An Evaluation of Neural Network Ensembles and Model Selection for Time Series Prediction

Devon K. Barrow, Sven F. Crone, and Nikolaos Kourentzes

Abstract— **Ensemble methods represent an approach to combine a set of models, each capable of solving a given task, but which together produce a composite global model whose accuracy and robustness exceeds that of the individual models. Ensembles of neural networks have traditionally been applied to machine learning and pattern recognition but more recently have been applied to forecasting of time series data. Several methods have been developed to produce neural network ensembles ranging from taking a simple average of individual model outputs to more complex methods such as bagging and boosting. Which ensemble method is best; what factors affect ensemble performance, under what data conditions are ensembles most useful and when is it beneficial to use ensembles over model selection are a few questions which remain unanswered. In this paper we present some initial findings using neural network ensembles based on the mean and median applied to forecast synthetic time series data. We vary factors such as the number of models included in the ensemble and how the models are selected, whether randomly or based on performance. We compare the performance of different ensembles to model selection and present the results.**

I. INTRODUCTION

In the computational intelligence world, significant progress has been made in the areas of time series prediction. Ensembles methods which use multiple models to obtain better predictive performance have received increased attention. Neural network ensembles have emerged as a popular method for time series forecasting. They work over a wide range of domains and applications with increased accuracy and robustness. A parallel approach occurs in the domain of operations research forecasting where extensive research has been conducted in the area of forecast combinations to increase the accuracy of time series prediction. Bates and Granger [1] was one of the first to show significant gains in accuracy through combination. Another early work by Newbold and Granger [2], combined various univariate time series forecasts and compared the combination against the performance of the standalone version of the individual models. They show that for set of forecasts *F*, a linear combination of these forecasts could be obtained which would also be unbiased and achieve a combined forecast error variance smaller than the individual forecasts. They found that the better combining procedures did produce an overall forecast superior to individual forecasts on the majority of tested time series. Neural network ensembles have been used significantly to improve the accuracy over single network models in time series forecasting [3].

How to combine models and under which conditions still remains a relatively open question. Makridakis and Winkler [4] suggest that using averages of forecasts provides improved forecasting accuracy and that the variability of accuracy among different combinations decreases, as the number of methods in the average increases. Makridakis et al. [5] show that taking a simple average outperforms taking a weighted average model combination. Elliott and Timmermann [6] however dispel the notion that equally-weighted combined forecasts lead to better performance than estimates of optimal forecast combination weights stating that this is directly linked to the use of the mean squared error loss as the loss function. More recently [7] use the weighted median because it is less sensitive to outliers than the weighted mean and gives better results under boosting. Evidence supporting the use of the arithmetic mean is extensive while taking the median over a number of time series forecasting models has emerged as a contender, with mixed results. No one has however evaluated the performance of the arithmetic mean and median combined ensembles for time series forecasting within an empirically sound and established methodology.

In this paper we use a multi-factorial approach to investigate two popular and widely used methods, the arithmetic mean and the median and evaluate their ensemble generating performance on time series with varying levels of noise and seasonality. These methods are applied to produce ensembles of neural networks which are used to forecast 45 artificial time series. These time series are designed to reflect four different levels of noise; no noise, low noise, medium noise and high noise data. This study seeks to compute benchmark results using a naïve methodology based on the performance of conventional model selection.

The paper is organised as follows. In section II we provide a brief introduction to neural networks and neural network ensembles. This is followed by a short discussion of ensemble methods, in particular the mean and median combination methods. In Section III an outline of the experimental setup is provided, followed by a presentation of the results and an analysis of the findings.

Devon K. Barrow is with the Department of Management Science, Lancaster University Management School, Lancaster LA1 4YX, United Kingdom (+44.1524.5-92991, e-mail: d.barrow@lancaster.ac.uk).

Sven F. Crone is with the Department of Management Science, Lancaster University Management School, Lancaster LA1 4YX, United Kingdom (+44.0844 307 6508, e-mail: s.crone@lancaster.ac.uk).

Nikolaos Kourentzes is with the Department of Management Science, Lancaster University Management School, Lancaster LA1 4YX, United Kingdom (+44.1524.5-92991, e-mail: n.kourentzes@lancaster.ac.uk).

978-1-4244-8126-2/10/$26.00 ©2010 IEEE

II. METHODS

A. Neural Networks for Forecasting

Forecasting time series with NNs is conventionally based on modelling a feed-forward topology in analogy to an non-linear autoregressive AR(p) model using the established Multilayer Perceptron (MLP), to which we will limit our analysis here. The functional form of these networks is

$$f(Y,w) = \beta_0 + \sum_{h=1}^{H} \beta_h g\left(\gamma_{0i} + \sum_{i=0}^{I} \gamma_{hi} y_i \right), \quad (1)$$

that describes a single layered MLP characterised by its input vector $Y = [y_t, y_{t-1}, \ldots, y_{t-I+1}]$, which captures the lagged observations of the time series in input nodes I, its number of hidden nodes H and a single output node. A non-linear transfer functions $g(\cdot)$ is used in the nodes of the hidden layer, conventionally this is the sigmoid logistic or the hyperbolic tangent functions [8]. The network parameters are denoted as weights $w = (\beta, \gamma)$ connecting input, hidden and output layer respectively and the biases β_0 and γ_{0i} of each neuron.

