# Implementation of a Neural Stochastic Volatility Model [4]

**Master's Memoir in Mathematics**

**Jacques Le Thuaut**[1]
**University of Franche-Comté**
**Under the supervision of Prof. Clement Dombry**[2]

**Abstract**

The paper introduces a new volatility model that integrates deep recurrent neural networks with statistical models to formulate a stochastic volatility model for financial time series. The proposed model uses a pair of complementary stochastic recurrent neural networks: a generative network that models the joint distribution of the volatility process and an inference network that approximates the conditional distribution of the latent variables given the observations. The model aims to improve volatility estimation and prediction over traditional deterministic and stochastic models.

## Acknowledgments

I would like to express my sincere gratitude to Prof. Clement Dombry for providing me the opportunity to delve into this new volatility tool. It was a challenging and truly interesting endeavor that significantly enriched my understanding of financial models and neural networks.

## Introduction

Volatility, which reflects the degree of variation in financial markets, is critical for investment decisions, risk management, security valuation, and monetary policy. Traditional volatility models like GARCH and stochastic volatility (SV) models have limitations due to their strong assumptions and difficulty in handling complex dependencies. This paper proposes a data-driven approach using neural networks to model volatility without heavy reliance on exogenous inputs.

## Related Work

The paper reviews existing volatility models, including deterministic models like ARCH and GARCH, and stochastic models such as the Heston model and MCMC-based frameworks. It highlights the limitations of these models and the potential of deep learning, particularly recurrent neural networks (RNNs), to address these limitations by modeling complex dependencies and nonlinear dynamics.

[1]Email: jacques.lethuaut@gmail.com

[2]Email: clement.dombry@univ-fcomte.fr

### Deterministic Models: ARCH and GARCH

The Autoregressive Conditional Heteroskedasticity (ARCH) model, introduced by Engle (1982), and its generalization, the Generalized Autoregressive Conditional Heteroskedasticity (GARCH) model by Bollerslev (1986), are widely used for modeling financial time series. These models assume that the current volatility is a function of past squared observations and past variances.

**ARCH Model** The ARCH model is defined as:

$$y_t = \sigma_t \epsilon_t, \quad \epsilon_t \sim N(0,1) \tag{1}$$

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i y_{t-i}^2 \tag{2}$$

where $y_t$ is the return at time $t$, $\sigma_t^2$ is the conditional variance, $\alpha_0$ and $\alpha_i$ are parameters, and $\epsilon_t$ is a white noise error term.

**GARCH Model** The GARCH model generalizes the ARCH model by including past conditional variances:

$$y_t = \sigma_t \epsilon_t, \quad \epsilon_t \sim N(0,1) \tag{3}$$

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i y_{t-i}^2 + \sum_{j=1}^{p} \beta_j \sigma_{t-j}^2 \tag{4}$$

where $\beta_j$ are parameters for past conditional variances.

### Stochastic Models: Heston Model and MCMC-Based Frameworks

Stochastic volatility models, such as the Heston model (Heston, 1993), introduce randomness in the volatility process itself, allowing for more flexibility. The Heston model assumes that the variance of the asset return follows a mean-reverting square-root process:

$$dV_t = \kappa(\theta - V_t)dt + \sigma\sqrt{V_t}dW_t \tag{5}$$

where $V_t$ is the variance at time $t$, $\kappa$ is the rate of mean reversion, $\theta$ is the long-term variance, $\sigma$ is the volatility of volatility, and $W_t$ is a Wiener process.

**MCMC-Based Frameworks** Markov Chain Monte Carlo (MCMC) methods are used to estimate the parameters of stochastic volatility models. MCMC methods generate samples from the posterior distribution of the parameters by constructing a Markov chain that has the desired distribution as its equilibrium distribution. The most commonly used MCMC method is the Metropolis-Hastings algorithm.

The Metropolis-Hastings algorithm works as follows:

1. Start with an initial value $\theta^{(0)}$ for the parameters.
2. At iteration $t$, generate a proposal $\theta^*$ from a proposal distribution $q(\theta^*|\theta^{(t-1)})$.
3. Calculate the acceptance probability:

$$\alpha = \min\left(1, \frac{p(\theta^*|y)q(\theta^{(t-1)}|\theta^*)}{p(\theta^{(t-1)}|y)q(\theta^*|\theta^{(t-1)})}\right) \tag{6}$$

4. Accept the proposal with probability $\alpha$; otherwise, keep the current value:

$$\theta^{(t)} = \begin{cases} \theta^* & \text{with probability } \alpha \\ \theta^{(t-1)} & \text{with probability } 1-\alpha \end{cases} \tag{7}$$

5. Repeat steps 2-4 for a large number of iterations.

MCMC methods are powerful but computationally intensive, and they may require careful tuning of the proposal distribution and the number of iterations to achieve good convergence.

