Financial predictions based on bootstrap-neural networks

A. Lombardi*, A. Vicino^{\(\disp\)}

* Linköping University ITN/Campus Norrköping, SE-601 74 Norrköping, Sweden e-mail: annlo@itn.liu.se

[⋄] Dipartimento di Ingegneria dell'Informazione Universit'à di Siena, Via Roma 56, 53100 Siena, Italy

Abstract. In this paper neural netw orks are applied to financial data in order to predict the daily price of the financial index LIFFE. Our attention is focused on the droice of the exogeneous variables and on the training of the netw ork itself. The first problem is solved by using the pre-whitening method that provides information on which variables are the most relevant for our prediction. The latter problem is due to the fact that data referring to a far past cannot be used because of the non-stationarit yof the financial indicators. This implies that the training set is relatively small and it is necessary to extract as much information as possible from recent data. The bootstrap approach is applied to the training set of the neural netw ork to improve the prediction capabilities of the system. This results in better prediction performances even when a limited number of data is available.

1. Introduction

F orecasting is the rational prediction of future even ts on the basis of information about past and current events. T raditional forecasting methods are almost all based on linear or linearized models such as autoregressive moving average methods. Sev eralinteresting approaches are based on neural net w orks([1], [2]).

Several researchers have attempted to use feed-forward neural networks [3] to predict future values of time series by extracting knowledge from the past. Neural networks have drawn considerable attention in recent years because of their in teresting learning abilities. Although back-propagation is probably the most widespread training algorithm, in this paper the Leven berg-Marquardt algorithm is used to train the network. This optimization method is more sophisticated than gradient descent and represents a very good alternative to it as it makes training times significantly shorter.

The bootstrap technique can be very useful when data collection is costly, i.e. expensive real measurements have to be performed, or when limited infor-

mation is available. In this situation, an algorithm generating its own training data is appropriate. The basic idea of bootstrap, first introduced by Efron [5], is that the relationship between the true distribution function F and the sample of size n is similar to the relationship betw een the sample distribution function F_n and a secondary sample drawn from it. It is, therefore, possible to replace the unknown distribution F by the empirical one F_n and use the sample drawn from F_n to train the netw ork. For a survey on bootstrap results see [6] and [7].

In multiv ariate analysis, an important problem is the choice of the exogeneous variables most significant for the variable to be predicted. In this paper the pre-whitening method is used ([8],[9]). It provides a criterion for choosing the variables that exhibits a higher correlation with the endogeneous variable as inputs to the model.

This paper describes the application of neural netw orks in conjunction with the bootstrap algorithm to the prediction of the financial index LIFFE. In the next section we give a brief overview of the method used for the prediction. In particular the bootstrap method applied to neural networks and the prewhitening technique will be illustrated. In Section 3 the implementation of the predictor is illustrated and the numerical results obtained on real data are discussed: the neural networks proach is compared with classical methods. Finally, Section 4 concludes the paper.

2. The estimation problem

2.1. Neural networks

Neural netw orksare well known and therefore we will not spend time in describing the nonlinear models they represent. How exer, it is worth to illustrate which net workshave been implemented in the experiments described in the following. Feedforwad networks have been used and they have been trained in batch mode using the Leven berg-Marquardt algorithm. This method is based on the Jacobian matrix of the cost function and and it increases the speed of the convergence quite significantly. In this paper we use a constructive procedure: hidden units are added to the network one-by-one starting from a single, fully connected hidden unit.

2.2. Bootstrap method

The bootstrap technique is a very general method to create measures of uncertain ty and bias in parameter estimation from independent and identically distributed (i.i.d.) variables (see [10] for an exhaustive review of the method and its properties.

Let the real problem be formulated as follows. Let x_1, \ldots, x_n be n independent observations from some distribution F(x). A parameter $\theta = g(F(\cdot))$ is estimated on the basis of the n observations $\widehat{\theta} = \widehat{\theta}(x_1, \ldots, x_n)$. When parameters

are estimated, though, the distribution F is unknown.

