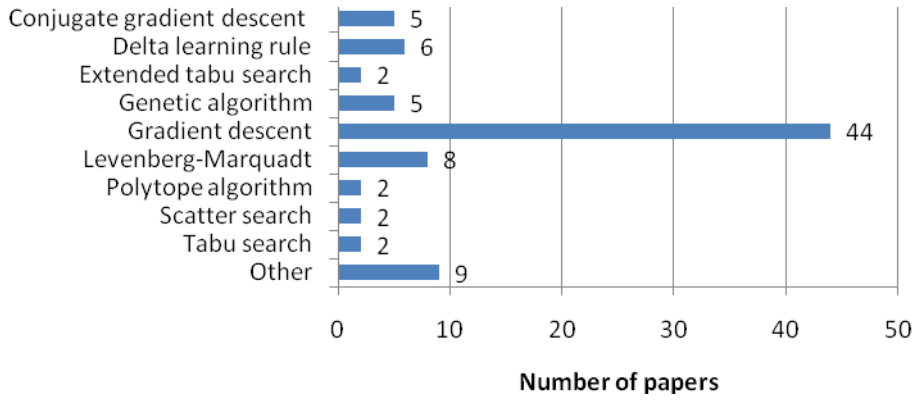

Fig. 9: Training algorithms employed in ANN forecasting literature

Several different training algorithms have been used in forecasting applications, as figure 9 summarises. The dominant algorithm is the gradient descent backpropagation training algorithm (52% of the papers). In figure 9, methods which are applied only to one paper are classified under the category "other" and include algorithms like BFGS quasi-Newton [70], Bayesian regularisation [87], simulated annealing [66], etc. Furthermore, there are 14 papers that do not record the training algorithm that was used. There are a number of papers that compare training algorithms for forecasting applications [25, 52, 66, 87-90]. Typically the gradient descend backpropagation algorithm is a benchmark in these studies and it is always outperformed. However, these studies should be viewed critically, since there is a publication bias. Gradient descent is an established algorithm so only papers that show improved results over it are expected to be published. Furthermore, there is an issue

of implementation validity, since the majority of these papers do not report the training parameters that were selected and use very few training initialisations, which are inadequate to overcome the problems caused by the stochastic nature of ANN training. The limited number of initialisations also limits the statistical analysis that can be done, as we discuss in more detail below. The high percentage of papers that use gradient descent can be explained by several factors; it has well studied and documented properties, the fact that the superiority of other alternatives is debatable and gradient descend has shown good performance in numerous studies and finally the limited selection of implemented algorithms in the widespread ANN software.

There are several cost functions that can be used to train ANNs. In this review numerous alternatives were identified, which are presented in figure 10. The measured cost is typically associated with the one step ahead in sample error. Teixeira and Rodrigues [91] use the four step ahead in sample error, which matches the forecasting horizon of their forecasting problem. This cost function is more appropriate as it minimises the error that is related with the objective of the forecasting exercise. The use of sum of squared errors (SSE), mean squared error and root mean squared error provide the same training result, but the latter two have higher computational cost, therefore there is no advantage in using them instead of the SSE. However a penalised for complexity version of SSE is bound to give different results. The same is true for cost functions that are based on different type of errors, like absolute errors, which are classified in figure 2.10 under the category "*other*", which includes all the cost functions that appear only once. The majority of the papers (57.4%) do not report the cost function that was used to train the ANNs.

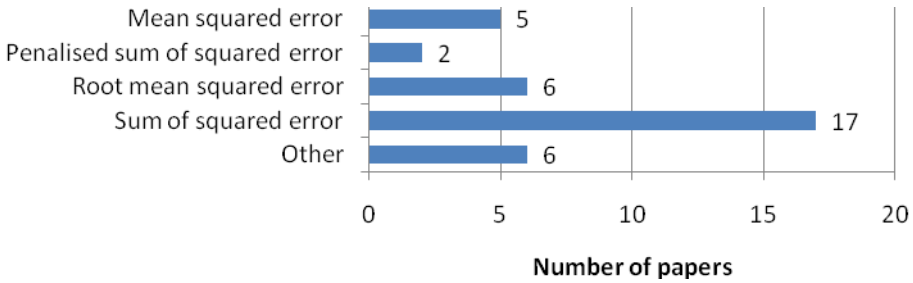

Fig. 10: Cost function in ANN forecasting literature.

Parameters like the training epochs/iterations, the learning parameters, the momentum and what stopping criterion was used, if any, are not recorded in many cases

either. Only 33% of the papers document for how many epochs the network was trained. The learning and the momentum is not documented in 75% of the papers, while the early stopping criterion is not discussed in 85% of the papers. For the latter, it is possible that in those papers that it is not discussed it is not used, as it is not necessary to produce forecasts. Ill documentations of these parameters harms the validity and the replicability of these papers [9, 10].

Another important parameter of the training of ANNs is the number of times that the network is initialised. Every time the network is initialised its weights are randomised and therefore produce a random starting point for the nonlinear optimisation that is performed during training. Because the training of the ANN can get stuck in local minima it is important that the networks are initialised several times to ensure a wide search of the error surface. If very few initialisations are evaluated then the reliability of the results is questionable, since they can be either good or bad due to randomness in the training and not due to the properties of the ANNs. On the other hand, if several initialisations are trained, the modeller can look at the distribution of the errors and evaluate if a good (or bad) solution is an outlier or close to the average behaviour of the model. Therefore it is important that the ANNs in forecasting studies are initialised multiple times and this number is reported. Table X summarises the reported multiple training initialisation in the literature.

Table X: Multiple training initialisations in the literature

Number of initialisations	Number of papers
3	1
5	2
10	4
15	1
20	1
50	1

Only 10 papers have multiple initialisations and from those only one [33] has over 30 initialisations that would typically allow statistical analysis of the results [92]. This represents a very small minority of the literature (11%). Liao and Fildes [60] do not initialise the training several times, but pick different initial weights with values between different bounds every time. The difference is that this does not guarantee that the ranges of the initial weights overlap, which therefore is equivalent to building a different model setups. For this reason this paper is not included in table X. The remaining papers do not report

multiple training initialisations. It is possible that more papers consider it, but it is not reported. This is a major problem for the literature. Considering that ANNs are extremely difficult to replicate, since the random seed used during training has to be identical to get the same results, it is principal that the robustness and the distribution of the errors of the ANNs due to training are evaluated. Results that are extracted after a single iteration of initialisation and training cannot be used to evaluate reliably the accuracy of the network and are impossible to replicate. On the other hand, if the behaviour of the network is examined over several initialisations, it can be expected that the results of the network, the next time it is trained, will be within easy to define bounds with a given confidence. This allows extracting valid and reliable conclusions. Note that in order to achieve full replication of ANN results several conditions must be satisfied; the software that simulates the ANNs must be identical, the random number generator that is used must be the same, the seed of the generator must be the same and the computer architecture, i.e. 32 or 64 bit, should be fixed and of course all the modelling parameters must be know. Therefore, it is unrealistic to expect replication of ANN papers results to the exact reported figures. However, it is relatively easy to ensure that the comparisons and the conclusions of a study hold with statistical confidence if the network is trained with multiple initialisations and the modelling parameters are reported fully and in detail. Naturally, in order to infer the level of confidence the number of initialisations must be known. Hence, to advance our understanding of ANNs it is imperative that multiple training initialisations become common practice.

