QUANTITATIVE TRADING OF GOLD AND SILVER USING NONLINEAR MODELS

by
Christian L. Dunis* and Anup Nathani**
(CIBEF* and Liverpool John Moores University)**

January 2007

Abstract

The main aim of this paper is to forecast gold and silver daily returns with advanced regression analysis using various linear and non-linear models.

ARMA models are used as a linear benchmark for comparison purposes with non-linear models such as Nearest Neighbours, MultiLayer Perceptron (MLP) and Higher Order Neural Networks (HONN). The models are assessed using statistical criteria such as correct directional change as well as financial criteria such as risk adjusted return. The main aim is to find which of these models generate the best returns and if nonlinear models can be used for generating excess returns in the precious metals market. This is achieved by implementing a trading simulation where the forecast is translated into a trading signal. Profit statistics are calculated taking into account transaction costs.

It is concluded that, for the January 2000-May 2006 period under review, nonlinear models like MLPs and HONNs did outperform the linear ARMA models. In the end, the performance of both MLP and HONN models showed the presence of nonlinearities in the gold and silver prices as it was found that nonlinear models can be effectively used for generating excess returns in these markets.

Keywords: Gold, Higher Order Neural Networks, MultiLayer Perceptron, Nearest Neighbours, Silver, Trading Models.

***Christian Dunis is Professor of Banking and Finance at Liverpool Business School, Liverpool John Moores University and director of CIBEF (Email: c.dunis@ljmu.ac.uk).*
*** Anup Nathani is an Associate Researcher with CIBEF (Email:) anup.nathani@gmail.com.*
****CIBEF: Centre for International Banking, Economics and Finance, JMU, John Foster Building, 98 Mount Pleasant, Liverpool, L3 5UZ.*

1 - INTRODUCTION

Gold and silver are two important traded commodities, with gold being used as a hedge against inflation (Ranson, 2005), a hedge against the variations of the US dollar (Capie *et al.*, 2004) apart from other uses in the jewellery industry, industrial uses, as an investment and a store of value. Since the breakdown of the Bretton Woods Agreement in 1971-1973 the official role of gold in the international monetary system has ended. After the de-linking of the US Dollar to gold there has been a wide fluctuation in gold prices. Since the start of 2000 precious metals prices have appreciated and there is speculation of further price rises. In the circumstances, forecasting the price of these commodities has important implications not only monetarily but also economically (Levin and Wright, 2006).

Precious metals have an economic value apart from being commodities and particularly gold is considered as an alternate investment to the US dollar and a 'safe heaven' investment. A rise in the value of the US Dollar will lead, all other things being equal, to a fall in the dollar value of gold so both assets exhibit a negative correlation. Economic indicators and political factors play an important role in the long run pricing of these commodities. Although fundamental analysis could be applied for the long term pricing of these metals, predicting daily returns is very difficult as argued by Baker and Van Tassel (1985) who also underline that there is a lot of speculative interest in the short run price movement of gold. Furthermore, there is evidence that the daily volatility of these commodities has increased with silver being one of the most volatile commodities. It would therefore be very difficult to forecast daily returns using fundamental analysis, which gives a very strong motivation for using technical analysis and/or quantitative methods.

A further motivation is drawn from Dunis and Williams (2003) who find that for the EUR/USD exchange rate, Neural Network Regression (NNR) techniques give better results than other alternative models. A similar methodology is followed in this paper and the results of the various models are compared. Most of past researches in these markets have concentrated on using linear ARMA models and nonlinear MultiLayer Perceptron (MLP) models. In this paper we apply linear ARMA models and nonlinear models such as Nearest Neighbours, MLP and Higher-Order Neural Networks (HONN). The prime aim here is to find out which quantitative model does best in forecasting the returns of these two metals in order to implement a daily quantitative trading strategy.

The paper is organised as follows: section 2 gives a brief overview of the relevant literature using the various models applied in this research. Section 3 describes the data used and its division into in-sample and out-of-sample periods, and discusses the issue of stationarity. Section 4 gives a brief theoretical overview of the various trading models retained and the methodology followed in this paper. The calculation of transaction costs is also presented in this section. Section 5 discusses the trading simulation results obtained from the various models and section 6 provides some concluding remarks.

2 - LITERATURE REVIEW

The importance of gold as a commodity and as a safe investment is evident from the study of Capie *et al.* (2004) showing how gold can be used as a hedge against the fluctuations of the US dollar in the foreign exchange market. Ranson (2005) demonstrates that gold is a better hedge against inflation than inflation indexed bonds. Levin and Wright (2006) conduct an extensive empirical analysis of gold's hedging capabilities and analyse the magnitude of the relationship between gold and US inflation.

Baker and Van Tassel (1985) argue that it is difficult to forecast the daily price of gold using fundamental analysis while Smith (2002) contends that the London gold price followed a random walk. Both studies conclude that the London gold price is unpredictable and only new information and news are responsible for changing gold prices and thus the study of past prices is useless.

Contrary to the above findings, Ntungo and Boyd (1998) use ARMA and MLP models for forecasting the returns of various commodities. They find that for silver returns both ARMA and MLP models are able to generate positive returns and the results of both approaches are nearly similar.

Many researchers have used nonlinear techniques, compared them with traditional linear regression analysis and concluded in favour of precious metals being nonlinear in nature. Frank and Stengos (1989) use gold and silver data from the 1970s and 1980s and find that there are nonlinearities in the rates of return of these two metals. Further, Stengos (1996) conducts a thorough study of gold rates of returns on daily data from January 1975 to April 1993 using nonparametric forecasting methods and comparing them with ARMA models. He finds that nonparametric methods work better although the overall forecasting results are not very good. More recently, Brauner *et al.* (1997), Agnon *et al.* (1999), Chatrath *et al.* (2001) and Parisi *et al.* (2003) also find significant nonlinearities in the gold and silver markets. While most researchers argue in favour of nonlinear modelling techniques for these metals, only Ntungo and Boyd (1998), as already mentioned, find that ARMA and NNR models give similar trading results.

