Bayesian Analysis of Neural Network Models for Conditional Return Distribution

Tatiana Miazhynskaia[1], Sylvia Frühwirth-Schnatter[2], Georg Dorffner[3]

[1] Department of Finance and Corporate Control, Vienna University of Technology,
Tatiana.Miazhynskaia@tuwien.ac.at
[2]Institute for Applied Statistics, Johannes Kepler University Linz, Austria
[3]Austrian Research Institute for Artificial Intelligence and Department of Medical Cybernetics and Artificial Intelligence, Medical University of Vienna, Austria

Abstract

We use neural networks (NN) as a tool for a non-linear autoregression to predict the second moment of conditional density of financial return series. The NN models are compared to the popular econometric model GARCH(1,1). We estimate the models in the Bayesian framework using Markov Chain Monte Carlo posterior simulations. The inter-linked aspects of the proposed Bayesian methodology are identification of NN hidden units and treatment of NN complexity based on model evidences. The empirical study includes the application of the designed strategy to the market data, where we found strong support for the non-linear NN model.

Keywords: Neural networks, Volatility modeling, Bayesian model selection, Markov Chain Monte Carlo (MCMC)

1. Introduction

One of the recent popular applications of NN is volatility modeling. Volatility (measured by the standard deviation of returns) is the key ingredient for the pricing of financial instruments and for risk estimation; and volatility modeling is of great importance in financial literature. The NN was found to provide a good tool to model non-linearity in volatility processes (see, e.g., [1] and references there).

In our study, we consider NN volatility models in the Bayesian framework. The Bayesian inference on NN models has gained its own place in literature (see, e.g. [2-6]). The autoregressive structure of our models limited our choice of methods. In our Bayesian implementation we combine the methodology [5] with a modified version of [4].

An open question in the NN community is the selection of the "optimal" size of the network. [1] and [4] among the others applied the reversible jump MCMC algorithm to obtain joint estimates of the number of neurons and weights. In our study, we use the bridge sampling technique ([7-8]) to compute model evidences for different NN sizes and choose the "optimal" size of the network based on posterior model probabilities. This strategy allows us to handle models independently that can reduce computations in the case of re-arranging of a model set.

The efficiency of the MCMC simulations for the NN parameters is closely connected to the identifiability problem of the network hidden units. Unlike the known literature (see [4]), we select identifiability constraints a posteriori, after the random permutation of the hidden node parameters.

For empirical analysis we used return series of the NIKKEI 225 stock index (Japan) over 16 years.

The key contributions of this study are:

- the full Bayesian analysis of the autoregressive NNs models, including prior specification, MCMC simulation, model selection, with application of this methodology to financial data;
- an objective strategy for NN hidden units identification;
- Bayesian treatment of neural network complexity based on model evidences.

2. Models

As a benchmark volatility model we use classical GARCH(1,1) model [9] with conditional normal distribution and AR(1) process for mean equation for financial daily returns r_t, i.e.

$$\begin{cases} r_t = a_0 + a_1 r_{t-1} + e_t, \quad t = 1,2,\ldots,N \\ e_t \mid I_{t-1} \sim \mathrm{N}(0, h_t), \\ h_t = \alpha_0 + \alpha_1 e_{t-1}^2 + \beta_1 h_{t-1}, \end{cases}$$

where I_{t-1} is an information set at time $t-1$ and h_t denotes volatility at time *t*.

This model captures several "stylized facts" of asset returns, namely heteroskedasticity, volatility clustering and excess kurtosis. But it frequently fails to

capture highly irregular phenomena, such as market crashes and other significant structural changes. One popular direction to extend the classical GARCH is to allow for non-linear dependencies in the conditional variance. As a tool for non-linear regression we use NN-based modeling, describing the conditional variance by a multi-layer perceptron (MLP) ([1]).

To model the conditional variance we adapted the dynamics of MLP in a recurrent fashion

$$h_t = \sum_{j=1}^{H} v_j \Psi(w_j e_{t-1}^2 + \gamma_j h_{t-1} + c_j) + \alpha_0 + \alpha_1 e_{t-1}^2 + \beta_1 h_{t-1}, \quad (1)$$

with the logistic activation function of the hidden units $\Psi(z) = \exp(z)/(1-\exp(z))$. H denotes the number of hidden units. We call such non-linear volatility model *MLP-model* and its complete specification is

$$\begin{cases} r_t = a_0 + a_1 r_{t-1} + e_t, \quad t = 1,2,\ldots,N \\ e_t \mid I_{t-1} \sim \mathrm{N}(0, h_t), \\ h_t \text{ is given by (1).} \end{cases}$$

In such a definition, the MLP-model is a non-linear generalization of the GARCH(1,1) model, where the GARCH specification for the conditional variance is replaced by the recurrent MLP (1).