3.5. ANN evaluation

The experimental design and evaluation framework of the papers that use ANN is strongly connected with designing a valid experiment and evaluation for any forecasting study. In forecasting literature there are several papers that discuss the design and the selection of the error measures [9, 11, 12, 93, 94]. What is important to evaluate in the case of the ANN forecasting literature is how closely these guidelines are followed and how valid are the comparisons.

One of the basic principles in forecasting evaluation is to use benchmarks to evaluate how good a model is. The majority of papers (85%) use non-ANN benchmarks to evaluate their models. Twelve papers do not use benchmarks. From those that use benchmarks only

5 include the random walk model. In forecasting studies it is important to include always a simple model like the random walk in order to have a desired accuracy minimum. If a model does not outperform a simple forecasting model such as the random walk, then there is no reason to use a more complicated model. Therefore, it is good practice to always include a random walk model or an equally simple model. Another important dimension of the evaluation is the error measure. Table XI includes the main error measure categories that can be found in the ANN literature. Note that most categories describe the family of the error measure, like *"absolute error measures*" and not the exact error metric, like mean absolute error, or median absolute error. This is done for economy of space, as there are 192 error measures employed in the literature. Note that under the category "*other"* measures several problem or domain specific measures are included, like the annualised returns or the Sharpe ratio.

Table XI: Error types in ANN literature

Error type	Number of papers
Absolute error measures	27
Absolute percentage error measures	30
AIC, BIC and variants	9
Correlation, R^2 and similar	12
Direction errors	8
Mean error	5
Relative absolute error measures	3
Squared error measures	53
Squared percentage error measures	1
Theil-U	3
Other	36

Table XII: Number of error measures used

Number of error measures	Number of papers
1	40
2	23
3	7
4	8
5	5
6	1
10	1
11	1

The most common error measures are based on some form of squared error. Forecasting literature has suggested using alternative measures [12, 93], since this family of errors is scale dependent, making them inappropriate for comparisons with several time series, and tends to overweight outliers due to the squaring. Absolute errors, which are the fourth most common family of errors, do not overemphasise outliers, but they still do not allow comparing across different time series. The most common error measure family to compare across different time series in the ANN literature is based on absolute percentage error metrics. Although these metrics are scale independent, and usually easy to interpret,

they have been criticised for being biased [12, 94]. The forecasting literature in order to remedy this has suggested a set of different error measures that are scale independent and less biased, like corrections on the common mean absolute percentage error [54], the absolute scaled errors [94] and the geometric root mean squared error [95, 96]. Such advances in error measures are not adopted in the ANN forecasting literature. On the other hand, there is a limited use of relative errors, which to some extent addresses the criticism to the other error measures [12]. One other positive of the evaluation metrics used in the ANN literature is that a lot of domain specific measures are used, which allow to make use of the dataset properties in order to get meaningful performance measures. Table XII summarises the number of error measures used in the ANN papers. About half of the papers (47%) use a single error measure, while a smaller portion uses several error measures, identifying that different accuracy calculations can provide different ranking of the models [54].

Adya and Collopy [9] investigated the validity of a number of ANN papers and suggested that it is important to provide both the in-sample and out-of-sample errors, since this way it can be assessed whether the ANN model has captured the structure of the time series and generalises well. In ANN literature only 32% of the papers report the errors in both subsets. The majority (64%) of the paper do not report the in-sample errors and a small part of papers (7%) do not provide out-of-sample errors.

Forecasting literature has stressed the importance of having a large sample of errors through multiple time series or rolling origin evaluation [12]. Both allow having more errors to construct the error summary statistics and therefore, better confidence in the results. Table VI and figure 5 illustrate the number of time series in the ANN literature and as discussed before the majority of papers use a single time series and only 12 papers consider 10 or more time series. Therefore one would expect the authors to use rolling origin evaluation in order to increase the sample of errors. However, only three papers state clearly that such an evaluation scheme was used. This limits considerably the confidence of the results of most ANN papers.

The ANN literature seems to be lagging in following the recommendations of the literature for designing an adequate experimental design for empirical evaluations [9, 11, 12, 93, 94]. This in conjunction with the problems discussed in the previous section regarding the reliability, robustness and replicability of the results limits the number of

papers from which safe conclusions can be drawn, something that was also identified by Adya and Collopy [9].

3.6. Findings regarding ANN forecasting performance

Adya and Collopy [9] found that ANNs outperform benchmarks 73% of the time, if only the papers that meet the criteria for valid evaluation are considered. In the M3 competition, which used 3003 time series, ANNs did not perform well and failed to outperform simpler models [54]. Armstrong [97] argues that too much research effort is devoted on ANNs, taking into consideration the modelling difficulties and their unproven performance. However he points out that there are studies that demonstrate good performance, referring to Liao and Fildes [60], and we need to identify the conditions under which ANNs are useful. Callen [98] advises caution on reading the positive results of ANN, warning of a possible publication bias, that usually the successful applications are published. Bunn [99] argues that even if there is empirical evidence in favour of ANNs, it will require advances in their explainability and robustness diagnostics before forecasters use them with confidence.

In this survey if the limitations stressed in the previous sections are not considered, ANNs outperform benchmarks in 70% of the papers. However, under stricter evaluation only a handful of papers can be considered and this percentage changes. By restricting the results to papers that use either reported rolling origin evaluation or more than 10 time series and follow a valid evaluation scheme only 14 papers can be considered, from which 64% report that ANNs outperform the benchmarks that were used in these studies. Callen et al. [98] forecast quarterly firm earnings and find ANNs unable to outperform linear models. Cao et al. [100] find that both the univariate and the multivariate ANNs perform better than linear models in forecasting daily stock returns from the Shanghai stock market. Heravi et al. [101] try to model the European industrial production and find that linear models perform better than ANN, but the latter can pick up directional changes more accurate. Hill et al. [42] use data from the M1 competition and find that ANN perform better for all time series apart from the annual data, for which the ANN were not significantly different, indicating an effect of the time series frequency on the ANN performance. Kotsialos et al. [21] find ANNs to perform marginally better, but due to their complexity they advise the use of exponential smoothing models instead. Liao and Fildes [60] use a large

telecommunication time series dataset and find that overall robust trend model is better, but ANNs have very similar accuracy outperforming all other benchmarks. Motiwalla and Wahab [72] find that ANN have better investment performance than linear regression models and a passive buy and hold strategy. Nelson et al. [43] revisit the M1 dataset and provide evidence that deseasonalising the time series helps to improve the forecasting performance of ANNs, validating the results of Hill et al. [42]. Terasvirta et al. [71] find that ANN models are better than the benchmarks at long forecasting horizons, but overall are worse, in forecasting monthly macroeconomic variables. Thomassey et al. [23] find that ANNs are better at predicting weekly textile sales than linear benchmarks. Zhang and Qi [46] evaluate the effect of detrending and deseasonlising time series for forecasting with ANN and find that this step helps and that ANN are able to outperform ARIMA models. Zhang et al. [102] find that ANN perform better than univariate and multivariate linear models at predicting the quarterly earnings per share. Jursa and Rohrig [103] find that ANNs are better than a nearest neighbourhood search forecasting model at predicting short term wind farm production. Moreno and Olmeda [61] do not find any clear advantage of ANNs against AR and ARX models in forecasting Morgan Stanley capital international indices. Note that the above papers do not consider the problem of multiple initialisations that was discussed before, with the exception of Liao and Fildes [60].

