Clustering and Classification in Option Pricing

NIKOLA GRADOJEVIC[*][†]

Lakehead University, The Rimini Center for Economic Analysis

DRAGAN KUKOLJ[‡][§]

University of Novi Sad

RAMAZAN GENÇAY[¶][‖]

Simon Fraser University, The Rimini Center for Economic Analysis

This paper reviews the recent option pricing literature and investigates how clustering and classification can assist option pricing models. Specifically, we consider non-parametric modular neural network (MNN) models to price the S&P-500 European call options. The focus is on decomposing and classifying options data into a number of sub-models across moneyness and maturity ranges that are processed individually. The fuzzy learning vector quantization (FLVQ) algorithm we propose generates decision regions (i.e., option classes) divided by 'intelligent' classification boundaries. Such an approach improves generalization properties of the MNN model and thereby increases its pricing accuracy.

Keywords: Option Pricing, Clustering, Parametric Methods, Non-parametric Methods, Fuzzy Logic

JEL Classifications: C45, G12

1 Introduction

It has been frequently documented that stock returns exhibit non-normality, stochastic volatility and jumps, which contrasts with standard assumptions of the Black and Scholes (1973) option

pricing model. For instance, Bakshi et al. (1997) report pricing biases for out-of-the-money put and call options, which are referred to as the "volatility smile." A related moneyness bias is the "volatility smirk," which represents a volatility smile that is skewed towards one side. In each of these biases of the Black-Scholes model, the implied volatility varies across the range of strike prices of otherwise identical options, thereby violating the model assumption of constant volatility. Less frequently noted pricing biases include biases across maturities. Black (1975) reports that options with less than three months to expiration tend to be overpriced by the Black-Scholes formula. On the other hand, Bakshi et al. (1997) find that out-of-sample pricing errors for the Black-Scholes model increase sharply with maturity, regardless of the moneyness. This phenomenon is more pronounced for the deepest out-of-the-money options.

To alleviate the biases of the Black-Scholes model, two strands of research have emerged in empirical option pricing literature: parametric and non-parametric models.[1] Bakshi et al. (1997) and Gençay and Gibson (2009) demonstrate the superiority of the stochastic volatility (SV), stochastic volatility random jump (SVJ) and stochastic interest rate (SI) parametric models over the Black-Scholes model in out-of-sample pricing and hedging exercises. More specifically, the evidence indicates that the SV characterization is of first-order importance in improving upon the Black-Scholes model. Extending the SV model to the SVJ model further improves the pricing of short term options. Finally, the SI feature improves the pricing and hedging of long-term options. However, the pricing improvements produced by these parametric models are not robust and exhibit some moneyness-related biases for short-term options. Therefore, the challenge of how to deal with the smile effects for short-term options still remains. Despite the theoretical appeal of parametric models, non-parametric models have been more effective in relaxing the assumptions of the Black-Scholes model (Gençay and Gibson, 2009; Gradojevic et al., 2009). For this reason, the largest part of this article focuses on non-parametric aspects of option pricing, sometimes called data-driven approaches.

The superiority of non-parametric models can be explained by their adaptive learning capabilities (e.g., artificial neural networks) and the fact that they do not constrain the distribution of the underlying returns. In particular, non-parametric models involve flexible functional forms that have the advantages of effectively dealing with jumps, non-stationarity, and negative skewness and kurtosis relative to their parametric counterparts. It is worthwhile to mention that non-parametric methods are generally based on a trade-off between smoothness and goodness-of-fit. This trade-off is usually controlled by the choice of a parameter in the estimation procedure, which is a difficult task. It may result in the lack of stability that is detrimental to the out-of-sample performance of non-parametric methods. This is the key reason why one may prefer a parsimonious parametric model. In addition, parametric models exhibit parametric transparency that non-parametric ('black-box') models are typically lacking. Notwithstanding the validity of this view, parametric models may also suffer from similar drawbacks. For exam-

---

[1]For some excellent recent surveys of option pricing literature see Garcia et al. (2010) and Renault (2010).

ple, certain parametric specifications of SV or SVJ may not be easily interpreted.

Overall, with respect to model implementation and interpretation, both approaches pose substantial difficulties. First, finding the initial values for local non-linear estimators of the SV and SVJ models emerges as a challenging task. The problem is that the optimization routines often arrive at inappropriate local solutions. Without substantial amount of time spent studying the data, ensuring convergence becomes a daunting process. Studies by Bakshi et al. (1997) and Gençay and Gibson (2009) point out these important issues that are frequently overlooked in the literature. Similarly, the pricing performance of non-parametric models such as neural networks (NN) heavily depends on the choice of the initial parameter values. In addition, the model structure, the types of non-linearities used in model's computational elements (nodes) and the selection of an appropriate estimation (training) set are equally important modeling decisions that have to be made. Without sufficient care, overfitting or data snooping biases might arise and deteriorate the out-of-sample performance or result in false inference.

It is also noteworthy that by relaxing the assumption of log-normality of the underlying price process or other assumptions of the Black-Scholes model, one can produce different parametric variants of the benchmark model.[2] This offers a possibility of exploiting more general cumulative distributions as well as more flexible functional forms. Some important papers along these lines are the mixture of distributions model by Melick and Thomas (1997) and the semi-parametric estimator by Aït-Sahalia and Lo (1998). These models have been shown to allow sizable improvements in option pricing accuracy compared to the Black-Scholes model, but Gradojevic et al. (2009) reveal their out-of-sample pricing inferiority against non-parametric modular neural network (MNN) models. Some recent non-parametric approaches also include the affine jump-diffusion models (Carr and Wu, 2004) and the normal inverse Gaussian models (Erikkson et al., 2009). Finally, Wu (2005) argues that fuzzy logic can be very useful for option pricing. Similar approaches can be found in Agliardi and Agliardi (2009) and Yoshida (2003).

The goal of this paper is, first, to review the relevant literature on empirical parametric and non-parametric option pricing. Sections 2 and 3 are dedicated, respectively, to those modeling strands. In Section 4, we present the fuzzy learning vector quantization (FLVQ) algorithm that is used to improve the pricing performance of the MNN model (Gradojevic et al., 2009). Specifically, as Gradojevic et al. (2009) rely on ad-hoc option data classification from Garcia and Gençay (2000), we refine the classification through the FLVQ algorithm that combines a Kohonen unsupervised NN and fuzzy logic c-means clustering. In contrast to the pre-set boundaries between clusters (i.e., modules), this methodology produces a set of 'intelligent' boundaries and improves data classification in each sample year (1987-1993). In turn, these improvements enhance the learning capabilities of the NNs in MNN's modules and result in an increased pricing accuracy. Section 5 discusses in detail the results of our pricing exercises.