With respect to the size of the NN we will test the MLP-model with different values H and perform Bayesian model selection to define the optimal NN size.

3. Bayesian inference

3.1. Basic concepts and notations

All complex models may be viewed as the specification of a joint distribution of observables (data) which we denote by Y and unobservables (model parameters) which we denote by θ. And the traditional approach to Bayesian model selection is concerned with the following situation:

Suppose the observed data Y are generated by a model M_i, one of a set **M** of competing models. Each model specifies the data likelihood $f(Y \mid \theta_i, M_i)$ as the distribution of Y apart from an unknown parameter vector θ_i of dimension n_i. Under prior densities $\pi(\theta_i \mid M_i)$ the marginal distribution of Y are found by integrating out the parameters

$$p(Y \mid M_i) = \int f(Y \mid \theta_i, M_i)\pi(\theta_i \mid M_i)d\theta_i \quad (2)$$

By analogy with data likelihood function, the quantity $p(Y \mid M_i)$ is called *model likelihood*.

We assume no prior preferences between our models (prior model probabilities are equal). Then the model likelihoods yield posterior model probabilities as

$$p(M_i \mid Y) = p(Y \mid M_i) / \sum_{k=1}^{M} p(Y \mid M_k)$$

The model with greater likelihood value is declared to have better performance.

The integral (2) is analytically tractable in only certain restricted problems and sampling based methods must be used to obtain estimates of the model likelihoods. We chose the bridge sampling technique for model likelihood computation (see, e.g., [7-8], [10] for the related information). As an input for the bridge sampling algorithm we used the samples from parameter posterior distributions, obtained with the help of MCMC simulations.

3.2. MCMC estimation

3.2.1. Priors

Because of the difficulty in interpreting the parameters for the neural network models we adopt a hierarchical prior structure ([5]) that enables us to treat the priors' parameters (hyperparameters) as random variables drawn from suitable distributions (hyperpriors). A convenient form for these hyperpriors is vague inverse Gamma distribution with some fixed shape and mean parameters.

To guarantee the positivity in the variance equation we worked with the logarithmic transformation of the parameters $(\alpha_0, \alpha_1, \beta_1, v_1, \ldots, v_H)$.

Thus, we adopt the following prior structure:

- N(0,10) for the mean parameters a_0 and a_1;
- $\log\mathrm{N}(\kappa_j, 1/\tau_j)$, $j = 1,2,3$, for three linear variance parameters $(\alpha_0, \alpha_1, \beta_1)$;
- $\mathrm{N}(0, 1/\tau_j)$, $j = 4,5,6$, for the input-hidden weights (w, γ) and biases c;
- $\log\mathrm{N}(0, 1/\tau_7)$ for the hidden-output weights v;
- the hyperpriors $\tau_j \sim \mathrm{Ga}(\varsigma_j, \omega_j)$, $j = 1,\ldots,7$.

Note that although GARCH parameters are rather tractable we used hierarchical prior structure for them as well in order to have more variability and apply universal approach over both models.

After preliminary tuning, we fixed the hyperprior shape parameters $\varsigma_j = 10$ for all groups of hyperparameters. The means $\omega_{1:7}$ of the hyperprior, controlling the width of the prior, were chosen to reach the maximal posterior model probability. The priors' centers κ were fixed reflecting our idea about GARCH parameters.

3.2.2. MCMC posterior simulation

Bayesian inference about unknown values are made via simulations from full conditional distributions $p(\theta \mid Y, \tau)$ and $p(\tau \mid Y, \theta)$ (θ denotes all unknown model parameters, τ - all hyperparameters).

Because of the conjugate hyperpriors' form, we obtain the posterior distribution $p(\tau \mid Y, \theta)$ to be again Gamma distribution with transformed shape and mean.