For parameterisation, data is presented to the MLP as an overlapping set of input vectors formed as a sliding window over the time series observations. Consequently, the specification of the network architecture determines the time series components that may be captured in the AR(p)-lags of the input vector and the capability of approximation. MLPs offer extensive degrees of freedom in modeling for prediction tasks. The modeler must decide upon the selection and sampling of datasets, the degrees of data pre-processing, the static architectural properties, and the learning algorithm that characterized through the objective function or error function. For a detailed discussion of these issues and the ability of NNs to forecast univariate time series, the reader is referred to [9].

B. Ensemble Methods

Both within the world of computational intelligence, where model combinations are referred to as ensembles [10] and the domain of forecasting where it is explicitly referred to as forecast combinations a number of methods have been developed for combining models to improve forecasting accuracy. Such methods predominantly have in common the generation of a set of weights which are assigned to each model. Methods can be classed under two rather broad headings; methods that assign weights based on a direct reduction in error variance, so called error-variance-based methods, linear or non-linear, and those that produce weights based on a conditional probability of model-data fit using Bayesian theory to assign probabilities of model fitness which are then converted to weights representing confidence in the given model. It is interesting to note however that the literature continues to support the use of simple model combinations such as the simple average and the median which have shown to outperform its more complex counterparts. As indicated in the introduction results of these have been somewhat mixed.

Makridakis and Winker [4] discussed the impact of the number of forecasts included in a simple average concluding that the forecasting accuracy improves, and that the variability of accuracy among different combinations decreases, as the number of methods in the average increases. Results of the M-competition involving seven experts, 24 methods and 1001 times series [5], showed that combining a simple average of six methods perform well overall and even better than the individual methods included in making the average. Furthermore, taking a weighted average based on the sample covariance matrix of fitting errors over the same methods, instead of the simple average also performs well, but not as well as the simple average. Palm and Zellner [11] go on to propose that the simple average is more robust with respect to specification uncertainty, time variation of parameters, and estimation errors than weighted averages and that perhaps it is more appropriate when there is insufficient information on past performances of individual forecasters. The sensitivity of weighted averages is considered by [12], who show how relatively small estimation errors can lead to negative weights or weights much greater than one, and to weighted averages that are considerably outside the range of the individual forecasts being combined. Jose and Winkler [13] propose to extend the robustness of simple averages by performing trimming and winsorizing of forecasts to avoid errors associated with extreme values. Makridakis et al. [14] mentioned that combining the exponential smoothing methods does not beat the best of the individual smoothing methods but later [15] found that the simple arithmetic average of three methods: Single, Holt and Dampen Trend Exponential Smoothing is more accurate than the three individual methods being combined for practically all forecasting horizons, although its difference from Dampen is small.

In the context of computational intelligence methods recent studies by [16] who apply boosted recurrent neural networks to predicting chaotic time series, found that the weighted median performed better than the weighted mean as it was more robust against noise. Avnimelech and Intrator [17] reveal several interesting results in the context of time series prediction based on neural network ensembles using boosting and bagging. They find that the median combination appears best though in Adaboost it shows no significant difference over mean. More interestingly tests found the non-weighted averages may be at least as good as the weighted averages and that the mean may be at least as good as the median. Deng et al. [18] observed that for boosting, weighted median is a better choice for combining the regressors than the weighted mean.

Out of the literature the two approaches found to be most widely used and accepted are the simple average and the median but mixed results have been obtained. Agnew [19] find that the median performed better than the mean using 16 macroeconomic forecasters while more recently Stock and Watson [20] found that the mean performed better when

forecasting output growth in a seven-country quarterly economic data set covering 1959 – 1999, with up to 73 predictors per country using AR model and a recursive random walk. Several of these claims are investigated in this paper, first to consider whether there is a significant difference in performance between the mean and median ensemble on noisy data and also on seasonal versus stationary time series. To consider to what degree model selection based on random selection versus a selection of the top performing models influences overall ensemble performance and finally to assess the impact of ensemble size on performance.

III. Experimental Design

A. Time series data

Forty five (45) synthetic time series are used in this experiment. These are equally split in three groups of 15 time series, simulating three different levels of Gaussian noise; low, medium and high noise, with sigma of 1, 5 and 10 respectively. The time series are generated following

$$y_t = \mu + \sum_{i}^{S} b_i d_{it} + z_t . \quad (2)$$

For the deterministic case, μ is the level of the time series, which was set to 100, and b_i the coefficients of seasonal dummies d_i, while S is the periodicity of seasonality, which was set to 12, simulating monthly data. The coefficients b_i were generated using a uniform distribution U(0,100) and consequently centered around zero. The noise z_t follows $N(0,\sigma^2)$. All time series have 480 observations, which are split into training, validation and test sets of 288, 96 and 96 observations each. Three time series, one for each noise level, are illustrated in figure 1.

B. Experimental setup

The forecasting horizon is set to 12 observations, or equally one complete year. To assess the forecasting performance of the NNs the symmetric mean absolute percent error (sMAPE) is used. This has the advantage of being scale independent, allowing to aggregate results over time series and is less biased than the commonly used mean absolute percentage error (MAPE). It computes the absolute error in percent between the actuals X_t and the forecast F_t for all periods t of the test set of size $n=h$ for each time origin:

$$SMAPE = \frac{1}{n}\sum_{t=1}^{n}\left(\frac{|X_t - F_t|}{(|X_t| + |F_t|)/2}\right)100 . \quad (3)$$

All models are evaluated using a rolling time origin evaluation, producing multiple forecasts for each time series, resulting in more accurate estimation of the forecasting error. Furthermore, the error estimation is robust against irregular origins [21].