While there have been several studies focussed on the gold market there have been very few dedicated to the silver market which is a further motivation for this paper.

Various studies have used neural network regression and alternative forecasting techniques, some concentrating on the foreign exchange market, eg. Dunis and Williams (2003), some concentrating on commodity spreads (Dunis *et al.*, 2005, 2006-a, -b). On the one hand, Dunis and Williams (2003) and Dunis *et al.* (2005) find that neural networks add value to the forecasting process and offer a rewarding alternative approach to more traditional modelling techniques whereas on the other hand Dunis *et al.* (2006-b) find that the ARMA model is a hard benchmark to beat on the WTI-Brent spread. Also, various studies (e.g. Knowles *et al.*, 2005 and Dunis *et al.*, 2006-a) argue that HONN is the best performing network. From past researches it is evident that in some markets neural networks work well while in others traditional methods may work better. This is an empirical issue.

Apart from neural networks there are few applications of nearest neighbours for forecasting. Stengos (1996) and Agnon *et al.* (1999) use nearest neighbours for forecasting commodity returns and find that they have better forecasting ability than traditional regression analysis noting that this model can forecast successfully even with small data sets where traditional methods fail. Alexandre et al. (1998) use the method for the high frequency forecasting of the DEM/FRF and USD/FRF exchange rates series. Guégan and Huck (2004) use this approach to forecast the Dow Jones Index.

The mixed results of previous researches, many of which stress the nonlinearity of the gold and silver markets, are a further incentive to find out which models perform best in these markets and whether the use of NNR models as an alternative to traditional forecasting techniques is justified.

3 - THE DATA

The data used in this study is the London Bullion Market Association (LBMA) daily gold and silver fixing in US dollars per troy ounce. This is the daily spot price fixed by major banks and is used as a benchmark all over the world. The data was obtained from Datastream.

The dataset is from 04/01/2000 to 31/05/2006 totalling 1620 observations. As shown in table 1 below, for linear models the data is split into an in-sample period of 1347 observations to optimize the models and an out-of-sample period of 273 observations where the best in-sample models are retained for out-of-sample forecasting.

Table 1: Data Division for Linear Models

Subset	Purpose	Period	Observations
In-sample	Optimize model	04/01/2000– 29/04/2005	1347
Out-of-sample	Validate Model	03/05/2005 – 31/05/2006	273

For Nearest Neighbours and NNR models the in-sample subset is further divided into training and test periods whereas the out-of-sample data is the same validation period as for linear models (Brauner *et al.*, 1997 and Dunis *et al.*, 2006-b). The data is divided as shown in table 2 below.

Table 2: Data Division for Nonlinear Models

Subset	Purpose	Period	Observations
Training	Optimize model	04/01/2000– 30/03/2004	1074
Test	Stop model optimization	31/03/2004– 29/04/2005	273
Validation	Validate model	03/05/2005– 31/05/2006	273

The training dataset is used to train the network, the criteria for training the network being profit maximisation over the test period: the training of the network is thus stopped when the profit on the test dataset is at a maximum. This model is then used for the out-of-sample analysis on the validation set.

The two data series are non-stationary as the mean of the series is not constant and it is biased towards a trend. To overcome this problem the level series are transformed into returns series. Given the price level P_1, P_2, P_3,…….P_t, the rate of return at time *t* is formed by $R_t = (P_t - P_{t-1})/P_{t-1}$. This series is a stationary series, a useful statistical property confirmed by stationarity tests not reported here in order to conserve space.

4 - METHODOLOGY

4.1 The Benchmark ARMA Model

ARMA models are particularly useful when information is limited to a single stationary series[1], or when there is no relevant economic theory. The ARMA model is a univariate time series model which combines a moving average process with a linear difference equation. The model is then explained purely in terms of its own past values. It does not assume any particular pattern in a time series, but uses an iterative approach to identify a possible model from a general class of models. Sometimes, it is possible that two or more models may approximate the series equally well, in this case the most parsimonious model should prevail.

In this paper, we use ARMA models as a benchmark[2] to compare the results of other models. For further details on the procedure, see Box *et al.* (1994) and Pindyck and Rubinfeld (1998).

4.1.1 The Gold ARMA Model

Initially an ARMA (10,10) model was estimated on the in-sample data, which is equivalent to 2 trading weeks. The ARMA (10, 10) model was unsatisfactory as several coefficients were insignificant at the 90% confidence interval. The non significant terms were then removed and a restricted ARMA (1, 4, 6, 8, 9, 10) model was obtained with coefficients significant at 99% confidence interval (see Appendix 1). This restricted model was then retained for out-of-sample forecasting.

4.1.2 The Silver ARMA Model

A similar approach was implemented for silver, leading to the selection of a restricted ARMA (3, 5, 7, 9, 10) model[3] that was then used for out-of sample forecasting (see Appendix 2).

4.2 The Nearest Neighbours Model

Nearest Neighbours is a nonlinear and non-parametric forecasting method. It is based on the idea that pieces of time series in the past have patterns which might have resemblance to pieces in the future. Similar patterns of behaviour

[1] The general class of ARMA models is for stationary time series. If the series is not stationary, an appropriate transformation is necessary.

[2] Technical MACD models and ARMA-GARCH models were also estimated but are not reported to conserve space as their in-sample results were less good than for ARMA models.