---

[2]For example, Bates (2000), Bakshi et al. (1997) and Pan (2002) utilize stochastic volatility and jump diffusion processes.

Section 6 concludes and suggests directions for future research.

2 Parametric Option Pricing Models

In a risk neutral economy, the underlying non-dividend-paying stock price $S(t)$ and its components are, for any t, given by (Bakshi et al., 1997):

$$\frac{dS(t)}{S(t)} = [R(t) - \lambda\mu_J] + \sqrt{V(t)}dw_S(t) + J(t)dq(t) \tag{1}$$

$$dV(t) = [\theta_v - \kappa_v V(t)]dt + \sigma_v \sqrt{V(t)}dw_v(t) \tag{2}$$

$$\ln[1 + J(t)] \sim N(\ln[1 + \mu_J] - 1/2 \, \sigma_J^2, \, \sigma_J^2) \tag{3}$$

where $R(t)$ is the time-t instantaneous spot interest rate; λ is the frequency of jumps per year; $V(t)$ is the diffusion component of stock return variance (conditional on no jump occuring); $w_s(t)$ and $w_v(t)$ are each a standard Brownian motion, with $\text{Cov}_t[dw_S(t), dw_v(t)] \equiv \rho dt$; $J(s)$ is the percentage jump size (conditional on a jump occuring) that is lognormally, identically, and independently distributed over time, with unconditional mean μ_J. The standard deviation of $\ln[1 + J(t)]$ is σ_J, $q(t)$ is a Poisson jump counter with intensity λ where $\Pr(dq(t) = 1) = \lambda dt$ and $\Pr(dq(t) = 0) = 1 - \lambda dt$. κ_v, θ_v/κ_v and σ_v are respectively the speed of adjustment, long-run mean, and variation coefficient of the diffusion volatility $V(t)$. $q(t)$ and $J(t)$ are uncorrelated with each other or with $w_s(t)$ and $w_v(t)$.

Under the assumed framework, the total return variance can be decomposed into two components

$$\frac{1}{dt}\text{Var}_t\left(\frac{dS(t)}{S(t)}\right) = V(t) + V_J(t) \tag{4}$$

where $V_J(t) \equiv (1/dt)\text{Var}_t[J(t)dq(t)] = \lambda[\mu_J^2 + (e^{\sigma_J^2} - 1)(1 + \mu_J)^2]$ is the instantaneous variance of the jump component. The discounting of future cash flows is according to the single-factor term structure model (Cox et al., 1985)

$$dR(t) = [\theta_R - \kappa_R R(t)]dt + \sigma_R \sqrt{R(t)}dw_R(t) \tag{5}$$

where κ_R, θ_R/κ_R and σ_R are respectively the speed of adjustment, long-run mean, and variation coefficient of the $R(t)$ process. $w_R(t)$ is the standard Brownian motion, uncorrelated with any other process in the model.

For a European call option written on the stock with strike price K and term-to-expiration τ, its time-t price $C(t, \tau)$ must solve

$$\frac{1}{2}VS^2\frac{\partial^2 C}{\partial S^2} \quad + \quad [R - \lambda\mu_J]S\frac{\partial C}{\partial S} + \rho\sigma_v VS\frac{\partial^2 C}{\partial S\partial V} + \frac{1}{2}\sigma_v^2 V\frac{\partial^2 C}{\partial V^2} + [\theta_v - \kappa_v V]\frac{\partial C}{\partial V}$$

$$+ \quad \frac{1}{2}\sigma_R^2 R\frac{\partial^2 C}{\partial R^2} + [\theta_R - \kappa_R R]\frac{\partial C}{\partial R} - \frac{\partial C}{\partial \tau} - RC \tag{6}$$

$$+ \quad \lambda E\{C(t, \tau, S(1 + J), R, V) - C(t, \tau, S, R, V)\} = 0.$$

subject to $C(t + \tau, 0) = \max\{S(t + \tau) - K, 0\}$. Bakshi et al. (1997) have shown that

$$C(t, \tau) = S(t)\Pi_1(t, \tau; S, R, V) - KB(t, \tau)\Pi_2(t, \tau; S, R, V) \tag{7}$$

where $B(t, \tau)$ is the current price of a zero-coupon bond that pays \$1 in τ periods from time t. The risk-neutral probabilities, Π_1 and Π_2, are recovered from inverting the respective characteristic functions:

$\Pi_j(t, \tau; S(t), R(t), V(t))$

$$= \frac{1}{2} + \frac{1}{\pi}\int_0^\infty Re\left[\frac{e^{-i\phi\ln[K]}f_j(t, \tau, S(t), R(t), V(t); \phi)}{i\phi}\right]d\phi \tag{8}$$

for $j = 1, 2$ with the characteristic functions f_j defined in the Appendix.

The European option pricing model in Equation 7 encompasses several of the most studied cases. For example, the Black-Scholes model is obtained when $\lambda = \theta_R = \kappa_R = \sigma_R = \theta_v = \kappa_v = \sigma_v = 0$; the stochastic volatility (SV) model when $\lambda = \theta_R = \kappa_R = \sigma_R = 0$; the stochastic interest rate (SIR) model when $\lambda = \theta_v = \kappa_v = \sigma_v = 0$; the stochastic volatility jump (SVJ) model when $\theta_R = \kappa_R = \sigma_R = 0$ and stochastic volatility & stochastic interest rate model (SVSI) when $\lambda = 0$.

As Gençay and Gibson (2009) demonstrate that in terms of their out-of-sample pricing performance NN models dominate the above parametric models, in the next section, we turn our attention to non-parametric option pricing. In general, there are two possible explanations for the superiority of NN models: their ability to capture non-normality of return distributions and adaptive learning.

3   Non-parametric Option Pricing

The Black-Scholes formula has been mostly criticized for its distributional assumptions of the underlying security. Non-parametric valuation models are a natural extension, as it is easier to relax the distributional assumptions. A natural non-parametric function for pricing a European call option on a non-dividend paying asset will relate the price of the option to the set of variables which characterize the option

$$C_t = f(S_t, K, \sigma_t, r_t, \tau) \tag{9}$$

where S_t is the price of the underlying asset, K is the strike price, σ_t is the volatility of the underlying asset, r_t is the interest rate and τ is the time-to-maturity.

To facilitate clustering and classification, we reduce our option pricing formula as in Hutchinson et al. (1994) and Garcia and Gençay (2000):

$$C_t = f(S_t, K, \tau) \qquad (10)$$

Assuming the homogeneity of degree one of the pricing function f with respect to S_t and K, one can write the option pricing function as follows:

$$\underbrace{\frac{C_t}{K}}_{c_t} = f(\underbrace{\frac{S_t}{K}}_{x_1}, 1, \underbrace{\tau}_{x_2}) = \phi(x_1, x_2). \qquad (11)$$

The pricing function f is approximated by an MNN with the number of modules and their location determined by the FLVQ algorithm (described in Section 4).