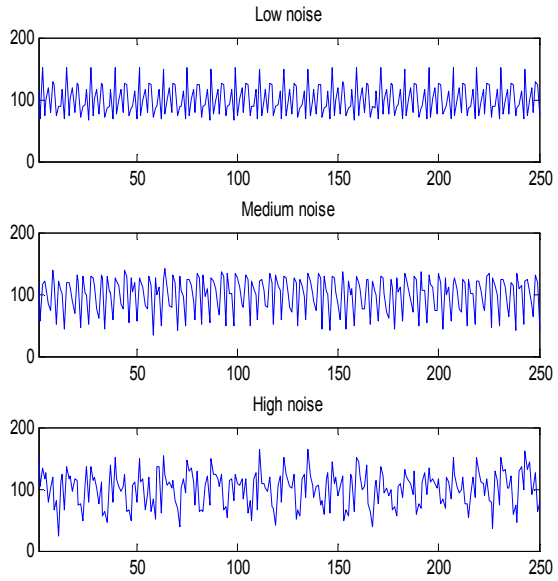

Fig. 1.First 250 observations of a low, medium and high noise time series

C. Methods

We construct nine different MLP models to forecast the times series. The rationale behind using multiple setups is to assess the performance of ensembles when models of varying fit to the data are available. The models differ in the specification of the input vector and the hidden layer. All other parameters remain constant. Given a time series Y_t three different input nodes are defined:

1. Use all Y_{t-1} up to Y_{t-12}. Essentially, a full season is used as inputs to forecast the next season (next 12 observations).
2. Use only Y_{t-1} and Y_{t-12}. This is a sparse specification of the previous setup. This setup can lead to underspecified models, given the deterministic nature of seasonality in the synthetic data.
3. Use Y_{t-1} up to Y_{t-6}. This is a shorter specification that aims to capture the same information more parsimoniously. This input vector is misspecified, as the seasonal lag Y_{t-12} is not included.

All these input vectors follow naive heuristics that have appeared in the literature [22]. Although these might be inadequate to be used in automatic forecasting of heterogeneous datasets and complex time series, and several alternatives have been proposed [23-24], they should be adequate to forecast the time series used in this experiment to a varying degree of accuracy. Note, that the first input vector can fully capture the data generating process of the time series, in contrast to the latter two. This is done in order to ensure that ensembles of models of varying performance can be constructed.

All MLPs use a single hidden layer with a varying number of hidden nodes. Three alternatives are considered, 2, 4 and 8 nodes. In all setups the nodes use the logistic activation function. A single output node with the identity activation function is used for all networks. The MLPs are all trained

using the simple back-propagation algorithm with momentum for 1,000 epochs and employing an early stopping criterion. The early stopping criterion evaluates the mean squared error (MSE) every epoch, stops training when no improvement is made for hundred epochs. The initial learning rate is set to η=0.4, applying a cooling factor Δη to reduce the learning rate by 0.05 per epoch; the momentum term is kept constant at φ=0.5. All data is pre-processed using linear scaling into the interval of [-0.5, 0.5] using the scaling function of on the minimum

$$z_t = -0{,}5 + \frac{\left(x_t - x_{t\,\min}\right)}{\left(x_{t\,\max} - x_{t\,\min}\right)}, \qquad (4)$$

$x_{t\ min}$ and maximum value $x_{t\ max}$ of x_t on the training and validation set. The data is then presented to the MLP using random sampling without replacement. Each MLP candidate is initialised 30 times with random starting weights in the interval of [-0.8, 0.8] in order to avoid local minima during the training and to provide an adequate error distribution using sufficient results.

D. Model ensembles

The aim of this paper is to evaluate the use of ensembles of neural networks to address the model selection problem. Therefore, it is important to determine which ensembles perform best and under which conditions. A large number of different candidate models are produced based on the 30 initialisations used in training the model. Following the standard implementations in literature, we perform model selection by choosing the best MLP based on minimum error in the validation set. This becomes our benchmark model selection procedure, which we use to assess the performance of the different ensembles of NNs. To build the ensembles we considered different combination methods, different ensemble sizes and different ensemble member selection criteria. The combination methods considered are those of the mean and median. Furthermore, model selection is done based on a selection of the top models (ranked from best to worst on the validation set) versus a random selection of models. Ensembles are then generated at different ensemble sizes ranging from 10% to 100% by increments of 10% over all 30 models. Forty ensembles are produced altogether based on method of combination, ensemble model selection criteria and number of models included in the ensemble i.e. ensemble size.

We perform two different sets of experiments. In the first set we use only MLPs that use as inputs all Y_{t-1} up to Y_{t-12} and 8 hidden nodes, which were found to be the most accurate for this dataset. The hypothesis that we want to test is whether ensembles can increase forecasting accuracy in comparison to conventional model selection, when there are only multiple initialisations of a single model that can fit well on the data generating process of the time series. The second set of experiments uses all nine different MLP setups, resulting in ensembles of heterogeneous models, which follows the suggestions of the literature as discussed in section II. Again, we contrast the use of ensembles versus model selection, in terms of forecasting accuracy and determine the impact of the various factors on ensemble performance.