Overall, ANNs show evidence of good performance, repeating the findings of previous reviews [2, 9] that reported ANNs being able to surpass in performance established benchmarks. However, an important finding is that the majority of ANN papers cannot be used in this meta-evaluation of ANNs due to several limitations in their experimental design. Addressing these limitations and raising the degree of replicability of the ANN studies should be important targets for ANN research.

4. Conclusions

This study aims to provide a critical overview of the advances in forecasting with ANNs. The contribution of the research is analysed in seven main axes and the current state-of-the-art in forecasting with ANN models is presented, along with the pressing research questions. More than a decade ago Zhang et al. [2] set a number of future research questions for the field of ANNs in forecasting. This study tries to see how these have been addressed since then. A key question set then was how do ANNs model time series that

allows them to outperform conventional methods. Unfortunately our understanding of the inner workings of ANNs is still incomplete and limited research effort has been put towards that target [70]. Another key question that was set was how to systematically build an ANN for a given problem. On this front there have been substantial advances. We know now that the input vector is the key determinant of ANN accuracy, followed by the specification of the hidden layer. There have been several papers that try to address these issues, yet no consensus on what is the best way has been reached. Other modelling decisions, like the choice of the transfer functions, have been less researched. There have been several papers that try to systematically build ANN models with relatively few arbitrary modelling choices; however there is still no fully systematic or automated modelling methodology. Furthermore, the majority of ANN papers do not address these modelling issues in a methodical way, resolving to trial and error approaches that do not advance our understanding of ANNs. Another question that was set was related to identifying the best training algorithm or method for time series forecasting. Although the standard gradient descent backpropagation is still the most widely applied training algorithm, different alternatives have been developed. There is some evidence that these algorithms perform better, but rigorous comparative evaluations that adhere to the criteria set by the established forecasting research do not exist. The last question posed was related to data pre-processing and sampling. The literature agrees that ANNs perform better when large samples are available, but the best way to pre-process the input data, if needed at all, is still debatable. The debate is mainly focused on the issue of how to best model trend and seasonality with ANNs. There is evidence that removing those as a pre-processing step, through first and seasonal differences, is beneficial to the accuracy of ANNs. However, there is also evidence that ANNs can forecast these time series at least as good as benchmarks without the need to pre-process the inputs. Other pre-processing methodologies, like using the logarithm of the time series to aid the training of the models or the Box-Cox transformation, have been proposed, but they have not been widely used.

This study identifies a set of problems in the ANN literature, which are outlined here. Key modelling issues are overlooked. Very few papers were found to address the issue of initialising multiple times the networks weights during initialisation. Multiple initialisations are necessary in order to evaluate the robustness and the reliability of the ANN model, due to the stochastic nature of the training and the problem of local minima. In addition to that,

multiple initialisations provide a better search for parameters. Furthermore, several parameters of the ANN models are set either arbitrarily or following a trial and error approach that does not advance out knowledge of ANNs and makes questionable the implementation validity of several papers.

Another principal problem is that several modelling decisions are not properly documented in the papers. This harms the reliability of the results, limits the contribution to our understanding of ANNs and makes the replication of experiments impossible. Furthermore, it hinders further meta-analysis of the results.

The ANN literature is lagging behind in implementing the suggestions of the forecasting literature on what constitutes a valid experimental design for empirical evaluation. Selecting a large number of time series, using rolling origin evaluation and selecting appropriate benchmarks and error measures is important in order to be able to provide valid and reliable conclusions. These decisions, like the ANN modelling decisions, must be clearly documented, to raise transparency in the literature and allow meta-analysis of the results in order to advance our understanding of ANNs. Once the experimental design allows producing detailed error data it is then possible to perform valid statistical analysis of the results, which will result in more reliable findings and evaluation of the conditions under which this results are valid.

Several open research questions are identified. There is evidence in the literature that the frequency of the time series is related to the performance of ANNs [32, 40, 42]. Furthermore, it has been long established that time series of different frequencies require different forecasting methodologies and exploration tools [37, 38]. Therefore, we need to explore whether ANNs are able to forecast both low and high frequency data, and what the required changes are in the modelling methodology, if any. This becomes especially important as there are more high frequency datasets available and the constant increase of computational resources allows us to use them [36]. Another key issue is the reconciliation of the literature that is addressing the issue of specifying the input variables and the hidden layers for ANNs. Several different methodologies have been proposed, most of which outperform all benchmarks in the limited number of studies that they have been applied. However, there is no direct comparison between them. It is necessary to rigorously evaluate the competing ANN modelling methodologies. This will reveal best practices and also allow us to better understand why some methods work better than others. Keeping in mind the

current findings of the literature that the most important determinant of ANN performance is the input vector, the specification of the ANNs' input variables should be addressed first, before other ANN modelling variables such as the hidden layers and nodes. Furthermore, one issue related to the time series frequency is whether these methodologies are equally applicable to different frequencies or not, and which are better suited for each problem. The issue of selecting the transfer functions has not been adequately researched either, leading most researchers to arbitrarily choose between the most common types. Their impact in forecasting is not well understood and should be explored further. The scaling of the inputs is also inadequately researched. In the literature there is no large scale empirical evaluation or a theoretical proof that answers how this problem should be tackled. There are several alternatives on how to scale the inputs of an ANN and also there is the option of restricting the bounds of the scaling more than what is required by the transfer functions. The effects of these choices are unclear, as is the magnitude of their impact in ANNs' forecasting accuracy. Finally, it is important to invest more research in the meta-analysis of the results in the literature in order to understand better how ANNs work and explain the evidence of superior performance over established benchmarks. This is a key step for making the use of ANNs more widespread and accepted.

