Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

Physica A 344 (2004) 162–167

PHYSICA A

www.elsevier.com/locate/physa

Confidence limits for data mining models of options prices

J.V. Healy, M. Dixon*[,1], B.J. Read, F.F. Cai

Computing, Communications Technology and Mathematics, London Metropolitan University, 31 Jewry Street, London, EC3N 2EY, UK

Received 15 December 2003
Available online 21 July 2004

Abstract

Non-parametric methods such as artificial neural nets can successfully model prices of financial options, out-performing the Black–Scholes analytic model (Eur. Phys. J. B 27 (2002) 219). However, the accuracy of such approaches is usually expressed only by a global fitting/error measure. This paper describes a robust method for determining prediction intervals for models derived by non-linear regression. We have demonstrated it by application to a standard synthetic example (29th Annual Conference of the IEEE Industrial Electronics Society, Special Session on Intelligent Systems, pp. 1926–1931). The method is used here to obtain prediction intervals for option prices using market data for LIFFE "ESX" FTSE 100 index options (http://www.liffe.com/liffedata/contracts/month_onmonth.xls). We avoid special neural net architectures and use standard regression procedures to determine local error bars. The method is appropriate for target data with non constant variance (or volatility).

PACS: 0.7.05.Mh; 89.65.Gh

Keywords: Option pricing; Neural nets; Data mining

*Corresponding author. Tel.: +44-207-320-1386; fax: +44-207-320-3009.
E-mail address: M.Dixon@Londonmet.ac.uk (M. Dixon).

[1] Visitor to BITD, Rutherford Appleton Laboratory, Chilton, DIDCOT, OX11 0Q.

0378-4371/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.physa.2004.06.112

1. Introduction

Data mining and computational methods such as artificial neural nets are increasingly used in finance. They provide an alternative non-parametric model of option prices, out-performing the Black–Scholes (BS) model [1–4]. These data mining methods are often generalisations of better known non-linear regression techniques. Confidence in the reliability of models and predictions is a key issue in finance. However, while techniques of statistical inference are well defined for the parametric regression methods traditionally employed in financial modelling, this is not the case for the non-parametric data mining techniques. Consequently, there is an absence from the literature of statistical hypothesis testing [5], and models are evaluated on summary criteria such as mean squared error and R^2. Also, prediction and confidence intervals are rarely constructed for option pricing models estimated using these techniques. In our work on option pricing, we have found that when input data is partitioned by moneyness/maturity, model price predictions may be unbiased for some partitions yet biased for others. Examination of pointwise prediction intervals is thus a necessity to obtain a fuller picture of model performance than is given by single summary statistics or hypothesis tests. We seek a method which is generally applicable, and can be used with any regression technique of sufficient flexibility. In this paper we describe a robust method for determining prediction intervals for neural nets and related techniques. We have tested the method empirically using a standard synthetic data set, and compared it with a method restricted to neural nets [6]. Here, our method is applied to obtain prediction intervals for pricing options, using a data set of 14,257 LIFFE European style FTSE 100 index 'ESX' options [7]. Our method uses standard regression procedures to determine local error bars. It is appropriate for target data with non-constant variance (or volatility).

2. Background

Neural nets are a generalisation of multivariate non-linear regression. Standard methods of computing confidence intervals such as the 'delta', and 'sandwich' methods as well as the 'naive' bootstrap apply in principle but have practical problems as discussed in Refs. [6,3]. Nix and Weigend [8] describe a method of using a neural net to estimate the variance of its own predictions of the target variable, thereby allowing prediction intervals to be constructed. They proposed a neural network, with two outputs, that simultaneously estimates the conditional mean of the specific target and its conditional variance. The architecture and cost functions used differed from a standard NN with fully connected hidden layers. The outcome is a model for both $y(x)$, the underlying true regression, and $\sigma^2(x)$ the true noise variance function. Nix and Weigend demonstrated good recovery of the target function and variance function, using the normalised mean squared error, and the mean cost, as performance criteria. The specific characteristics of the method [8] mean that it is restricted to NNs and is not generally applicable.