[3] The 9th AR term was further removed as it was statistically insignificant.

are located in terms of nearest neighbours using a distance called the Euclidean distance and these patterns are used to predict behaviour in the immediate future. Alexandre *et al.* (1998) and Guégan and Huck (2004) recommend this approach mainly when making short term predictions. It only uses local information to forecast and makes no attempt to fit a model to the whole time series at once. It is similar to technical analysis as it tries to find out patterns based on certain parameters.

The user defines parameters such as the number of neighbours *K*, the length of the nearest neighbours pattern *m* (also called the ‘embedding dimension’) and the weighting of final prices in a neighbour *α*. When *α* is greater than 1, a greater emphasis is given to similarity between the more recent observations. The nearest neighbours algorithm weights the nearest to furthest Neighbour sets according to the Euclidean distance from the actual time series. The Euclidean distance can be interpreted as the fit of an historic pattern to the actual pattern. These three parameters are very important for neighbours and very easy to apply computationally.

The main aspect of the nearest neighbours algorithm is the Euclidean distance, in its simplest form it can be considered the distance (or similarity) between true patterns and selected neighbours. The Euclidean distance used in this example is a function of this distance and the standard deviation. For a series of 1620 observations with *α*=1 and *K*=1, the “similarity” between neighbours or $S_{m,k}$ is defined as:

$$S_{m,k} = \frac{1}{1620 * K} \sum_{t=T-1620+1}^{T} \sum_{k=1}^{K} \frac{\left(\frac{1}{m} \sum_{i=1}^{m} \left(w_{k,t-i+1}^{m} - w_{t-i+1} \right)^2 \right)^{\frac{1}{2}}}{\sigma_t}$$

where:

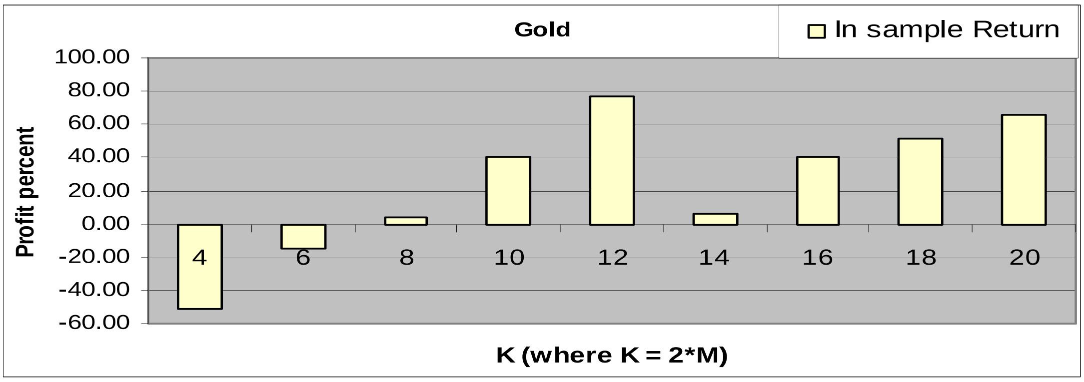
σ_t is the *n*-day moving standard deviation (this is used in order to account for the changing standard deviation of the series over time);
K is the number of neighbours,
m is the embedding dimension,
$w_{k,t-i+1}^{m}$ refers to the i^{th} element of the k^{th} neighbour with dimension *m* at time *t*.

Intuitively the smaller the value for $S_{m,k}$ the smaller the distance between the neighbours and the actual pattern and hence the better the fit of the model. Guégan and Huck (2004) also suggest that a good approximation for choosing the parameters *K* and *m* is dependant on the size of the information set. We choose m from the interval:
m = [R(log(T)), R(log(T)+2)]
where R is the rounding function rounding to the immediate lower figure. So for an information set T=1620, the choice of *m* lies between 7 and 9. A standard heuristic is to select *K* to be approximately twice the value of *m*. It is useful to decide on these parameters based on the profit maximization of the training and test datasets, fixing *K* and *m* within the intervals defined above according to the highest profit of the test sample set.

4.2.1 The Gold Nearest Neighbours Model

An in-sample analysis of gold returns was conducted with various combinations of *K* and *m* and α=1. To start with, the value of *K* was taken as twice the value of *m*. Figure 1 below shows the results of various combinations of values of *K* and *m* for gold.

Figure 1: Gold in-sample profitability for various values of *K* and *m*

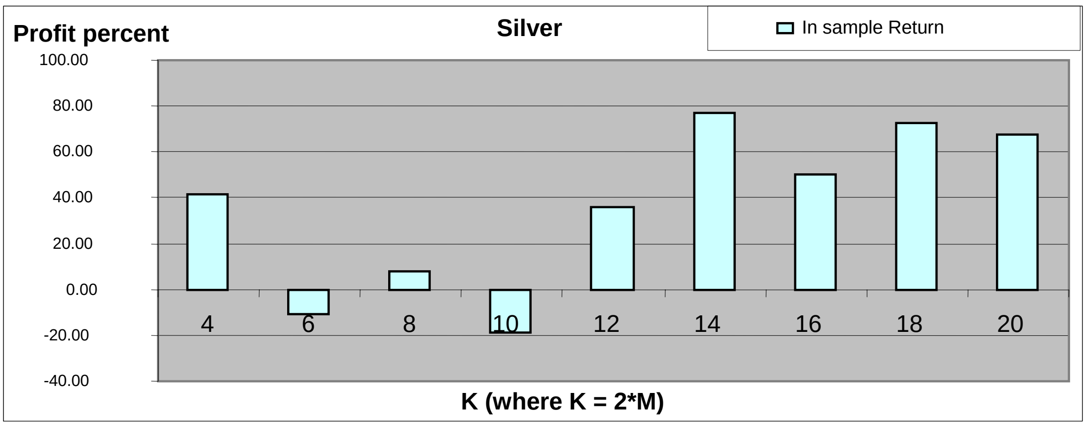


Based on the in-sample analysis the values of *K*=12 and *m*=6 with α=1 were chosen as the optimized in-sample model. As evident from figure 1 above, and in contrast with Stengos (1996), the value of *K* did affect the forecast. Increasing the value of α from 1 to 1.05 gave a better in-sample profit of 83.16% and this model was retained for out-of-sample analysis.