The basic idea of bootstrap is to replace the unknown distribution F(x) by the sample distribution $F_n(x)$. The bootstrap problem can be summarized in the following two steps:

• Replace the parameter θ with the parameter $\tilde{\theta}$ dependent on the sample distribution $F_n(x)$

$$F_n(x) = \frac{\#\{x_i \le x\}}{n}$$

$$\tilde{\theta} = g(F_n(\cdot))$$
(1)

Notice that the variables in (1) are known since x_1, \ldots, x_n represent measurements.

• Simulate the same number of independent observations from $F_n(x)$ and get the bootstap sample

$$x_1^*, \dots, x_n^*. \tag{2}$$

The bootstrap elements are randomly selected from the original ones with replacement. This means that some elements may appear several times while others may be missing.

Then the parameter estimate is computed as

$$\widehat{\theta}^* = \widehat{\theta}(x_1^*, \dots, x_n^*) \tag{3}$$

by the same estimator as in the real problem.

In this paper, the bootstrap technique is used to improve the prediction capability of neural netw orks, rather than for measuring the precision of a well defined estimate. Specifically, the bootstrap approach is used in our context to generate an extended training set for the neural netw ork. V etor resampling has been applied to the neural netw ork models. It is merely the resampling of n rows of data drawn with replacement from the original n rows.

The bootstrap loop is applied to the original data set NB times so that a training set that is NB times the size of the original one is generated. The netw orkis thus trained on a bigger set and consequently it is expected that the prediction ability of the netw ork improves as the netw ork can extract more information during the training process.

2.3. Pre-whitening

Many variables related to this financial problem can be considered as exogeneous inputs to the model. Nevertheless, the use of many variables would require a neural net wok with an excessive number of neurons and possibly many hidden layers. In our problem, it is not possible to train such a netw ork because few data are available for each time series. In order to choose which of these v ariables are more significant for prediction purposes, the pre-whitening method is applied to our data. This technique is based on the analysis of the residuals of the predictions based on the best univariate model. The exogeneous variables showing a higher correlation with the residual endogenous

variable are chosen as the most appropriate for prediction. The result of this analysis allows us to choose as exogeneous input for our model the future price of the bund index.

3. Numerical results

The time series to be predicted is the future price of the financial index LIFFE (London International Financial F utures Exchange). The series contains the daily closing-time prices from June to October 1998 and it is reported in Figure 1

The data set is composed of about one hundred data which have been divided into two sets: the first 80 items of the data were used to train the network while the last 21 items were used to validate the network whose parameters are generated during the training process. All data have been differentiated and all the results of the simulations refer to differentiated data even if it is not stated explicitly

First, the net w orkis trained using the Leven berg-Marquardt algorithm. When the netw ork training is over, a simulation is run using the validation set and the output variable is predicted.

The sum-squared error is computed on these predictions and it is used as a figure of merit to compare results

$$E = \sqrt{\frac{1}{N_v} \sum_{i=1}^{N_v} \left[\hat{y}(t_0 + i) - y(t_0 + i) \right]^2},$$
 (4)

where $y(\cdot)$ denotes the real value and $\widehat{y}(\cdot)$ the corresponding one step ahead prediction. N_v represents the number of data contained in the validation set. Note that validation data are never used during the training process so that the netw ork is always tested on data that have not been used in the parameter estimation step of the procedure.

The neural netw ork predictor has been implemented in MATLAB by using the *Neural Network Toolbox*.

Our aim is to choose an architecture which minimizes the prediction error (4). To this purpose, a systematic search over the space of one hidden layer perceptrons was performed. Many simulations with a different number of neurons in the hidden layer have been run in order to find out the new ork giving the smallest prediction error. The resulting optimal network has 3 neurons in the hidden layer.

