

Modelling, Forecasting and Trading of Commodity Spreads

**Thesis submitted in accordance with the requirements of the University of Liverpool for the degree of
Doctor in Philosophy.**

by

Peter William Middleton

March 2014

Table of Contents

CHAPTER 1: Introduction.....	9
1.1 Introduction	9
1.2 Motivation and Contribution to Knowledge	12
1.3 Structure of Thesis	12
CHAPTER 2: Models	15
2.1 Naive Trading Strategy	15
2.2 Buy and Hold Strategy	15
2.3 MACD Model	16
2.4 ARMA Model	17
2.5 Cointegration Model	17
2.6 Neural Networks	18
2.6.1 The Multi-layer Perceptron Model	19
Figure 1. A single output, inter-connected MLP model (2 neurons / nodes).....	20
2.6.2 The Recurrent Network	21
Figure 2. Elman recurrent neural network architecture with two neurons / nodes for the hidden layer.	
.....	22
2.6.3 The Higher Order Neural Network	23
Figure 3. Second order HONN with three inputs (1 neuron / node).....	24
2.6.4 The PSO Radial Basis Function Neural Network Model	25
Figure 4. Radial basis function neural network (with two hidden nodes)	27
3.0 Genetic Programming Algorithm (GPA).....	30
Figure 5. Generic tree structure	32
Figure 6. Mutation of a tree structure	34
Figure 7. Crossover family tree-like structure	35
CHAPTER 3: Modelling and Trading the Corn/Ethanol Crush Spread with Neural Networks.....	37
1.0 Introduction	38
2.0 Literature Review	40
2.1 Spread Trading Agricultural Futures.....	41
2.2 Seasonality of Agricultural Futures.....	43
Table 1. Typical corn cycle - seasonal sub-periods.....	43
2.3 Application of Neural Network Architectures	44
3.0 Descriptive Statistics and Data.....	45
3.1 Statistical Behaviour of Commodity Prices	46
Figure 8. The corn-ethanol crush CBOT daily closing prices (23/03/2005 – 31/12/2009)	47
3.2 Descriptive Statistics	48
Figure 9. Histogram of corn/ethanol spread return series.....	48
Table 2. Explanatory variables.....	49
Table 3. Data segregation for the full sample period.....	50
3.3 Rolling Forward Procedure	50
3.4 Discounting the Existence of Seasonality	51

4.0 Methodology	52
4.1 Benchmark Models.....	52
4.2 MACD Model.....	52
4.3 ARMA Model.....	53
5.0 Empirical Results	53
5.1 Statistical Performance.....	53
Table 4. Out-of-sample statistical performance.....	53
5.2 Trading Performance	54
Table 5. Out of sample trading performance results (unfiltered).....	54
Table 6. Out of sample trading performance results (filtered)	54
6.0 Concluding Remarks	56
CHAPTER 4: Trading and Hedging the Corn/Ethanol Crush Spread using Time Varying Leverage and Nonlinear Models	58
1.0 Introduction	59
2.0 Literature Review	61
3.0 Descriptive Statistics and Related Financial Data	62
3.1 Statistical Behaviour of the Crush.....	64
Figure 10. The corn-ethanol crush CBOT daily closing prices (23/03/2005 – 31/12/2010).....	65
3.2 Descriptive Statistics and Explanatory Variables.....	66
Figure 11. Histogram of corn/ethanol spread return series.....	66
Table 7. Explanatory variables for the neural networks	67
Table 8. Data segregation for the full sample period.....	67
3.3 Rolling Forward Procedure	67
4.0 Methodology	68
4.1 Benchmark Models.....	68
4.2 MACD Model.....	69
4.3 ARMA Model.....	69
5.0 Empirical Results	70
5.1 Statistical Performance	70
Table 9. In-sample statistical performance	70
Table 10. Out-of- sample statistical performance.....	70
5.2 Trading Performance	71
Table 11. Out-of-sample (unleveraged trading performance).....	71
Table 12. Leverage structure	73
Table 13. Out-of-sample (leveraged trading performance)	74
6.0 Concluding Remarks	75
CHAPTER 5: Non-Linear Forecasting of the Gold Miner Spread: An Application of Correlation Filters.....	77
1.0 Introduction	78
2.0 Literature Review	80
3.0 Descriptive Statistics and Related Financial Data	81
3.1 Statistical Behaviour of the Spread	83

Figure 12.The gold miners spread ETF daily closing prices (23/05/2006 – 30/06/2011).....	83
3.2 Descriptive Statistics and Explanatory Variables.....	83
Figure 13.Histogram of GLD/GDX spread return series (full sample)	84
Table 14.Neural data segregation for the full sample period	84
Table 15.GPA data segregation for the full sample period	86
Table 16.Explanatory variables for the neural networks and the GPA	86
4.0 Methodology	87
4.1 Benchmark Strategies and Models	87
4.1.1 ARMA	87
4.1.2 Cointegration Model – The Long Run Relationship	88
Figure 14.In Sample residuals of the long-run cointegrating relationship (μ)	88
Table 17. Augmented dickey fuller unit root test (in-sample)	89
Figure 15.In sample price series	89
5.0 Empirical Results	90
5.1 Statistical Measures	91
Table 18.In sample statistical accuracy	91
Table 19.Out-of-sample statistical accuracy	91
Figure 16.MAE in-sample statistics	93
5.2 Summary Trading Performance	93
Table 20.Out of sample trading performance (unfiltered).....	94
Table 21.Volatility of spread returns (annualised).....	95
Figure 17.RiskMetrics of returns (in-sample)	96
Figure 18.In sample correlation.....	97
Table 22.Out of sample trading performance (filtered).....	98
6.0 Concluding Remarks	99
CHAPTER 6: An Analysis of ETF Commodity Spread Portfolios – A Case of Mean Reversion..	101
1.0 Introduction	102
Figure 19. In sample GLD vs. GDX relative price performance (23/05/2006 = 100)	104
Figure 20. In sample USO vs. OIH relative price performance (23/05/2006 = 100).....	105
Figure 21. In sample SLV vs. GDX relative price performance (23/05/2006 = 100)	105
2.0 Literature Review	105
3.0 Descriptive Statistics and Related Financial Data	108
Table 23. ETF contract specifications	108
Figure 22. GLD vs. GDX spread (in-sample)	110
Figure 23. USO vs. OIH spread (in-sample).....	111
Figure 24. SLV vs. GDX spread (in-sample)	112

Figure 25. GM spread (14/06/06-31/07/08)	112
Figure 26. GM spread (01/05/09-20/05/10)	113
Figure 27. GM spread in-sample (23/05/06-20/05/10)	113
Table 24. Augmented dickey-fuller test statistics	113
Figure 28. Oil spread (23/04/07-16/10/07).....	114
Figure 29. Oil spread (02/01/09-20/05/10).....	115
Figure 30. Oil spread in-sample (23/05/06-20/05/10).....	115
Table 25. Augmented dickey-fuller test statistics	115
Table 26. Augmented dickey-fuller test statistics	116
Figure 31. Silver spread in-sample spread (23/05/06 – 20/05/10)	116
Figure 32. Gold miner - full sample distribution of spread returns.....	117
Figure 33. Oil - full sample distribution of spread returns	118
Figure 34. Silver - full sample distribution of spread returns	118
3.1 The Commodity Spreads and Related Financial Data	118
Table 27. Full dataset	119
Table 28. Explanatory variables for the MLP neural network.....	119
4.0 Methodology	119
4.1 ARMA	119
4.2 Cointegration Models	120
Table 29. Johansen (1988) test results GLD/GDX.....	122
Table 30. Johansen (1988) test results USO/OIH	122
Table 31. Johansen (1988) test results SLV/GDX.....	123
5.0 Empirical Results	123
5.1 Trading Rules and Strategies.....	123
Table 32. Trading parameters	124
Table 33. Half-life days	125
5.2 Results	125
Table 34. Unfiltered trading statistics (in-sample) (rankings: * = 1 st , ** = 2 nd , *** = 3 rd)	126
Table 35. Unfiltered trading statistics (out-of-sample) (rankings: * = 1 st , ** = 2 nd , *** = 3 rd)	126
Table 36. Unfiltered out-of-sample portfolio performance	127
Table 37. Filtered out-of-sample portfolio performance	128
Figure 35. ARMA riskmetrics volatility (in-sample)	129
Figure 36. Cointegration riskmetrics volatility (in-sample)	129
Figure 37. MLP riskmetrics volatility (in-sample)	129