In contrast to parametric models such as the Black-Scholes that assume cumulative normal shapes, non-parametric models such as MNNs make minimal assumptions about the shape or the properties of the underlying price process. In addition, non-parametric pricing functions are convenient for estimating state price densities (SPDs) and its partial derivatives (the "Greeks"). For instance, the MNN estimator can be directly differentiated numerically (or analytically) to extract the SPD or option delta. Since the partial derivatives of an option pricing formula are risk pricing tools, an accurate estimation of the unknown option pricing function is essential for pricing and hedging.

Similar to linear regression models, NNs map a set of explanatory variables (x_i, for example, $i = 1, 2$) into a dependent variable (c_t). The major difference between NNs and other non-parametric estimators is that NNs are based on a layered structure where each layer transforms time series x_i's using special functions known as sigmoid logistic or hyperbolic tangent functions. The number of layers and the structure of each layer have to be determined before estimation, like the bandwidth in kernel regression methodologies. In the case of NNs, this is typically performed through a cross-validation method on the validation part of the data (Garcia and Gençay, 2000). Then, the parameters of an NN are estimated by minimizing the loss function defined as the sum of squared differences between the observed c_t and the one predicted by an NN. The NN-based approaches in option pricing were used by Garcia and Gençay (2000), Hutchinson et al. (1994), Gençay and Gibson (2009), Gradojevic et al. (2009) and Gençay and Altay-Salih (2003).

A popular non-parametric alternative to NNs to estimate the pricing function is kernel regression (Aït-Sahalia and Lo, 1998). In contrast to NNs, where observational errors are reduced by averaging the data based on a recursive error minimization procedure, for kernel regression this is achieved by local averaging (smoothing). The kernel method estimates the price based

on the weighted sum of the information from the in-sample data. More specifically, given some time series of explanatory variables (e.g., x_1 and x_2), to estimate the price, more weight is assigned to the observations at locations that are closer to the vector. For instance, our option pricing model has two regressors and the appropriate kernel pricing function would be constructed as a product of two univariate kernels. The usual trade-off between smoothness and goodness-of-fit is achieved by the choice of the bandwidth of the kernel function. Aït-Sahalia and Lo (1998) show that the Nadaraya-Watson kernel estimator's errors for the SPD and option delta lie within one per cent of the theoretical values. The out-of-sample pricing performance of their model is also impressive and superior over the NN model. It is noteworthy that the kernel pricing function from Aït-Sahalia and Lo (1998) that is used for forecasting is semi-parametric. It is informed by dividend yield, τ, K, S_t, risk-free interest rate, and non-parametrically estimated volatility. In contrast, the variables in the MNN model are the ratio of the asset price to the strike price and the time to maturity. Nevertheless, Gradojevic et al. (2009) find that the MNN model outperforms the kernel estimator from Aït-Sahalia and Lo (1998) in out-of-sample pricing exercises.

A novel approach to option pricing pioneered by Barndorff-Nielsen (1998) and Barndorff-Nielsen and Shephard (2001), and extended by Erikkson et al. (2009), employs the normal inverse Gaussian model to approximate an unknown distribution risk-neutral density. This model's pricing errors are smaller than the errors of several option pricing models with known densities. The benefit of using the normal inverse Gaussian family of distributions is that they are characterized by the first four moments, which are important for derivative pricing. Another relatively recent theoretical research direction is the affine jump-diffusion models. These models rely on the compound Poisson processes to model jumps which may be a finite or an infinite number of jumps within a finite time interval. An advanced application of such models can be found in Carr and Wu (2004). It uses a generalization of the affine Poisson jump diffusions, i.e., time-changed Lévy processes that relax the affine structure and allow more general specifications of the jump structure. The empirical results of applying time-changed Lévy processes to the S&P-500 options signify the role of jumps and stochastic volatility in option pricing (Carr et al., 2003; Huang and Wu, 2004). However, the results also illustrate the importance of choosing the right jump structure and the appropriate stochastic volatility specification. Specifically, a good option pricing model should include a high-frequency jump component and should allow stochastic volatility to be generated separately from the diffusion component and the jump component.

Other non-parametric and semi-parametric estimators that can be employed for option pricing include nearest-neighbour estimators (Yakowitz, 1987), constrained splines (Bates, 2000), average derivative estimators (Hardle and Stoker, 1989), local polynomial regression (Aït-Sahalia and Duarte, 2003), and orthogonal series expansion (Chiarella et al., 1999).

115

4   Fuzzy learning vector quantization (FLVQ)

Fuzzy learning vector quantization (FLVQ) algorithm belongs to the class of batch unsupervised learning methods (Bezdek and Pal, 1995). The FLVQ algorithm represents a combination of the weight adaptation rule used in the Kohonen unsupervised neural network and the fuzzy set membership function proposed by the batch clustering fuzzy c-means (FCM) algorithm. In comparison to similar clustering methods, the main advantages of the FLVQ algorithm are: 1) smaller number of input (user-defined) parameters is required, 2) less frequently trapped in local minima, and 3) final solution is not affected by the order of the input data sequence. The clustering process of the FLVQ algorithm is based on the assumption that each data class may contain several clusters defined by their representative points called *prototypes*. In general, the algorithm contains two phases: the initialization of the prototype coordinates of the clusters (prototype generation) and the improvement of classification error by adapting the prototypes' coordinates (prototype adaptation).

4.1   Prototype generation

The FLVQ is a self-organizational, non-sequential and competitive clustering algorithm. Using the training data, after the initial number of classes and prototypes are defined, for each input observation from the training set, the density of distribution of other input data in its neighborhood is determined from the FCM algorithm.[3] The output of this algorithm represents the coordinates of the data with the highest density, which are adopted as the initial values of the prototypes.

Classes that should be extended with new prototypes are identified in an iterative process. Those classes are considered to have the largest number of misclassified data. In other words, if an input pattern has the nearest prototype from the same class it is correctly classified and vice-versa. In each iteration, a partition matrix U of the current prototypes is calculated by:

$$U = u_{i,j} = \frac{\left(\frac{1}{d(x_j(e),C_i(e))}\right)^{\frac{1}{m(e)-1}}}{\sum_{k=1}^{K}\left(\frac{1}{d(x_j(e),C_k(e))}\right)^{\frac{1}{m(e)-1}}}, \quad i = 1,\ldots,K, \quad j = 1,\ldots,N. \tag{12}$$

where N is the number of observations in the training data, K is total number of prototypes, x_j is the jth input pattern from the training set, C is the matrix of prototypes coordinates, $d(x_j(e),C_k(e))$ is the Euclidean distance between j^{th} input pattern and the kth prototype in each training (iteration) epoch e and the weighting exponent $m(e)$ is called the "degree of fuzziness" in epoch e.