IV. Experimental Results

A. Single model experiments

Firstly, we discuss the experiments that involve a single MLP setup. The purpose of this experiment is to assess the utility of ensembles for model selection when there is only a single NN producing the forecasts. This follows the first input vector option as presented in section III.C and 8 nodes in the hidden layer. This network was chosen being the most accurate overall. The results are presented in table I and summarized in figure 2. It is observed that ensemble performance based on a single NN remains stable across different ensemble sizes. For example, under low noise and top and random selection, mean and median performance error remain within 0.01% point accuracy across ensemble size. Under high noise and top and random selection, mean and median ensemble are within 0.04% range with the only exception being the best ensemble which was 0.08% from the worst performing ensemble. Mean and median ensembles are similarly robust across ensemble sizes. Moreover, mean and median ensembles perform similarly within ensemble model selection. At all noise levels and within top ensemble model selection, mean and median ensemble performance are within 0.01% on average and at worst. However this robust performance appears to degrade under random selection where the maximum differences are noted. For example, under high noise and random selection, and at 40% ensemble size, mean ensemble performance is 10.65% while median ensemble performance is 10.75%. Top selection of ensemble models therefore appears slightly more robust against ensemble method than random selection.

Across all levels of noise and all levels of ensemble sizes, whether considered separately or together, top selection outperforms random selection. Best ensemble based on top selection at low noise produces mean error of 1.11% while the best ensemble based on random selection produces 1.12%. At the medium noise level the mean performance errors are 5.06% to 5.15% respectively and at the high noise level it is 5.59% to 5.64%. While performance varies minimally across mean and median ensembles, there is a relationship between ensemble model selection and noise level whichwhich appears to impact forecast accuracy.

With regards to the performance of ensembles relative to model selection, the best ensemble based on top selection, both mean and median combined, outperform model selection at all levels of noise though not significantly. Where random selection of models is made, model selection outperforms the best ensemble created using either the mean or the median. The gains over model selection are minimal and model selection would be preferred where computation of ensembles is an issue. If ensembles are used however,

TABLE I
sMAPE ACCURACY OF SINGLE MODEL ENSEMBLES

Mean Errors

			10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	MS	Aver. ENS
Low noise	Top	Mean	1.11%	1.11%	1.11%	1.11%	1.12%	1.12%	1.12%	1.12%	1.12%	1.12%	1.11%	1.12%
		Median	1.11%	1.11%	1.12%	1.12%	1.12%	1.12%	1.12%	1.12%	1.12%	1.12%	1.11%	1.12%
	Random	Mean	1.12%	1.13%	1.12%	1.13%	1.13%	1.12%	1.12%	1.12%	1.12%	1.12%	**1.11%**	1.13%
		Median	1.13%	1.13%	1.12%	1.12%	1.13%	1.13%	1.12%	1.13%	1.12%	1.12%	**1.11%**	1.13%
Medium noise	Top	Mean	5.06%	5.07%	5.09%	5.11%	5.12%	5.14%	5.14%	5.15%	5.16%	5.17%	5.09%	5.12%
		Median	5.08%	5.07%	5.08%	5.11%	5.12%	5.13%	5.14%	5.15%	5.16%	5.17%	5.09%	5.18%
	Random	Mean	5.11%	5.17%	5.16%	5.17%	5.16%	5.17%	5.17%	5.18%	5.16%	5.17%	**5.09%**	5.16%
		Median	5.24%	5.18%	5.15%	5.19%	5.19%	5.18%	5.17%	5.18%	5.18%	5.17%	**5.09%**	5.18%
High noise	Top	Mean	10.64%	10.60%	10.64%	10.64%	10.64%	10.65%	10.66%	10.67%	10.68%	10.70%	10.61%	10.65%
		Median	10.65%	10.60%	10.64%	10.64%	10.65%	10.66%	10.67%	10.67%	10.68%	10.70%	10.61%	10.66%
	Random	Mean	10.68%	10.75%	10.73%	10.65%	10.71%	10.71%	10.69%	10.70%	10.68%	10.70%	**10.61%**	10.70%
		Median	10.65%	10.72%	10.69%	10.75%	10.69%	10.74%	10.71%	10.71%	10.71%	10.70%	**10.61%**	10.71%
All	Top	Mean	5.60%	5.59%	5.61%	5.62%	5.63%	5.64%	5.64%	5.65%	5.65%	5.66%	5.60%	10.65%
		Median	5.61%	5.59%	5.61%	5.62%	5.63%	5.64%	5.64%	5.65%	5.65%	5.67%	5.60%	5.63%
	Random	Mean	5.64%	5.68%	5.67%	5.65%	5.66%	5.67%	5.66%	5.67%	5.66%	5.66%	**5.60%**	5.66%
		Median	5.68%	5.68%	5.65%	5.69%	5.67%	5.68%	5.67%	5.67%	5.67%	5.67%	**5.60%**	5.67%