4.2.2 The Silver Nearest Neighbours Model

Various combinations of *K* and *m* were also tried on the in-sample data for silver. The optimisation protocol was basically the same as with gold. The values of *K* and *m* and the resulting profits are shown in figure 2. The optimal values of *K* and *m* were found to be 14 and 7 respectively, which maximized in-sample profits. It was further found that α= 0.9 was giving the best result for silver. Also, following Guégan and Huck (2004) who argue that the value of *K* should be between two to three times that of *m*, the optimum value of *m* was finally set at 6 with *K*=14. This model gave an in-sample profit of 78.11% and was retained for out-sample-forecasting.

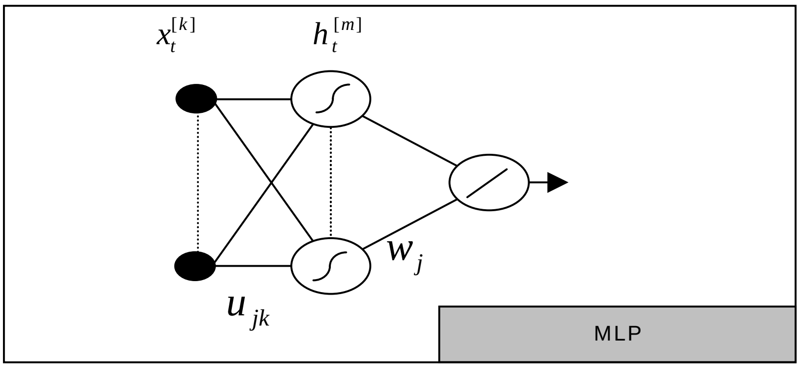
Figure 2: Silver in-sample profitability for various values of *K* and *m*



4.3 The MultiLayer Perceptron Model

The most basic type of NNR model is the MLP model[4]. An MLP consists of three layers with a collection of input nodes known as the input layer, the output layer and the hidden layer: the input and the output layers are separated by a hidden layer consisting of one or more hidden nodes. The nodes in adjacent layers are fully connected. The number of nodes in the hidden layer defines the amount of complexity that the model can fit. Each neuron at the input layer receives the data, multiplies them by a connection strength called a ‘weight’ and sends them to a neuron in the next layer. The input layer transmits a weighted sum of its inputs to the hidden layer. The hidden layer passes the information through a nonlinear activation function and this information is passed onto the output layer if the calculated value is above a threshold. The relationship between each neuron and other neurons in the next layer follows a ‘feed-forward’ process. The input and hidden layers also include a bias node (similar to the intercept for standard regression) with a fixed value of 1. The connection between neurons for a single output neuron in the network is shown in figure 3 below.

Figure 3: A single output, fully connected MLP model



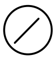
where:

$x_t^{[n]}$ $(n = 1,2,\cdots,k+1)$ are the model inputs (including the input bias node) at time *t*.

$h_t^{[m]}$ $(m = 1,2,...,m+1)$ are the hidden nodes outputs (including the hidden bias node)

u_{jk} and w_j are the network weights

 is the transfer sigmoid function: $S(x) = \dfrac{1}{1+e^{-x}}$,

 is a linear function: $F(x) = \sum_i x_i$

The error function to be minimised is:

$$E(u_{jk}, w_j) = \frac{1}{T}\sum(\Delta S_t - \Delta\tilde{S}_t(u_{jk}, w_j))^2$$

[4] For further details on neural networks, refer to Haykin (1999), Kaastra and Boyd (1996), Kingdon (1997) or Zhang *et al.* (1998).

with ΔS_t being the target value (the actual asset return at time *t*).

At the beginning, the modelling process is initialized with random weights. The output of the processing unit is then passed on to the single output node of the output layer. The NNR error is then analysed and systematically minimised by adjusting the weights. The training of the network is of utmost importance, since it is possible for the network to learn the training data subset exactly (commonly referred to as overfitting). For this reason the network training must be stopped early. Thus, as per standard heuristics, the data set is divided into three subsets, using roughly 2/3 of the data for training the model, 1/6 for testing and the remaining 1/6 for validation.

The training subset is used to optimize the model, the 'back propagation of errors' algorithm is used to establish optimal weights from the initial random weights. The test subset is used to stop the training subset from overfitting. The optimization on the training subset is stopped when profitability is maximized on the test subset. The validation subset is used to simulate future values of the times series, which for comparison purposes is the same as the out-of-sample subset of the other models.

4.3.1 The Gold MLP Model

As mentioned by Parisi *et al.* (2003), there are only a few general rules about the selection of the learning parameters and the time at which the learning must be stopped, and hence experimenting plays an important role in the training of networks. For gold, the ACF and PACF of the returns suggested 19 as an appropriate number of lags and thus an MLP model with 19 lags was developed. Various levels of learning rate and momentum for the network were tried. All models with 19 lags encountered problems of excessive learning or inadequate learning. The number of lags was subsequently reduced and we found that 10 lags gave overall better models. Thus, 10 lags were chosen as the model input and various models were found. The final model retained for out-of-sample forecasting had 10 inputs 5 hidden nodes and 1 output node and was fully connected. Also, it was found that a learning rate of 0.001 and momentum of 0.0001 were best to prevent excess or inadequate learning of the gold network. Training was stopped when profit was maximized in the test period.