3. Our model/algorithm

Our model, which is described fully in Ref. [6], has two outputs and simultaneously estimates the regression function and the noise variance function. However, it differs from the model described by Nix and Weigend [8]. This is because it uses a standard fully connected NN architecture. It employs a least-squares cost function. And, in contrast to Ref. [8], it uses independent training and validation sets, rather than interchanging validation sets. Finally, we use the following three-Phase training algorithm:

Phase I: Randomly split the training data into two data sets, Set A and Set B. Using Set A, train a NN model on the target variable $d(x)$. Run the trained NN model on Set B, to obtain a set of squared residuals. By using squared residuals on a test set (Set B) as the second target for Phase II, over fitting and consequent underestimation of the standard error is avoided.

Phase II: Using Set B for training, train a second NN with two output nodes. The target for the first output node is the variable $d(x)$; the target for the second output node is the squared residuals obtained in Phase I from set B using the model trained on Set A.

Phase III (*optional*): Using Set A for training, train a further NN with two output nodes. The target for the first output node is the variable $d(x)$; the target for the second output node is squared residuals for the estimate $d^*(x)$ obtained in Phase II using the model trained on Set B. Set A and B are randomly split into a training and a validation portion; testing of each Phase is performed on an independent test set, Set C.

4. Results

We first applied our method to a univariate synthetic example defined by Nix and Weigend [8], called Example 1 in their paper and here. This uses a one-dimensional data set where $y(x)$ the true regression, and $\sigma^2(x)$, the noise variance function, are known. The true regression is $y(x) = \mathrm{Sin}(3x)\mathrm{Sin}(5x)$, where x is a uniformly distributed random number from the interval [0, $\pi/2$]. The noise $n(x)$ consists of numbers from the normal distribution $N[0, \sigma^2(x)]$, where $\sigma^2(x) = 0.02 + 0.25[1 - \mathrm{Sin}(5x)]^2$ is the noise variance function. The target value for training is $d(x) = y(x) + n(x)$. Fig. 1 shows a plot of the data points, the estimated and true regression $y(x)$, and the true and estimated prediction band obtained using our Phase III model. Fig. 1 illustrates, that our method can recover a good estimate of the prediction intervals given very noisy data.

5. Option pricing performance

In Example 1 the noise is artificially large and variable, and the data points widely dispersed compared to option prices. We therefore applied our method to Example

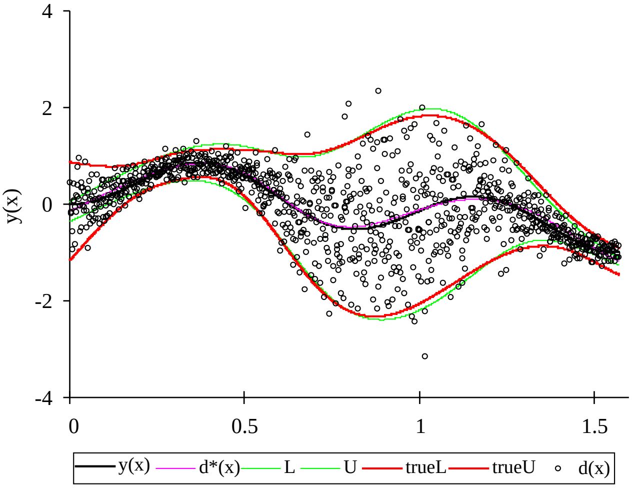

Fig. 1. Prediction bands for Example 1.

2, fully described in [6], based on synthetic option prices. Tests showed the network was able to recover the true regression function and noise variance function analogously to Example 1. In the more realistic, less noisy, context of Example 2 it was found Phase III training was superfluous, and did not improve the estimate of the true noise variance function. Here we present Example 3, which uses the same data set as Example 2, but uses actual observed prices of options in place of the synthetic prices of Example 2, so there is no underlying (smooth) true noise variance function. Hence, the estimated noise variance can only be compared with the actual squared residuals for the fit to the observed option prices. We follow [2] and use moneyness and time to maturity as input variables. For Example 3, a hidden layer of 10 nodes in Phase I and 20 nodes in Phase II were used giving values of $R^2 = 0.96$ and 0.93, respectively, for the option price. Example 4 has the same targets as Example 3 but uses as inputs the set of Black–Scholes variables and has corresponding $R^2 = 0.997$ and 0.991.