Median Errors

			10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	MS	Aver. ENS
Low noise	Top	Mean	1.15%	1.16%	1.15%	1.16%	1.16%	1.16%	1.16%	1.16%	1.16%	1.16%	1.15%	1.16%
		Median	1.16%	1.16%	1.16%	1.16%	1.16%	1.16%	1.16%	1.16%	1.16%	1.16%	**1.15%**	1.16%
	Random	Mean	1.16%	1.16%	1.15%	1.16%	1.16%	1.15%	1.16%	1.16%	1.16%	1.16%	1.15%	1.16%
		Median	1.20%	1.14%	1.18%	1.17%	1.16%	1.16%	1.16%	1.16%	1.16%	1.16%	1.15%	1.17%
Medium noise	Top	Mean	5.13%	5.11%	5.10%	5.09%	5.10%	5.13%	5.14%	5.15%	5.17%	5.19%	5.19%	5.13%
		Median	5.11%	5.10%	5.11%	5.10%	5.10%	5.14%	5.16%	5.16%	5.18%	5.20%	5.19%	5.14%
	Random	Mean	5.04%	5.25%	5.11%	5.16%	5.19%	5.21%	5.18%	5.19%	5.18%	5.19%	5.19%	5.17%
		Median	5.18%	5.18%	5.13%	5.25%	5.18%	5.19%	5.19%	5.20%	5.20%	5.20%	5.19%	5.19%
High noise	Top	Mean	10.83%	10.80%	10.82%	10.80%	10.82%	10.80%	10.81%	10.82%	10.81%	10.79%	**10.74%**	10.81%
		Median	10.85%	10.82%	10.83%	10.80%	10.83%	10.82%	10.83%	10.84%	10.82%	10.80%	**10.74%**	10.82%
	Random	Mean	10.76%	10.82%	10.76%	10.75%	10.84%	10.81%	10.78%	10.78%	10.78%	10.79%	**10.74%**	10.79%
		Median	10.74%	10.83%	10.80%	10.82%	10.86%	10.77%	10.84%	10.81%	10.80%	10.80%	10.74%	10.81%
All	Top	Mean	5.70%	5.69%	5.69%	5.68%	5.69%	5.70%	5.70%	5.71%	5.71%	5.71%	5.69%	5.70%
		Median	5.71%	5.69%	5.70%	5.69%	5.70%	5.71%	5.72%	5.72%	5.72%	5.72%	5.69%	5.71%
	Random	Mean	5.65%	5.74%	5.67%	5.69%	5.73%	5.72%	5.71%	5.71%	5.71%	5.71%	5.69%	5.71%
		Median	5.71%	5.72%	5.70%	5.75%	5.73%	5.71%	5.73%	5.72%	5.72%	5.72%	**5.69%**	5.72%

Table 1: Ensemble performance (test set error – sMAPE) across different noise levels for the single model experiments. Each column presents results of ensembles of different sizes (10% - 100% of ensemble members). MS presents the results of model selection and Aver.ENS contains the average ensemble performance across different sizes. Underlined values represent the best model for each row. Boldface model selection results represent cases that ensembles are outperformed.

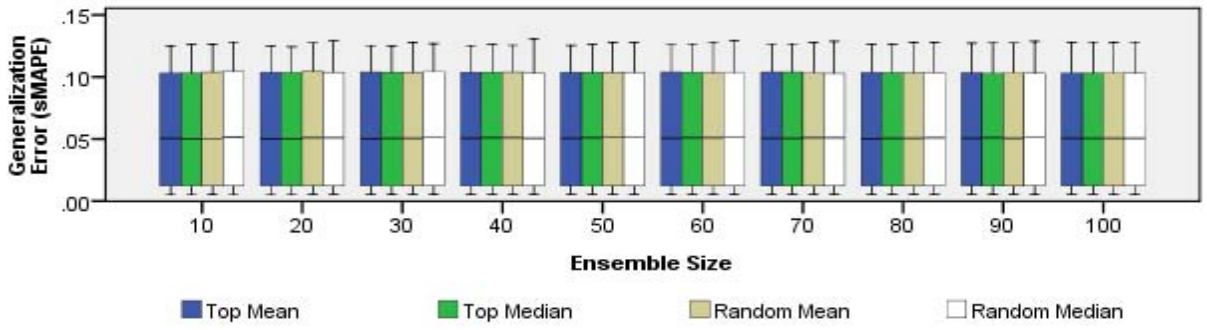

Fig. 2. Boxplots of ensemble performance (test set error – sMAPE) across different noise levels for the single model experiments.

performance of the models across noise level should be considered.

B. Multiple models experiments

Following the findings of the single model experiments, we evaluate the benefits of ensembles when several NN models are considered. Now, models have a varying degree of fit to the time series and produce heterogeneous forecasts. First it is noticed that across almost all noise levels and ensemble sizes, median ensembles outperform mean ensembles. At the low noise level and random selection we see than the best median ensemble performance (see data underlined in table II) has an error of 1.13% (mean measure) across ensemble sizes of 20%, 30,%, 60%,70% and 90% compared to the mean performance error of 1.66% only at the 30% ensemble size. Best mean and median ensemble

