2009 International Joint Conference on Computational Sciences and Optimization

A Statistical Neural Network Approach for Value-at-Risk Analysis

Xiaoliang Chen*, Kin Keung Lai
Department of Management Sciences, City University of Hong Kong
Tat Chee Avenue, Kowloon, Hong Kong
msxlchen@cityu.edu.hk

Jerome Yen
Department of Finance, The Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong

Abstract

This study develops a new methodology based on ANN for Value-at-Risk (VaR) modeling. Specifically, we propose a statistical procedure for ANN model selection. The statistical ANN deals with each layer individually and estimates the weights of subsequent layer with those of preceding layers fixed. This allows the derivation of statistical theory for model selection, which reduces the need to fit a comprehensive set of models. Experiment results show that the statistical ANN approach performs well on stock index return series compared to traditional forecasting methods.

1. Introduction

VaR calculations aim to measure the worst expected loss over a given time interval under normal market conditions at a given confidence level [1]. As a comprehensive risk measure, VaR summarizes the overall market risk exposure through one single quantitative parameter. Accurate estimation of VaR is of critical importance for companies to control and manage risk-bearing business activities, and for financial institutions to comply with requirements of national regulatory authorities.

Originally, VaR is defined as the value that a portfolio might lose with a given probability over a certain time horizon (usually one or ten days). Standard VaR estimates take the mathematical form as in Equation 1, which can be translated into narrative terms as 'we are α percent certain that we will not lose more than VaR of our investment in the next t days under normal market conditions [1]'.

$$P(\Pi(t) - \Pi(0) \leq -VaR) = 1 - \alpha \qquad (1)$$

Where $\Pi(\cdot)$ denotes the value of the portfolio at time t, and α is the confidence level.

However, although the concept is simple and straightforward, VaR estimation is in fact a very tough statistical proposition, and, unfortunately, none of the traditional estimation methods has achieved convincing results. The main reason is that they overlook the stylized facts of financial time series, such as heavy-tailedness, skewness, heteroskedasticity, etc., and then misspecify model assumptions. With globalization of capital markets, empirical distributions of modern asset returns have become more and more complicated, making their market risk more difficult to capture. Under such circumstances, ANN has been widely advocated as a new alternative modeling methodology, which has gained increasing acceptance in the fields of econometric and financial forecasting and risk management.

Specifically, several attempts have been made to predict VaR, as well as volatility and future value, with ANN. Locarek-junge and Prinzler [2] illustrated how VaR estimates can be obtained, using a neural network approach. Using an US-Dollar portfolio, they showed that ANN may offer an alternative to existing VaR models that assume normality in market returns. Schittenkopf et al. [3] proposed to estimate conditional densities in a neural network framework. Empirical analysis of daily FTSE 100 data showed that the out-of-sample forecasting performance of neural networks sightly dominates those of GARCH models. Dunis and Huang [4] examined non-parametric neural network models for forecasting volatility in trading currency, applying them to GBP/USD and USD/JPY exchange rates. Experiment results with daily data indicated that ANN appears to be a better single modeling approach than the GARCH alternative. Hamid and Iqbal [5] compared volatility forecasts from neural networks with forecasts of implied volatility from S&P 500 index futures options, using the Barone-Adesi and Whaley (BAW) American futures options pricing model. Forecasts from neural networks outperformed

978-0-7695-3605-7/09 $25.00 © 2009 IEEE
DOI 10.1109/CSO.2009.350

implied volatility forecasts. Yu et al. [6] presented an ANN based approach to predict foreign exchange rates. Empirical analysis revealed that the novel forecasting model outperforms the other comparable models. Yu et al. [7] introduced an empirical mode decomposition (EMD) based neural network ensemble learning paradigm for world crude oil spot price forecasting and empirical results demonstrated attractiveness of the proposed method. In summary, these pilot studies indicate a large potential for ANN in financial applications, including VaR estimation.