4.3.2 The Silver MLP Model

For silver no definite number of lags could be arrived at looking at the ACF and PACF of returns so various networks were trained with 8, 9 and 10 lags as model inputs and 9 lags were finally retained. Also, it was found that a learning rate of 0.0003 and momentum of 0.0001 gave optimum training to the network. Various networks were run with 9 lags and the model which had optimum training and produced best in-sample results was retained for out-of-sample forecasting. The selected model had 9 input nodes, 5 hidden nodes and 1 output node.

4.4 The Higher Order Neural Network Model

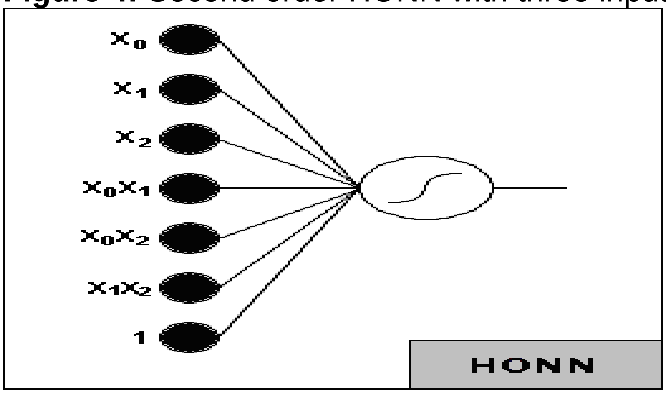
HONNs were first introduced by Giles and Maxwell (1987) and were called "tensor networks". While they have already experienced some success in the field of pattern recognition and associative recall[5], their use in finance has been so far limited. The architecture of a three input second order HONN is shown below in figure 4.

HONNs use joint activation functions; this technique reduces the need to establish the relationships between inputs when training. Furthermore this reduces the number of free weights and means that HONNs can be faster to train than even MLPs. However, because the number of inputs can be very large for higher order architectures, orders of 4 and over are rarely used.

Another advantage of the reduction of free weights means that the problems of overfitting and local optima affecting the results can be largely avoided as noted by Zhang *et al.* (2002), Knowles *et al.* (2005) and Dunis *et al.* (2006-a).

The HONN in this study uses the transfer sigmoid function as described for the MLP architecture in section 4.3. This has been done in order to be able to draw direct comparisons between the architectures of the models.

Figure 4: Second order HONN with three inputs



 are the model inputs.

 is the transfer sigmoid function: $S(x) = \dfrac{1}{1+e^{-x}}$,

4.4.1 The Gold HONN Model

For gold various numbers of lags and orders 2 and 3 were tried as inputs to model. Starting from 2 lags various numbers of lags up to 18 were tried. It was found that 18 was the appropriate number of lags giving the best results. Various networks with 18 lags were estimated: the best network was of order 3 with a training parameter of 0.000001 and this model was retained for out-of-sample forecasting.

4.4.2 The Silver HONN Model

For silver also various lags were tried as inputs to the model with orders 2 and 3. Up to 10 lags were tried and it was found that 10 was the appropriate number of lags with order 3. The training parameter for the chosen model was 0.0000001 and this model was retained for out-of-sample forecasting.

4.5 Transaction Costs

[5] Associative recall is the act of associating two seemingly unrelated entities, such as smell and colour. For more information see Karayiannis and Venetsanopoulos (1995).

Transaction costs for trading bullion futures involves a bid-ask spread and a commission fee. Since the price involved in the research, LBMA daily PM fixing, is a mid-price, the bid-ask spread is taken as the cost which would be incurred when trading on an exchange[6]. It has been taken as a percentage of the average price of gold and silver over the entire data set. This would imply the cost of trading being averaged out in the entire period. The bid-ask spread for gold is 0.0278% and for silver is 0.0862%.

Also, a commission fee of $20 per round trip trade (i.e. opening and closing a position on a single contract) has been taken from www.ghco.com. This is 0.0278% for gold and 0.0345% for silver. This gives a total cost for a round trip (i.e. bid-ask spread plus commission fee) of 0.0556% for gold and 0.1207% for silver.

5 - EMPIRICAL RESULTS

5.1 Trading Simulation

To evaluate the performance of all the trading models a trading simulation model was selected. The premise of the model was that at the close of each day a forecast for the next day was generated and a hypothetical buy or sell position was initiated. The trading strategy used here was to buy (sell) if a rising (falling) price trend was forecasted. If no change in the trend from the previous day was predicted then the current position was maintained. The trading rule can be summarized as:

If $\hat{P}_{t+1} < P_t$, then sell/go or stay short

If $\hat{P}_{t+1} > P_t$, then buy/go or stay long

where P_t is today's closing price and $\hat{P}_{t+1}$ is the closing price forecasted for the next trading day. These trading rules reflect turning points in price movements and, as noted by Ntungo and Boyd (1998) they place an emphasis on the correct directional forecast. If a model predicts the correct directional change for the next trading day then at the end of the following day the gross profit will equal the return of that day or vice versa. The magnitude of the forecast is not considered as important as predicting the correct directional change. For net profit calculations, transaction costs are deducted from gross profits whenever a new trade is initiated. The integration of transaction cost gives a realistic assessment of the returns of each model. Profit statistics are calculated for in- and out-of-sample periods separately. Among the list of conventional trading performance measures used by the fund management industry to analyse trading results (see, among others, Dunis and Williams (2003) for details), we focus on the annualised return, the Information ratio (a measure of risk-adjusted return) and the maximum drawdown (a measure of downside risk showing the maximum cumulative loss that could have been incurred on a portfolio). Since profit statistics are calculated for all models after deducting transaction costs, the number of

[6] COMEX is considered here for trading bullion and the bid-ask spread is taken on the basis of minimum price fluctuation specified by the exchange. Please refer to www.nymex.com for details.

trades is very important for a model's performance as more active models will attract higher transaction costs reducing the overall profitability.