Table XIII: List of journal papers retrieved for the survey

Computers and Operations Research	Vroomen et al. [30]	Amaral et al. [104]
Desilets et al. [105]	El-Fallahi [90]	Cancelo et al. [106]
Markham and Rakes [32]	Zhang and Qi [46]	Jursa and Rohrig [103]
Condon et al. [14]	Bodyanskiy and Popov [53]	Soares and Medeiros [107]
Leung et al. [108]	Casqueiro and Rodrigues [109]	**Journal of Forecasting**
Lind and Sulek [110]	Curry and Morgan [89]	Lachtermacher and Fuller [44]
Motiwalla and Wahab [72]	Freitas and Rodrigues [111]	Connor [112]
Zhang [34]	Lin and Chen [113]	Donaldson and Kamstra [114]
Zhang et al. [35]	Curry [47]	Haefke and Helmenstein [115]
Curry et al. [25]	Landajo et al. [116]	Adya and Collopy [9]
Chen et al. [117]	Moreno and Olmeda [61]	Anders et al. [63]
Chen and Leung [118]	Nikolopoulos et al. [16]	Cottrell et al. [119]
Marti and El-Fallahi [120]	Andreou et al. [121]	Li et al. [122]
Cao et al. [100]	Carbonneau et al. [123]	Nelson et al. [43]
Gupta and Singh [124]	Hahn et al. [41]	Qi and Maddala [58]
Liao and Fildes [60]	**International Journal of Forecasting**	Refenes and Zapranis [65]
Torres et al. [52]	Gorr et al. [125]	Venkatachalam and Sohl [126]
Yu et al. [127]	Hill et al. [128]	Bentz and Merunka [24]
Setzler et al. [17]	Callen et al. [98]	Lam and Lam [129]
Decision Sciences	Church and Curran [4]	Moshiri and Cameron [130]
Jain and Nag [131]	Dougherty and Cobbett [19]	Schittenkopf et al. [132]
Swanson and White [59]	Kirby et al. [20]	Taylor [133]
Desai and Bharati [134]	Kim and Chun [135]	Swanson and Zeng [83]
Hu et al. [33]	Zhang et al. [2]	Dunis and Huang [136]
Jiang et al. [26]	Balkin and Ord [45]	Kaashoek and Dijk [69]
Papatla and Zahedi [29]	Darbellay and Slama [3]	Nag and Mitra [73]
Sexton et al. [82]	Leung et al. [79]	Amilon [86]
Zhang et al. [102]	Thomas [137]	Kanas [138]
European Journal of Operational Research	Gencay and Selcuk [139]	Dahl and Hylleberg [57]
Hruschka [140]	Qi [141]	Lindemann et al. [142]
Bunn [99]	Tkacz [143]	Moshiri and Brown [68]
Wang [144]	Corcoran et al. [13]	Chen and Leung [145]
Wittkemper and Steiner [146]	Olson and Mossman [80]	Kajitani et al. [67]
Wood and Dasgupta [51]	Heravi et al. [101]	Kotsialos et al. [21]
Teixeira and Rodrigues [91]	Conejo et al. [147]	Pantelidaki [28]
Badiru and Sieger [148]	Ghiassi et al. [55]	Gradojevic and Yang [149]
Sexton et al. [88]	Hippert et al. [40]	Medeiros et al. [64]
Sexton et al. [87]	Novales [150]	Hruschka [31]
Prybutok et al. [15]	Terasvirta et al. [71]	Bekiros and Georgoutsos [151]
Dia [18]	Terasvirta et al. [152]	**Management Science**
Kuo [22]	Armstrong [97]	Hill et al. [42]
Qi and Zhang [62]	de Menezes and Nikolaev [39]	Kim et al. [27]
Sahin et al. [81]	Taylor et al. [38]	
Setiono and Thong [70]	Preminger and Frank [153]	
Thomassey et al. [23]	da Silva et al. [66]	

References

[1] M. I. Jordan, and C. M. Bishop, “Neural networks,” *Acm Computing Surveys,* vol. 28, no. 1, pp. 73-75, Mar, 1996.

[2] G. Q. Zhang, B. E. Patuwo, and M. Y. Hu, “Forecasting with artificial neural networks: The state of the art,” *International Journal of Forecasting,* vol. 14, no. 1, pp. 35-62, Mar, 1998.

[3] G. A. Darbellay, and M. Slama, “Forecasting the short-term demand for electricity - Do neural networks stand a better chance?,” *International Journal of Forecasting,* vol. 16, no. 1, pp. 71-83, Jan-Mar, 2000.

[4] K. B. Church, and S. P. Curram, “Forecasting consumers' expenditure: A comparison between econometric and neural network models,” *International Journal of Forecasting,* vol. 12, no. 2, pp. 255-267, Jun, 1996.

[5] K. Hornik, “Approximation capabilities of multilayer feedforward networks,” *Neural Networks,* vol. 4, no. 2, pp. 251-257, 1991.

[6] K. Hornik, M. Stinchcombe, and H. White, “Multilayer feedforward networks are universal approximators ” *Neural Networks,* vol. 2, no. 5, pp. 359-366, 1989.

[7] V. d. H. f. B. e.V. "Vienna List," 07/09/2009, 2009; http://bach.wu-wien.ac.at/bachapp/cgi-bin/fides/fides.aspx/fides.aspx?journal=true;lang=DE.

[8] I. WoK. "Journal Citation Reports," 07/08/2009, 2009; http://admin-apps.isiknowledge.com.ezproxy.lancs.ac.uk/JCR/JCR?SID=1FAPBHeEN42HLAIn%40bo .

[9] M. Adya, and F. Collopy, “How effective are neural networks at forecasting and prediction? A review and evaluation,” *Journal of Forecasting,* vol. 17, no. 5-6, pp. 481-495, Sep-Nov, 1998.

[10] S. F. Crone, and D. B. Preßmar, "An Extended Evaluation Framework for Neural Network Publications in Sales Forecasting."

[11] F. Collopy, M. Adya, and J. S. Armstrong, “Principle for examining predictive validity: The case of information systems spending forecats,” *Information Systems Research,* vol. 5, no. 2, pp. 170-179, Jun, 1994.

[12] L. Tashman, “Out-of-sample tests of forecasting accuracy: an analysis and review,” *International Journal of Forecasting,* vol. 16, pp. 437-450, 2000.

[13] J. J. Corcoran, I. D. Wilson, and J. A. Ware, “Predicting the geo-temporal variations of crime and disorder,” *International Journal of Forecasting,* vol. 19, no. 4, pp. 623-634, Oct-Dec, 2003.

[14] E. M. Condon, B. L. Golden, and E. A. Wasil, “Predicting the success of nations at the Summer Olympics using neural networks,” *Computers & Operations Research,* vol. 26, no. 13, pp. 1243-1265, Nov, 1999.

[15] V. R. Prybutok, J. S. Yi, and D. Mitchell, “Comparison of neural network models with ARIMA and regression models for prediction of Houston's daily maximum ozone concentrations,” *European Journal of Operational Research,* vol. 122, no. 1, pp. 31-40, Apr, 2000.

[16] K. Nikolopoulos, P. Goodwin, A. Patelis *et al.*, “Forecasting with cue information: A comparison of multiple regression with alternative forecasting approaches,” *European Journal of Operational Research,* vol. 180, no. 1, pp. 354-368, Jul, 2007.

[17] H. Setzler, C. Saydam, and S. Park, “EMS call volume predictions: A comparative study,” *Computers & Operations Research,* vol. 36, no. 6, pp. 1843-1851, Jun, 2009.

[18] H. Dia, “An object-oriented neural network approach to short-term traffic forecasting,” *European Journal of Operational Research,* vol. 131, no. 2, pp. 253-261, Jun, 2001.

[19] M. S. Dougherty, and M. R. Cobbett, “Short-term inter-urban traffic forecasts using neural networks,” *International Journal of Forecasting,* vol. 13, no. 1, pp. 21-31, Mar, 1997.

[20] H. R. Kirby, S. M. Watson, and M. S. Dougherty, “Should we use neural networks or statistical models for short-term motorway traffic forecasting?,” *International Journal of Forecasting,* vol. 13, no. 1, pp. 43-50, Mar, 1997.

[21] A. Kotsialos, M. Papageorgiou, and A. Poulimenos, “Long-term sales forecasting using Holt-Winters and neural network methods,” *Journal of Forecasting,* vol. 24, no. 5, pp. 353-368, Aug, 2005.

[22] R. J. Kuo, “A sales forecasting system based on fuzzy neural network with initial weights generated by genetic algorithm,” *European Journal of Operational Research,* vol. 129, no. 3, pp. 496-517, Mar, 2001.