6.0 Concluding Remarks	130
CHAPTER 7: Modelling, Forecasting and Trading the Crack – A Sliding Window Approach to Training Neural Networks.....	133
1.0 Introduction	134
Figure 38. Full sample price series of the 3:2:1 crack spread	136
Figure 39. Trading dataset price performance. The ‘crack’ vs. large cap refiners equity	138
Figure 40. Trading dataset price performance (rebased to 100). The ‘crack’ vs. small / medium cap refiners equity	139
2.0 Literature Review	140
2.1 Modelling the Crack	140
2.2 Training of Neural Networks.....	141
3.0 Descriptive Statistics and Related Financial Data.....	143
Table 38. Segregation of Dataset	144
Table 39. PSO RBF input selection during the training windows	149
Table 40. Most significant explanatory variables.....	149
Figure 41. Spread returns (out of sample)	150
4.0 Methodology	150
5.0 Empirical Results	151
5.1 Statistical Accuracy	151
Table 41. Out-of-sample trading statistics	151
5.2 Trading Performance	152
Table 42. Out-of-sample unfiltered trading performance	152
Table 43. Out-of-sample filtered trading performance.....	153
Figure 42. PSO RBF unfiltered trading performance (380 days sliding window).	155
Figure 43. PSO RBF unfiltered trading performance (500 days sliding window).	155
Figure 44. PSO RBF filtered trading performance (380 days sliding window).....	156
Figure 45. PSO RBF filtered trading performance (500 days sliding window).....	156
6.0 Concluding Remarks	157
CHAPTER 8: Final Remarks and Future Research	160
CHAPTER 9: Appendix.....	163
A.1 Performance Measures	163
Table 44. Statistical and trading performance measures.....	163
A.2. CHAPTER 3.....	163
A.2.1 Contract specifications	163
Table 45. Contract specifications	164
A.2.2 Network input criteria and selection	164
Table 46. Correlation matrix of neural inputs (in-sample correlation of returns)	164
A.2.3 Correlogram of spread returns.....	164
A.2.4 Networks characteristics	165

Table 48. Network characteristics	165
A.2.5 ARMA modelling	165
Table 49. ARMA output	166
A.2.6 Empirical results in the training and test sub-periods	166
Table 51. In sample trading performance (unfiltered)	167
Table 52. In sample trading performance results (filtered)	167
A.3 CHAPTER 4	167
A.3.1 Contract specifications	167
Table 53. Contract specifications	167
A.3.2 Network input criteria and selection	167
Table 54. Correlation matrix of neural inputs (in-sample return correlations)	167
A.3.3 Model parameters	168
Table 55. Neural network characteristics	168
Table 56. GPA characteristics	168
A.3.4 ARMA modelling	168
Table 57. ARMA results	169
A.3.5 Empirical results in the training and test sub-periods	169
Table 58. In sample (unleveraged trading performance)	169
Table 59. In sample (leveraged trading performance)	169
A.4 CHAPTER 5	169
A.4.1 Non-linear network parameters	170
Table 60. Neural network parameters	170
Table 61. Genetic programming parameters	170
A.4.2 ARMA modelling	170
A.4.3 In-sample trading performance	171
Table 63. In-sample trading performance (unfiltered)	172
Table 64. In-sample trading performance (filtered)	172
A.5 CHAPTER 6	172
A.5.1 ARMA results	172
Table 65. GM spread ARMA results	173
Table 66. Oil spread ARMA results	173
Table 67. Silver spread ARMA results	174
A.5.2 MLP training parameters	174
Table 68. Training parameters	174
A.5.3 Portfolio results	174
Table 69. Unfiltered in-sample portfolio trading performance	174
Table 70. Filtered in-sample portfolio trading performance	175

A.6 CHAPTER 7.....	175
A.6.1 Supplementary information.....	175
Table 71. The refiners market capitalisation.....	175
A.6.2 ARMA equations and estimations.....	175
Table 72. ARMA equations.....	176
A.6.3 GARCH equations and estimations.....	176
Table 73. GARCH model # 1.....	177
Table 74. GARCH model # 2.....	178
A.6.4 PSO parameters.....	178
Table 75. PSO RBF parameters.....	178
Table 76. Neural characteristics.....	179
A.6.5 Best weights over the training windows.....	179
Table 77. Best weights obtained from the 380-day training window.....	180
Table 78. Best weights obtained from the 500-day training window.....	181
CHAPTER 10: References.....	182

CHAPTER 1: Introduction

1.1 Introduction

Historically, econometric models have been developed to model financial instruments and markets however the vast majority of these ‘traditional’ models have one thing in common, linearity. While this is convenient and sometimes intuitive many linear models fail to fully capture the dynamic and complex nature of financial instruments and markets. More recently, ‘sophisticated’ methodologies have been evolved to accurately capture ‘non-linear’ relationships that exist between financial time series. This rapidly advancing field in quantitative finance is known as Artificial Intelligence.

The earliest forms of artificial intelligence are Neural Networks however these have since been developed using more accurate learning algorithms. Neural networks are also of particular use because of their capability of being able to continually learn as new information is fed into the network. In this research new data is introduced using both fixed and sliding window approaches for training each of the networks. Furthermore, Genetic Programming Algorithms are also highly regarded in the financial industry and have been increasingly applied as an optimisation technique. Therefore, each of the non-linear models are supported by existing research and as a result these methodologies have become practical tools for optimising existing models and predicting future movements in financial assets.

In the absence of computational algorithms to rationalise large amounts of data, investors are confronted with a difficult and seemingly impossible task of trying to comprehend large datasets of information. Nevertheless, advancements in computing technology have enabled market participants to benefit from the use of neural networks (NN) and genetic programming (GP) algorithms in order to optimise and identify patterns and trends between explanatory variables and

target outputs. This is of particular importance in the agricultural market such as grains, precious metals and other commodities are informationally rich with large amounts of data being readily available to evaluate.

Among the first to use neural networks for financial analysis were Rumelhart and McClelland (1986), Lippman (1987), and Medsker *et al.* (1993). More recently, neural networks and genetic programming algorithms have been extensively applied to the foreign exchange market (Hornik *et al.*, 1989; Lawrenz and Westerhoff, 2003), for credit analysis (Tam and Kiang, 1992), volatility forecasting (Ormoneit and Neuneier, 1996; Donaldson and Kamstra, 1997), option pricing, (Hutchinson *et al.*, 1994), portfolio optimisation (Chang *et al.*, 2000; Lin *et al.*, 2001), to both developed (Swales and Yoon, 1992) and emerging (Kimoto *et al.*, 1990) stock markets, and for optimisation of technical trading rules (Tsai *et al.*, 1999; Neely *et al.*, 2003). The application of non-linear methodologies to futures contracts and in particular, commodity spread trading, is limited. Trippi and DeSieno (1992) and Kaastra and Boyd (1995), however were among the first to explore and apply neural networks to forecast futures markets.

Financial markets and assets are influenced by an array of factors including but not limited to; human behaviour, economic variables, and many other systematic and non-systematic factors . As a result, many academics and practitioners have devised numerous approaches and models to explain financial time series such as fundamental analysis, technical analysis and behavioural finance. The purpose of this research however is to identify, forecast and trade daily changes in commodity spreads using a combination of novel nonlinear modeling techniques and performance enhancing trading filters. During the research process, non-linear models such as neural networks and genetic algorithms are used to identify trends in complex and expansive commodity datasets. Each of the methodologies are used to produce predictions for future time periods. In this research forecasts for $t+1$ horizons are examined.

Progressively, each chapter presents an evolution of research in the area of non-linear forecasting to address inefficiencies associated with more traditional neural architectures. In total a collection of five non-linear methodologies are proposed and analysed to trade commodity ‘spreads’. These non-linear methodologies are benchmarked against linear models which include Naïve strategies, Moving Average Convergence Divergence (MACD) strategies, buy and hold strategies, Autoregressive Moving Average (ARMA) models, and Cointegration models. In the final chapter of the research a mixed model approach is employed to include linear outputs from benchmark models as inputs during the training of each neural network. The research includes various adaptations of existing non-linear methodologies such as neural networks and genetic programming. Through historical data input, each non-linear methodology is trained to construct ‘optimal’ trading models. Models are selected to trade commodity spreads using data from Exchange Traded Funds (ETFs) and Futures contracts. In all cases the reader is presented with results from both unfiltered and filtered trading simulations.

The aim of this thesis is to benefit both hedgers and speculators who are interested in applying non-linear methodologies to the task of forecasting changes in commodity spreads. By allowing market participants to input numerous explanatory variables, non-linear methodologies such as neural networks and genetic programming algorithms can become a valuable tool for predicting changes in commodity spreads. Empirical evidence reveals that non-linear methodologies are statistically superior compared to existing linear models and they also produce higher risk adjusted returns. Moreover, by including output from linear models in the input dataset to train non-linear models, market participants are also able benefit from a ‘synergy’ of information using a ‘mixed model’ approach. In order to improve trading results the research also offers examples of numerous trading filters which can also be of use to hedgers and speculators. On the whole the research contributes a wealth of knowledge to academic studies as it offers conclusive

evidence to support the widespread integration and use of non-linear modelling in the form of artificial intelligence. Empirical results are evaluated by statistical measures as well as financial performance measures which are widely used by financial institutions.

1.2 Motivation and Contribution to Knowledge

The purpose and motivation of the research is to develop quantitative trading models that are intuitive, powerful, robust, and accommodate specific investment objectives. This study is designed to promote widespread use of stochastic methodologies as a tool in quantitative finance to disseminate complex information which involves an intensive search through large amounts of data.

Although Neural Networks and Genetic Algorithms have been applied and reviewed by numerous existing papers the use of these methodologies in the area of Spread Trading is limited. Furthermore, many of the established models which exist in finance only present a limited analysis of non-linear methodologies. For one, many only use autoregressive returns as inputs. In this research each paper provides evidence of a more practical and thorough application. A variety of training algorithms and trading strategies are applied to improve on the results produced by more ‘traditional neural networks’. Results from the research provide evidence of alternative methodologies available to market participants when trading commodity spreads. These methodologies are more sophisticated techniques and each of the trading strategies and filters aim to maximise profitability while reducing risk. Furthermore, use of alternative training algorithms, multi-objective fitness functions and input selection criteria are all investigated in the research. Different approaches to ‘training’ each of the neural networks are also investigated and in particular the final chapter displays results obtained from sliding windows of 380 and 500 days.