5.2 ARMA Model Performance

ARMA models for both gold and silver gave good in-sample and out-of-sample trading profits. In contrast with Stengos (1996), the gold ARMA model gives consistent high returns both in- and out-of-sample. Table 3 below gives the trading simulation statistics of this model.

The reason may be attributed to the fact that the ARMA model was capable enough of tracking the bull run. The silver ARMA model also performed reasonably well both in- and out-of-sample, even if it suffers from a large maximum drawdown in-sample. Table 4 shows the results.

Table 3: Gold ARMA Trading Results

Gold ARMA forecast	in-sample	out-of-sample
Cumulative return	116.37%	53.66%
Annualised Return	21.95%	49.53%
Standard Deviation	13.83%	18.07%
Information Ratio	1.59	2.74
Maximum drawdown	-16.64%	-10.98%
Total Trading Days	1314	267
No. of Trades	579	24
Correct Directional Change	52.21%	57.30%

Table 4: Silver ARMA Trading Results

Silver ARMA forecast	in-sample	out-of-sample
Cumulative return	151.95%	52.04%
Annualised Return	28.66%	48.04%
Standard Deviation	23.61%	35.60%
Information Ratio	1.21	1.35
Maximum drawdown	-44.07%	-19.50%
Total Trading Days	1308	271
No. of Trades	818	48
Correct Directional Change	51.07%	54.98%

The results of these ARMA models are in agreement with Ntungo and Boyd (1998) although our results are much better than those reported in their study. By contrast, they are in total disagreement with Stengos (1996) who found that linear models had no success in forecasting such nonlinear systems.

5.3 Nearest Neighbours Model Performance

Among those arguing that gold and silver series are nonlinear, such as Frank and Stengos (1998) and Agnon *et al.* (1999), the use of nearest neighbours was found to give better results than traditional regression analysis by Stengos (1996). The results of neighbours we obtain are very ordinary for gold and not good for silver. Tables 5 and 6 show the neighbours performance statistics for gold and silver respectively.

Although the gold model is profitable in-sample, the information ratio of 0.71 does not compare well with the ARMA benchmark. The silver model just manages to cover transaction costs in the in-sample period, where it predicts correctly less than 50% of the time and registers a massive maximum drawdown.

Surprisingly, both models give better out-of-sample results. But it would be highly unlikely that these models would have been selected and put to trading based on their in-sample performance statistics when there were other models with promising results. Further investigation into the forecasting ability of nearest neighbours applied to precious metals is thus required.

Table 5: Gold Nearest Neighbours Trading Results

Gold Neighbours forecast	in-sample	out-of-sample
Cumulative return	47.99%	30.95%
Annualised Return	9.51%	28.57%
Standard Deviation	13.30%	18.23%
Information Ratio	0.71	1.57
Maximum drawdown	-22.20%	-9.60%
Total Trading Days	1256	266
No. of Trades	633	129
Correct Directional Change	51.67%	55.64%

Table 6: Silver Nearest Neighbours Trading Results

Silver Neighbours forecast	in-sample	out-of-sample
Cumulative return	1.59%	20.24%
Annualised Return	0.32%	18.68%
Standard Deviation	24.02%	35.57%
Information Ratio	0.01	0.53
Maximum drawdown	-93.87%	-32.25%
Total Trading Days	1222	269
No. of Trades	635	126
Correct Directional Change	48.28%	55.76%

5.4 MLP Model Performance

The MLP models for both gold and silver are reliable as can be concluded from the performance statistics that these models generated, as can be seen from tables 7 and 8 below.

Table 7: Gold MLP Trading Results

Gold MLP forecast	in-sample	out-of-sample
Cumulative return	189.65%	37.53%
Annualised Return	35.80%	34.65%
Standard Deviation	13.79%	18.18%
Information Ratio	2.60	1.91
Maximum drawdown	-12.07%	-7.17%
Total Trading Days	1307	266
No. of Trades	353	83
Correct Directional Change	55.09%	56.39%

Table 8: Silver MLP Trading Results

Silver MLP forecast	in-sample	out-of-sample
Cumulative return	273.84%	50.41%
Annualised Return	51.65%	46.53%
Standard Deviation	23.18%	35.24%
Information Ratio	2.23	1.32
Maximum drawdown	-24.94%	-29.80%
Total Trading Days	1297	269
No. of Trades	721	139
Correct Directional Change	57.98%	51.67%

After transaction costs, both models give robust risk-adjusted profits in the in- and out-of-sample periods. The gold MLP model performs significantly well out-of-sample with an information ratio of 1.91, a maximum drawdown of -7.17% and a correct directional change of 56.39%. The silver model also performs well out-of-sample with an information ratio of 1.32 although the higher risk involved with trading silver is evident from the higher standard deviation and the large maximum drawdown of -29.80%. Finally, although MLP models give slightly lower profits than their ARMA counterparts in the out-of-sample period, this judgement is made with the benefit of hindsight as, in the real world, model selection can only be achieved on the basis of in-sample performance results: overall, the performance of MLP models was better and more consistent than that of the benchmark ARMA models.

5.5 HONN Model Performance

As can be seen in tables 9 and 10 below, the performance of HONN models is also very good. The Gold HONN model was actually second best of all the gold models in-sample and it produces slightly better results than the MLP in the out-of-sample period.