---

[3]We experimented with training data from the first two quarters of each year and the validation data from the third quarter representing the trial testing data. The most accurate pricing for the third quarter data was obtained by using eight classes and thirty prototypes (i.e., on average, about 3-4 prototypes per class). These input parameters are then used in pricing the fourth quarter options (true out-of-sample data).

The degree of fuzziness $m = m(e)$ is monotonically decreasing with respect to the processing time e, i.e., the number of training epochs:

$$m = m_0 - \frac{e(m_0 - m_f)}{e_{max}} \tag{13}$$

where m_0 and m_f are the initial and final values of the degree of fuzziness m, whose values are limited by the heuristically determined constraint: $7 > m_0 > m_f > 1.1$.[4] Parameter e_{max} determines the maximum number of epochs.

The coordinates of prototypes are modified using FLVQ learning rule given by following expression:

$$C_i(e + 1) = \sum_{j=1}^{N} \alpha_{i,j}(e)x_j(e), \quad i = 1, \ldots, K. \tag{14}$$

where $\alpha_{i,j}(e)$ is the learning rate of the ith prototype in epoch e. This learning factor is set by:

$$\alpha_{i,j}(e) = \frac{(u_{i,j}(e))^{m(e)}}{\sum_{j=1}^{N}(u_{i,j}(e))^{m(e)}} \tag{15}$$

Equations 12 - 15 are repeated until the given number of prototypes K is reached.

4.2   Prototype adaptation

This phase of the algorithm involves modifications of the prototypes' coordinates. The key parts of the algorithm are equations 14 and 15, where the elements of the partition matrix U and the coordinates of the prototypes C_i are updated. Based on the distances among all input patterns and the current position of the prototypes, new coordinates of the prototypes are calculated in each iteration. The parameter $m(e)$ and current epoch e determine the prototype's neighborhood, while only patterns belonging to same class are considered. After this step, the total number of prototypes, as well as their number per each class are not changed. In the end, after e_{max} iterations, the best prototype coordinates, i.e., the prototypes with the highest number of correctly classified patterns are recorded and assigned to the classes.

5   Results

5.1   Data

The data are daily S&P-500 index European call option prices taken from the Chicago Board Options Exchange. For each available complete year, over the period from January 1987 to October 1994, options across different strike prices and maturities are considered. Being one

---

[4]Bezdek and Pal (1995) show that this range is appropriate for a monotonically decreasing FLVQ algorithm. More specifically, these limits prevent numerical instability that may arise when values of m are close to 1 or are very large (i.e., when m tends to infinity).

of the deepest and the most efficient option markets in the United States, the S&P-500 index option market is an adequate choice for our investigation. This also makes our study directly comparable to Garcia and Gençay (2000) and Gradojevic et al. (2009).

Options with zero volume are not used in the estimation. For NN estimation purposes, the data for each year are divided into three parts: first two quarters (training set), third quarter (validation set) and fourth quarter (out-of-sample, testing set). This produced the following non-overlapping sub-samples:

- **1987:** Training sample: 3610, Validation sample: 2010, Testing sample: 2239

- **1988:** Training sample: 3434, Validation sample: 1642, Testing sample: 1479

- **1989:** Training sample: 3052, Validation sample: 1565, Testing sample: 1515

- **1990:** Training sample: 3605, Validation sample: 2075, Testing sample: 2166

- **1991:** Training sample: 4481, Validation sample: 1922, Testing sample: 2061

- **1992:** Training sample: 4374, Validation sample: 1922, Testing sample: 1848

- **1993:** Training sample: 4214, Validation sample: 1973, Testing sample: 2030

The optimal NN architecture for each module is determined from the out-of-sample performance on the validation set with respect to the mean-squared prediction error (MSPE). To keep the complexity of the NNs close to Garcia and Gençay (2000), the modules are single hidden layer NNs with either sigmoid logistic or hyperbolic tangent activation functions. Hence, the search for an optimal NN architecture involves specifying the number of hidden nodes (we choose from 1-15 hidden nodes) and their activation functions. The parameters are estimated using the standard Levenberg-Marquardt algorithm. Overfitting is prevented by early stopping, i.e., stopping the training process when the validation set error starts to increase. To control for the possible sensitivity of the NNs to the initial parameter values, the training is performed from ten different random seeds and the average MSPE values are reported. The predictive performance on the testing part of the sample is finally assessed with the MSPE criterion.

The out-of-sample pricing performance of an MNN model is compared to the Black-Scholes model. The Black-Scholes call prices (C_t) are computed using the standard formula:

$$C_t = S_t N(d) - Ke^{-r\tau} N(d - \sigma \sqrt{\tau}) \quad \text{where} \ \ d = \frac{\ln(S_t/K) + (r + 0.5\sigma^2)\tau}{\sigma \sqrt{\tau}} \tag{16}$$

where N is the cumulative normal distribution, S_t is the price of the underlying asset, K is the strike price, τ is the time to maturity, r is the risk-free interest rate, and σ is the volatility of the underlying assets' continuously-compounded returns estimated from the last sixty days

preceding the first day of the last quarter. The risk-free rate is approximated by the monthly yield of the U.S. Treasury bills.

The statistical significance of the prediction performance is assessed with the Diebold-Mariano statistic (Diebold and Mariano, 1995), distributed as standard normal in large samples.[5] When the calculated values of the Diebold-Mariano statistic are large and positive, our model is viewed as being able to significantly improve upon the competing models.

5.2   Pricing performance

		NN with hint	MNN model	MNN-FLVQ model	BS model	DM
$\overline{MSPE}$						
	1987	16.7	4.1204	3.0880	4.38	20.66
	1988	0.7114	0.6107	0.5277	2.07	6.63
	1989	0.4138	0.4008	0.3954	1.42	3.11
	1990	0.6761	0.5579	0.4859	2.62	4.10
	1991	0.3498	0.3293	0.3107	1.73	3.15
	1992	0.1511	0.1285	0.1176	1.36	4.28
	1993	0.1054	0.0546	0.0475	0.74	4.43

Table 1: PREDICTION PERFORMANCE OF THE OPTION PRICING MODELS.