ANN offers a particularly powerful choice since they can efficiently represent non-linear multivariate functions with no serious limitations on the form of functions to be approximated. However, despite the advantages of using neural networks, there are currently no widely accepted procedures for determining the network architecture for a given application. Considering this shortcoming, we propose a new architecture selection process with statistical inference to facilitate the application of neural networks, specifically, to VaR estimation.

The remainder of this paper is organized as follows. Section 2 proposes the statistical ANN method by deriving its underlying mathematical theory and developing a practical procedure for VaR estimation. Section 3 presents empirical analysis on Hong Kong's stock index. It also compares the prediction performance of the statistical ANN model and the classical ARMA-GARCH method. Section 4 draws the conclusion and discusses future research directions.

2. The proposed model

In this section, we present the statistical ANN method, derive its underlying mathematical theory and develop a practical approach for its application for VaR estimation.

2.1. Statistical ANN model selection

The motivation for using traditional statistical selection procedures (such as the likelihood ratio and Wald tests) is that multilayer neural networks are essentially nonlinear models, and the same procedures should be used for testing parameter significance. Unfortunately, this kind of approaches is problematic because of the non-identifiable feature of a neural network model. For example, if the connection weight between a hidden and output unit is zero, the input to hidden layer weights associated with that connection can take any value. From this perspective, it is evident that the identification problem is mainly caused by the dependence of the weights between layers. One possible solution might be to estimate the weights for each layer independently. Since a single hidden layer is adequate for most applications, this procedure could be divided into two steps, i.e. estimating the input to hidden layer weights and then estimating the hidden to output layer weights. Below we derive results which interpret how independent estimation of weights for each layer works.

In neural network training, the first step is to take linear combinations of the input variables. Suppose that there are m vectors each with p elements $(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m)$. Then the set of all possible combinations of these vectors (i.e. $k_1\mathbf{x}_1 + k_2\mathbf{x}_2 + \cdots + k_m\mathbf{x}_m$) forms a linear space and the $\mathbf{x}$ vectors span or generate it. This is analogous to creating a new vector space in statistical terminology. Provided we have a set of vectors which are different lagged versions of the original time series, the linear combinations of these vectors also constitute a basis for the predictor space. Proper selection of this basis can help solve many problems related to using neural networks. Specifically, we hope to select a basis which could capture the underlying characteristics of the input variables and at the same time keep away from the computational problems induced by parameter non-uniqueness or multicollinearity. A natural motivation is the use of PCA, which is a dimension reduction technique with the goal of finding a few principal components that explain a large proportion of the total sample variance. In PCA, the original variables are transformed into new ones, which are mutually orthogonal. It is computationally very beneficial if the original variables are highly correlated. Mathematically, if $\mathbf{x}$ is a random vector with mean μ and covariance matrix Σ, the principal component analysis takes the transformation as follows

$$\mathbf{x} \to \mathbf{z} = \mathbf{\Gamma}'(\mathbf{x} - \mu), \tag{2}$$

where $\mathbf{\Gamma}$ is orthogonal, and $\mathbf{\Gamma}'\mathbf{\Sigma}\mathbf{\Gamma} = \mathbf{\Lambda}$ is the diagonal matrix of eigenvalues λ_j ($\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0$). Then the ith principal component of $\mathbf{x}$ may be defined as the ith element of vector $\mathbf{z}$:

$$z_i = \gamma'_{(i)}(\mathbf{x} - \mu). \tag{3}$$

Here, $\gamma'_{(i)}$ is the ith column of $\mathbf{\Gamma}$, and is called the ith vector of principal component loadings. In principal components, the first one has the largest variance among all the linear combinations of $\mathbf{x}$ and $var(z_j) = \lambda_j$.

The second part of the statistical ANN model entails weighted summation of activation functions whose parameters are determined by nonlinear least squares. First, we observe that in univariate time series forecasting, we are primarily concerned with mapping $f : \mathbf{R}^p \to \mathbf{R}^1$, i.e. we approximate function f, which has p inputs and a single output. Note that the mapping refers to the 'signal' component, while the random error, by definition, cannot be approximated by past values. This result translates into taking a linear combination of the input variables, transforming them through a weighted logistic function, and summing the values to derive the output.