[23] S. Thomassey, M. Happiette, and J. M. Castelain, “A short and mean-term automatic forecasting system - application to textile logistics,” *European Journal of Operational Research,* vol. 161, no. 1, pp. 275-284, Feb, 2004.

[24] Y. Bentz, and D. Merunka, “Neural networks and the multinomial legit for brand choice modelling: A hybrid approach,” *Journal of Forecasting,* vol. 19, no. 3, pp. 177-200, Apr, 2000.

[25] B. Curry, P. Morgan, and M. Silver, “Neural networks and non-linear statistical methods: an application to the modelling of price-quality relationships,” *Computers & Operations Research,* vol. 29, no. 8, pp. 951-969, Jul, 2002.

[26] J. J. Jiang, M. S. Zhong, and G. Klein, “Marketing category forecasting: An alternative of BVAR - Artificial neural networks,” *Decision Sciences,* vol. 31, no. 4, pp. 789-812, Fal, 2000.

[27] Y. Kim, W. N. Street, G. J. Russell *et al.*, “Customer targeting: A neural network approach guided by genetic algorithms,” *Management Science,* vol. 51, no. 2, pp. 264-276, Feb, 2005.

[28] S. Pantelidaki, and D. W. Bunn, “Development of a multifunctional sales response model with the diagnostic aid of artificial neural networks,” *Journal of Forecasting,* vol. 24, no. 7, pp. 505-521, Nov, 2005.

[29] P. Papatla, and M. Zahedi, “Leveraging the strengths of choice models and neural networks: A multiproduct comparative analysis,” *Decision Sciences,* vol. 33, no. 3, pp. 433-468, Sum, 2002.

[30] B. Vroomen, P. H. Franses, and E. van Nierop, “Modeling consideration sets and brand choice using artificial neural networks,” *European Journal of Operational Research,* vol. 154, no. 1, pp. 206-217, Apr, 2004.

[31] H. Hruschka, “Using a heterogeneous multinomial probit model with a neural net extension to model brand choice,” *Journal of Forecasting,* vol. 26, no. 2, pp. 113-127, Mar, 2007.

[32] I. S. Markham, and T. R. Rakes, “The effect of sample size and variability of data on the comparative performance of artificial neural networks and regression,” *Computers & Operations Research,* vol. 25, no. 4, pp. 251-263, Apr, 1998.

[33] M. Y. Hu, G. Q. Zhang, C. Z. Jiang *et al.*, “A cross-validation analysis of neural network out-of-sample performance in exchange rate forecasting,” *Decision Sciences,* vol. 30, no. 1, pp. 197-216, Win, 1999.

[34] G. P. Zhang, “An investigation of neural networks for linear time-series forecasting,” *Computers & Operations Research,* vol. 28, no. 12, pp. 1183-1202, Oct, 2001.

[35] G. P. Zhang, B. E. Patuwo, and M. Y. Hu, “A simulation study of artificial neural networks for nonlinear time-series forecasting,” *Computers & Operations Research,* vol. 28, no. 4, pp. 381-396, Apr, 2001.

[36] R. F. Engle, “The econometrics of ultra-high-frequency data,” *Econometrica,* vol. 68, no. 1, pp. 1-22, Jan, 2000.

[37] C. W. J. Granger, “Extracting information from mega-panels and high-frequency data,” *Statistica Neerlandica,* vol. 52, no. 3, pp. 258-272, Nov, 1998.

[38] J. W. Taylor, L. M. de Menezes, and P. E. McSharry, “A comparison of univariate methods for forecasting electricity demand up to a day ahead,” *International Journal of Forecasting,* vol. 22, no. 1, pp. 1-16, Jan-Mar, 2006.

[39] L. M. de Menezes, and N. Y. Nikolaev, “Forecasting with genetically programmed polynomial neural networks,” *International Journal of Forecasting,* vol. 22, no. 2, pp. 249-265, Apr-Jun, 2006.

[40] H. S. Hippert, D. W. Bunn, and R. C. Souza, “Large neural networks for electricity load forecasting: Are they overfitted?,” *International Journal of Forecasting,* vol. 21, no. 3, pp. 425-434, Jul-Sep, 2005.

[41] H. Hahn, S. Meyer-Nieberg, and S. Pickl, "Electric load forecasting methods: Tools for decision making." pp. 902-907, 2009.

[42] T. Hill, M. O'Connor, and W. Remus, “Neural network models for time series forecasts,” *Management Science,* vol. 42, no. 7, pp. 1082-1092, Jul, 1996.

[43] M. Nelson, T. Hill, W. Remus *et al.*, “Time series forecasting using neural networks: Should the data be deseasonalized first?,” *Journal of Forecasting,* vol. 18, no. 5, pp. 359-367, Sep, 1999.

[44] G. Lachtermacher, and J. D. Fuller, “Backpropagation in Time-Series Forecasting,” *Journal of Forecasting,* vol. 14, no. 4, pp. 381-393, Jul, 1995.

[45] S. D. Balkin, and J. K. Ord, “Automatic neural network modeling for univariate time series,” *International Journal of Forecasting,* vol. 16, no. 4, pp. 509-515, Oct-Dec, 2000.

[46] G. P. Zhang, and M. Qi, “Neural network forecasting for seasonal and trend time series,” *European Journal of Operational Research,* vol. 160, no. 2, pp. 501-514, Jan, 2005.

[47] B. Curry, “Neural networks and seasonality: Some technical considerations,” *European Journal of Operational Research,* vol. 179, no. 1, pp. 267-274, May, 2007.

[48] S. F. Crone, and R. Dhawan, "Forecasting seasonal time series with neural networks: A sensitivity analysis of architecture parameters." pp. 2099-2104.

[49] G. P. Zhang, and D. M. Kline, “Quarterly time-series forecasting with neural networks,” *Ieee Transactions on Neural Networks,* vol. 18, no. 6, pp. 1800-1814, Nov, 2007.

[50] E. Ghysels, and D. R. Osborn, *The Econometric Analysis of Seasonal Time Series*, Cambridge: Cambridge University Press, 2001.

[51] D. Wood, and B. Dasgupta, “Classifying trend movements in the MSCI USA Capital market index - A comparison of regression, arima and neural network methods,” *Computers & Operations Research,* vol. 23, no. 6, pp. 611-622, Jun, 1996.

[52] M. Torres, C. Hervas, and F. Amador, “Approximating the sheep milk production curve through the use of artificial neural networks and genetic algorithms,” *Computers & Operations Research,* vol. 32, no. 10, pp. 2653-2670, Oct, 2005.

[53] Y. Bodyanskiy, and S. Popov, “Neural network approach to forecasting of quasiperiodic financial time series,” *European Journal of Operational Research,* vol. 175, no. 3, pp. 1357-1366, Dec, 2006.

[54] S. Makridakis, and M. Hibon, “The M3-Competition: results, conclusions and implications,” *International Journal of Forecasting,* vol. 16, no. 4, pp. 451-476, Oct-Dec, 2000.

[55] M. Ghiassi, H. Saidane, and D. K. Zimbra, “A dynamic artificial neural network model for forecasting time series events,” *International Journal of Forecasting,* vol. 21, no. 2, pp. 341-362, Apr-Jun, 2005.

