Can the *wavelet-kernel* methodology improve other kernel techniques?

M. Gago & E. Juaristi
Department of Finance and Economics,
Mondragon University, Spain

Abstract

Our aim in this paper is to compare different ways of forecasting using the wavelet transform and the kernel regression. We consider that working with block (or segment) of data is richer than working with individual data (as in traditional kernel), as we assume there is some kind of pattern inside each block which will improve the estimation, and therefore the prediction. We choose the wavelet transform because this transform is able to separate components of data in different locations and with different location in time and frequency. To test the performance of the different methodologies we have carried out a Monte Carlo Study in which we have compared the four methodologies: Ordinary Least Square (OLS), Traditional Kernel (TK), Block Kernel (BK) and Wavelet-Kernel (WK). Two real life applications have been realized. On the one hand, volatility smile has been forecast and on the other, the rated temperature of the steel coils' furnace is predicted. Surprisingly contradictory results had been obtained.
Keywords: wavelets, Kernel regression, bandwidth selection, implied volatility, option pricing models, steel coils' furnace's rated temperature.

1 Introduction

The wavelet transform is a mathematical tool that is very used in several fields such as engineering, mathematics, physics, economics or finance. It allows us to separate data evolving in time in different frequency-time components. This way, we will be able to identify in data its peaks and discontinuities using high scales components and its long-trend or pattern using the low ones, because as it's known at low scales the wavelet transform has a large time support,whereas at high scales has a small one.

WIT Transactions on Information and Communication Technologies, Vol 40, © 2008 WIT Press
www.witpress.com, ISSN 1743-3517 (on-line)
doi:10.2495/DATA080081

One of the possible applications of wavelets is forecasting time series data. In Antoniadis et al. [1], they consider the prediction problem of a time series on a whole time-interval depending on its past utilizing a non-parametric technique: the kernel estimation. In this paper, we apply this technique but adapted to cross section data.

Like in Antoniadis et al. [1] we consider that working with block (or segment) of data is richer than working with individual data (as in traditional kernel), because we assume that there is some kind of pattern inside each block of data which will improve the estimation and therefore the prediction. Besides, blocks have a time structure, so we can only take into account the previous blocks and in no way future blocks. We use wavelet transform to find similarities between different blocks and apply a kernel-wavelet technique to make the regression.

Other possibility in order to extract the regression curve from the data is directly the application of the kernel nonparametric estimator although we are going to adapt it to the structure of the data we have, the blocks.

Our aim is to compare the predictions obtained via different methods:

1. The ordinary least square estimation (OLS).
2. The *traditional* kernel estimation (TK).
3. The *blocked* kernel estimation(BK).
4. The wavelet-kernel estimation (WK).

Before applying the proposed methodologies to real dataset, we do a Monte Carlo study to present, in an empirical framework, the behavior of the methodologies. Finally, we apply the methodologies to two real life dataset, in particular to

1. 1-day-ahead prediction of volatility depending on moneyness.
2. 1-block-ahead prediction of the rated temperature in the furnace for steel coils.

In the first application, we have all call and put options on the IBEX-35 index futures traded daily on MEFF during the period January 1996 through November 1998. A very important point in the option pricing models is whether the actual distribution of the underlying asset implied by the option market data is consistent with the distribution assumed by the theoretical option pricing model. In an option pricing framework, the main theoretical model is the Black–Scholes (1973) model, see [2], which establishes that all option prices on the same underlying asset with the same expiration date but with different exercise prices should have the same implied volatility. However, the well known volatility smile pattern suggest that the BS formula tends to misprice deep in-the-money and deep out-of-the-money options. There have been various attempts to deal with this apparent failure of the BS valuation model. Our objective is to estimate the implied volatility appeared in the Spanish market with the proposed methodologies.

In the second application, we work with steel coils in galvanized furnaces. Our interest is focused on predicting the rated temperature in the furnace so that the steel coil goes out of the galvanization process with the temperature needed. For that purpose we have data of 100 steel coils with their temperatures at the entrance of the furnace and inside it.

This work is organized as follows. In Section 2 the nonparametric kernel estimator of the unknown regression function is presented. In Section 3 the wavelet method is described. Technical aspects in the two methodologies are explained in Section 4. A simulation based comparison between wavelets and kernels appears in Section 5. Finally, Section 6 presents the applications to option pricing and industrial data and Section 7 concludes.

2 Nonparametric kernel methodology

In this section we describe the general framework in which the nonparametric kernel estimation procedure is applied.