Notes: This table reports the out-of-sample average mean-squared prediction errors ($\overline{MSPE}$) for the following option pricing models: Garcia and Gençay (2000)'s feedforward neural network model with the hint (NN with hint), Gradojevic et al. (2009)'s modular neural network (MNN) model with nine modules, the MNN model with eight modules determined by the FLVQ algorithm (MNN-FLVQ model), and the Black-Scholes model (BS model). The average MSPEs for the MNN models have been obtained as averages across ten different random training seeds. DM denotes the Diebold and Mariano (1995) test statistic. This test is used to assess the statistical significance of the model's forecast gains relative to the NN with hint model. All $\overline{MSPE}$ figures have been multiplied by 10^4.

Table 1 compares the out-of-sample pricing performance of the NN model with the hint (Garcia and Gençay, 2000), the MNN model with nine modules from Gradojevic et al. (2009), the MNN model with eight modules determined by the FLVQ algorithm and the Black-Scholes model (in terms of the MSPE performance measure). The MNN models are estimated ten times from ten different sets of starting values and the average MSPEs are reported. Clearly, both

---

[5]We compute the test statistic as the average of the forecast error differences for the ten estimations of the optimal NN architectures.

MNN models outperform the NN model with the hint as well as the Black-Scholes model in all years. Further, the MNN-FLVQ model represents a robust pricing improvement relative to the basic MNN model from Gradojevic et al. (2009) and it demonstrates that an adequate (i.e., intelligent) classification can benefit option pricing. In general, the pricing improvements over the NN with hint model are statistically significant according to the Diebold-Mariano statistic for all sample years. The improvements are particularly sizeable in 1987 and 1988 (about 25% and 15% reductions in the average MSPE, respectively), which were more volatile than the other years. Such findings emphasize the importance of classification and clustering in risk management.

Next, by comparing the module-by-module performance of the basic MNN model (Gradojevic et al., 2009) and the MNN-FLVQ model, we will investigate the underpinnings of the observed pricing improvements. Figure 1 displays the clusters obtained by classifying the options with the FLVQ algorithm in 1988 and 1993. The optimal classification is based on eight modules (clusters) marked with ordinal numbers 1-8. The FLVQ algorithm classifies the options into the following types: out-of-the-money, short term options (1), near-the-money, short term options (2), in-the-money, short-medium term options (3), out-of-the-money, medium term options (4), near-the-money, medium term options (5), in-the-money, medium-long term options (6), out-of-the-money, long term options (7), and near-the-money, long term options (8). Note that the FLVQ-determined boundaries between the clusters are not given by straight lines. In contrast, the option types selected by the so-called "hard" boundaries in the Gradojevic et al. (2009)'s MNN model are set as follows:[6]

- **Type 1:** Out-of-the-money, short term options: $(S_t/K) < 0.97$ and $\tau < 0.1$;

- **Type 2:** Near-the-money, short term options: $0.97 \leq (S_t/K) \leq 1.05$ and $\tau < 0.1$;

- **Type 3:** In-the-money, short term options: $(S_t/K) > 1.05$ and $\tau < 0.1$;

- **Type 4:** Out-of-the-money, medium term options: $(S_t/K) < 0.97$ and $0.1 \leq \tau \leq 0.2$;

- **Type 5:** Near-the-money, medium term options: $0.97 \leq (S_t/K) \leq 1.05$ and $0.1 \leq \tau \leq 0.2$;

- **Type 6:** In-the-money, medium term options: $(S_t/K) > 1.05$ and $0.1 \leq \tau \leq 0.2$;

- **Type 7:** Out-of-the-money, long term options: $(S_t/K) < 0.97$ and $\tau > 0.2$;

- **Type 8:** Near-the-money, long term options: $0.97 \leq (S_t/K) \leq 1.05$ and $\tau > 0.2$;

- **Type 9:** In-the-money, long term options: $(S_t/K) > 1.05$ and $\tau > 0.2$.

---

[6]The same pre-determined boundaries are used in Garcia and Gençay (2000).

Figure 1: FLVQ option classification
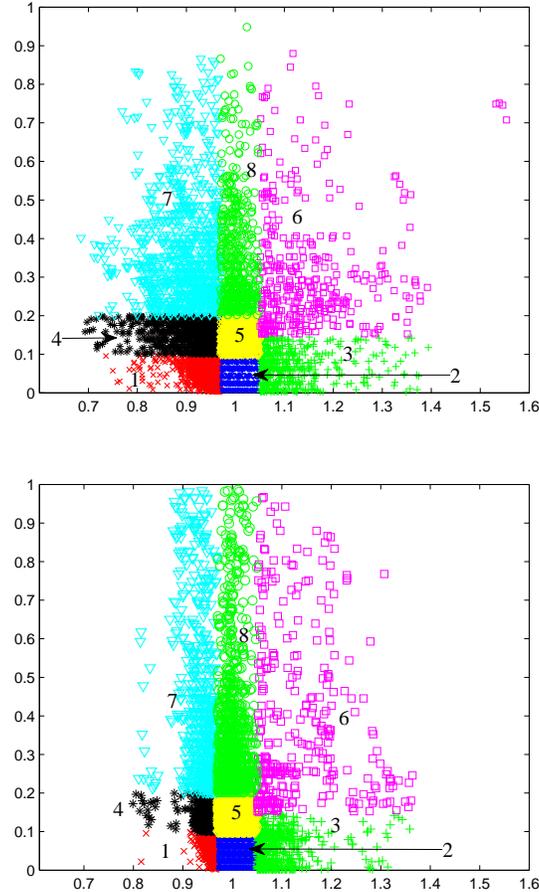
The moneyness (S_t/K) of the S&P-500 call options is plotted on the x-axis and their time-to-expiration (τ) in years on the y-axis. Top panel: Eight modules (clusters; marked with ordinal numbers 1-8) received by the FLVQ classification algorithm of all available call options (3610 observations) in the training sample of 1988. Bottom panel: Eight modules (clusters; marked with ordinal numbers 1-8) received by the FLVQ classification algorithm of all available call options (4214 observations) in the training sample of 1993.