TABLE II
sMAPE ACCURACY OF MULTIPLE MODELS ENSEMBLES

Mean Errors

			10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	MS	Aver. ENS
Low noise	Top	Mean	1.11%	1.12%	1.12%	1.13%	1.14%	1.14%	1.14%	1.22%	1.36%	1.80%	**1.10%**	1.23%
		Median	1.11%	1.12%	1.12%	1.13%	1.14%	1.15%	1.15%	1.14%	1.13%	1.13%	**1.10%**	1.13%
	Random	Mean	1.97%	1.84%	1.66%	1.89%	1.95%	1.77%	1.78%	1.81%	1.79%	1.80%	**1.10%**	1.83%
		Median	1.14%	1.13%	1.13%	1.13%	1.14%	1.13%	1.13%	1.13%	1.13%	1.13%	**1.10%**	1.13%
Medium noise	Top	Mean	5.12%	5.16%	5.19%	5.23%	5.28%	5.33%	5.31%	5.30%	5.42%	5.73%	5.12%	5.31%
		Median	5.12%	5.16%	5.19%	5.23%	5.29%	5.38%	5.43%	5.36%	5.31%	5.29%	5.12%	5.28%
	Random	Mean	6.13%	5.87%	5.52%	5.65%	5.70%	5.66%	5.67%	5.71%	5.72%	5.73%	**5.12%**	5.74%
		Median	5.25%	5.29%	5.24%	5.33%	5.33%	5.32%	5.29%	5.26%	5.29%	5.29%	**5.12%**	5.29%
High noise	Top	Mean	10.62%	10.66%	10.71%	10.81%	10.94%	11.05%	11.05%	11.08%	11.34%	11.79%	10.71%	11.01%
		Median	10.63%	10.66%	10.73%	10.79%	10.91%	11.10%	11.23%	11.08%	11.02%	11.01%	10.71%	10.91%
	Random	Mean	11.72%	11.58%	11.63%	11.86%	11.85%	11.86%	11.86%	11.92%	11.81%	11.79%	**10.71%**	11.79%
		Median	11.05%	10.99%	11.17%	11.03%	11.07%	11.01%	11.00%	11.02%	11.02%	11.01%	**10.71%**	11.03%
All	Top	Mean	5.62%	5.64%	5.68%	5.73%	5.79%	5.84%	5.83%	5.86%	6.04%	6.44%	5.65%	5.85%
		Median	5.62%	5.65%	5.68%	5.72%	5.78%	5.88%	5.93%	5.86%	5.82%	5.81%	5.65%	5.77%
	Random	Mean	6.61%	6.43%	6.27%	6.47%	6.50%	6.43%	6.44%	6.48%	6.44%	6.44%	**5.65%**	6.45%
		Median	5.81%	5.80%	5.84%	5.83%	5.85%	5.82%	5.80%	5.80%	5.81%	5.81%	**5.65%**	5.82%

Median Errors

			10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	MS	Aver. ENS
Low noise	Top	Mean	1.15%	1.15%	1.16%	1.17%	1.17%	1.18%	1.17%	1.21%	1.23%	1.64%	**1.14%**	1.22%
		Median	1.15%	1.16%	1.16%	1.17%	1.18%	1.19%	1.19%	1.18%	1.17%	1.17%	**1.14%**	1.17%
	Random	Mean	1.82%	1.58%	1.55%	1.73%	1.82%	1.62%	1.58%	1.59%	1.65%	1.64%	**1.14%**	1.66%
		Median	1.17%	1.17%	1.16%	1.17%	1.17%	1.18%	1.17%	1.16%	1.17%	1.17%	**1.14%**	1.17%
Medium noise	Top	Mean	5.11%	5.17%	5.23%	5.28%	5.32%	5.37%	5.36%	5.32%	5.34%	5.62%	5.30%	5.31%
		Median	5.11%	5.16%	5.21%	5.26%	5.33%	5.43%	5.47%	5.36%	5.28%	5.28%	5.30%	5.29%
	Random	Mean	5.82%	5.73%	5.51%	5.55%	5.59%	5.60%	5.59%	5.62%	5.61%	5.62%	**5.30%**	5.62%
		Median	5.25%	5.29%	5.22%	5.33%	5.33%	5.32%	5.28%	5.26%	5.28%	5.28%	5.30%	5.28%
High noise	Top	Mean	10.73%	10.73%	10.77%	10.86%	10.95%	11.08%	11.08%	11.11%	11.27%	11.70%	10.73%	11.03%
		Median	10.76%	10.75%	10.76%	10.82%	10.95%	11.14%	11.25%	11.14%	11.09%	11.07%	**10.73%**	10.97%
	Random	Mean	11.69%	11.43%	11.71%	11.76%	11.78%	11.81%	11.78%	11.86%	11.73%	11.70%	**10.73%**	11.73%
		Median	11.08%	11.07%	11.23%	11.01%	11.08%	10.99%	11.04%	11.08%	11.05%	11.07%	**10.73%**	11.07%
All	Top	Mean	5.66%	5.68%	5.72%	5.77%	5.81%	5.88%	5.87%	5.88%	5.95%	6.32%	5.72%	5.85%
		Median	5.67%	5.69%	5.71%	5.75%	5.82%	5.92%	5.97%	5.89%	5.85%	5.84%	5.72%	5.81%
	Random	Mean	6.44%	6.25%	6.26%	6.35%	6.40%	6.34%	6.32%	6.36%	6.33%	6.32%	**5.72%**	6.34%
		Median	5.83%	5.84%	5.87%	5.84%	5.86%	5.83%	5.83%	5.83%	5.83%	5.84%	**5.72%**	5.84%

Table 2: Ensemble performance (test set error – sMAPE) across different noise levels for the multiple model experiments. Each column presents results of ensembles of different sizes (10% - 100% of ensemble members). MS presents the results of model selection and Aver.ENS contains the average ensemble performance across different sizes. Underlined values represent the best model for each row. Boldface model selection results represent cases that ensembles are outperformed.

performance is the same under top selection. At medium noise and top selection performance of best mean and median ensemble is again the same while at random selection best mean ensemble error is 5.52% while the median is 5.24%. Again, at high noise level performance of the best mean and median top ensemble is the same while the median outperforms the mean on random selection 10.99% to 11.58%.