A graphical demonstration is provided in Figs. 2 and 3 of Examples 3 and 4. Here, the method is used to price a previously unseen option price series. In Figs. 2 and 3, the crosses are the observed option prices. The solid line ‘Estimated’ is the NN estimate of the prices. The dashed lines U*, L* are the prediction intervals constructed from the NN estimate of the noise variance. The dotted lines U, L, are confidence intervals constructed using actual residuals from the NN estimate of the option prices. In Fig. 3, BS is the Black–Scholes price estimate. Fig. 2 shows that the observed prices lie within the prediction band, which is consistent with unbiased price estimates. In Fig. 3, the improved fit provided by the additional input variables is reflected by the much narrower prediction intervals. The NN estimates of prices are particularly good, compared to BS, for the thinly traded deep in the money options.

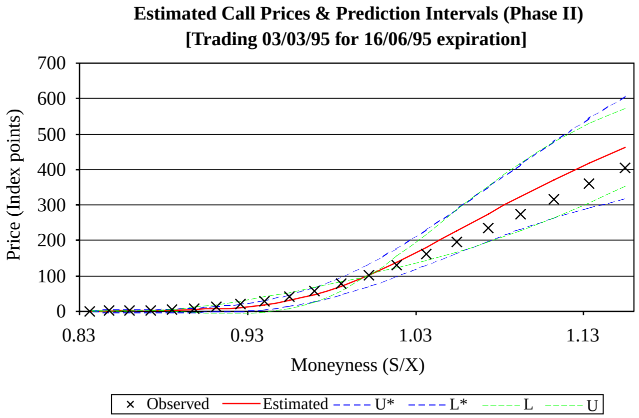

Fig. 2. Example 3: Application to option price series—inputs $(t, S/X)$.

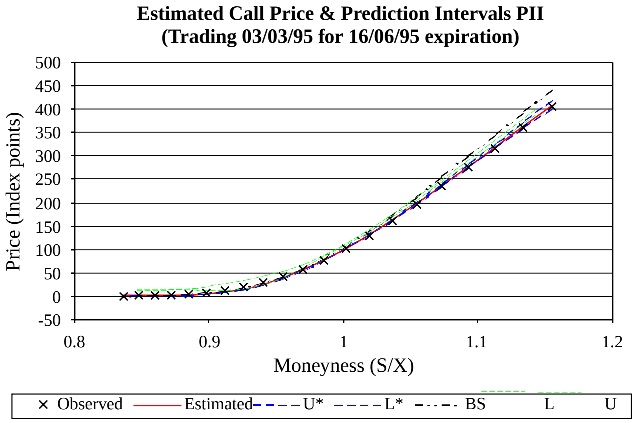

Fig. 3. Example 4: Application to option price series—inputs are BS variables.

6. Conclusions

This paper describes a robust method for determining prediction intervals for models derived by non-parametric forms of non-linear regression. The method can be used with general purpose modelling tools and is not restricted to application specific software. We have demonstrated its utility by constructing prediction intervals for option prices estimated using neural nets and shown their width is a much better indicator of model quality than summary statistics such as R^2. The method is also applicable to any comparable regression technique of sufficient flexibility.

References

[1] J. Benell, C. Sutcliffe, Black–Scholes versus artificial neural networks in pricing FTSE 100 options, Discussion Paper 00-156, School of Management, Southampton University, 2000.

[2] J. Hutchinson, A. Lo, T. Poggio, A non-parametric approach to pricing and hedging derivative securities via learning networks, J. Finance 49 (3) (1994) 851–889.

[3] J. Healy, M. Dixon , B. Read, F.F. Cai, 2003. Confidence & prediction in generalised non-linear models: an application to option pricing, Paper No 03-6, Centre for International Capital Markets Discussion papers, London Metropolitan University.

[4] J.V. Healy, M. Dixon, B.J. Read, F.F. Cai, A data-centric approach to understanding the pricing of financial options, Eur. Phys. J. B 27 (2002) 219–227.

[5] C. Sutclife, Private communication, January 2003 (We thank Professor Sutcliffe for confirming this observation.).

[6] J.V. Healy, M. Dixon, B.J. Read, F.F. Cai, in: Confidence in data mining model predictions: a financial engineering application; 29th Annual Conference of the IEEE Industrial Electronics Society, Special Session on Intelligent Systems, 2003, pp. 1926–1931.

[7] London International Financial Futures and Options Exchange, http://www.liffe.com/liffedata/contracts/month_onmonth.xls.

[8] D.A. Nix, A.S. Weigend, Learning local error bars for non-linear regression, in: Proceedings of NIPS 7, 1995, pp. 489–496.