The above results show that pre-determining boundaries deteriorate the MNN model's pricing performance relative to the MNN-FLVQ model. It can be observed that the FLVQ-generated clusters correspond to merging type 3 and a subset of type 6 options into cluster 3. In addition, cluster 6 roughly combines the remaining type 6 and also type 9 options. Table 2 lists the average MSPE figures for all modules of the MNN models across all years. We find that both MNN models produce on average the highest pricing errors for modules 3, 6 , 8 and 9 i.e., for the

Year	Cluster →	1	2	3	4	5	6	7	8	9
	Model:									
1987	MNN	3.53	2.01	5.69	1.15	4.69	6.87	5.17	10.17	8.87
	MNN-FLVQ	1.32	1.91	6.44	2.19	5.03	8.22	2.96	10	
1988	MNN	0.06	0.35	0.35	0.49	1.18	0.68	0.49	1.49	0.56
	MNN-FLVQ	0.11	0.33	0.36	0.10	0.11	0.86	0.10	0.69	
1989	MNN	0.13	0.33	0.58	0.17	0.48	1.85	0.23	0.51	0.42
	MNN-FLVQ	0.05	0.34	0.10	0.17	0.48	0.60	0.19	0.56	
1990	MNN	0.05	0.19	0.47	0.30	0.97	0.76	1.08	0.87	0.70
	MNN-FLVQ	0.09	0.20	0.49	0.42	0.10	0.78	0.55	0.77	
1991	MNN	0.02	0.09	0.54	0.14	0.34	0.39	0.18	0.50	1.53
	MNN-FLVQ	0.02	0.11	0.52	0.14	0.25	1.41	0.14	0.45	
1992	MNN	0.00	0.05	0.17	0.03	0.11	0.20	0.12	0.23	0.46
	MNN-FLVQ	0.00	0.05	0.15	0.04	0.11	0.45	0.07	0.20	
1993	MNN	0.00	0.03	0.10	0.01	0.04	0.19	0.02	0.08	0.12
	MNN-FLVQ	0.00	0.03	0.12	0.01	0.03	0.12	0.03	0.04	
Average	MNN	0.55	0.44	1.13	0.33	1.12	1.57	1.05	1.98	1.81
	MNN-FLVQ	0.23	0.43	1.18	0.44	0.88	1.78	0.58	1.82	

Table 2: PREDICTION PERFORMANCE OF THE MNN MODEL MODULES.

Notes: Each cell of this table contains the out-of-sample average mean-squared prediction errors ($\overline{MSPE}$) for the modules (clusters) of the following option pricing models: Gradojevic et al. (2009)'s modular neural network (MNN) model with nine modules (marked with ordinal numbers 1-9) and the MNN model with eight modules (marked with ordinal numbers 1-8) determined by the FLVQ algorithm (MNN-FLVQ). The MNN-FLVQ model's cluster 3 corresponds to merged clusters 3 and (part of) 6 of the MNN model. The MNN-FLVQ model's cluster 6 corresponds to merged clusters 9 and (part of) 6 of the MNN model. In addition, cluster 6 combines the remaining type 6 and also type 9 options. The average MSPEs for the MNN models have been obtained as averages across ten different random training seeds. All figures have been multiplied by 10^4.

in-the-money (3, 6, 9) and long term (8, 9) options. Overall, near-the-money, long term options appear to be the most difficult to price, while pricing the short term options is the most accurate. The fact that the MSPE figures are the smallest for the out-of-the-money, short term options is generally in accord with Bakshi et al. (1997) and their evidence for the Black-Scholes and the competing parametric models (SVJ, SI and SV). Figure 2 also indicates that the pricing biases increase with both the time to expiration and the moneyness of the options. As noted before, the errors are more pronounced in modules 3 (in-the-money options), 6 and 8 (long term options). This is visible as more scatter in the corresponding panels of Figure 2.[7]

Figure 2: Error plots for the modules in the MNN-FLVQ model (1991)
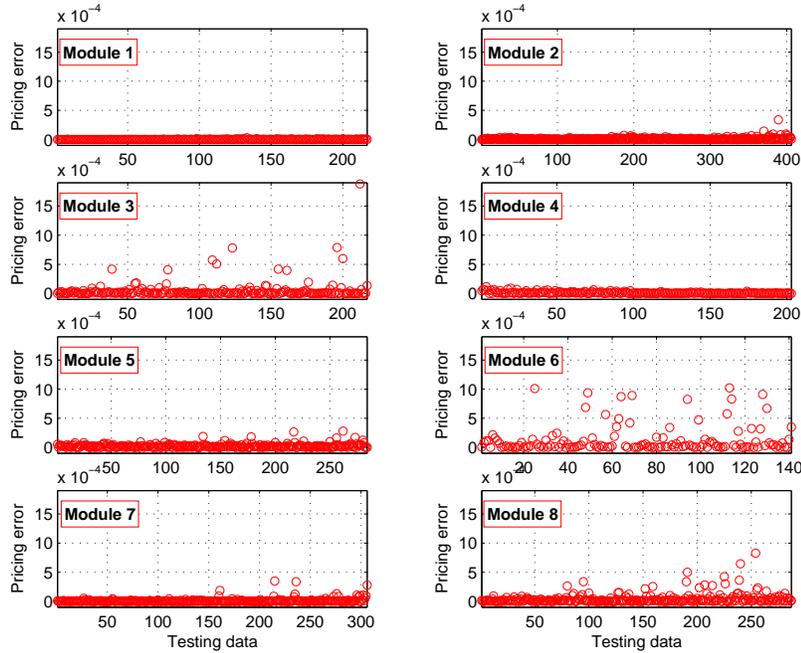
Notes:

Out-of-sample squared errors $((c_t - \hat{c}_t)^2, t = 1, \ldots, 2061)$ for 1991 are plotted for each observation classified in modules 1-8. The observation number of the S&P-500 call options is plotted on the x-axis and out-of-sample squared errors on the y-axis. All call options in 1991 are classified using the FLVQ algorithm into the following types (modules): 217 out-of-the-money, short term options (module 1), 406 near-the-money, short term options (module 2), 217 in-the-money, short-medium term options (module 3), 203 out-of-the-money, medium term options (module 4), 284 near-the-money, medium term options (module 5), 141 in-the-money, medium-long term options (module 6), 306 out-of-the-money, long term options (module 7), and 287 near-the-money, long term options (module 8).

---

[7]The shapes of the plots for all other years resemble the ones for 1991 and can be available upon request from the authors.

The MNN-FLVQ model consistently improves the pricing accuracy of the MNN model in modules 7 and 8, which represent long term options. As long term options are difficult to price and they also represent a good indicator of model's accuracy (Bakshi et al., 2000), this in turn emphasizes the effectiveness of our approach. Module 5 (near-the-money, medium term options) also exhibits substantial pricing improvements, while the large reduction in MSPE in module 1 is driven by 1987, the year of the stock market crash. In all, we attribute the success of the MNN-FLVQ model to its 'intelligent' option classification that results in a better fit of the non-linearities in the NN modules.

6   Conclusions

The goal of this study is to propose a new research direction in option pricing that is based on the clustering and classification of the S&P-500 European call options over the 1987-1993 data span. The method we utilize for such purpose is the FLVQ algorithm that provides sensible and 'intelligent' boundaries between option classes. The optimal classification consists of eight clusters that are used as modules in an MNN model. The resulting model is called the MNN-FLVQ model and it represents a methodological improvement of the Gradojevic et al. (2009)'s MNN architecture. Empirical pricing exercises show that the MNN-FLVQ model outperforms both the standard MNN and the NN with the hint models. As the latter two non-parametric models dominate their parametric counterparts including the Black-Scholes model, the MNN-FLVQ model proves to be an effective option pricing tool. More precisely, the success of the MNN-FLVQ model mainly lies in its accurate classification of long term option. Also, the model exhibits remarkable pricing ability in volatile years - 1987 and 1988.