When ensemble models are selected based on top performance, mean and median ensembles perform similar across all noise level as shown above and illustrated in table I. However the median ensemble outperforms mean ensembles when models are selected randomly to be included in the ensemble. This adds something very valuable to the literature indicating that where model selection is done at random or as is the case in practice model performance is not know priori, it is best advised to use median ensembles which appear to be more robust against the performance of such models. If the performance of models are known in advance and known to be quite good, then using mean or median ensemble makes no difference. Ultimately, the choice however will depend on the ensemble size relative to the model population as we explain in the following paragraph.

It is observed that median ensemble is also robust against ensemble sizes. One case was already referenced above where median outperforms mean across several ensemble sizes at low noise level and random selection. Even at top selection where performance of the best mean and median ensemble is the same, the performance error of the mean ensemble ranges from 1.11% at 10% ensemble size to 1.80% obtained at 100% ensemble size. This can be compared to the median ensemble where the range is 1.11% at 10% ensemble size with worst performance at 1.15% at 70% ensemble size. While median performance appears more robust against ensemble size this does not imply that the median ensemble performs better on accuracy than the mean. At random selection this is clearly the case as eluded to previously, however at top selection it is noted that mean ensemble performance is as robust as median ensemble

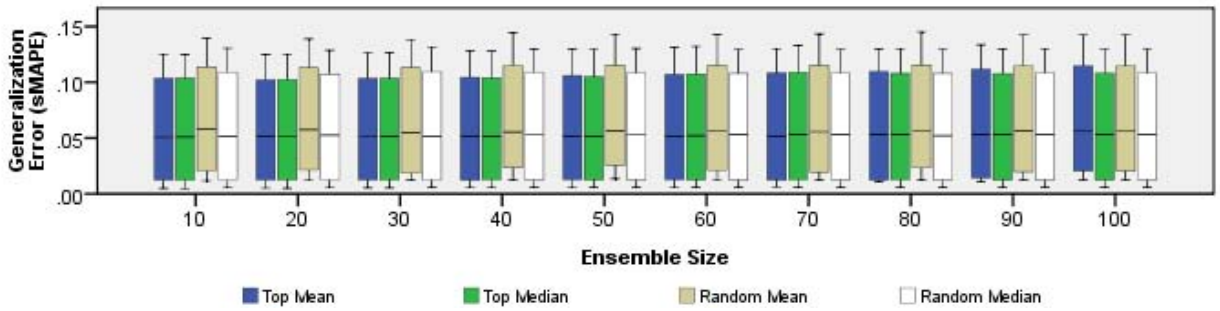

Fig. 3. Boxplots of ensemble performance (test set error – sMAPE) across different noise levels for the multiple models experiments.

performance and slightly more accurate. This is only up to an ensemble size of 60% of model population however. For ensemble sizes 70% and greater mean ensemble performance significantly decreases e.g. at medium noise and top selection best mean ensemble performance error is 5.12% while at 70% it is 5.43% and at 100% it is 5.73%. Therefore, in addition to taking into consideration the type of selection, top or random, or prior knowledge of model accuracy as would usually be the case in practice, one also needs to consider the ensemble size as a percentage of the model population. If the best models are used from a larger mode population of models it would be advisable to include 50% of less of the total model population in the ensemble at which point the choice of method should be the mean. However as ensemble sizes become larger relative to model population, one runs the risk of including badly performing models which can significantly degrade the performance of the mean combiner.

At this point it can be summarized that selection of ensembles really does not depend on the level of noise but rather on the combination of ensemble model selection and method of model combination. If the best models are selected to be included in the ensemble and the ensemble size is small enough relative to the model population, then mean and median ensembles perform similarly with only marginally better performance by the mean.

Results on the performance of model selection and ensembles indicate that at low levels of noise, model selection outperforms ensembles regardless of ensemble type. Ensembles based on random selection perform even worst against model selection with a performance error of 1.13% (median) and 1.66% (mean) compared to 1.10% for model selection. At other levels of noise, while model selection outperforms ensembles based on random model selection, it is outperformed by ensembles based on top selection whether combined according to the mean or the median. It can be further summarised that ensembles offer no benefit over model selection on low noise data where the time series is less erratic. However the best ensembles, based on top selection, mean or median outperform model selection as noise level increases and the time series becomes more difficult to predict. The results are summarised in figure 3.

V. Conclusion

This paper investigates the performance of different neural network ensembles across several factors and compares them to model selection. The investigation was divided into two experiments, the first investigating the performance of homogenous ensembles based on a single neural network with different initializations and the second based on heterogeneous ensembles with networks having differing architectures. In both experiments ensembles created were evaluated across several factors:

1. Ensemble size ranging from 10% of total model population to 100% by increments of 10%
2. Ensemble model selection of two types; random selection where the models are chosen randomly from the population and top selection where the models are chosen based on top rank performance as evaluated by their validation set error.
3. and ensemble method which were one of either the mean or median combination.