[56] U. Anders, and O. Korn, “Model selection in neural networks,” *Neural Networks,* vol. 12, no. 2, pp. 309-323, Mar, 1999.

[57] C. M. Dahl, and S. Hylleberg, “Flexible regression models and relative forecast performance,” *International Journal of Forecasting,* vol. 20, no. 2, pp. 201-217, Apr-Jun, 2004.

[58] M. Qi, and G. S. Maddala, “Economic factors and the stock market: A new perspective,” *Journal of Forecasting,* vol. 18, no. 3, pp. 151-166, May, 1999.

[59] N. R. Swanson, and H. White, “Forecasting economic time series using flexible versus fixed specification and linear versus nonlinear econometric models,” *International Journal of Forecasting,* vol. 13, no. 4, pp. 439-461, Dec, 1997.

[60] K. P. Liao, and R. Fildes, “The accuracy of a procedural approach to specifying feedforward neural networks for forecasting,” *Computers & Operations Research,* vol. 32, no. 8, pp. 2151-2169, Aug, 2005.

[61] D. Moreno, and I. Olmeda, “Is the predictability of emerging and developed stock markets really exploitable?,” *European Journal of Operational Research,* vol. 182, no. 1, pp. 436-454, Oct, 2007.

[62] M. Qi, and G. P. Zhang, “An investigation of model selection criteria for neural network time series forecasting,” *European Journal of Operational Research,* vol. 132, no. 3, pp. 666-680, Aug, 2001.

[63] U. Anders, O. Korn, and C. Schmitt, “Improving the pricing of options: A neural network approach,” *Journal of Forecasting,* vol. 17, no. 5-6, pp. 369-388, Sep-Nov, 1998.

[64] M. C. Medeiros, T. Terasvirta, and G. Rech, “Building neural network models for time series: A statistical approach,” *Journal of Forecasting,* vol. 25, no. 1, pp. 49-75, Jan, 2006.

[65] A. P. N. Refenes, and A. D. Zapranis, “Neural model identification, variable selection and model adequacy,” *Journal of Forecasting,* vol. 18, no. 5, pp. 299-332, Sep, 1999.

[66] A. P. A. da Silva, V. H. Ferreira, and R. M. G. Velasquez, “Input space to neural network based load forecasters,” *International Journal of Forecasting,* vol. 24, no. 4, pp. 616-629, Oct-Dec, 2008.

[67] Y. Kajitani, K. W. Hipel, and I. McLeod, “Forecasting nonlinear time series with feed-forward neural networks: A case study of Canadian lynx data,” *Journal of Forecasting,* vol. 24, no. 2, pp. 105-117, Mar, 2005.

[68] S. Moshiri, and L. Brown, “Unemployment variation over the business cycles: a comparison of forecasting models,” *Journal of Forecasting,* vol. 23, no. 7, pp. 497-511, Nov, 2004.

[69] J. F. Kaashoek, and H. K. Van Dijk, “Neural network pruning applied to real exchange rate analysis,” *Journal of Forecasting,* vol. 21, no. 8, pp. 559-577, Dec, 2002.

[70] R. Setiono, and J. Y. L. Thong, “An approach to generate rules from neural networks for regression problems,” *European Journal of Operational Research,* vol. 155, no. 1, pp. 239-250, May, 2004.

[71] T. Terasvirta, D. van Dijk, and M. C. Medeiros, “Linear models, smooth transition autoregressions, and neural networks for forecasting macroeconomic time series: A re-examination,” *International Journal of Forecasting,* vol. 21, no. 4, pp. 755-774, Oct-Dec, 2005.

[72] L. Motiwalla, and M. Wahab, “Predictable variation and profitable trading of US equities: a trading simulation using neural networks,” *Computers & Operations Research,* vol. 27, no. 11-12, pp. 1111-1129, Sep-Oct, 2000.

[73] A. K. Nag, and A. Mitra, “Forecasting daily foreign exchange rates using genetically optimized neural networks,” *Journal of Forecasting,* vol. 21, no. 7, pp. 501-511, Nov, 2002.

[74] J. D. Hamilton, “A Parametric Approach to Flexible Nonlinear Inference,” *Econometrica,* vol. 69, no. 3, pp. 537-73, 2001.

[75] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, *Time series analysis: forecasting and control*, 3rd ed., New Jersey: Prentice Hall Inc., 1994.

[76] J. Moody, and J. Utans, "PRINCIPLED ARCHITECTURE SELECTION FOR NEURAL NETWORKS - APPLICATION TO CORPORATE BOND RATING PREDICTION," *Advances in Neural Information Processing Systems 4*, Advances in Neural Information Processing Systems J. E. Moody, S. J. Hanson and R. P. Lippmann, eds., pp. 683-690, 1992.

[77] G. Rech, T. Terasvirta, and R. Tschernig, “A simple variable selection technique for nonlinear models,” *Communications in Statistics-Theory and Methods,* vol. 30, no. 6, pp. 1227-1241, 2001.

[78] W. J. M. Levelt, “Are multilayer feedforward networks effectively Turing Machines?,” *Psychological Research,* vol. 52, no. 2-3, pp. 153-157, 1990.

[79] M. T. Leung, H. Daouk, and A. S. Chen, “Forecasting stock indices: a comparison of classification and level estimation models,” *International Journal of Forecasting,* vol. 16, no. 2, pp. 173-190, Apr-Jun, 2000.

[80] D. Olson, and C. Mossman, “Neural network forecasts of Canadian stock returns using accounting ratios,” *International Journal of Forecasting,* vol. 19, no. 3, pp. 453-465, Jul-Sep, 2003.

[81] S. O. Sahin, F. Ulengin, and B. Ulengin, “Using neural networks and cognitive mapping in scenario analysis: The case of Turkey's inflation dynamics,” *European Journal of Operational Research,* vol. 158, no. 1, pp. 124-145, Oct, 2004.

[82] R. S. Sexton, R. S. Sriram, and H. Etheridge, “Improving decision effectiveness of artificial neural networks: A modified genetic algorithm approach,” *Decision Sciences,* vol. 34, no. 3, pp. 421-442, Sum, 2003.