Consider the following data generating process (DGP):

$$Y_i = m(X_i) + \epsilon_i \quad \text{for } i = 1, \ldots, n, \tag{1}$$

where the function $m(\cdot)$ is unknown, the values X_i $(i = 1, \ldots, n)$ are observations of the explanatory variable with density function $f(X)$, and the perturbation errors are assumed to be $i.i.d.$, with zero mean and variance equal to σ_ϵ^2.

Consider a realization $\{x_i, y_i\}_{i=1}^n$ from the DGP described above. For any point x in the domain of the one-dimensional explanatory variable, the general one-dimensional kernel estimator of $m(x)$ can be written as:

$$m(x) = \frac{\sum_{i=1}^n K_h(x - x_i) y_i}{\sum_{i=1}^n K_h(x - x_i)} \tag{2}$$

where h is the bandwidth and K is the univariate kernel or weight function; being, $\int K(x)dx = 1$.

The estimation of $m(\cdot)$ is carried out after choosing the weight function K and the smoothing parameter or bandwidth, h. It is well-known that the selection of the kernel function is not relevant in the estimation (in this paper we have decided to use the most famous one, the gaussian) but as long as the selection of the bandwidth is important, we will explain with more detail in section 4 how we have selected it.

As we said in the introduction, we are interested in extracting the signal from the data and predicting in one block using the information in the previous blocks. So, actually, instead of using the equation 2, we have used another two expressions, explained below.

Let be x_i^J the i^{th} observation of the J^{th} block. We are interested in predicting in every observation in block J, x_i^J, for $i = 1, \ldots, m$ (where m is the number of observations in each block), using the information in all the previous blocks $(p = 1, \ldots, J - 1)$.

We propose two estimators for doing so:

1. In the traditional one (TK), we are going to estimate using all the information in the previous blocks, as follows:

$$m(x_i^J) = \frac{\sum_{p=1}^{J-1}\sum_{k=1}^{m} K\left(\frac{x_i^J - x_k^p}{h}\right) y_k^p}{\sum_{p=1}^{J-1}\sum_{k=1}^{m} K\left(\frac{x_i^J - x_k^p}{h}\right)} \tag{3}$$

2. In the blocked kernel (BK), we are going to compare the data in the block where we want to predict with all the data in each of the previous blocks, as follows:

$$m(x_i^J) = \frac{\sum_{p=1}^{(J-1)} K\left(\frac{x_1^J - x_1^p}{h}\right) K\left(\frac{x_2^J - x_2^p}{h}\right) \cdots K\left(\frac{x_m^J - x_m^p}{h}\right) y_i^p}{\sum_{p=1}^{(J-1)} K\left(\frac{x_1^J - x_1^p}{h}\right) K\left(\frac{x_2^J - x_2^p}{h}\right) \cdots K\left(\frac{x_m^J - x_m^p}{h}\right)} \tag{4}$$

3 Wavelet methodology

The wavelet transform used in this work is the discrete dyadic one. More details about the wavelet transform could be obtained in Chui [3], Daubechies [4] Vidakovic [5] among others.

The wavelet transform is widely used in many fields of economics and finance: non-stationary time series, time-scale decompositions, forecasting, density estimation, etc. Our work is centered in forecasting. There have been different approaches to reduce and predict time series using wavelets. For instance: in Aussem and Murtagh [6], Gonghui et al. [7] and Lotric [8] the wavelet is combined with neural networks; in Cristi and Tummala [9], Hong et al. [10] and Renaud et al. [11] with Kalman filter, in Renaud et al. [12] with autoregressive model and in Antoniadis et al. [1] with kernel methodology, among others.

We are going to focus our attention in the combination between kernel and wavelet for the forecasting problem, as in Antoniadis et al. [1], where the authors use kernel to predict in time series domain.They measure the similarity between intervals by means of the wavelet transform. We have adjusted that technique to be able to apply to cross section data, so we will be doing kernel regression. Below, we will explain this methodology.

According to Antoniadis et al. [1], we consider a continuous stochastic process $\mathbf{X} = \mathbf{X}(t)_{t\in R}$. If the process $\mathbf{X}$ is observed in a closed interval $[0, T]$ our aim is to describe what will happen in the next interval $[T, T+\delta]$, $\delta > 0$; this will be much richer than to get an estimate for a single datum.

In order to predict, a stochastic process is defined associated to each subinterval, $[j\delta, (j+1)\delta]_{j=0,1,\ldots,k-1}$ and $\delta = \frac{T}{k}$ for interval $[0, T]$ as follows:

$$\mathbf{Z}_i(t) = \mathbf{X}(t + (i-1)\delta) \quad i = 1, \ldots, k \quad \forall t \in [0, \delta) \tag{5}$$

This will cover the full range of $[0, T]$.