Table 9: Gold HONN Trading Results

Gold HONN forecast	in-sample	out-of-sample
Cumulative return	166.14%	38.49%
Annualised Return	31.53%	35.53%
Standard Deviation	13.87%	18.13%
Information Ratio	2.27	1.96
Maximum drawdown	-20.36%	-10.36%
Total Trading Days	1310	266
No. of Trades	680	136
Correct Directional Change	54.89%	57.14%

The silver HONN model performs rather well both in- and out-of-sample. The in-sample performance is nearly identical to the MLP model, albeit slightly lower. Out-of-sample it has the best return performance of all models, nevertheless its maximum drawdown is significantly higher than that for the MLP model. Based on the in-sample analysis, one would have selected the silver MLP model for trading as it was marginally better than its HONN counterpart. In reality, one would have probably chosen a model combination strategy where the forecasts of both MLP and HONN models would be used, but such a combination strategy is beyond the scope of this paper.

Table 10: Silver HONN Trading Results

Silver HONN forecast	in-sample	out-of-sample
Cumulative return	268.97%	57.36%
Annualised Return	50.73%	52.95%
Standard Deviation	23.25%	35.29%
Information Ratio	2.18	1.50
Maximum drawdown	-25.47%	-39.63%
Total Trading Days	1297	269
No. of Trades	662	140
Correct Directional Change	57.29%	52.79%

Overall, our results are in line with those of Brauner *et al.* (1997) who find that neural network predictions perform well for gold. Parisi *et al.* (2003) also conclude that neural networks are good alternatives to technical analysis and conventional time series analysis in the gold market. Stengos (1996) and Agnon *et al.* (1999) concur arguing in favour of nonlinearities in the commodity markets. The performance of the MLP and HONN models used in this research vindicates the above findings for both gold and silver.

6 - CONCLUDING REMARKS

In this paper, nonlinear modelling techniques for gold and silver were used and benchmarked against linear ARMA models. As evident from the performance results of both MLP and HONN models, and for the period under review, it is indeed possible to use advanced quantitative methods to forecast the returns of these metals and to trade them profitably on a daily basis.

It should be stressed however that ARMA models provided a tough benchmark to beat, they actually were far superior to traditional technical models such as MACD models. Still, although ARMA models had slightly better results out-of-sample, this was achieved with the benefit of hindsight which one does not have in the real world. A thorough model selection can only be achieved on the basis of in-sample performance results and this alone favours the use of NNR models. Overall, MLP and HONN models were more reliable than the benchmark with robust profits both in-and out-of-sample, MLPs being marginally better.

Our results are in line with the literature on nonlinear financial markets, eg. Brauner *et al.* (1997) and Dunis and Williams (2003). Nonlinearities in the gold and silver markets are evident from this study and give a strong motivation to use NNR models for trading them and experiment with more alternative architectures such as recurrent networks.

In summary, and in contrast with Smith (2002), it seems that, for the period considered, it was possible to successfully forecast the daily gold and silver returns using advanced quantitative methods, even if further investigation into the forecasting ability of nearest neighbours applied to precious metals is required. Our results confirm the nonlinear nature of commodity markets and vindicate the use of NNR models as an alternative to traditional forecasting techniques for gold and silver trading.

REFERENCES

Agnon, Y., Golan, A., and Shearer, M. (1999), 'Nonparametric, Nonlinear, Short-Term Forecasting: Theory and Evidence for Nonlinearities in the Commodity Markets', *Economic Letters*, Vol. 65, pp. 293-299.

Alexandre, H., Girerd-Potin, I. and Tarramasco, O. (1998), 'High Frequency Exchange rate Forecasting by the Nearest Neighbours Method', in Dunis, C. and Zhou, B. [eds.], *Nonlinear Modelling of High frequency Financial Time Series*, John Wiley, Chichester, pp. 279-291.

Baker, S.A. and Van Tassel, R.C. (1985), 'Forecasting the Price of Gold: A Fundamentalist Approach', *Atlantic Economic Journal*, Vol. 13, pp. 43-51.

Box, G. E. P., Jenkins, G. M. and Gregory, G. C. (1994), *Time Series Analysis: Forecasting and Control*, Prentice-Hall, New Jersey.

Brauner, E.O., Dayhoff, J.E. Xiaoyun S. and Hormby, S. (1997), 'Neural Network Training Techniques for a Gold Trading Model', *Computational Intelligence for Financial Engineering* (CIFEr), Proceedings of the IEEE/IAFE, pp. 57-63.

Capie, F.; Mills, T.C. and Wood, G. (2004), **'**Gold as a Hedge against the US Dollar**',** World Gold Council Report, Research Study No. 30, Cass Business School, London.

Chatrath, A., Adrangi, B. and Shank, T. (2001), 'Nonlinear Dependence in Gold and Silver Futures: Is it Chaos?', *The American Economist*, Vol. 45 (2), pp. 25-32.

Dunis, C. and Williams, M. (2003), 'Application of Advanced Regression Analysis for Trading and Investment' in Dunis, C., Laws, J. and Naïm, P. [eds.], *Applied Quantitative Methods for Trading and Investment*, John Wiley, Chichester, pp. 1-39.

Dunis, C. L, Laws, J. and Evans, B. (2005), 'Modelling with Recurrent and Higher Order Networks: A Comparative Analysis', *Neural Network World*, Vol. 6 (5), pp. 509-523.

Dunis, C. L., Laws, J. and Evans, B. (2006-a), 'Modelling and Trading the Soybean-Oil Crush Spread with Recurrent and Higher Order Networks: A Comparative Analysis', *Neural Network World*, Vol. 3 (6), pp. 193-213.