1.3 Structure of Thesis

The rest of the research is organized in chapters designed to provide readers with an insight into various methodologies and trading strategies. Chapter 2 provides details about each of the models which were used during the analysis. This includes both non-linear methods and more conventional models. Much of the content in this research has either been accepted for publication, presented at international conferences or is currently undergoing review for publication in academic journals. All of these papers were also approved by the University of Liverpool's internal board in order to obtain funding to attend conferences. The research is structured into self contained chapters with each focusing on different methodologies, trading strategies and time periods. The beginning of each chapter offers an introduction to the investigated spreads as well as a review of existing literature. While the focus of each chapter follows a logical progression of research there are some unavoidable repetitions throughout, however this has been kept to a minimum with each model only being described once in Chapter 2. The references and appendix have also been merged at the end of the thesis. The layout of the thesis is the presentation of five research papers. Each of the chapters are briefly explained below:

Chapter 2 details each of the models used during the research providing particulars for both linear and non-linear methodologies. Throughout, the linear models are referred to as 'benchmark' models which are gauged against each of the nonlinear methodologies.

Chapter 3 is the first essay of five entitled "Modelling and Trading the Corn/Ethanol Crush Spread with Neural Networks". Results from this analysis were presented at the 2011 Financial Forecasting Markets (FFM) conference in Marseille, France. This has also been **accepted** for publication in the forthcoming book (January 2014) entitled 'Computational Intelligence Techniques for Trading and Investment.'

Chapter 4 is the second essay of five entitled “Trading and Hedging the Corn/Ethanol Crush Spread Using Time Varying Leverage and Nonlinear Models”. Results from this analysis were also presented at the 2011 Financial Forecasting Markets (FFM) conference in Marseille, France. This has been submitted as part of the conference proceedings has been **accepted** for special edition publication in the **European Journal of Finance**.

Chapter 5 is the third essay of five entitled “Non-linear Forecasting of the Gold Miner Spread: An Application of Correlation Filters”. Results from this analysis were presented at the ‘2012 Higher Moments in Finance and Actuarial Science’ in Venice, Italy as well as the 2012 Financial Forecasting Markets (FFM) conference in Marseille, France. This has been submitted to the **Intelligent Systems in Accounting, Finance and Management** journal and has been **accepted** for publication.

Chapter 6 is the penultimate essay of five entitled “An Analysis of ETF Commodity Spread Portfolios – A Case of Mean Reversion”. Results from this analysis were presented at the 2012 Financial Forecasting Markets (FFM) conference in 2012. This has been submitted to the journal ‘*Quantitative Finance*’ and is currently under review for potential publication.

Chapter 7 is the final essay of the series and is entitled “Modelling, Forecasting and Trading the Crack – A Sliding Window Approach to Training Neural Networks”. This has been submitted to the journal ‘*Operations Research*’ and is currently under review for potential publication.

CHAPTER 2: Models

2.1 Naive Trading Strategy

This strategy is known as ‘naive’ due to its simplistic nature: trading signals for $t+1$ are determined by the directional change in the spread at time t . This is also known as a random walk model as the signals which are produced for trading incorporate an element of ‘randomness’. The Naive strategy is widely used as a benchmark to test the robustness of more technically calibrated models. Hence, the forecasted time series is based on a one day autoregressive linear method which assumes that the ‘best’ forecast is determined by the most recently observed data. This is as depicted in equation 1.

$$\hat{Y}_{t+1} = Y_t \tag{1}$$

Where: Y_t is the actual rate of return at period t
 \hat{Y}_{t+1} is the forecasted rate of return for the next period

2.2 Buy and Hold Strategy

Much of the existing ‘buy and hold’ literature examines optimal holding periods for stocks. For instance, Jegadeesh and Titman (1993) who analyse a period extending from 1965 – 1989, find trading strategies that buy past winners and sell past losers realise significant abnormal returns. In particular, strategists who selected winners based on 6 months prior performance realised average profits in excess of 12% when holding these ‘winners’ for 6 month periods at a time. More recently, Vanstone *et al* (2010) compare the performance of a buy and hold strategy to that of an Artificial Neural Network (ANN). In this research, both strategies are tasked with forecasting the Australian stock market (ASE200) and the ANN is found to produce superior results with higher

average daily profit and a higher Sharpe ratio. Research carried out by Niaki and Hoseinzade (2013) also compares an ANN to a buy and hold strategy when trading the S&P 500 index from 01/03/04 – 30/06/08. Results corroborate those produced by Vanstone *et al* (2010) as Niaki and Hoseinzade (2013) also find that ANN's are more profitable than buy and hold strategies when trading equities.

Many asset managers and stock pickers adopt this strategy by buying/selling undervalued / overvalued assets, holding them until a price target is reached, and then the asset(s) are sold/bought to realise a profit or to limit a loss in the event that a price target is not achieved.

2.3 MACD Model

Introduced by Appel (1979), the Moving Average Convergence/Divergence (MACD) indicator is one of the most widely used indicators in technical analysis and has since established itself as a prominent technique in forecasting. As a result, the signals produced from this indicator have been used to form a benchmark model. The MACD model is mathematically defined in equation 2.

$$M_t = \frac{(Y_t + Y_{t-1} + Y_{t-2} + \dots + Y_{t-n+1})}{n} \tag{2}$$

Where:

- M_t is the moving average at time t
- n is the number of terms in the moving average
- Y_t is the actual rate of return at period t

The MACD strategy is also a fairly pragmatic model as two moving average series are created with different moving average lengths. One of the moving averages is considered to be a short term moving

average while the other is a longer term moving average. These moving averages are arbitrarily determined based on which combination performed best over each of the in-sample period in terms of annualised returns.

2.4 ARMA Model

Autoregressive moving average models (ARMA) assume that the value of a time series depends on its previous values (the autoregressive component) and on previous residual values (the moving average component). A typical ARMA model takes the below form:

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \dots + \phi_p Y_{t-p} + \varepsilon_t - w_1 \varepsilon_{t-1} - w_2 \varepsilon_{t-2} - \dots - w_q \varepsilon_{t-q} \quad (3)$$

Where:

Y_t	is the dependent variable at time t
Y_{t-1} , Y_{t-2} , and Y_{t-p}	are the lagged dependent variable
ϕ_0 , ϕ_1 , ϕ_2 , and ϕ_p	are regression coefficients
ε_t	is the residual term
ε_{t-1} , ε_{t-2} , and ε_{t-p}	are previous values of the residual
w_1 , w_2 , and w_q	are weights

2.5 Cointegration Model

Cointegration between non-stationary variables occurs when a linear combination of the variables results in a stationary time series (Engle and Granger, 1987). There are two methodologies which are commonly employed to test whether or not a pair is cointegrated. The first is known as the Engle and Granger (1987) 2-step approach and the second is known as the

Johansen (1988) approach. The latter was selected for analysis of the Gold Miner spread in chapters 5 and 6 because the Engle and Granger (1987) approach has various limitations. In particular, its results are sensitive to the ordering of variables which may create unreliable output with residuals having different sets of statistical properties. In addition, if the pair is not found to be cointegrated then as discovered by Lim and Martin (1995) the resulting cointegrating equation estimates a spurious time series. Furthermore, the Johansen (1988) test can be directly used to estimate the Vector-Error-Correction model from which the coefficients are obtained in order to build a cointegrated time series. As a result, the Johansen test was seen as the most appropriate test for this research.

2.6 Neural Networks

Neural networks exist in a variety of different architectures and have been implemented in numerous financial applications. However, the architecture that is most widely used for the analysis of stock markets is known as the Multi-Layer Perceptron (MLP) neural network.

A generic neural network is built with at least three layers comprising of an input, hidden and output layer. The structure of the input layer is determined by the number of explanatory variables depicted as nodes in the architecture. The hidden layer represents the capacity of complexity in which the model can support or 'fit'. Moreover, both the input and hidden layers contain what is known as a bias node. The value attributed to this node is a fixed value and is equal to one. Its purpose is similar to the functionality of which the intercept serves in more traditional regression models. The final and third layer of a standard neural network, the output layer, is governed by a structure of nodes corresponding to a number of response variables. Furthermore, each of these layers is linked via a node to node interconnecting system enabling a functional network of 'neurons'.

On the whole, neural networks learn the relationships in data using neurons similar to how the human brain works. They are a non-parametric tool and use a series of waves and neurons to capture even

very complex relationships between the predictor inputs and the target variables¹. They can overcome messy data such as noise and imprecision in the measurement system. Neural networks are appropriate for regression as well as classification, time series analysis and clustering.

The functionality of a simple network can be surmised as a step by step process as follows:

- i. Inputs are determined and entered into the network for analysis. Target outputs (variables) are also set to enable the network to proceed and develop a learning ability.
- ii. The input data are then processed by the input nodes which contain a value of explanatory variables.
- iii. Furthermore, due to the fact that each node connection represents a weight factor the information then reaches a hidden layer node as a weighted calculation of its inputs.
- iv. The nodes of the hidden layer then pass the processed data through a nonlinear activation function.
- v. This is then processed by the output layer providing the calculated value is above the threshold (determined by the back propagation of errors algorithm).
- vi. Finally, the processed outputs are then validated to measure whether the network needs to be retrained in order to better fit the data series.