Thus, the continuous stochastic process $\mathbf{X} = \mathbf{X}(t)_{t\in R}$ is covered by a discrete partition: $\mathbf{Z} = (\mathbf{Z}_i)_{i\in N}$. This representation of the stochastic process is common

in statistics as it facilitates the understanding of how evolves the process $\mathbf{X}$; it is particularly advantageous if $\mathbf{X}$ has a seasonal component with period δ or if it is intended to predict the future behavior of $\mathbf{X}$ on an interval of length δ.

The kernel estimator that we will use is the one defined in equation (2), but in this case both predictor and response variables are discrete time functions, so our variables will be blocks since the above partition is applied to all variables. Thus, the prediction of block $\mathbf{Z}_n^Y$ is obtained via kernel regression on all the previous blocks $\mathbf{Z}_i^Y i = 1, \ldots, n-1$.

$$\mathbf{Z}_n^Y(\cdot) = \sum_{i=1}^{n-1} \omega_{n,i}\, \mathbf{Z}_i^Y(\cdot) \tag{6}$$

where the weights, $\omega_{n,i}$, measure the similarity between the block $\mathbf{Z}_n^Y$ and all the previous ones: $\mathbf{Z}_i^Y$ with $i = 1, 2, \ldots, n-1$.

Therefore, the predicted block is seen as a weighted average of past blocks where the blocks more similar to the one predicting (the similarity is looked for between $\mathbf{X}$ blocks) will have more weight and the less similar less weight. The analysis of being more or less similar, is done by a distance between the discrete wavelet coefficients. The different properties needed to apply this method may be looked at Antoniadis et al. [1].

Among the reasons that we can highlight to choose the wavelet transform, is the fact that the data collected in most of the processes are inherently multiscalar due to contributions from events taking place at different locations and with different location in time and frequency. Thus, the data analysis and modeling approaches that represent the measured variables at various scales are better suited to extract information from the data than methods that represent variables in a single scale.

In short, the forecasting technique has two steps: in the first place, blocks' weights will be fixed analyzing the similarity between the predicting block and the previous ones (as said before, this similarity is found between explanatory variables' blocks), and the second step, in which is implemented a weighted average as the one described in equation 6.

Let us see how is defined the similarity between two blocks (see Antoniadis et al. [1]). To this end we are going to define the similarity between two series:

Given two series, the wavelet coefficients of the discrete wavelet transform for each time series at scale $j = j_0, \ldots, J-1$ and location $k = 0, 1, \ldots, 2^j - 1$, with $j_0 \geq 0$ are denoted by $\theta_{j,k}^{(i)} i = 1, 2$.

In each scale, $j \geq j_0$, the distance is defined as it follows:

$$d_j(\theta^{(\mathbf{1})}, \theta^{(\mathbf{2})}) = \left(\sum_{k=0}^{2^j-1} (\theta_{jk}^{(1)} - \theta_{jk}^{(2)})^2 \right)^{1/2} \tag{7}$$

This metric measure the similarity between two series in scale j. To take into account all scales, we use:

$$D(\theta^{(\mathbf{1})}, \theta^{(\mathbf{2})}) = \sum_{j=j_0}^{J-1} 2^{-j} d_j(\theta^{(\mathbf{1})}, \theta^{(\mathbf{2})}) \tag{8}$$

The distance defined in j^{th} scale is weighted by 2^{-j}. This is due to the fact that successive scales have half as many discrete wavelet coefficients as the previous scale, therefore distances of different scales are not directly comparable. This problem is corrected by weighting the distance of a scale by twice the weight of the next higher level. Thus, the distances of all scales are comparable and makes that lowest scales (smoothed versions) have greater weight, something that is suitable for the type of process we use, stationary processes (more details in Antoniadis et al. [1]).

Once we have calculated the distances between blocks using wavelet scaling coefficients ($\Xi_i = \{\xi_i^{(J,k)}/k = 0, 1, \ldots, 2^J - 1\}$ are the scaling coefficients at scale J of block $\mathbf{Z}_i$), the next step is the introduction of such distances within the kernel function. Thus, the prediction kernel is the one described in equation 6 with weights:

$$\omega_{n,i} = \frac{K(D(C(\Xi_n), C(\Xi_i))/h_n)}{\frac{1}{n} + \sum_{i=1}^{n-1} K(D(C(\Xi_n), C(\Xi_i))/h_n)} \tag{9}$$

where $C(\Xi_i)$ are the wavelets coefficients obtained via the "pyramid algorithm" (see Mallat [13]), h_n is the smoothing parameter of the kernel regression and K is the kernel function.