For ANNs, the inner summation, say h_{pj}, has unknown weights that are estimated by the backpropagation algorithm. In the proposed method, the weights in the inner summation

$$h_{pj} = \alpha_j + \sum_i w_{ji} x_{pi} \tag{4}$$

are determined by construction of the basis presented previously. Thus, in the univariate time series framework, we are left with

$$\hat{y}_t = \alpha_0 + \sum_{j=1}^{k} W_j \varphi_1(h_{pj}) \tag{5}$$

where φ_1 is the logistic function, h_{pj} is the pth value from the jth basis variable, W_j represents the weight from hidden node j to the output unit, and α_0 is a scalar bias.

Now that the modeling structure has been developed, the next consideration is model selection. In common with all neural network studies [8], the model parameters are estimated by nonlinear least squares. This type of algorithm modifies the model parameters to minimize the sum of squared errors. For testing purposes, when employing least squares, we assume an additive and Gaussian error structure written as

$$y_t = g(z_t, \theta) + \varepsilon_t \tag{6}$$

where g is some possibly nonlinear function of data z_t and parameters θ.

In the model selection process, we may want to assess the importance of each of the basis variables in terms of its predictive ability. We can do this for one variable at a time by choosing between two model specifications:

$$H : y_t = g(z_t, \theta) + \varepsilon_t \tag{7}$$

and

$$A : y_t = g(z_t, \theta) + \beta h(z_t, \omega) + \varepsilon_t \tag{8}$$

where θ, β and ω are unknown parameters. The asymptotic level α likelihood ratio test for

$$H : \beta = 0 \text{ against } A : \beta \neq 0$$

is given by

$$L = \frac{(SSE(\hat{\theta}) - SSE(\hat{\theta}, \hat{\omega}))/q}{SSE(\hat{\theta}, \hat{\omega})/(P - s)} \tag{9}$$

where $(SSE(\hat{\theta})$ and $SSE(\hat{\theta}, \hat{\omega})$ are the sums of squared errors associated with the full and reduced models, respectively. Under H_0, L is F-distributed with q (numerator) and $P - s$ (denominator) degrees of freedom (q is the number of weight parameters which are constrained to be zero in the reduced model). Therefore, whenever L exceeds the $\alpha \times 100$ critical point of the F-distribution, the reduced model is rejected.

2.2. VaR estimation procedure

Interpreting VaR as the quantile of future asset returns, conditional on current information, this section deals with a new approach to quantile estimation which does not require any of the questionable assumptions adopted by traditional VaR estimation methods. In the first place, we define formally the univariate process for asset return, r_t, as follows

$$\begin{aligned} r_t &= \mu_t + \varepsilon_t \\ \varepsilon_t &= z_t \sigma_t \\ \mu_t &= \mu(\Omega_{t-1}; \theta) \\ \sigma_t^2 &= \sigma^2(\Omega_{t-1}; \theta) \end{aligned} \tag{10}$$

Where r_t represents the financial return of the portfolio at time t, μ_t is the conditional expected return, σ_t represents the conditional volatility of the portfolio change or innovation ε_t, Ω_{t-1} is the information set available at time $t-1$, θ is a finite-dimensional vector of true parameter values, and z_t is an iid random variable satisfying $E(z_t|\Omega_{t-1}; \theta) = 0$ and $Var(z_t|\Omega_{t-1}; \theta) = 1$.

Assume a portfolio of one asset position worth \$1. The VaR could be defined parametrically as:

$$VaR_\alpha^t = \mu_t + q_\alpha \sigma_t \tag{11}$$

Where q_α is the α quantile of z_t.

The definition of parametric VaR says to get an estimate one needs to quantify the expected return or conditional mean μ_t, and conditional volatility σ_t of the asset values, and the conditional quantile of the martingale difference sequence z_t.