2.6.1 The Multi-layer Perceptron Model

The multi-layer perceptron allows the user to select a set of activation functions to explore including identity, logistic, hyperbolic tangent, negative exponential and sine². These activation functions can be used for both hidden and output neurons. MLP also trains networks using a variety of algorithms such as gradient descent, conjugant descent and BFGS (Broyden, Fletcher, Goldfarb and Shanno). Here the logistic activation function and gradient descent algorithm are used.

The network architecture of a conventional MLP network can best be illustrated as seen in figure 1.

¹ As such, neural networks are often considered 'black box' as they fail to show the significance of each input. The way the network weights independent variables to form the forecasted outputs is also unclear.

² This activation function is considered to be non-monotonic in that it is difficult to make weights vary sufficiently from their initial position. Therefore, this can result in much larger number of local minima in the error surface (Sopena *et al* (1999)).

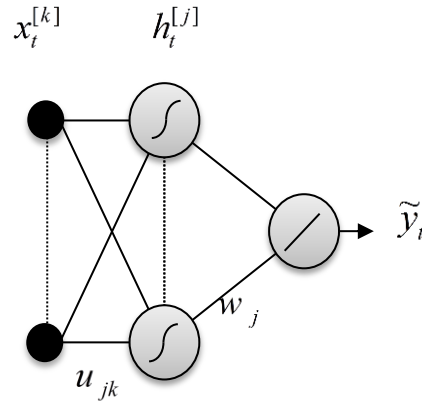


Figure 1. A single output, inter-connected MLP model (2 neurons / nodes)

where:

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

$h_t^{[m]}$ ($m = 1, 2, \dots, j + 1$) hidden node outputs (including the hidden bias node)

\tilde{y}_t MLP model output

u_{jk} and w_j network weights



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

(4)



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

(5)

The error function to be minimised is: $E(u_{jk}, w_j) = \frac{1}{T} \sum_{t=1}^T (y_t - \tilde{y}_t(u_{jk}, w_j))^2$

(6)

with y_t being the target value and T the number of trading days.

Training and selection of a network is halted once profit (in the form of an annualised return) is at its greatest during the in-sample period. Once the network weights are optimised during in sample training, these are then applied to an out-of-sample trading simulation. Parameters for the MLP network models are presented in the appendix.

2.6.2 The Recurrent Network

Recurrent Networks are an adaptive neural network with asymmetric connections that are related to the ‘Hopfield network’. While a complete review of RNNs is beyond the scope of this study the architecture that was adopted for this investigation can be seen in figure 2. It is however important to understand its differences when compared to the other two networks (MLP and HONN). Recurrent networks consist of both feed-forward and feed-back connections which enable them to retain information for later use (Draye *et al.* (1996)). As a consequence, a standard recurrent network comprises of a greater amount of neuron connections compared to the other two models implying longer computational times during the training process. Each of the neurons is graded using an activation feedback function to create forecasts (see Tenti (1996)). Furthermore, forecasts are created by adaptively modifying the synaptic weights within this model utilizing the generalisation of the delta rule introduced by Rumelhart *et al.* (1986).

Past papers have evaluated the effectiveness of RNNs and discovered that, due to their additional memory inputs, they can sometimes yield better annualised returns in comparison to simple MLP networks. Additive neural networks have been extensively studied in both their continuous-time and discrete-time versions. More recently, Tino *et al.* (2001) and Haschke and Steil (2005) elaborate further on the benefits of using RNNs.

A simple illustration of the architecture of an Elman (1990) RNN is presented in figure 2.

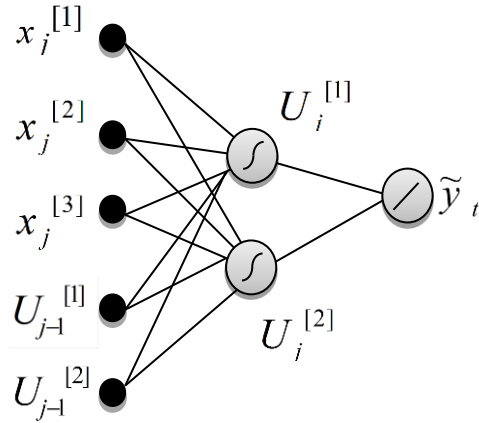


Figure 2. Elman recurrent neural network architecture with two neurons / nodes for the hidden layer.

where:

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

\tilde{y}_t recurrent model output

$d_t^{[f]}$ ($f = 1, 2$) and $w_t^{[n]}$ ($n = 1, 2, \dots, k + 1$) network weights

$U_t^{[f]}$ ($f = 1, 2$) output of the hidden nodes at time t

 transfer sigmoid function: $S(x) = \frac{1}{1 + e^{-x}}$

(7)

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

(8)

The error function to be minimised is:
$$E(d_t, w_t) = \frac{1}{T} \sum_{t=1}^T (y_t - \tilde{y}_t(d_t, w_t))^2$$
 (9)

In summary, the RNN architecture has the potential to provide more accurate outputs because the inputs are (potentially) taken from all previous values (see inputs $U_{j-1}^{[1]}$ and $U_{j-1}^{[2]}$ in figure 2 above).

2.6.3 The Higher Order Neural Network

Higher Order Neural Networks (HONNs) were first introduced by Giles and Maxwell (1987) and were called “Tensor Networks”. Although the extent of their use in finance has so far been limited, Knowles *et al.* (2005) show that, with shorter computational times and limited input variables, “the best HONN models show a profit increase over the MLP of around 8%”. Fulcher *et al.* (2006) elevate HONNs forecasting ability to be distinctly superior in comparison to other types of neural networks as they are considered to be more ‘open box’ whereas the majority of neural networks are commonly classified as ‘black box’ methodologies. As explained further by Giles and Maxwell (1987), HONNs exhibit adequate learning and storage capabilities due to the fact that the order of the network can be structured in a manner which resembles the order of the problem.

While they have already experienced some success in the field of pattern recognition and associative recall³, HONNs have not yet been widely used in finance. The architecture of a three input second order HONN is shown below:

³ Associative recall is the act of associating two seemingly unrelated entities, such as smell and colour. For more information see Karayiannis and Venetsanopoulos (1994).

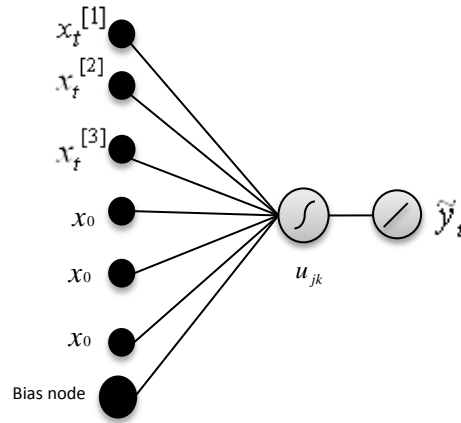


Figure 3. Second order HONN with three inputs (1 neuron / node).

where:

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

\tilde{y}_t HONNs model outputs

u_{jk} network weights

model inputs.



transfer sigmoid function: $S(x) = \frac{1}{1 + e^{-x}}$

(10)



a linear function:

$$F(x) = \sum_i x_i$$

(11)

The error function to be minimised is: $E(u_{jk}, w_j) = \frac{1}{T} \sum_{t=1}^T (y_t - \tilde{y}_t(u_{jk}))^2$

(12)

with y_i being the target value.

HONNs use joint activation functions to reduce the need to establish the relationships between inputs when training. Furthermore, this function also reduces the number of free weights and as a consequence the training procedure for HONNs is less time consuming compared to other neural networks. Due to the nature of HONNs and the fact that the number of inputs can be numerous, orders of 4 and over are rarely used. Another benefit of reducing free weights is that issues of ‘over fitting’ and local optima which are known to affect neural network results can be largely avoided. For a more comprehensive and thorough investigation into HONNs, please refer to Zhang and Qi (2005) and Knowles *et al.* (2005).

The HONN methodology was estimated in line with parameters also used for both MLP and RNN networks. Therefore the training process was stopped once satisfactory annualised returns were produced during the in-sample simulation. Parameters for the HONN network are presented in the appendices of each chapter.

2.6.4 The PSO Radial Basis Function Neural Network Model

A Radial Basis Function (RBF) Neural Network is a feed-forward neural network where hidden layers do not implement an activation function, but instead a radial basis function. As discussed by Park *et al.* (2002), input values in an RBF network are each assigned to a node in the input layer and then passed directly through to the hidden layer without weights. On the other hand, traditional neural networks such as the MLP pass inputs through to the hidden layer as weighted computations.

The Particle Swarm Optimization (PSO) aspect introduces a hybrid approach to the training of a network and hence the refinement of its forecasting accuracy has been compared to that achieved by Genetic Programming Algorithms. PSO was first introduced by Kennedy and Eberhart (1995) as a stochastic optimiser during the neural network training process. Kennedy and Eberhart

(1995) developed the PSO algorithm based on observations found within nature such as the social behavior found within a flock of birds or a school of fish. With these observations as a basis, the algorithm is developed to search a fixed space in attempt to identify optimal positions within this space to best solve a predefined problem. In particular, PSO optimization reduces the time it takes to train neural networks by simplifying the complex calculations found within traditional Neural Networks and determining the optimal number of hidden layers⁴. Many academics have previously researched standard Radial Basis Function Neural Networks however the combination of PSO and Neural Networks is relatively new to time series analysis. As explained by Chen and Qian (2009), PSO optimizes parameters within a traditional RBF. In particular, this optimization helps overcome inefficiencies associated within the standard back propagation algorithm.