4 Technical aspects

In this section, we will described the technical features that had been used in the methods presented.

Regarding to the discrete wavelet transform there are several wavelets basis (see Daubechies [4]) we could choose from, although prior simulations showed that the method is robust with respect to the wavelet filter chosen. Thus, we have chosen for our applications the wavelet basis Symmlet 9. In the case where the number of elements of our blocks or segments is not a power of 2 (necessary for the dyadic discrete wavelet transform, see Daubechies [4]), each block of the explanatory variables is extended by mirror to the closest power of two.

Whilst, as said before, the selection of the kernel function is not relevant, the selection of the smoothing parameter is an important issue. We have used two criteria to select the smoothing parameter: the GCV and the RICE criteria, every time a kernel is employed both criteria are applied. Consider penalized least error measure $G(h)$, where h denotes the smoothing parameter, $G(h) = p(h)\phi(n^{-1}h^{-1})$ where $p(h) = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{m}(x_i))^2$ is the prediction error and $\phi(\cdot)$ is the penalized function. Different proposal for $\phi(\cdot)$ lead to different criteria. We will use the *Generalized Cross-Validation criterion* (GCV), where

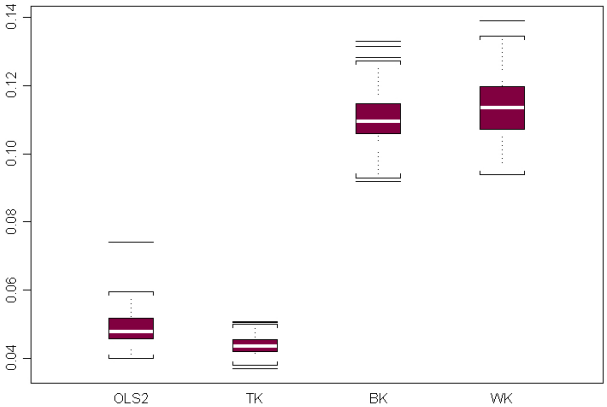

Figure 1: MSE values for all methods except OLS.

$\phi_{GCV}(n^{-1}h^{-1}) = (1 - \frac{1}{n}\mathbf{tr}K(h))^{-2}$ and the *Rice criterion* (RICE), where $\phi_R(n^{-1}h^{-1}) = (1 - \frac{2}{n}\mathbf{tr}K(h))^{-1}$.

5 Monte Carlo study

There is one main objective in this section: the analysis of the differences in practice between the estimators of the different methods. For this purpose we have simulated the following univariate model: $m(x) = 6(1 - e^{\frac{-x^2}{10}})$ and the data are generated as $Y_i = m(X_i) + \epsilon_i$, where X_i is a random sample from a standard normal distribution and the error terms are also i.i.d. from a zero mean normal distribution with standard deviation equal to 0.2. The number of replications is 100, and each of them consist of 42 blocks or segments of length 24, which altogether makes for series of size 1008.

Two measures of error have been computed to calculate the differences between the true regression function and the estimated ones, in the four proposed methods:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{10}$$

$$RME = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{11}$$

In Figure 1 the empirical distribution of the MSE error measure is represented via boxplots, for OLS (linear), OLS2 (quadratic), TK,BK,WK and only for the GCV smoothing parameter proposal.

Unfortunately the traditional kernel method provides a better estimation. In spite of these results, we checked out the methodologies in two real life datasets.

We have removed the OLS errors from the Figure 1 because they distorted the results.

6 Applications in real life datasets

6.1 1-day-ahead prediction of volatility

As said in the introduction, we apply the methodologies in a financial database, with the aim of explaining better the volatility implied in the option prices. Although our database is comprised of all call and put options on the IBEX-35 index futures traded daily on MEFF during the period January 1996 through November 1998, in this paper we are only going to present the comparison between the methodologies for the calls in 1996 due to lack of space, but the other results are available upon request. Also, we eliminate the observations before 10:30 and after 16:45 to avoid data which may reflect trades to influence market maker margin requirements.

In 1998, Aït-Sahalia and Lo [14] estimated the implied volatility function using a multivariate kernel estimator, where the BS implied volatility is replaced by a nonparametric function which depend upon the explanatory variables: moneyness degree and time-to-expiration. In 2002, with the same database as ours Ferreira et al. [15] estimated the same function using a multivariate SNN kernel estimator, including the liquidity as an explanatory variable. Although they found that liquidity was important in the in-sample pricing, in the out-of-sample performance only the moneyness seems to be important to explain the smile.