One crucial issue with neural netw orks is the fact that the solution achiev ed depends on the initial values c hosen for the wights. In fact, when the netw ork is being trained, at every step a minimization problem is solved; the problem is not con vex and heman y local minima are present. Several simulations have been run starting from different initial conditions. The smallest prediction error obtained is

$$E = 0.2592 (5)$$

Bootstrap has been applied to the training set. Experiments with different values for NB have been carried out using the same initial conditions corrisponding to the prediction error (5). The smallest prediction error was obtained when NB=3 and the prediction error is

$$E = 0.1577$$
 . (6)

When bootstrap is applied, all the simulations are carried out with the same starting point for the parameters. Due to the resampling of the training set according to the bootstrap algorithm, each repetition of the same experiment provides, possibly, a different prediction error. Bootstrap is applied only to the training set, so that during the validation process, the netw ork is always tested on the same data. Consequently, it is meaningful to compare the prediction error obtained in the different repetitions of the experiment.

Comparison with statistical techniques

To evaluate the performance level of neural netw ork models, a standard linear regression model was computed. The predictions produced by the linear model are compared to those produced by the neural netw ork. Sev eral models have been estimated and for each model, different initial v alues for the parameters have been considered. The best selected model is an ARIMAX model

$$A(d)\Delta y(t) = B(d)u(t) + C(d)e(t), \tag{7}$$

with $n_a = 3$, $n_b = 5$, $n_c = 5$.

The obtained error computed on the validation set is

$$E = 0.3914.$$
 (8)

In Figure 2 the predictions obtained by bootstrap-neural netw ork tec hnique and the linear models are reported and compared to the real data. It can be oserved that neural networksoutperform linear models and in addition, application of the bootstrap consistently reduces the prediction error.

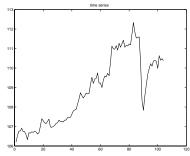
4. Conclusions

In this paper we have discussed an application of neural networks to time series prediction for financial purposes.

Models based on neural netw orks can be used as an alternative to linear models. Although non-linear models are more demanding from a computational point of view, results show how they can improve the predictive performance compared to classical linear statistical models

The bootstrap approach has been applied to the training set of the neural netw ork. This technique is very useful when few data are available because it allows the netw ork to extract more information from the data. The prediction error produced by a networktrained on a resampled data set is significantly smaller than the error produced by a networktrained on the original data.

This can be explained by the fact that the netw ork istrained on a new cost function with, possibly, different minima.



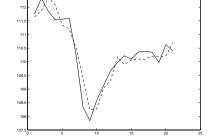


Figure 1: Time series

Figure 2: Predictions on the validation set

References

- [1] R.G. Hoptroff. "The principles and practice of time series forecasting and business modeling using neural nets." Neural Computing & Applications Journal, vol. 1, pp. 59–66, 1993.
- [2] A.N. Refenes, A. Zapranis and G. Francis. "Stock performance modelling using neural netw orks: a comparative study with regression models." *Neural Networks*, vol. 7, n. 2, pp. 375–388, 1994.
- [3] T. Kimoto et al.. "Stock market prediction with modular neural networks." Proceedings of IJCNN-90 San Diego, CA, 1990.
- [4] S. Haykin. Neural Networks. A Comprehensive Foundation Maxwell Macmillan International, New York.
- [5] B. Efron. "Bootstrap Methods: Another Look at the Jac kknife." The Annals of Statistics vol. 7, pp. 1–26, 1979.
- [6] T.J. DiCiccio and J.P. Romano. "A Review of Bootstrap Confidence Intervals." Journal of the Royal Statistical Society, Ser. B, vol. 50, pp. 338–354, 1988.
- [7] D. Hinkley. "Bootstrap Methods." *Journal of the Royal Statistical Society*, Ser. B, vol. 50, pp. 321–337, 1988.
- [8] , L. Ljung. System Identification Theory for the User. PTR Prentice hall, 1999.
- [9] W. Vandaele. Applied Time Series and Box-Jenkins Models. Academic Press, 1983.
- [10] J.S.U. Hjorth. Computer Intensive Statistical Methods. V alidation model selection and bootstrap Chapman & Hall, London, 1994.