The RBF neural network approximates a desired function by the superposition of non-orthogonal, radially symmetric functions as discussed in more detail by Theofilatos *et al.* (2010). The networks architecture is depicted below in figure 4 (Sermpinis *et al.*, 2013).

⁴For the purpose of forecasting, the proposed PSO RBF model utilises a constant layer of 10 neurons. Tests were conducted using the algorithm to search for the ‘optimal’ number of hidden neurons. Results from these tests produce a lot more than 10 neurons and as a result the PSO RBF was found to ‘over-fit’ the data in most cases. This can be checked by observing the best weights output and comparing training using fewer fixed neurons with what the algorithm would use if it was tasked with identifying the ‘optimal’ number of neurons. With this in mind, a number of experiments were run using varying numbers of hidden neurons. All of the PSO RBF parameters are provided in appendix A.4. The best weights for each of the models are included in appendix A.5.

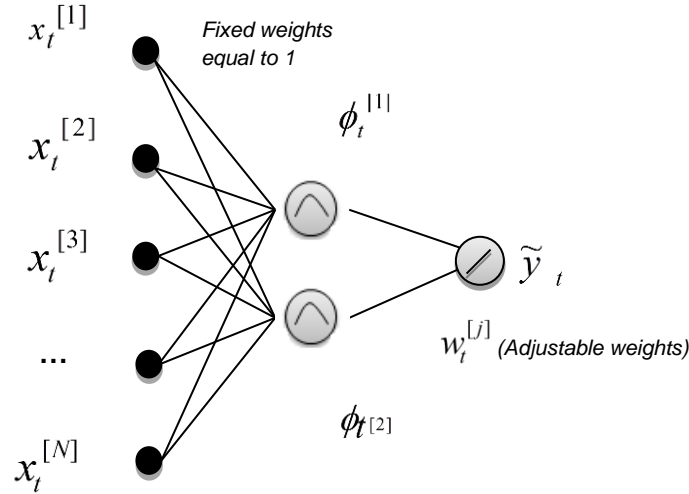



Figure 4. Radial basis function neural network (with two hidden nodes)

Here, the Gaussian radial basis function is used in the hidden layer (as seen in equation 13) as this is the most common found in existing financial time series literature.

x_t ($n = 1, 2, \dots, N + 1$) are the model inputs (including the input bias node)

\tilde{y}_t RBF model's output

$\phi_t^{[j]}$ ($j=1,2$) are the adjustable weights

 is the Gaussian function: $\phi_t^i(x) = e^{-\frac{\|x-C_i\|^2}{2\sigma_i^2}}$

(13)

where: C_i is a vector indicating the centre of the Gaussian Function and σ_i is a value indicating its width. C_i , σ_i and the weights w_i are parameters which are optimized by the PSO algorithm during a learning phase while training the RBF neural network.



is the linear output function: $U(x) = \sum_i x_i$

(14)

The error function to be minimised is: $E(C, \sigma, w_t) = \frac{1}{T} \sum_{t=1}^T (y_t - \tilde{y}_t(w_t, C, \sigma))^2$

(15)

with y_t being the target value and T the number of trading days.

In order to maximise annualised returns an additional fitness function is employed as defined below in equation 16. This approach was first introduced by Sermpinis *et al.* (2013).

The annualised return function to be maximized is:

$$R^A - MSE - (n * 10^{-2})$$

(16)

where: R^A = annualised return

MSE = mean square error defined in equation 10.⁵

n = number of inputs.

The R^A terms range from -0.4 to 0.5 while experimental results indicated that the maximum value for the MSE term is 0.01. These parameters are established so that the algorithm can primarily search for profitable forecasts with statistical performance becoming of secondary importance.

⁵The number of hidden neurons is multiplied with 10^{-2} because the simplicity of the derived neural network is of secondary importance compared to the other two objectives (maximize the annualized return and minimizing the MSE).

The hybrid methodology of combining a Particle Swarm Optimizer with an RBF Neural Network was first inspired by Li *et al.* (2008) and is also an extension of the PSO RBF methodology proposed by Sermpinis *et al.* (2013). The Particle Swarm Optimization PSO methodology is used to locate the parameters C_i , of the RBF neural network, while at the same time locating the optimal feature subset which should be used as inputs to the RBF network.

The complexity of a traditional neural network is reduced by applying the PSO algorithm to refine the training process. As applied by Theofilatos *et al.* (2010), the PSO algorithm encodes network weights as particle components with each particle evaluating inputs based on minimizing the error function in equation 15. PSO parameters are also ‘adaptive’ as depicted in equations 17 – 19. This proves beneficial to a wider range of users. Therefore ‘velocity’ as described originally by Kennedy and Eberhart (1995) is adaptable with the algorithm retaining knowledge of an input’s (particle) best position within the population (swarm).

With the PSO algorithm the traditional neural network weight matrix is reorganized as an array of randomly initialized particles to commence the optimization procedure. During this search the PSO algorithm is assessing ‘global’ and ‘local’ variants. A local variant is an individual particle’s best solution achieved thus far while the global variant is the best solution achieved in the entire population of particles. Furthermore, Mohaghegi *et al.* (2005) note that particles have a tendency to repeat their past behavior (cognitive) as well as follow the behavior of those particles deemed ‘fit’ (socialization). The eventuality of this behavior is that the population of particles converges to create an optimal solution. Upon the completion of iterations the particles return to their best position which is identified during the search / training process. Predefined parameters for the PSO algorithm can be found in the appendix. For a more detailed explanation please refer to Eberhart *et al.* (1996) and Theofilatos *et al.* (2010).

$$W_{(T)} = (0.4/N^2) * (T-N)^2 + 0.4 \tag{17}$$

$$c1_{(T)} = (-2) * T/N + 2.5 \tag{18}$$

$$c2_{(t)} = (2) * T/N + 0.5 \tag{19}$$

where: T is the current iteration.
N is the total number of iterations.

Weights are decreased from 1.0 to 0.4 during the training phase in search of a candid solution to the proposed problem. In selecting the appropriate training set the termination criterion applied to the PSO algorithm is 10^{-3} . Ultimately, training is stopped once the number of iterations reaches 100 or the profit in the form of annualised returns is at its maximum.

3.0 Genetic Programming Algorithm (GPA)

Evolutionary algorithms have been applied to financial time series since the early 90's however in more recent years further developments of these algorithms have been witnessed. A timeline of Genetic Algorithms (GA) has seen a progression from fixed length character strings (Holland (1975)) to hierarchical variable length strings (Koza (1992)), to Genetic programming algorithms (GPA) represented in tree like structures, and more recently Genetic expression programming (GEP) has been added to this evolutionary family. In particular, GPA as an application used for predicting financial time series is a relatively new forecasting methodology.

Neely *et al.* (1997) explore the use of genetic programming to search for optimal technical trading rules and encode these rules in the form of non-recombining trees. Li and Tsang (1999) use an earlier evolutionary Genetic Algorithm (Koza (1992)) in order to forecast the Dow Jones Industrial Average (DJIA).

The GP application is programmed and implemented to evolve tree based structures which represent models (sub-trees) of input-output (see figure 5 below). The application builds algebraic expressions in order to calculate next day returns from a variety of inputs. Once the GP application arrives at an optimal expression during the in sample period this is then carried forward and applied during an out of sample period. In the design phase of the GP application the focus is primarily on execution time optimisation as well as limiting the ‘bloat effect’. The bloat effect is similar to the issue of ‘over fitting’ experienced in neural networks. In the case of a GP application the risk of continuously increasing and expanding the tree size is instead present. This algorithm is run in a ‘steady state’ with a single member of the population being replaced at a time. In comparison to other algorithms (such as typical generational GAs) steady state algorithms have greater selection abilities and lower genetic drift. Steady state algorithms also offer exceptional multiprocessing capabilities (Ferreira (2006)). In principle, the GP application reproduces newer models replacing the weaker ones in the population based on ‘fitness’.

The genetic tree structure consists of nodes (depicted as circles in figure 5) which represent functions that exist to perform actions within each structure. The purpose of each function is to generate output signals whereas the square-like symbols are terminal functions representing the end of a function and indicating that the most superior sub-tree (model) has been reached. For example, the tree structure in figure 5 is characterised by the algebraic expression $4 / x_1(t-1) + \ln(x_2(t-2))$. In this example there are 3 terminal nodes expressed as $x_1(t-1)$, $x_2(t-2)$ and 4. The non-terminal nodes however represent the functions $/$, \ln and $+$. Furthermore, each individual in the population

corresponds to a single sub-tree structure with each being limited by a predefined maximum tree size of 6 in order to avoid the ‘bloat effect’. During the in sample back test a tree size of 6 is decided based on trial and error optimisation. Furthermore, this is in line with existing literature as Iba (1999) also use a maximum tree depth of 6 in order to forecast Japanese stock market prices. Santini and Tettamanzi (2001) who use a genetic algorithm to predict the Dow Jones set their expression max depth to 5.