[83] N. R. Swanson, and T. Zeng, “Choosing among competing econometric forecasts: Regression-based forecast combination using model selection,” *Journal of Forecasting,* vol. 20, no. 6, pp. 425-440, Sep, 2001.
[84] T. Terasvirta, C.-F. Lin, and C. W. J. Granger, *Power of the Neural Network Linearity Test*, Department of Economics, UC San Diego, 1991.
[85] H. White, "An additional hidden unit test for neglected non-linearity in multilayer feedforward networks." pp. 451-455.
[86] H. Amilon, “A neural network versus Black-Scholes: A comparison of pricing and hedging performances,” *Journal of Forecasting,* vol. 22, no. 4, pp. 317-335, Jul, 2003.
[87] R. S. Sexton, R. E. Dorsey, and J. D. Johnson, “Optimization of neural networks: A comparative analysis of the genetic algorithm and simulated annealing,” *European Journal of Operational Research,* vol. 114, no. 3, pp. 589-601, May, 1999.
[88] R. S. Sexton, B. Alidaee, R. E. Dorsey *et al.*, “Global optimization for artificial neural networks: A tabu search application,” *European Journal of Operational Research,* vol. 106, no. 2-3, pp. 570-584, Apr, 1998.
[89] B. Curry, and P. H. Morgan, “Model selection in neural networks: Some difficulties,” *European Journal of Operational Research,* vol. 170, no. 2, pp. 567-577, Apr, 2006.
[90] A. El-Fallahi, R. Marti, and L. Lasdon, “Path relinking and GRG for artificial neural networks,” *European Journal of Operational Research,* vol. 169, no. 2, pp. 508-519, Mar, 2005.
[91] J. C. Teixeira, and A. J. Rodrigues, “An applied study on recursive estimation method, neural networks and forecasting,” *European Journal of Operational Research,* vol. 101, no. 2, pp. 406-417, Sep, 1997.
[92] A. H. Kvanli, R. J. Pavur, and K. Keeling, *Introduction to Business Statistics*, 6th ed., Mason, Ohio: Thomson/South-Western, 2002.
[93] J. S. Armstrong, and R. Fildes, “On the selection of error measures for comparisons among forecasting methods,” *Journal of Forecasting,* vol. 14, no. 1, pp. 67-71, Jan, 1995.
[94] R. J. Hyndman, and A. B. Koehler, “Another look at measures of forecast accuracy,” *International Journal of Forecasting,* vol. 22, pp. 679-688, 2006.
[95] A. A. Syntetos, and J. E. Boylan, “The accuracy of intermittent demand estimates,” *International Journal of Forecasting,* vol. 21, no. 2, pp. 303-314, Apr-Jun, 2005.
[96] R. Fildes, “The evaluation of extrapolative forecasting methods,” *International Journal of Forecasting,* vol. 8, no. 1, pp. 81-98, Jun, 1992.
[97] J. S. Armstrong, “Findings from evidence-based forecasting: Methods for reducing forecast error,” *International Journal of Forecasting,* vol. 22, no. 3, pp. 583-598, 2006.
[98] J. L. Callen, C. C. Y. Kwan, P. C. Y. Yip *et al.*, “Neural network forecasting of quarterly accounting earnings,” *International Journal of Forecasting,* vol. 12, no. 4, pp. 475-482, Dec, 1996.
[99] D. W. Bunn, “Non-traditional methods of forecasting,” *European Journal of Operational Research,* vol. 92, no. 3, pp. 528-536, Aug, 1996.
[100] Q. Cao, K. B. Leggio, and M. J. Schniederjans, “A comparison between Fama and French's model and artificial neural networks in predicting the Chinese stock market,” *Computers & Operations Research,* vol. 32, no. 10, pp. 2499-2512, Oct, 2005.

[101] S. Heravi, D. R. Osborn, and C. R. Birchenhall, “Linear versus neural network forecasts for European industrial production series,” *International Journal of Forecasting,* vol. 20, no. 3, pp. 435-446, Jul-Sep, 2004.
[102] W. Zhang, Q. Cao, and M. J. Schniederjans, “Neural network earnings per share forecasting models: A comparative analysis of alternative methods,” *Decision Sciences,* vol. 35, no. 2, pp. 205-237, Spr, 2004.
[103] R. Jursa, and K. Rohrig, “Short-term wind power forecasting using evolutionary algorithms for the automated specification of artificial intelligence models,” *International Journal of Forecasting,* vol. 24, no. 4, pp. 694-709, Oct-Dec, 2008.
[104] L. F. Amaral, R. C. Souza, and M. Stevenson, “A smooth transition periodic autoregressive (STPAR) model for short-term load forecasting,” *International Journal of Forecasting,* vol. 24, no. 4, pp. 603-615, Oct-Dec, 2008.
[105] L. Desilets, B. Golden, Q. W. Wang *et al.*, “Predicting Salinity in the Chesapeake Bay Using Backpropagation,” *Computers & Operations Research,* vol. 19, no. 3-4, pp. 277-285, Apr-May, 1992.
[106] J. R. Cancelo, A. Espasa, and R. Grafe, “Forecasting the electricity load from one day to one week ahead for the Spanish system operator,” *International Journal of Forecasting,* vol. 24, no. 4, pp. 588-602, Oct-Dec, 2008.
[107] L. J. Soares, and M. C. Medeiros, “Modeling and forecasting short-term electricity load: A comparison of methods with an application to Brazilian data,” *International Journal of Forecasting,* vol. 24, no. 4, pp. 630-644, Oct-Dec, 2008.
[108] M. T. Leung, A. S. Chen, and H. Daouk, “Forecasting exchange rates using general regression neural networks,” *Computers & Operations Research,* vol. 27, no. 11-12, pp. 1093-1110, Sep-Oct, 2000.
[109] P. X. Casqueiro, and A. J. L. Rodrigues, “Neuro-dynamic trading methods,” *European Journal of Operational Research,* vol. 175, no. 3, pp. 1400-1412, Dec, 2006.
[110] M. R. Lind, and J. M. Sulek, “A methodology for forecasting knowledge work projects,” *Computers & Operations Research,* vol. 27, no. 11-12, pp. 1153-1169, Sep-Oct, 2000.
[111] P. S. A. Freitas, and A. J. L. Rodrigues, “Model combination in neural-based forecasting,” *European Journal of Operational Research,* vol. 173, no. 3, pp. 801-814, Sep, 2006.
[112] J. T. Connor, “A robust neural network filter for electricity demand prediction,” *Journal of Forecasting,* vol. 15, no. 6, pp. 437-458, Nov, 1996.
[113] C. J. Lin, and C. H. Chen, “A compensation-based recurrent fuzzy neural network for dynamic system identification,” *European Journal of Operational Research,* vol. 172, no. 2, pp. 696-715, Jul, 2006.
[114] R. G. Donaldson, and M. Kamstra, “Forecast combining with neural networks,” *Journal of Forecasting,* vol. 15, no. 1, pp. 49-61, Jan, 1996.
[115] C. Haefke, and C. Helmenstein, “Forecasting Austrian IPOs: An application of linear and neural network error-correction models,” *Journal of Forecasting,* vol. 15, no. 3, pp. 237-251, Apr, 1996.
[116] M. Landajo, J. de Andres, and P. Lorca, “Robust neural modeling for the cross-sectional analysis of accounting information,” *European Journal of Operational Research,* vol. 177, no. 2, pp. 1232-1252, Mar, 2007.