The autoregressive structure of the variance equation results in no property of conjugacy for all model parameters. To sample from the posterior $p(\theta \mid Y, \tau)$ we applied the random walk Metropolis algorithm with Student t-distribution as a proposal for the mean parameters and with Gaussian proposal for the variance parameters, where the variances of these proposal distributions were tuned to come near "optimal" acceptance rate in the range 20-40%. After initial exploratory runs it was checked for the correlation between the parameters and the blocking update of highly correlated parameters was implemented to improve the convergence of simulations.

Finally, we used the resulting posterior output as an input to the bridge sampling algorithm ([10]) to compute model likelihood.

4. Illustration

In the empirical illustration we use daily closing values of the Japan index NIKKEI 225. The time interval taken for all data sets was 16 years from January, 1986, to December, 2001. All data were transformed into continuously compounded returns (in percent) in the standard way by the natural logarithm of the ratio of consecutive daily closing levels.

4.1. Identification of hidden units in MLP-model

One of the most important issues for neural network models with $H \geq 2$ is their unidentifiability due to the invariance to relabeling the hidden units. To cope with this problem, we follow the approach by [8] for mixture models: during posterior simulations we performed random permutation of the hidden units' weights and then by constructing scatter plots of MCMC output we checked for the possible identification and the identification conditions. This helps to avoid multimodality in the posterior due to labeling of the units.

We want to note that our approach in dealing with the NNs identification problem differs from the one known in literature (see [4]), where an identification condition is defined before observing posteriors. Such formal identifiability introduces an obvious bias toward an NN structure with assumed number of hidden units even if the true number of units is smaller. Moreover, if the number of units is correct, clear separability of all groups of parameters is not always the case.

The posterior scatter plots for the MLP-model with two hidden units ($H = 2$) are presented in Fig.1:

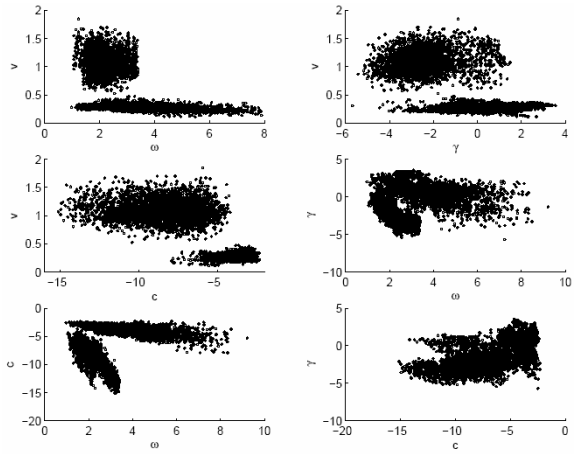

Fig. 1: Posterior paired plots of the non-linear parameters of the MLP-model with H=2.

The plots show two separated nodes with respect to the hidden-output parameter v. To remove multimodality, we reorder the posterior output according to the condition $v_1 < v_2$. The marginal posterior plots of the non-linear parameters before and after identification are given in Fig.2.

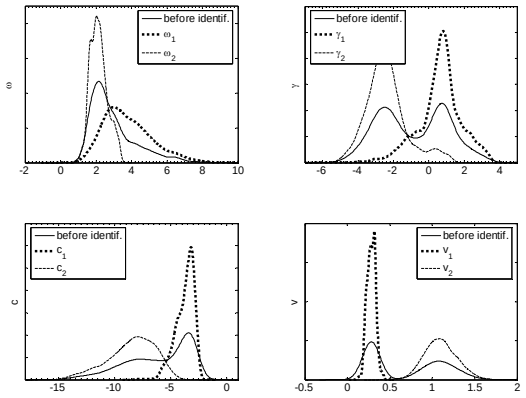

Fig. 2: Posterior distribution of the non-linear parameters of the MLP-model (H=2) before and after identification based on the condition $v_1 < v_2$. The solid line denotes the posterior before applying the identification condition. The dotted and dashed lines correspond to the parameters of the first and second hidden units, respectively, after identification.

Note that it is hard to identify the nodes based on the other weights' constraints. The reordering of the output chain with respect to, e.g., w-weights would not remove multimodality due to units' labeling.

When we increase the number of hidden units in the MLP-model and take $H = 3$, the posterior paired

plots give no identification for the third hidden unit and in the posterior we see only two modes (Fig.3).