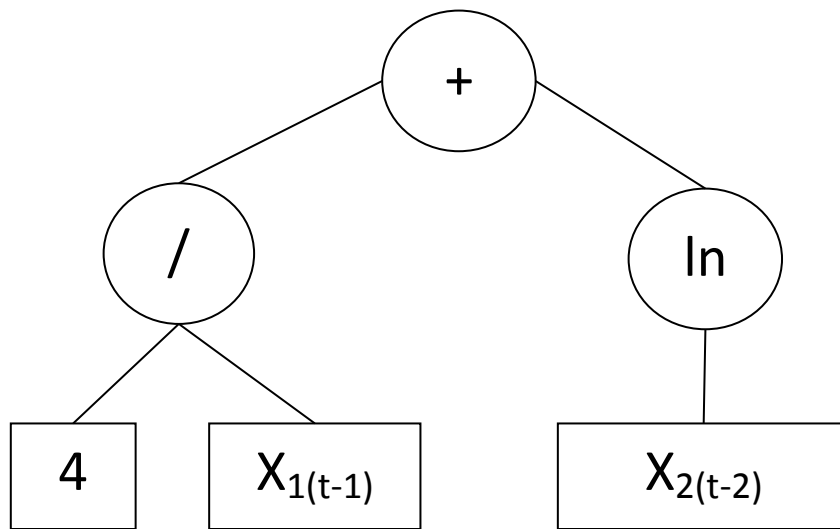




Figure 5. Generic tree structure

where:

 denotes a function symbol/non-terminal node
 denotes a terminal symbol/terminal node

Koza (1998) summarises the functionality aspect of the GP algorithm in the following steps:

- (1) The generation of an initial population of randomly constructed models (Generation 0) is developed with each model being represented in a tree like structure of functions and terminals suitable to the

problem. This initial generation serves as a basis for any future creations of generations therefore it is important that it provides an adequate amount of solutions that are spread out across as much of the ‘search space’ as possible. Thus, our initial population is created by executing basic functions and terminals in order to initiate the process of evolution in search of optimal models which offer solutions to the problem. Additionally, each individual (tree structure) of the population is of variable length (i.e. total number of functions and terminals) and of different structure. In most cases, it is normal for the majority of these models to be considered ‘unfit’ solutions to the problem however ideally the model should also present a valuable array of fitness cases. This variety of fitness cases enables the algorithm to establish which individuals are fitter than others. Ultimately, it is the nature of Genetic Programming which enables the exploitation and manipulation of these different fitness cases until the best fitting models, in terms of least error, are produced.

- (2) Following this initial generation of randomly selected models a random subset (sub tree) of the population is then selected for a tournament in the tournament selection phase. This process (tournament procedure) is essentially a selection mechanism in order to decipher which individuals from the population are to be chosen for reproduction to develop the next generation.
- (3) An evaluation of the members of this subset is then carried out and assigned a fitness value. As stated by Koza (1998) the fitness cases are either selected at random or in some structured manner (e.g. at regular intervals). In our application, as mentioned briefly in the first step, the fitness value is defined as the mean squared error (MSE) with the lowest MSE being targeted as the best.⁶
- (4) Following the establishment of fitness values the tournament winners are then determined in order to create a new population. To reiterate, the winners of this scenario are the models with the lowest MSE.

⁶ Other statistical measures that can be used in order to determine the fitness value are the sum of the absolute value of the differences between the output produced by the model and/or the desired output (i.e. the Minkowski distance) or, alternatively, the square root of the sum of the squared errors (i.e. the Euclidean distance). It is also worth noting that on occasions when individuals provide suitable solutions and arrive at terminal nodes then it suffices to assume that these are ‘fit individuals’.

(5) Having identified the tournament winners in the previous step we then proceed by exposing the models to two genetic operators known as mutations and crossovers. Both operators are discussed in more detail below:

(5a) Genetic Operators (Generation of new populations):

The two genetic operators that are used in this algorithm are mutations and crossovers. In principle the mutation operator creates a new model from an existing one (traditionally known as a unary operator) while the cross-over model creates a new model from two existing models. The latter is therefore traditionally considered a binary operator.

i.) **Mutation:** In this process one mutation point is indiscriminately chosen as an independent point and the resulting sub-tree is to be omitted. From this resulting sub-tree, another new sub-tree is then reproduced using the same procedure that was initially implemented to create the original random population. Although this was the procedure implemented for mutation during this study there are also a number of alternative methods which are explored in other research.

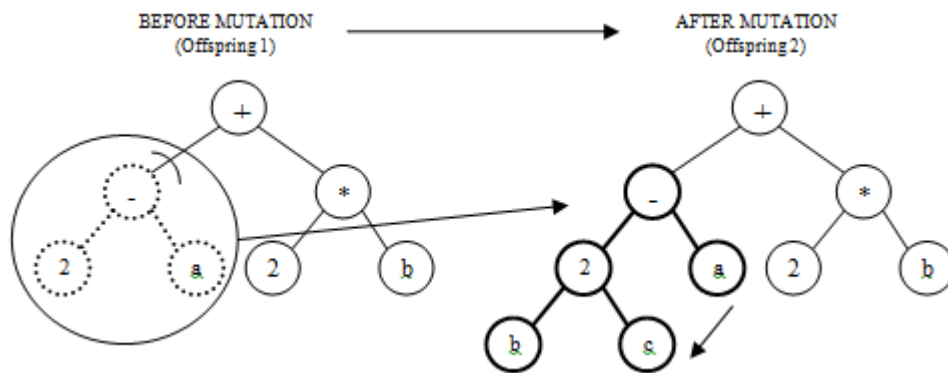




Figure 6. Mutation of a tree structure

where:
 denotes the original sub-tree (model)
 denotes the new sub-tree (model)

⌋ denotes the mutation point

ii.) **Crossover:** This operator creates two new models from existing models by genetically recombining randomly chosen parts. A random crossover point is chosen from each 'fit individual' and recombined with another to create superior offspring. More specifically, the models are selected based on their fitness and the crossover allocates future trials to regions of the search space whose models contain parts from superior models. As a full explanation of crossovers is beyond the scope of this paper, please refer to Koza (1992) for more details.

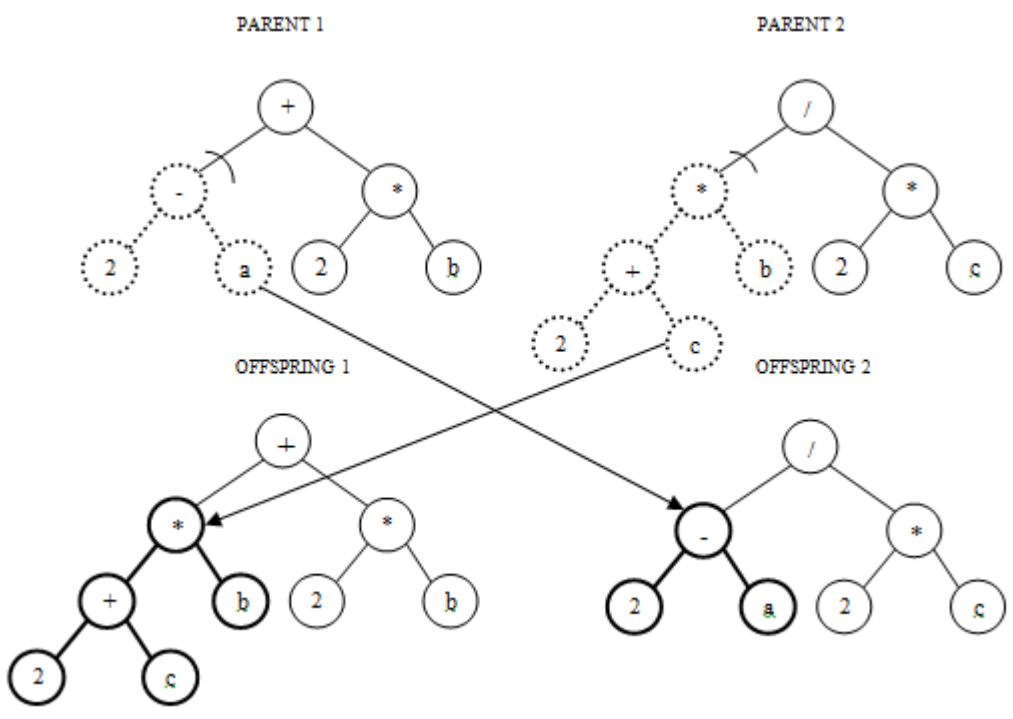




Figure 7. Crossover family tree-like structure

where:  denotes the original sub-tree (model)
 denotes the new sub-tree (model) produced from the crossover operator.

denotes the crossover point

- (6) The population is then altered with the tournament losers being replaced by the winners (superior) offspring. Parallels may be drawn to that of natural selection found in nature.
- (7) Provided the termination criterion is not reached, the algorithm returns to step 2 and these steps are repeated until the predefined termination criterion for genetic programming is satisfied. The termination criterion for this study is set to 100,000 generations at which point the cycles are stopped and forecasted results can be obtained.
- (8) Ultimately, optimal models from the population emerge offering a forecast for next day returns specific to the problem.

Given that the generation of the initial population is randomly constructed as discussed above, forecasts may differ between GP algorithms. In order to eliminate any variance between our GP forecasts, an average was derived from a committee of 10 GP algorithms all of which produced the highest profit during the training sub-period. Taking the average from a number of GP results is commonplace in GP literature. For one, Aranha and Iba (2008) forecast stock market returns using an average of 30 returns derived from 30 different models.