[117] A. S. Chen, M. T. Leung, and H. Daouk, “Application of neural networks to an emerging financial market: forecasting and trading the Taiwan Stock Index,” *Computers & Operations Research,* vol. 30, no. 6, pp. 901-923, May, 2003.
[118] A. S. Chen, and M. T. Leung, “Regression neural network for error correction in foreign exchange forecasting and trading,” *Computers & Operations Research,* vol. 31, no. 7, pp. 1049-1068, Jun, 2004.
[119] M. Cottrell, B. Girard, and P. Rousset, “Forecasting of curves using a Kohonen classification,” *Journal of Forecasting,* vol. 17, no. 5-6, pp. 429-439, Sep-Nov, 1998.
[120] R. Marti, and A. El-Fallahi, “Multilayer neural networks: an experimental evaluation of on-line training methods,” *Computers & Operations Research,* vol. 31, no. 9, pp. 1491-1513, Aug, 2004.
[121] P. C. Andreou, C. Charalambous, and S. H. Martzoukos, “Pricing and trading European options by combining artificial neural networks and parametric models with implied parameters,” *European Journal of Operational Research,* vol. 185, no. 3, pp. 1415-1433, Mar, 2008.
[122] X. Li, C. L. Ang, and R. Gray, “An intelligent business forecaster for strategic business planning,” *Journal of Forecasting,* vol. 18, no. 3, pp. 181-204, May, 1999.
[123] R. Carbonneau, K. Laframboise, and R. Vahidov, “Application of machine learning techniques for supply chain demand forecasting,” *European Journal of Operational Research,* vol. 184, no. 3, pp. 1140-1154, Feb, 2008.
[124] N. Gupta, and M. P. Singh, “Estimation of software reliability with execution time model using the pattern mapping technique of artificial neural network,” *Computers & Operations Research,* vol. 32, no. 1, pp. 187-199, Jan, 2005.
[125] W. L. Gorr, D. Nagin, and J. Szczypula, “Comparative-Study of Artificial Neural-Network and Statistical-Models for Predicting Student Grade-Point Averages,” *International Journal of Forecasting,* vol. 10, no. 1, pp. 17-34, Jun, 1994.
[126] A. R. Venkatachalam, and J. E. Sohl, “An intelligent model selection and forecasting system,” *Journal of Forecasting,* vol. 18, no. 3, pp. 167-180, May, 1999.
[127] L. Yu, S. Wang, and K. K. Lai, “Neural network-based mean-variance-skewness model for portfolio selection,” *Computers & Operations Research,* vol. 35, no. 1, pp. 34-46, Jan, 2008.
[128] T. Hill, L. Marquez, M. Oconnor *et al.*, “Artificial Neural-Network Models for Forecasting and Decision-Making,” *International Journal of Forecasting,* vol. 10, no. 1, pp. 5-15, Jun, 1994.
[129] K. Lam, and K. C. Lam, “Forecasting for the generation of trading signals in financial markets,” *Journal of Forecasting,* vol. 19, no. 1, pp. 39-52, Jan, 2000.
[130] S. Moshiri, and N. Cameron, “Neural network versus econometric models in forecasting inflation,” *Journal of Forecasting,* vol. 19, no. 3, pp. 201-217, Apr, 2000.
[131] B. A. Jain, and B. N. Nag, “Artificial neural network models for pricing initial public offerings,” *Decision Sciences,* vol. 26, no. 3, pp. 283-302, May-Jun, 1995.
[132] C. Schittenkopf, G. Dorffner, and E. J. Dockner, “Forecasting time-dependent conditional densities: A semi-non-parametric neural network approach,” *Journal of Forecasting,* vol. 19, no. 4, pp. 355-374, Jul, 2000.
[133] J. W. Taylor, “A quantile regression neural network approach to estimating the conditional density of multiperiod returns,” *Journal of Forecasting,* vol. 19, no. 4, pp. 299-311, Jul, 2000.

[134] V. S. Desai, and R. Bharati, “The efficacy of neural networks in predicting returns on stock and bond indices,” *Decision Sciences,* vol. 29, no. 2, pp. 405-425, Spr, 1998.
[135] S. H. Kim, and S. H. Chun, “Graded forecasting using an array of bipolar predictions: application of probabilistic neural networks to a stock market index,” *International Journal of Forecasting,* vol. 14, no. 3, pp. 323-337, Sep, 1998.
[136] C. L. Dunis, and X. H. Huang, “Forecasting and trading currency volatility: An application of recurrent neural regression and model combination,” *Journal of Forecasting,* vol. 21, no. 5, pp. 317-354, Aug, 2002.
[137] L. C. Thomas, “A survey of credit and behavioural scoring: forecasting financial risk of lending to consumers,” *International Journal of Forecasting,* vol. 16, no. 2, pp. 149-172, Apr-Jun, 2000.
[138] A. Kanas, “Non-linear forecasts of stock returns,” *Journal of Forecasting,* vol. 22, no. 4, pp. 299-315, Jul, 2003.
[139] R. Gencay, and F. Selcuk, “Neural network toolbox 3.0 for use with MATLAB (TM),” *International Journal of Forecasting,* vol. 17, no. 2, pp. 305-317, Apr-Jun, 2001.
[140] H. Hruschka, “Determining Market Response Functions by Neural Network Modeling - a Comparison to Econometric Techniques,” *European Journal of Operational Research,* vol. 66, no. 1, pp. 27-35, Apr, 1993.
[141] M. Qi, “Predicting US recessions with leading indicators via neural network models,” *International Journal of Forecasting,* vol. 17, no. 3, pp. 383-401, Jul-Sep, 2001.
[142] A. Lindemann, C. L. Dunis, and P. Lisboa, “Probability distributions, trading strategies and leverage: An application of Gaussian mixture models,” *Journal of Forecasting,* vol. 23, no. 8, pp. 559-585, Dec, 2004.
[143] G. Tkacz, “Neural network forecasting of Canadian GDP growth,” *International Journal of Forecasting,* vol. 17, no. 1, pp. 57-69, Jan-Mar, 2001.
[144] S. H. Wang, “Nonparametric econometric modelling: A neural network approach,” *European Journal of Operational Research,* vol. 89, no. 3, pp. 581-592, Mar, 1996.
[145] A. S. Chen, and M. T. Leung, “Performance evaluation of neural network architectures: The case of predicting foreign exchange correlations,” *Journal of Forecasting,* vol. 24, no. 6, pp. 403-420, Sep, 2005.
[146] H. G. Wittkemper, and M. Steiner, “Using neural networks to forecast the systematic risk of stocks,” *European Journal of Operational Research,* vol. 90, no. 3, pp. 577-588, May, 1996.
[147] A. J. Conejo, J. Contreras, R. Espinola *et al.*, “Forecasting electricity prices for a day-ahead pool-based electric energy market,” *International Journal of Forecasting,* vol. 21, no. 3, pp. 435-462, Jul-Sep, 2005.
[148] A. B. Badiru, and D. B. Sieger, “Neural network as a simulation metamodel in economic analysis of risky projects,” *European Journal of Operational Research,* vol. 105, no. 1, pp. 130-142, Feb, 1998.
[149] N. Gradojevic, and J. Yang, “Non-linear, non-parametric, non-fundamental exchange rate forecasting,” *Journal of Forecasting,* vol. 25, no. 4, pp. 227-245, Jul, 2006.
[150] A. Novales, “Comments on: "Linear models, smooth transition autoregressions, and neural networks for forecasting macroeconomic time series: A re-examination",” *International Journal of Forecasting,* vol. 21, no. 4, pp. 775-780, Oct-Dec, 2005.
[151] S. D. Bekiros, and D. A. Georgoutsos, “Direction-of-change forecasting using a volatility-based recurrent neural network,” *Journal of Forecasting,* vol. 27, no. 5, pp. 407-417, Aug, 2008.

[152] T. Terasvirta, D. van Dijk, and M. C. Medeiros, “Comments on: "Linear models, smooth transition autoregressions, and neural networks for forecasting macroeconomic time series: A re-examination" - Reply,” *International Journal of Forecasting,* vol. 21, no. 4, pp. 781-783, Oct-Dec, 2005.
[153] A. Preminger, and R. Franck, “Forecasting exchange rates: A robust regression approach,” *International Journal of Forecasting,* vol. 23, no. 1, pp. 71-84, 2007.