For estimation of the conditional mean μ_t of the asset return model, let the information set Ω_t be generated by

$$Z_s := (r_s, \ldots, r_{s-\tau+1}), s \leq t \tag{12}$$

Here, Z_s represents the most recent returns. Based on this information, one can estimate the expected conditional return

$$\mu_t = E(r_t|\Omega_{t-1}) \tag{13}$$

As suggested by Bishop [9], for a neural network trained by least-squares method, the network function is given by the conditional average of the target data, conditioned on the input vector. Therefore, by training the network using our proposed method with data set (r_t, Z_{t-1}), one could obtain the one-day-ahead prediction of the conditional mean directly.

As in the case of the expected return, one can use statistical ANN for estimating conditional volatilities. Considering the stochastic process that describes the asset returns dynamics, the conditional variance is given by

$$\begin{aligned}\sigma_t^2 &= \left(\frac{r_t - u_t}{z_t}\right)^2 \\ &= E(r_t^2|\Omega_{t-1}) - [E(r_t|\Omega_{t-1})]^2 \\ &= E(r_t^2|\Omega_{t-1}) - \mu_t^2\end{aligned} \tag{14}$$

Therefore, one could estimate conditional volatility σ_t^2 by estimating the conditional second moment and subtracting the squared neural network estimate of the conditional expected return $E(r_t|\Omega_{t-1})$. The procedure described above is applicable to (r_t^2, Z_{t-1}) instead of (r_t, Z_{t-1}).

For conditional quantile estimation, we follow an approach similar to FHS so as not to impose any theoretical distribution on the data. To bring returns close to a stable distribution we subtract the expected mean estimate from the original daily return and divide the result by the corresponding volatility estimate. Thus, the standardized residual return is given by

$$\hat{e}_{t-i} = \frac{r_{t-i} - \hat{\mu}_{t-i}}{\hat{\sigma}_{t-i}}, \quad i = 1, \ldots, t-1 \tag{15}$$

This makes the set of standardized innovations almost independently and identically distributed (i.i.d.) and, therefore, suitable for historical simulation. Then historical standardized innovation e^*, drawn randomly (with replacement), after being scaled with conditional volatility and added with conditional mean, could generate a pathway for future return, which is given by

$$r_t^* = \hat{\mu}_t + e^* \hat{\sigma}_t \tag{16}$$

The distribution of evolved returns at the end of the pathway (i.e. one day ahead) for the single asset is obtained by replicating this procedure a large number of times (e.g. 5000). Then the VaR could be estimated at designated quantile or confidence level with the simulated distribution.

3. Empirical analysis

3.1. Data description

The data set used in this paper consists of 913 log returns derived from daily observations of the Hang Seng Index (HSI) from 2 May 2003 to 29 December 2006. In our experiment, the one-day-ahead VaR estimates are obtained with the rolling-window approach in which training set constains 500 data points.

Table 1. Descriptive statistics

Statistics	Value
Mean	0.0009
Maximum	0.0352
Minimum	-0.0418
Median	0.0008
Standard Deviation	0.0092
Skewness	-0.2358
Kurtosis	4.5233

Descriptive statistics for logarithm return series of HSI in the sample period are summarized in Table 1. In the table we can find some stylized facts as spotted in typical asset return series. The skewness of -0.2358 denotes that more returns lie in the left tail than would be expected in a normal distribution. Also, the high kurtosis implies that the possibility of encountering extreme events is comparatively large. These facts suggest that advanced risk management instruments should be in place.

3.2. Experiment results

In this study, the proposed statistical ANN model is benchmarked on classical ARMA-GARCH model for VaR estimation. The ARMA, GARCH and ANN models, as well as PCA, are all implemented through Matlab software produced by Mathworks Laboratory Corporation. For ARMA-GARCH VaR model, the complete set of daily expected returns and volatilities are generated by ARMA(0, 1) and GARCH(2, 3) whose model orders are chosen via AIC. The VaRs are calculated with different quantiles derived from standard normal distribution.