CHAPTER 3: Modelling and Trading the Corn/Ethanol Crush Spread with Neural Networks

April 2011

Abstract

The recent addition of ethanol futures to the CBOT exchange has provided market participants with further hedging and speculative opportunities. In particular, this spread provides farmers, commodity processors and grain elevators with a hedging tool to directly manage price risk exposure. This paper investigates the ‘Crush’ spread trade between corn and ethanol futures commonly known as the ‘Corn Crush’ spread. A spread trading system based on daily closing prices for each of the commodities over a 5 year horizon is constructed using various neural network architectures. Multilayer Perceptron (MLP) Neural Network (NN), Recurrent NN (RNN), and Higher Order NN (HONN) architectures are all applied to the task of forecasting next day spread returns. Results produced by each of these NN models are compared to linear trading models such Naive and MACD trading strategies as well as an ARMA model in order to measure effectiveness.

From the analysis the HONN outperforms all of the other forecasting methods in terms of both trading performance and statistical accuracy. The RiskMetrics volatility market timing filter also enhances annualised returns while reducing volatility and maximum drawdowns. Furthermore, the corn crush spread is found to display similar characteristics to the soybean crush spread as observed by Dunis *et al.* (2006-b) and as stated by the CBOT (2007).

Keywords

Spread Trading, Corn Futures, Ethanol Futures, Higher Order Neural Networks, Multilayer Perceptron Neural Networks, Recurrent Neural Networks.

1.0 Introduction

The motivation behind this paper derives from the recent surge in prices for agricultural commodities. Arguably, this is only the beginning of drastically rising and volatile prices to be experienced in the agricultural industry. For the most part, this is due to rising global populations with the improving economies of China and India exerting the most authority on an upward trend in world food prices. With this in mind, a whole new tier of middle class is beginning to emerge in these developing economies consuming greater quantities of meats and grains. Another factor such as the impact of climate change on agriculture is also considered to be one of the key reasons for previously unseen price swings. In addition, the use of ethanol as an alternative fuel has also had its effect on food prices. These influential forces have a global impact affecting both developing and developed economies.

Rising volatile commodity prices have also lead to an increase in the number of market participants in these agricultural markets. For instance, farmers, commodity processors and grain elevators all use these financial markets to manage risk and hedge against adverse price movements. On the other hand, speculators are also drawn to these markets primarily to make profits and to take advantage of diversified investment strategies. These opportunities present themselves as a result of growing world populations and climate change, as mentioned above, as well as technological advances in bio fuels. Ultimately, the increase in demand for agricultural commodities coupled with an uncertainty of supply and ever increasing investment opportunities are all to blame for the most recent surges in prices across agricultural commodities markets.

This investigation aims to rigorously evaluate the profitability of the Corn – Ethanol spread. The profit margin created from the Corn - Ethanol spread is achieved from the process of converting corn into ethanol. This procedure involves extracting the carbohydrates stored in corn to create simple sugars in order to produce the valuable by-product known as ethanol. As a consequence, the ethanol industry is one of the fastest growing industries in the United States with production growing from 175 million gallons in the 1980's to almost 6.2 billion gallons in 2007⁷. The future prospects of the ethanol market appear to be extremely prosperous with the Energy Independence and Security Act (EISA) of 2007 being passed

⁷ Ethanol Facilities: U.S. *Ethanol Production*, AMERICAN COALITION FOR ETHANOL, <http://www.ethanol.org/index.php?id=37&parentid=8> (last visited on November 25, 2009)

encouraging the additional construction of ethanol plants to accommodate for the sharp rise in demand for ethanol as an alternative bio fuel. Furthermore, this act sets forth a mandate that gasoline consumption must include 15 billion gallons of ethanol to be produced in the United States by the year 2015. The underlying stimuli behind increasing the production of ethanol include the of lessening U.S. dependence on foreign oil imports as well as efforts to quell pressures from environmental activists who call for the use of alternative cleaner renewable energy. Moreover, with U.S. crude oil prices reaching an all-time high in July 2008 at \$147.27⁸ a barrel it has become apparent that alternative cheaper bio fuels are essential.

Although corn based ethanol as a bio fuel has many virtues its efficiency as a renewable energy has also been open to widespread criticism. Many feel that it is not as efficient as other sources such as soy biodiesel and sugar cane based bio fuels (Shapouri *et al.*, 2002). In particular, sugar cane based bio fuels are widely produced in Brazil however the United States have imposed high tax levies on these imports to make them less attractive in an effort to suppress international competition. As a result of widespread production, ethanol as a bio fuel has also become highly controversial within the US creating a ‘tug of war’ scenario. On one side, cattle farmers are arguing that corn is more valuable as feed for livestock, while on the other, ethanol manufacturers and politicians are steadfast in their promotion of corn as feed for bio fuel mass production. This has developed into what is now known as the ‘food for fuel’ debate. At the heart of this dispute is the fact that increasing ethanol production induces a higher demand for corn and hence increases the average price of corn as highlighted by Shapouri *et al.* (1995). In effect, this then makes it more expensive for farmers to feed their livestock as corn is one of the main grains used in the feed process. As a knock on effect this is also reflected in the prices of meats, dairies and various other related products. Additional mandates such as those set out by the Environmental Protection Agency (EPA) currently dictate regulations that require oil producers to maintain a ‘blend’ of 90% gasoline / 10% ethanol mix. As long as these types of mandates are in place it seems only sensible to assume that the agricultural industry will experience ever increasing feed prices as fuel is now competing with food for cropland. For instance, this

⁸ Financial Times: Commodity markets in worst annual fall. December 31, 2008. <http://www.ft.com/cms/s/0/52ea3658-d72d-11dd-8c5c-000077b07658.html>: [Accessed:13/11/2009]

issue recently surfaced in the 2006 – 2007 harvest when unprecedented high grain prices were experienced, in part, due to the pressure for additional corn acres to meet the growing needs of the ethanol industry.

The main objective of this paper is to develop profitable trading simulations from speculatively trading the corn / ethanol ‘Crush’ spread. The analysis covers a 5 year horizon commencing when the ethanol futures contract was first traded on the Chicago Board of Trade (CBOT) exchange (March 23, 2005). The relationship between the two commodities is investigated by analysing the spread created from their daily closing prices with the application of optimal neural network forecasting architectures. The objective is to exploit and evaluate the relationship between the two underlying commodities in order to model, forecast and profitably trade the Crush. In addition, this investigation also aims to build on earlier work carried out by Dunis *et al.* (2006-b), who investigate the soybean-oil crush spread comparing the effectiveness of various neural network architectures to more conventional forecasting techniques. With the motivations for carrying out this research reviewed above, further investigation into the mentioned commodity futures is warranted.

The remainder of this paper is organised as follows. Section 2 provides a review of past and current literature. Section 3 discusses the descriptive statistics behind the ‘Crush’ spread. Section 4 details the various methodologies employed for this investigation, section 5 presents the empirical results and finally section 6 concludes with a summary of observations and limitations.

2.0 Literature Review

Numerous studies test the application of technical trading rules to trading financial assets and provide evidence that they are valuable tools for manipulation of financial time series. For instance and perhaps more specific to this paper, studies carried out by Kaastra and Boyd (1995), Trippi and DeSieno (1992), and Dunis (1989) all justify the use of technical trading rules as effective avenues to trading financial futures markets. Bessimber and Chan (1995) and Beja and Goldman (1980) also provide justification for the use of technical trading rules especially during times when the market is experiencing ‘informational inefficiencies’. During these times savvy investors have a limited window of opportunity in which to benefit

from econometric analysis. Moreover, due to their effectiveness trading strategies have been around for decades with some proving to be more rewarding at times than others. However, the scope of this paper is primarily focused on a trading strategy that has become increasingly popular due to the emergence of a wider range of financial products. This trading strategy is commonly referred to as spread trading and now commodity processors and farmers can directly hedge risks associated with processing margins.

The earliest literature on spread trading is by Working (1949). In his research, inter-temporal price relationships of futures contracts are evaluated with the application of a spread trading system. From this analysis, he uncovers numerous opportunities for traders to speculate on price irregularities between different futures contracts. In effect, this provided an insight into the strategy's initial purpose that involves speculating on the 'cost of carry' between different futures contracts.

More recently however, there have been subsequent studies which have uncovered additional benefits of the strategy. Most notably, Meland (1981) provides evidence that spread trading is also a valuable way of creating market liquidity. He indicates that spread trading not only provides speculative opportunities but can also be utilised by arbitrageurs and hedgers alike. Peterson (1977) discusses the benefits of spread trading further by highlighting the fact that it also increases the amount of investment opportunities. Dunis *et al.* (2006-a) discuss that spreads offer an 'affordable alternative approach' to investing. Additionally, they also explain that spreads are less likely to suffer from information shocks with the movement of the two participating legs acting to effectively eliminate this by offsetting each other in such circumstances. With this in mind, 'speculative bubbles' as explained by Sweeney (1988) tend not to be associated with investment strategies such as spread trading.

2.1 Spread Trading Agricultural Futures

Historically, agricultural futures markets were primarily used as platforms in which one could hedge against price risk exposures associated with price movements in the cash market. This view was first established by Working (1953, 1954, 1960 and 1962) who argued that agricultural futures markets are used mainly for hedging purposes and that speculation is dictated by the volume of this hedging activity.

However, with increased participants and technological advances, speculation plays a more prominent role in today's agricultural futures markets.