Our approach can also be understood as a generalization and extension of the methodologies such as the Markov switching model or the threshold regression model. Similarly, the FLVQ algorithm involves splitting and sorting the data based on certain criteria. However, it does not make use of any special threshold or state variables to classify the data, but, rather relies on its clustering (learning) algorithm applied on the input space. In addition, it generates more than two regimes that characterize the relationship between the variables of interest. In this context, our approach could also be applied to other non-linear problems in time series financial econometrics. For example, financial models that are potentially susceptible to regime switching include foreign exchange rate, interest rate and derivative pricing models.

In closing, we conclude that the MNN-FLVQ model is a promising future research avenue in option pricing. Furthermore, the current approach can be extended to the notion of 'soft' boundaries where data points could belong to any cluster with a given probability. An example of this approach are Gaussian mixture models (Behr and Pötter, 2009). This class of models employs all data points in the input space and allows for interaction between clusters when predicting the output variable.

Appendix

The characteristic functions are:

$$
\begin{aligned}
f_1(t,\,\tau) \;=\; & \exp\Bigg\{ -\frac{\theta_R}{\sigma_R^2}\left[2\ln\left(1-\frac{[\xi_R-\kappa_R](1-e^{-\xi R\tau})}{2\xi_R}\right)+[\xi_R-\kappa_R]\tau\right] \\
& -\frac{\theta_v}{\sigma_v^2}\left[2\ln\left(1-\frac{[\xi_v-\kappa_v+(1+i\,\phi)\rho\,\sigma_v](1-e^{-\xi v\tau})}{2\xi_v}\right)\right] \\
& -\frac{\theta_v}{\sigma_v^2}[\xi_v-\kappa_v+(1+i\,\phi)\rho\,\sigma_v]\,\tau+i\,\phi\,\ln[S(t)] \\
& +\frac{2\,i\,\phi\,(1-e^{-\xi R\tau})}{2\xi_R-[\xi_R-\kappa_R](1-e^{-\xi R\tau})}R(t) \\
& +\lambda(1+\mu_J)\tau[(1+\mu_J)^{i\,\phi}e^{(i\,\phi/2)(1+i\,\phi)\sigma_J^2}-1]-\lambda\,i\,\phi\mu_J\,\tau \\
& +\frac{i\,\phi\,(i\,\phi+1)(1-e^{-\xi v\tau})}{2\xi_v-[\xi_v-\kappa_v+(1+i\,\phi)\rho\,\sigma_v](1-e^{-\xi v\tau})}V(t)\Bigg\},
\end{aligned}
\tag{A1}
$$

$$
\begin{aligned}
f_2(t,\,\tau) \;=\; & \exp\Bigg\{ -\frac{\theta_R}{\sigma_R^2}\left[2\ln\left(1-\frac{[\xi_R^*-\kappa_R](1-e^{-\xi^* R\tau})}{2\xi_R^*}\right)+[\xi_R^*-\kappa_R]\tau\right] \\
& -\frac{\theta_v}{\sigma_v^2}\left[2\ln\left(1-\frac{[\xi_v^*-\kappa_v+i\,\phi\rho\,\sigma_v](1-e^{-\xi^* v\tau})}{2\xi_v^*}\right)+[\xi_v^*-\kappa_v+i\,\phi\rho\,\sigma_v]\,\tau\right] \\
& +i\,\phi\,\ln[S(t)]-\ln[B(t,\,\tau)]+\frac{2\,(i\,\phi-1)(1-e^{-\xi^* R\tau})}{2\xi_R^*-[\xi_R^*-\kappa_R](1-e^{-\xi^* R\tau})}R(t) \\
& +\lambda\tau[(1+\mu_J)^{i\,\phi}e^{(i\,\phi/2)(i\,\phi-1)\sigma_J^2}-1]-\lambda\,i\,\phi\mu_J\,\tau \\
& +\frac{i\,\phi\,(i\,\phi-1)(1-e^{-\xi^* v\tau})}{2\xi_v^*-[\xi_v^*-\kappa_v+i\,\phi\rho\,\sigma_v](1-e^{-\xi^* v\tau})}V(t)\Bigg\},
\end{aligned}
\tag{A2}
$$

where

$$
\xi_R = \sqrt{\kappa_R^2-2\,\sigma_R^2\,i\,\phi}, \qquad \xi_v = \sqrt{[\kappa_v-(1+i\,\phi)\rho\,\sigma_v]^2-i\,\phi\,(i\,\phi+1)\sigma_v^2},
$$

$$
\xi_R^* = \sqrt{\kappa_R^2-2\,\sigma_R^2\,(i\,\phi-1)}, \quad \text{and} \quad \xi_v^* = \sqrt{[\kappa_v-i\,\phi\rho\,\sigma_v]^2-i\,\phi\,(i\,\phi-1)\sigma_v^2}.
$$

The stochastic interest rate (SI), the stochastic volatility (SV), stochastic volatility and stochastic interest rate (SVSI) and the stochastic volatility random jump (SVJ) models are all nested within the general formula in Equation 7. In the case of SVJ, $R(t) = R$ is a constant, $B(t, \tau) = e^{-R\tau}$ and the characteristic functions are:

$$\hat{f}_1 = \exp\left\{ -i\,\phi\ln[B(t,\,\tau)] - \frac{\theta_v}{\sigma_v^2}\left[2\ln\left(1 - \frac{[\xi_v - \kappa_v + (1 + i\,\phi)\rho\,\sigma_v](1 - e^{-\xi v\tau})}{2\xi_v}\right)\right]\right.$$

$$-\frac{\theta_v}{\sigma_v^2}[\xi_v - \kappa_v + (1 + i\,\phi)\rho\,\sigma_v]\,\tau + i\,\phi\,\ln[S(t)]$$

$$+\lambda(1 + \mu_J)\tau[(1 + \mu_J)^{i\phi}e^{(i\phi/2)(1+i\phi)\sigma_J^2} - 1] - \lambda\,i\,\phi\,\mu_J\,\tau$$

$$\left. + \frac{i\,\phi\,(i\,\phi + 1)(1 - e^{-\xi v\tau})}{2\xi_v - [\xi_v - \kappa_v + (1 + i\,\phi)\rho\,\sigma_v](1 - e^{-\xi v\tau})}V(t)\right\},$$