## Limitations of Traditional Models

Both deterministic and stochastic models have limitations. Deterministic models like GARCH often fail to capture the long-range dependencies and nonlinearities in financial time series. Stochastic models, while more flexible, are computationally demanding and may still rely on unrealistic assumptions. Moreover, both types of models may struggle to adapt to structural changes in the market.

## Potential of Deep Learning

Deep learning, particularly RNNs, offers a powerful alternative by directly learning the dependencies and patterns in the data without stringent assumptions. RNNs can model complex, nonlinear relationships and capture long-range dependencies, making them well-suited for volatility modeling. The integration of neural networks with traditional statistical methods can lead to more accurate and robust volatility forecasts.

Recurrent Neural Networks (RNNs) are a class of neural networks designed to recognize patterns in sequences of data. An RNN maintains a hidden state that captures information about previous time steps, allowing it to model temporal dependencies. The mathematical formulation of an RNN is as follows:

For a given input sequence $\{x_t\}$, the hidden state $h_t$ and the output $y_t$ at each time step $t$ are updated as:

$$h_t = \sigma(W_h x_t + U_h h_{t-1} + b_h) \tag{8}$$

$$y_t = \sigma(W_y h_t + b_y) \tag{9}$$

where:

- $x_t$ is the input at time step $t$,
- $h_t$ is the hidden state at time step $t$,
- $y_t$ is the output at time step $t$,
- $W_h$, $U_h$, and $W_y$ are weight matrices,
- $b_h$ and $b_y$ are bias vectors,
- $\sigma$ is an activation function (such as the hyperbolic tangent $\tanh$ or the logistic sigmoid).

RNNs can be enhanced with Long Short-Term Memory (LSTM) cells or Gated Recurrent Units (GRUs) to better capture long-range dependencies and alleviate issues like vanishing gradients. The LSTM cell introduces gates to control the flow of information:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad \text{(forget gate)} \tag{10}$$
$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad \text{(input gate)} \tag{11}$$
$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad \text{(output gate)} \tag{12}$$
$$\tilde{c_t} = \tanh(W_c x_t + U_c h_{t-1} + b_c) \quad \text{(cell input)} \tag{13}$$
$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c_t} \quad \text{(cell state)} \tag{14}$$
$$h_t = o_t \odot \tanh(c_t) \quad \text{(hidden state)} \tag{15}$$

where $f_t$, $i_t$, $o_t$, and $c_t$ are the forget, input, output gates, and cell state, respectively, and $\odot$ denotes element-wise multiplication.

By leveraging these architectures, RNNs and their variants are capable of modeling the intricate dynamics of financial time series data, providing a robust tool for volatility forecasting.

## Neural Stochastic Volatility Models

The proposed model consists of:

- **Generative Network**: Uses RNNs to model the joint distribution of observable and latent variables. It predicts the next state of the system based on past observations and latent variables.
- **Inference Network**: Uses bidirectional RNNs to approximate the conditional distribution of latent variables given the observable data. This network helps infer the latent variables that explain the observed data.

The model leverages variational inference to handle the intractability of exact inference in the posterior distribution of latent variables.

## Training and Inference

The training process involves maximizing the evidence lower bound (ELBO) to learn the model parameters. The inference network provides an approximation to the true posterior distribution, facilitating efficient training. The model also supports recursive forecasting, enabling multi-step predictions.

## Experiments

The paper conducts experiments on real-world stock price datasets, comparing the proposed model's performance against various deterministic and stochastic models. The results show that the proposed model achieves higher accuracy in volatility estimation and prediction, validated through lower average negative log-likelihood.

## Conclusion

The paper presents a neural stochastic volatility model that integrates RNNs with statistical models for improved volatility estimation and prediction. The model demonstrates superior performance over traditional models, offering a flexible and powerful framework for financial time series analysis. Future work will explore extending the model to handle non-Gaussian residual distributions and other complex scenarios.

## Key Contributions

- Introduction of a neural stochastic volatility model leveraging RNNs.
- Use of variational inference for efficient training and inference.
- Demonstration of improved volatility estimation and prediction on real-world datasets.

## References

[1] Engle, R. F. 1982. Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation. *Econometrica*, 50(4): 987-1007.

[2] Bollerslev, T. 1986. Generalized Autoregressive Conditional Heteroskedasticity. *Journal of Econometrics*, 31(3): 307-327.

[3] Heston, S. L. 1993. A Closed-Form Solution for Options with Stochastic Volatility with Applications to Bond and Currency Options. *The Review of Financial Studies*, 6(2): 327-343.

[4] Luo, R.; Zhang, W.; Xu, X.; and Wang, J. 2018. A Neural Stochastic Volatility Model. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1).