For the statistical ANN model, to predict conditional mean we choose six lags as input variables of the network, based on autocorrelation analysis. Then six principal components are derived using PCA. The likelihood ratio test described in the last section, with $\alpha = 99\%$, is used to determine whether a component is to remain in the model. As a result, five components are kept in the network. The procedure for conditional volatility estimation is designed in a similar way. Eight lags are selected as input variables and three components remained in the network structure. The training epochs are set to 1000 due to the problem complexity. To assess the forecasting ability of the new method, we compute mean squared error (MSE), Kupiec's statistic and its p-value for the estimates.

Table 2 presents the experiment results of the benchmark model and the proposed model. The results show that ARMA-GARCH models are accepted at 97.5% and 95% confidence levels, and the statistical ANN models are accepted on all conditions. Careful analysis suggests that the statistical ANN model improves the reliability (as measured

Table 2. Experiment results

Models	Confidence Level	VaR Exceedance	MSE	Kupiec Statistic	P-value	Model Acceptance
ARMA	99.0%	12	0.0004	10.0505	0.0015	×
GARCH	97.5%	18	0.0003	4.8408	0.0277	√
Model	95.0%	28	0.0003	2.5280	0.1118	√
Statistical	99.0%	5	0.0011	0.1777	0.6733	√
ANN	97.5%	12	0.0006	0.2735	0.6010	√
Model	95.0%	19	0.0004	0.1342	0.7142	√

by p-value) significantly while retaining a similar level of accuracy (as measured by MSE) compared to the benchmark model. Besides, in our experiment the overall estimation process of the the statistical ANN model needs about half time as that of the ARMA-GARCH model does. Therefore, we argue that it could be used as an alternative model for VaR estimation due to its good performance and fewer distribution assumptions required.

4. Conclusion

A neural network is a nonlinear model with the ability to approximate practically any function to any desired degree of accuracy. However, using a neural network for modeling and forecasting proved difficult for a number of reasons, such as the lack of a statistical basis and the non-identifiability of model parameters. These difficulties motivate the need for a modeling paradigm that can learn the connection between variables and can select a model using consistent statistical procedures. This study derives and presents such a method, and demonstrates its ability in VaR estimation, using real data.

Further investigation to evaluate the applicability of the proposed method, and further improvement of its performance, is desirable. One possibility is to include exogenous variables in the estimation process. In fact, the stock market is often influenced by many external factors, such as interest rates, exchange rates, etc. Thus, containing exogenous factors as extra explanatory variables may strengthen the model's forecasting ability.

Acknowledgments

This project is supported by the grant from the NSFC of China and RGC of Hong Kong Joint Research Scheme (Project No. N_CityU110/07).

References

[1] P. Jorion. *Value at risk : the new benchmark for managing financial risk.* McGraw-Hill, New York, 2nd. edition, 2001.

[2] H. Locarek-junge and R. Prinzler. Estimating value-at-risk using neural networks. *Informations Systeme in der Finanzwirtschaft*, 1998.

[3] C. Schittenkopf, G. Dorffner, and E.J. Dockner. Forecasting time-dependent conditional densities: a semi-nonparametric neural network approach. *Journal of Forecasting*, 19:355–374, 2000.

[4] C.L. Dunis and X. Huang. Forecasting and trading currency volatility: An application of recurrent neural regression and model combination. *Journal of Forecasting*, 21(5):317–54, August 2002.

[5] S.A. Hamid and Z. Iqbal. Using neural networks for forecasting volatility of S&P 500 index futures prices. *Journal of Business Research*, 57(10):1116–1125, 2004.

[6] L.A. Yu, S.Y. Wang, and K.K. Lai. Adaptive smoothing neural networks in foreign exchange rate forecasting. *Lecture Notes in Computer Science*, 3516:523–530, 2005.

[7] L.A. Yu, S.Y. Wang, and K.K. Lai. Forecasting crude oil price with an emd-based neural network ensemble learning paradigm. *Energy Economics*, 30(5):2623–2635, Sep 2008.

[8] G. Zhang, B.E. Patuwo, and M.Y. Hu. Forecasting with artificial neural networks: the state of the art. *International Journal of Forecasting*, 14:35–62, 1998.

[9] C.M. Bishop. *Neural networks for pattern recognition.* Oxford University Press, 1995.