We have estimated the implied volatility depending on the moneyness with the described methodologies. In Table 1, we present the results, in terms of the mean square error (MSE) and the relative mean error (RME). In this case, the wavelet-kernel methodology is the best (statistically proved) to explain the volatility smile in 1996.

Table 1: Errors of different estimators for volatility and temperature.

Prediction Method	MSEVolat	RMEVolat	MSETemp	RMETemp
OLS	6.254	0.117	539.63	0.02325
OLS2	6.258	0.117	538.96	0.02521
TK	6.589	0.132	333.20	0.01547
BK	5.875	0.119	357.45	0.01563
WK	4.824	0.111	293.81	0.01447

6.2 1-block-ahead prediction of the temperature needed in the furnace for steel coils

In industrial plants it is important to reduce the costs derived from manufacturing process failures. One of those failures could be found in galvanized furnaces for steel coils: for example, if the steel coil goes out of the galvanization process with the wrong temperature, the coil is not accepted. In our case, our principal aim is to predict the rated temperature in the furnace to get the needed temperature of the steel coil. In order to explain this variable, we re going to use the variables that Martínez de Pisón (2003) [16] considered significant in his thesis. Thus, we have chosen two variables: Temperature of the steel coil at the entrance of the furnace and the rated temperature of the furnace, the second variable being the response variable. We had 100 steel coils, each with 16 points. The results obtained are shown also in Table 1. In this case as well, the wavelet kernel methodology has better (statistically) results than the others.

7 Conclusions

We have applied four different methodologies in our work. In the Monte Carlo study the traditional kernel estimator has shown better results than the others, but on the other hand in real life applications the wavelet-kernel estimator performs better. This has led us to question if the worse outcome obtained in simulations are due to mirror extended data as explained in Section 4 or that the data simulated are not good for this type of blocks' application. Further research has to be developed in these issues.

References

[1] Antoniadis, A., Paparoditis, E. & Sapatinas, T., A functional wavelet-kernel approach for continuous-time prediction. *JRSS series B*, pp. 837–857, 2006.
[2] Black, F. & Scholes, M., The pricing of options and corporate liabilities. *Journal of Political Economy*, **81**, pp. 637–659, 1973.
[3] Chui, C., *An introduction to wavelets*. Wavelet Analysis and Its Applications, Elsevier, 1992.
[4] Daubechies, I., *Ten Lectures on Wavelets*. SIAM: Society for Industrial and Applied Mathematics, 1992.
[5] Vidakovic, B., *Statistical Modeling by Wavelets*. JW & Sons, 1999.
[6] Aussem, A. & Murtagh, F., A neuro-wavelet strategy for web traffic forecasting. *Journal of Official Statistics 1*, **1**, pp. 65–87, 1998.
[7] Gonghui, Z., Starck, J., Campbell, J. & Murtagh, F., The wavelet transform for filtering financial data streams. *Journal of Computational Intelligence in Finance*, **7-3**, pp. 18–35, 1999.
[8] Lotric, U., Wavelet based denoising integrated into multilayered perceptron.

Neurocomputing, **62**, pp. 179–196, 2004.
[9] Cristi, R. & Tummala, M., Multirate, multiresolution, recursive kalman filter. *Signal Processing*, **80**, pp. 1945–1958, September 2000.
[10] Hong, L., Chen, G. & Chui, C.K., A filter-bank-based kalman filtering technique for wavelet estimation and decomposition of random signals. *IEEE transactions on circuits and systems 2, Analog and digital signal processing*, **45**, pp. 237–241, 1998.
[11] Renaud, O., Starck, J. & Murtagh, F., Wavelet-based combined signal filtering and prediction. *Systems, Man and Cybernetics, Part B, IEEE Transactions on*, 2005.
[12] Renaud, O., Starck, J. & Murtagh, F., Prediction based on a multiscale decomposition. volume 1, pp. 217–232, 2003.
[13] Mallat, S., A theory for multiresolution signal decomposition - the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**, pp. 674–693, 1989.
[14] Aït-Sahalia, Y. & Lo, A., Nonparametric estimation of state-price densities implicit in financial asset prices. *Journal of Finance*, **53**, pp. 499–547, 1998.
[15] Ferreira, E., Gago, M. & Rubio, G., A semiparametric estimation of liquidity effects on option pricing. *Spanish Economic Review*, **5**, pp. 1–24, 2003.
[16] Martínez de Pisón, F., *Optimización mediante tcnicas de minera de datos del ciclo de recocido de una lnea de galvanizado*. Ph.D. thesis, Univ. de la Rioja, 2003.