Dunis, C. L., Laws, J. and Evans, B. (2006-b), 'Trading Futures Spreads: An Application of Correlation and Threshold Filters', *Applied Financial Economics*, Vol. 16, pp. 1-12.

Frank, M. and Stengos, T. (1989), 'Measuring the Strangeness of Gold and Silver Rates of Return', *Review of Economic Studies*, Vol. 56 (188), pp. 553-567.

Giles, L. and Maxwell, T. (1987), "Learning Invariance and Generalization in High-Order Neural Networks", *Applied Optics*, 26 (23), pp 4972-4978.

Guégan, D. and Huck, N. (2004), 'On the Use of Nearest Neighbours to Forecast in Finance', Note de Recherche IDHE-MORA n 10-2004, ENS Cachan.

Haykin, S. (1999), *Neural Networks: A Comprehensive Foundation*, 2nd edition, Prentice-Hall, New Jersey.

Kaastra, I. and Boyd, M. (1996), 'Designing a Neural Network for Forecasting Financial and Economic Time Series', *Neurocomputing,* 10, pp. 215-236.

Karayiannis, N. B. Venetsanopoulos, A. N. (1995), "On The Training and Performance of Higher-Order Neural Networks" Mathematical Biosciences, 129 (2), pp. 143-168.

Kingdon, J. (1997), *Intelligent Systems and Financial Forecasting,* Springer, London.

Knowles, A. Hussein, A. Deredy, W. Lisboa, P. and Dunis, C. L. (2005), 'Higher-Order Neural Networks with Bayesian Confidence Measure for Prediction of EUR/USD Exchange Rate', *CIBEF Working Papers*, www.cibef.com.

Levin, E.J. and Wright, R.E. (2006), 'Short-run and Long-run Determinants of the Price of Gold', World Gold Council Report, Research Study No. 32, Cass Business School, London.

Ntungo C. and Boyd M. (1998), 'Commodity Futures Trading Performance Using Neural Network Models versus ARMA Models', *The Journal of Futures Markets*, 18, pp. 965-983.

Parisi, F.; Parisi, A. and Guerrero J.L. (2003), 'Rolling and Recursive Neural Network Models: The Gold Price', Working Paper, Universidad de Chile.

Pindyck, R. S. and Rubinfeld, D. L. (1998), *Econometric Models and Economic Forecasts*, Irwin McGraw-Hill, Singapore.

Ranson, D.; (2005), 'Inflation Protection: Why Gold Works Better Than "Linkers"', World Gold Council Report, Wainwright H.C., World Gold Council Publications, London.

Smith, G. (2002), 'Tests of the Random Walk Hypothesis for London Gold Prices', *Applied Economics Letters*, 9, pp. 671-674.

Stengos, T. (1996), 'Nonparametric Forecasts of Gold Rates of Return', International Symposia in Economic Theory and Econometrics Series, Cambridge University Press, pp 393-406.

Zhang, G., Patuwo, B. E. and Hu, M. Y. (1998), 'Forecasting with Artificial Neural Networks: The State of The Art', *International Journal of Forecasting*, 14, pp. 35-62.

Zhang, M., Xu, S. and Fulcher, J. (2002), 'Neuron-Adaptive Higher Order Neural-Network Models for Automated Financial Data Modelling', *IEEE Transactions on Neural Networks*, 13 (1), pp. 188-204.

Appendix 1: Gold restricted ARMA model

Dependent Variable: D(GOLD)
Included observations: 1336 after adjustments
White Heteroskedasticity-Consistent Standard Errors & Covariance

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	0.113568	0.080277	1.414701	0.1574
AR(1)	-0.205473	0.039175	-5.244990	0.0000
AR(4)	0.235490	0.027399	8.594901	0.0000
AR(6)	0.261428	0.027752	9.420287	0.0000
AR(8)	-0.153028	0.034903	-4.384387	0.0000
AR(9)	-0.235874	0.035426	-6.658222	0.0000
AR(10)	-0.753840	0.042495	-17.73929	0.0000
MA(1)	0.152152	0.033290	4.570551	0.0000
MA(4)	-0.197105	0.021428	-9.198479	0.0000
MA(6)	-0.307942	0.020067	-15.34581	0.0000
MA(8)	0.155463	0.028977	5.365064	0.0000
MA(9)	0.223946	0.030280	7.395844	0.0000
MA(10)	0.819380	0.036803	22.26405	0.0000

Adjusted R-squared	0.045864	S.D. dependent var	3.019177
S.E. of regression	2.949130	Akaike info criterion	5.010580
Log likelihood	-3334.068	F-statistic	6.347576
Durbin-Watson stat	1.979814	Prob(F-statistic)	0.000000

Appendix 2: Silver restricted ARMA model

Dependent Variable: D(SILVER)
Included observations: 1336 after adjustments
White Heteroskedasticity-Consistent Standard Errors & Covariance

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	0.135715	0.232184	0.584515	0.5590
AR(3)	-0.344278	0.052310	-6.581441	0.0000
AR(5)	0.367922	0.039811	9.241801	0.0000
AR(7)	-0.444335	0.043265	-10.27001	0.0000
AR(10)	-0.786046	0.063641	-12.35118	0.0000
MA(3)	0.298375	0.045940	6.494846	0.0000
MA(5)	-0.374874	0.027357	-13.70298	0.0000
MA(7)	0.407318	0.033112	12.30139	0.0000
MA(9)	-0.048833	0.016279	-2.999786	0.0028
MA(10)	0.842174	0.052580	16.01704	0.0000

Adjusted R-squared	0.035469	S.D. dependent var	8.996388
S.E. of regression	8.835404	Akaike info criterion	7.202867
Log likelihood	-4801.515	F-statistic	6.454640
Durbin-Watson stat	2.035614	Prob(F-statistic)	0.000000