(A3)

$$\hat{f}_2 = \exp\left\{ -i\,\phi\ln[B(t,\,\tau)] - \frac{\theta_v}{\sigma_v^2}\left[2\ln\left(1 - \frac{[\xi_v^* - \kappa_v + i\,\phi\rho\,\sigma_v](1 - e^{-\xi^* v\tau})}{2\xi_v^*}\right)\right]\right.$$

$$-\frac{\theta_v}{\sigma_v^2}[\xi_v^* - \kappa_v + +i\,\phi\rho\,\sigma_v]\,\tau + i\,\phi\,\ln[S(t)]$$

$$+\lambda\tau[(1 + \mu_J)^{i\phi}e^{(i\phi/2)(i\phi-1)\sigma_J^2} - 1] - \lambda\,i\,\phi\,\mu_J\,\tau$$

$$\left. + \frac{i\,\phi\,(i\,\phi - 1)(1 - e^{-\xi^* v\tau})}{2\xi_v^* - [\xi_v^* - \kappa_v + i\,\phi\rho\,\sigma_v](1 - e^{-\xi^* v\tau})}V(t)\right\},$$

(A4)

The characteristic functions for the SV model can be obtained by setting $\lambda = 0$ in Equations A3-A4.

References

Agliardi, E. and Agliardi, R. (2009), Fuzzy defaultable bonds, *Fuzzy Sets and Systems* 160(18), 2597–2607.

Aït-Sahalia, Y. and Duarte, J. (2003), Nonparametric Option Pricing Under Shape Restrictions, *Journal of Econometrics* 116, 9–47.

Aït-Sahalia, Y. and Lo, A. (1998), Nonparametric Estimation of State-Price Densities Implicit in Financial Asset Prices, *Journal of Finance* 53, 499–547.

Bakshi, G., Cao, C. and Chen, Z. (1997), Empirical Performance of Alternative Option Pricing Models, *Journal of Finance* 52(5), 2003–2049.

Bakshi, G., Cao, C. and Chen, Z. (2000), Pricing and Hedging Long-Term Options, *Journal of Econometrics* 94, 277–318.

Barndorff-Nielsen, O. (1998), Processes of normal inverse Gaussian type, *Finance and Stochastics* 2, 41–68.

Barndorff-Nielsen, O. and Shephard, N. (2001), Non-Gaussian Ornstein-Uhlenbeck-Based Models and Some of Their Uses in Financial Economics, *Journal of the Royal Statistical Society B* 63, 167–241.

Bates, D. S. (2000), Post-'87 Crash Fears in the S&P 500 Futures Option Market, *Journal of Econometrics* 94(1/2), 181–238.

Behr, A. and Pötter, U. (2009), Alternatives to the normal model of stock returns: Gaussian mixture, generalised logF and generalised hyperbolic models, *Annals of Finance* 5(1), 49–68.

Bezdek, J. and Pal, N. (1995), Two soft relatives of learning vector quantization, *Neural Networks* 8(5), 729–743.

Black, F. (1975), Fact and Fantasy in the Use of Options, *Financial Analysts Journal* 31, 36–41, 61–72.

Black, F. and Scholes, M. (1973), The Pricing of Options and Corporate Liabilities, *Journal of Political Economy* 81, 637–659.

Carr, P. and Wu, L. (2004), Time-changed Lévy processes and option pricing, *Journal of Financial Economics* 71, 113–141.

Carr, P., Geman, H., Madan, D. and Yor, M. (2003), Stochastic Volatility for Lévy Processes, *Mathematical Finance* 13, 345–382.

Chiarella, C., El-Hassana, N. and Kucera, A. (1999), Evaluation of American option prices in a path integral framework using Fourier-Hermite series expansions, *Journal of Economic Dynamics and Control* 23(9-10), 1387–1424.

Cox, J. C., Ingersoll, J. E. and Ross, S. A. (1985), A Theory of the Term Structure of Interest Rates, *Econometrica* 53, 385–407.

Diebold, F. X. and Mariano, R. S. (1995), Comparing Predictive Accuracy, *Journal of Business and Economic Statistics* 13, 253–263.

Erikkson, A., Ghysels, E. and Wang, F. (2009), The normal inverse Gaussian distribution and the pricing of derivatives, *Journal of Derivatives* 16, 23–38.

Garcia, R. and Gençay, R. (2000), Pricing and Hedging Derivative Securities with Neural Networks and a Homogeneity Hint, *Journal of Econometrics* 94(1/2), 93–115.

Garcia, R., Ghysels, E. and Renault, E. (2010), *The Econometrics of Option Pricing*, Vol. 2, Elsevier-North Holland, Amsterdam, pp. 479–552.

Gençay, R. and Altay-Salih, A. (2003), Degree of Mispricing with the Black-Scholes Model and Nonparametric Cures, *Annals of Economics and Finance* 4, 73–101.

Gençay, R. and Gibson, R. (2009), Model Risk for European-Style Stock Index Options, *IEEE Transactions on Neural Networks* 18(1), 193–202.

Gradojevic, N., Gençay, R. and Kukolj, D. (2009), Option Pricing with Modular Neural Networks, *IEEE Transactions on Neural Networks* 20(4), 626–637.

Hardle, W. and Stoker, T. (1989), Investigating Smooth Multiple Regression by the Method of Average Derivatives, *Journal of the American Statistical Association* 84, 986–995.

Huang, J. and Wu, L. (2004), Specification analysis of option pricing models based on time-changed Lévy processes, *Journal of Finance* 59(3), 1405–1440.

Hutchinson, J. M., Lo, A. W. and Poggio, T. (1994), A Nonparametric Approach to Pricing and Hedging Derivative Securities via Learning Networks, *Journal of Finance* 49(3), 851–889.

Melick, W. R. and Thomas, C. P. (1997), Recovering an Asset's Implied PDF from Option Prices: An Application to Crude Oil During the Gulf Crisis, *Journal of Financial and Quantitative Analysis* 32(1), 91–115.

Pan, J. (2002), The Jump-risk Premia Implicit in Options: Evidence from an Integrated Time Series Study, *Journal of Financial Economics* 63, 3–50.

Renault, E. (2010), *Econometrics of Option Pricing*, Vol. 2, Wiley, New York, pp. 518–528.

Wu, H.-C. (2005), European Option Pricing Under Fuzzy Environment, *International Journal of Intelligent Systems* 20(1), 89–102.

Yakowitz, S. J. (1987), Nearest Neighbour Methods for Time Series Analysis, *Journal of Time Series Analysis* 8(2), 235–247.

Yoshida, Y. (2003), The valuation of European options in uncertain environment, *European Journal of Operational Research* 145(1), 221–229.