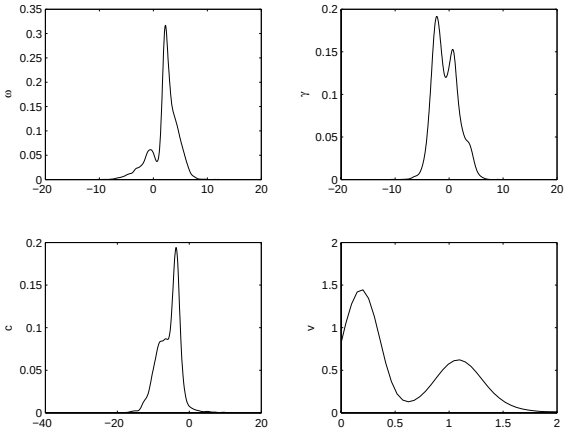

Fig. 3: Posterior distribution of the non-linear parameters for the MLP-model with $H = 3$.

It seems that in this case we deal with node duplication and two hidden units are enough to describe the non-linearity being in the volatility of the NIKKEI 225.

4.2. Model Selection

Now we can test for the optimal size of neural network fitted to the market data. We compute the model likelihoods (ML) according to (2) and posterior model probabilities (MP) exhibited by the linear GARCH and the non-linear MLP-model for three cases H=1,2 and 3.

Model	*Log ML*	*MP*
GARCH(1,1)	-6809.8	0.0000
$MLP_{H=1}$-model	-6794.5	0.0034
$MLP_{H=2}$-model	**-6788.9**	**0.9137**
$MLP_{H=3}$-model	-6791.3	0.0829

We see clear preference for non-linearity in the NIKKEI 225 return series. The posterior model probability of the linear GARCH model is close to 0%. Introducing one hidden unit leads to a significant improvement in the model performance. The MLP-model with two hidden units dominates with the probability of 91%. But the model with three hidden units becomes over-complex and is punished by a smaller model evidence.

5. Conclusions

We applied the full Bayesian procedure to the autoregressive NN volatility models, including hierarchical prior specification, MCMC posterior simulation and model evidence computation. We proposed a new methodology to identify hidden units, consisting of random permutation of units to increase MCMC efficiency and identification a posteriori. To treat NN model complexity, we performed Bayesian model selection based on model evidences.

The application of the proposed methodology to market data showed its reliability. In the econometric context, we get the strong support for non-linearity in volatility process modeled by MLP with two hidden units. Adding one more hidden unit leads to overparametrization and worse generalization.

Future work will deal with extension of empirical study of linear versus non-linear volatility models in two directions: under different assumptions about returns' conditional distribution and for different financial data.

6. References

[1] C. Schittenkopf, G. Dorffner, and E. Dockner "Forecasting time-dependent conditional densities: a seminonparametric neural network approach," *Journal of Forecasting*, 19, pp. 355-374, 2000.

[2] C. Andrieu, N. de Freitas, and A. Doucet, "Robust full Bayesian methods for neural networks," *Advances in Neural Information Processing Systems*, v. 12, pp. 379-385, 2000.

[3] C. Bishop, "Neural Networks for Pattern Recognition," Clarendon Press, Oxford, 1995.

[4] P. Müller, and D.R. Insua, "Issues in Bayesian analysis of neural network models," *Neural Computation*, 10, pp. 571-592, 1998.

[5] R. Neal, "Bayesian Learning for Neural Networks," Springer, 1996.

[6] A. Vehtari, and J. Lampinen, "Bayesian model assessment and comparison using crossvalidation predictive densities," *Neural Computation*, 14, pp. 2439-2468, 2002.

[7] X. Meng, and W. Wong, "Simulating ratios of normalizing constants via a simple identity," *Statistical Sinica*, 6, pp. 831-860, 1996.

[8] S. Frühwirth-Schnatter, "MCMC estimation of classical and dynamic switching and mixture models," .*Journal of the American Statistical Association*, 96, pp. 194-209, 2001.

[9] T. Bollerslev, "A generalized autoregressive conditional heteroskedasticity," *Journal of Econometrics*, 31, pp. 307-327, 1986.

[10] T. Miazhynskaia, and G. Dorffner, "A comparison of Bayesian model selection based on MCMC with an application to GARCH-type models," *Statistical Papers*, to appear, 2005