In the first experiment based on the single network modeled over 30 initializations, no significant changes were noted across ensemble factors. The mean ensemble performed on average as well as the median ensemble across noise level and seasonality. It was however noted that ensembles based on random model selection perform worst than those based on top selection and that this performance worsens as noise level increases. When compared to model selection it was noted that mean and median ensembles based on top selection perform comparatively across all time series and therefore no benefits of ensembles over model selection were noted.

In the second experiment involving different networks, significant differences were noted across ensemble factors. Mean and median performance on top selection were similar, however if the prediction performance of individuals models are not known in advance, then the median is the preferred ensemble method. It is also the preferred ensemble method across noise level where the mean is more volatile and also across ensembles sizes. As noise level increases mean and median ensembles based on top selection also outperform model selection.

This research used synthetic time series to evaluate the

performance of ensembles of NNs. Although this aids in the design of the experiments, allowing us to run the single model ensemble experiments using correctly specified NN, it also limits our confidence in generalizing the findings of this study. Similar experiments should be performed on real time series. These will have unknown data generating process, therefore testing the accuracy of single model ensembles/model selection schemes in practice, against multiple model ensembles that seem to perform at least equally well, without assuming that the true data generating process has been captured.

REFERENCES

[1] J. M. Bates and C. W. J. Granger, "Combination of Forecasts," *Operational Research Quarterly,* vol. 20, pp. 451-&, 1969.

[2] P. Newbold and C. W. J. Granger, "Experience with Forecasting Univariate Time Series and Combination of Forecasts," *Journal of the Royal Statistical Society Series a-Statistics in Society,* vol. 137, pp. 131-165, 1974.

[3] S. Crone. (2007, 20/08/2009). *NN3 Results*. Available: http://www.neural-forecasting-competition.com/NN3/results.htm

[4] S. Makridakis and R. L. Winkler, "Averages of Forecasts - Some Empirical Results," *Management Science,* vol. 29, pp. 987-996, 1983.

[5] S. Makridakis*, et al.*, "The Accuracy of Extrapolation (Time-Series) Methods - Results of a Forecasting Competition," *Journal of Forecasting,* vol. 1, pp. 111-153, 1982.

[6] G. Elliott and A. Timmermann, "Optimal forecast combinations under general loss functions and forecast error distributions," *Journal of Econometrics,* vol. 122, pp. 47-79, Sep 2004.

[7] M. Assaad*, et al.*, "A new boosting algorithm for improved time-series forecasting with recurrent neural networks," *Information Fusion,* vol. 9, pp. 41-55, Jan 2008.

[8] G. Zhang*, et al.*, "Forecasting with artificial neural networks: The state of the art," *International Journal of Forecasting,* vol. 14, pp. 35-62, 1998/3/1 1998.

[9] G. Q. Zhang*, et al.*, "Forecasting with artificial neural networks: The state of the art," *International Journal of Forecasting,* vol. 14, pp. 35-62, Mar 1998.

[10] L. K. Hansen and P. Salamon, "Neural Network Ensembles," *Ieee Transactions on Pattern Analysis and Machine Intelligence,* vol. 12, pp. 993-1001, Oct 1990.

[11] F. C. Palm and A. Zellner, "To combine or not to combine - Issues of combining forecasts," *Journal of Forecasting,* vol. 11, pp. 687-701, Dec 1992.

[12] R. L. Winkler and R. T. Clemen, "Sensitivity of weights in combining forecasts," *Operations Research,* vol. 40, pp. 609-614, May-Jun 1992.

[13] V. R. R. Jose and R. L. Winkler, "Simple robust averages of forecasts: Some empirical results," *International Journal of Forecasting,* vol. 24, pp. 163-169, Jan-Mar 2008.

[14] S. Makridakis*, et al.*, "The M2-Competition - A Real-Time Judgmentally Based Forecasting Study," *International Journal of Forecasting,* vol. 9, pp. 5-22, Apr 1993.

[15] S. Makridakis and M. Hibon, "The M3-Competition: results, conclusions and implications," *International Journal of Forecasting,* vol. 16, pp. 451-476, Oct-Dec 2000.

[16] M. Assaad*, et al.*, "Predicting chaotic time series by boosted recurrent neural networks," in *Neural Information Processing, Pt 2, Proceedings*. vol. 4233, I. King*, et al.*, Eds., ed Berlin: Springer-Verlag Berlin, 2006, pp. 831-840.

[17] R. Avnimelech and N. Intrator, "Boosting regression estimators," *Neural computation,* vol. 11, pp. 499-520, Feb 1999.

[18] Y. F. Deng*, et al.*, *Ensemble SVR for prediction of time series*. New York: Ieee, 2005.

[19] C. E. Agnew, "Bayesian Consensus Forecasts of Macroeconomic Variables," *Journal of Forecasting,* vol. 4, pp. 363-376, Oct-Dec 1985.

[20] J. H. Stock and M. W. Watson, "Combination forecasts of output growth in a seven-country data set," *Journal of Forecasting,* vol. 23, pp. 405-430, Sep 2004.

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

[22] S. D. Balkin and J. K. Ord, "Automatic neural network modeling for univariate time series," *International Journal of Forecasting,* vol. 16, pp. 509-515, Oct-Dec 2000.

[23] U. Anders and O. Korn, "Model selection in neural networks," *Neural Networks,* vol. 12, pp. 309-323, Mar 1999.

[24] G. Lachtermacher and J. D. Fuller, "Backpropagation in Time-Series Forecasting," *Journal of Forecasting,* vol. 14, pp. 381-393, Jul 1995.