On review of existing literature it becomes apparent that the virtues offered from hedging a Corn Crush spread over short term horizons are investigated by some however there is limited literature regarding spread trading of agricultural commodity markets as a vehicle to hedge or speculate in the longer run. For one, Dahlgran (2009) investigates the effectiveness of one-through eight-week hedges over a three year horizon. In particular, part of his investigation examines the effectiveness of corn crush hedging as a risk management vehicle covering the period of March 23rd, 2005 to December 31st, 2008. Dahlgran (2009) concludes that the effectiveness of hedging a Corn Crush is comparable to results yielded from a soybean crush. Hence, as a risk management tool, the corn crush hedge offers ethanol producers similar price risk protection as experienced by soybean processors who utilize the soybean crush hedge. Finally, he implies that the corn crush hedge may cater for more widespread use other than hedging. In support of his findings, the CBOT (2007) also promotes the 'corn crush' hedge as analogous to the soybean crush hedge. The limited literature review regarding speculation of the corn crush spread can perhaps be attributed to the fact that ethanol has only been traded on the CBOT as a futures contract since early 2005. Franken and Parcell (2003) explain that prior to the availability of ethanol futures contracts on the CBOT, ethanol price risk was cross-hedged with unleaded gasoline futures. However, with the recent creation of an ethanol specific futures contract, opportunities have arisen that enable direct hedging. One can now hedge against the price risk associated with holding ethanol stock as well as safeguarding against price adversities linked with processing corn into ethanol.

Dunis *et al.* (2006-b) explore agricultural spreads in the form of the soybean-oil crush spread. They investigate the profitability of spread trading a soybean-oil spread over a horizon of 10 years spanning from 01/01/1995 – 01/01/2005. The effectiveness of various neural network architectures is benchmarked against more conventional forecasting techniques such as the fair value co-integration model. The analysis concludes that profitability is present when trading such a spread with Higher Order Neural Networks (HONNs) proving to produce the highest annualised out-of-sample returns. Hence, HONNs possess superior

forecasting abilities to those of Recurrent Neural Networks (RNN), Multilayer Perceptron (MLP), and Fair Value Co-integration when tasked with forecasting next day returns for the soybean-oil spread.

2.2 Seasonality of Agricultural Futures

Weather uncertainty is a fundamental aspect in commodity trading as significant weather events can dramatically impact the supply of a commodity. Such seasonal behaviour is inherent in the grain market and pricing is dictated by factors such as when the commodity is planted, pollinated and harvested. Due to the nature of agricultural commodities seasonal behaviour is experienced in annual cycles with markets tending to move in given directions throughout certain times of the year.

Corn as an agricultural grain commodity is broken down into three periods over a typical year as displayed in table 1.

Period	Months	General Market Sentiment
Late Spring to Mid-Summer	March – June	Bullish (weather premium)
Mid-Summer Harvest	July – September	Bearish (supply is plentiful and major threats to crop supply have now passed)
Post Harvest	October - February	Bearish (A sharp decline during the ‘February Break ⁹ ’ is often experienced in Corn prices)

Table 1. Typical corn cycle - seasonal sub-periods

Till and Eagleeye (2004) discuss the nature of corn futures prices and the price pressure effect that is prevalent in commodity futures contracts. Their investigation uncovers the nature of commodity contracts, such as corn, discovering that a fear premium is commonly added into their pricing. This is particularly common in the build-up to a harvest season in circumstances when adverse weather conditions are forecasted. It is common for grain markets to assume a rather pessimistic view which creates this ‘premium’ during times when real or perceived threats to food supply are forecasted. With this in mind, the seasonality of corn is such that grain futures prices are inclusive of this ‘premium’ especially when approaching the U.S. harvest season in the months foregoing that of July. However, as the harvest season progresses and providing adverse weather does not occur then the ‘weather premium’ in corn gradually diminishes from the

⁹ One of the most common seasonal patterns experienced in grains and soybeans. Usually follows a short rally which occurs during the post-harvest months of December and January.

fair value price of the contract and this is reflected by a gradual decrease in the futures price of corn. For instance, at the end of July, the weather conditions affecting the corresponding year's harvest would have either already occurred or may not have been present at all. Therefore, late July futures prices are less inclined to include a weather premium. Furthermore, in months following July the crop is entering its harvest season where seasonal lows are prevalent as a result of supplies being plentiful. This bolsters confidence and perceived security in the commodity.

Till and Eagleeye (2004) discuss various methods for trading commodities however they highlight that by using futures spreads the risk in of experiencing large losses are limited. Spread trading allows market participants to hedge for 'first-order' or exogenous risk. For example, when certain events occur such as Hurricane Katrina then both legs of the spread will be affected. As a result, a loss in one position is offset by a gain in the other due to the different views being taken on each leg. Generally this is when one leg is long and the other is short. Spread trading does however run the risk of experiencing timing differences when inventory cycles between the two participating commodities are not the same.

A full explanation of how seasonality is accounted for can be seen in section 3.4.

2.3 Application of Neural Network Architectures

Neural networks are computationally powerful and intuitive modelling tools that can be applied to financial time series in order to rationalise masses of data into knowledge useful for making investment decisions. As a consequence, the forecasting of time series' future trends has been analysed by many in the academic world. Fama and French (1986) determined that market prices exhibit, to some extent, a form of memory pattern within them and as a result future price trends do in fact contain an element of predictability based on historic prices. In recognition of this, a variety of forecasting techniques have been applied throughout the years. In more recent years however, there has been a noticeable increase in popularity for artificial intelligence with its rapid growth in financial time series analysis and forecasting. In particular, there has been a resurgence of neural networks in a variety of architectures such as the Multilayer Perceptron (MLP) (Lisi and Schiavo, 1999; Faraway and Chatfield, 1998; Hill *et al.*, 1996; Lachtermacher and Fuller,

1995; Jayawardena and Fernando, 1995); Recurrent Networks RNN (Freedman, 1994); Radial Basis Functions (RBF) (Fernando and Jayawardena, 1998; Hutchinson, 1994); and a comparative analysis of MLP and RBF (Fernando and Jayawardena, 1998; Jayawardena *et al.*, 1996). Neural Networks have become particularly popular in finance because they are a fairly robust computing tool with learning and adaptive capabilities. These characteristics have been utilised to accurately predict financial assets by successfully capturing and interpreting nonlinear relationships between explanatory variables and target outputs. Traditional statistical methods have proven to be less accurate due to the fact that in most cases they fail to disseminate nonlinear data and discontinuities which are both common in financial time series.

3.0 Descriptive Statistics and Data

The daily closing prices for each of the commodities contract months were obtained from Datastream for the period covering March 23, 2005 – December 31st, 2009. Corn futures¹⁰ are the most heavily traded agricultural commodity and have been traded on the CBOT¹¹ exchange since the mid 1800's. On the other hand, ethanol futures¹² are a more recent addition to the CBOT exchange having only been traded as a futures contract since March 23, 2005. As a result, ethanol is not traded as frequently as Corn and is therefore less liquid. Both contracts are traded from 09:30am to 13:15pm and 18:00pm to 7:15am CST¹³. As a result, the issue of non-synchronous pricing that plagues many other investigations does not affect the construction of a reliable trading dataset.

The Corn 'Crush' spread is calculated taking the fact that both commodity futures contracts are priced and traded in different units into consideration. Corn is priced in cents per bushel whereas ethanol is traded in dollars per gallon. Therefore, a conversion of prices into equal units is required. Currently, one bushel of corn yields approximately 2.8 gallons of ethanol (CME, 2010). To create a tradable spread between the two contracts the price of ethanol must be multiplied by 2.8 in order to convert it into dollars

¹⁰ CBOT corn contract specifications can be found in appendix

¹¹ Chicago Board of Trade

¹² CBOT ethanol contract specifications can be found in appendix

¹³ Central Standard Time (CST)

per bushel. Lastly, to calculate the corn ‘crush’ spread, the price of corn is subtracted from the price of ethanol (in dollars per bushel). This calculation is mathematically depicted in equation 20.

$$C_t = P_C - [(2.8 * P_E) / 100] \tag{20}$$

where: C_t = Price of the crush spread at time t (in cents per bushel)
 P_C = Price of the corn contract at time t (in cents per bushel)
 P_E = Price of the ethanol contract at time t (in dollars per gallon)

There are various other ways to calculate the spread depending on what the market participant is attempting to achieve. Other Corn Crush spread combinations may include distillers dried grains (another by product of corn) and natural gas as this is consumed during the ‘crushing’ process. However for the purpose of this investigation the analysis is solely focused on the relationship between Corn and Ethanol.

The methodology applied throughout this investigation to calculate the returns of the corn crush spread can be seen in equation 21 as also used by Butterworth and Holmes (2002) and more recently by Dunis *et al.* (2006-b):

$$\Delta S_t = \left[\frac{(P_{C(t)} - P_{C(t-1)})}{(P_{C(t-1)})} - \frac{(P_{E(t)} - P_{E(t-1)})}{(P_{E(t-1)})} \right] \tag{21}$$

where: ΔS_t = percentage change in the spread at time t
 $P_{C(t)}$ = is the price of corn at time t (in cents per bushel)
 $P_{C(t-1)}$ = is the price of corn at time $t-1$ (in cents per bushel)
 $P_{E(t)}$ = is the price of ethanol at time t (in cents per bushel)
 $P_{E(t-1)}$ = is the price of ethanol at time $t-1$ (in cents per bushel).

3.1 Statistical Behaviour of Commodity Prices

The spread time series for the full sample period can be seen below:

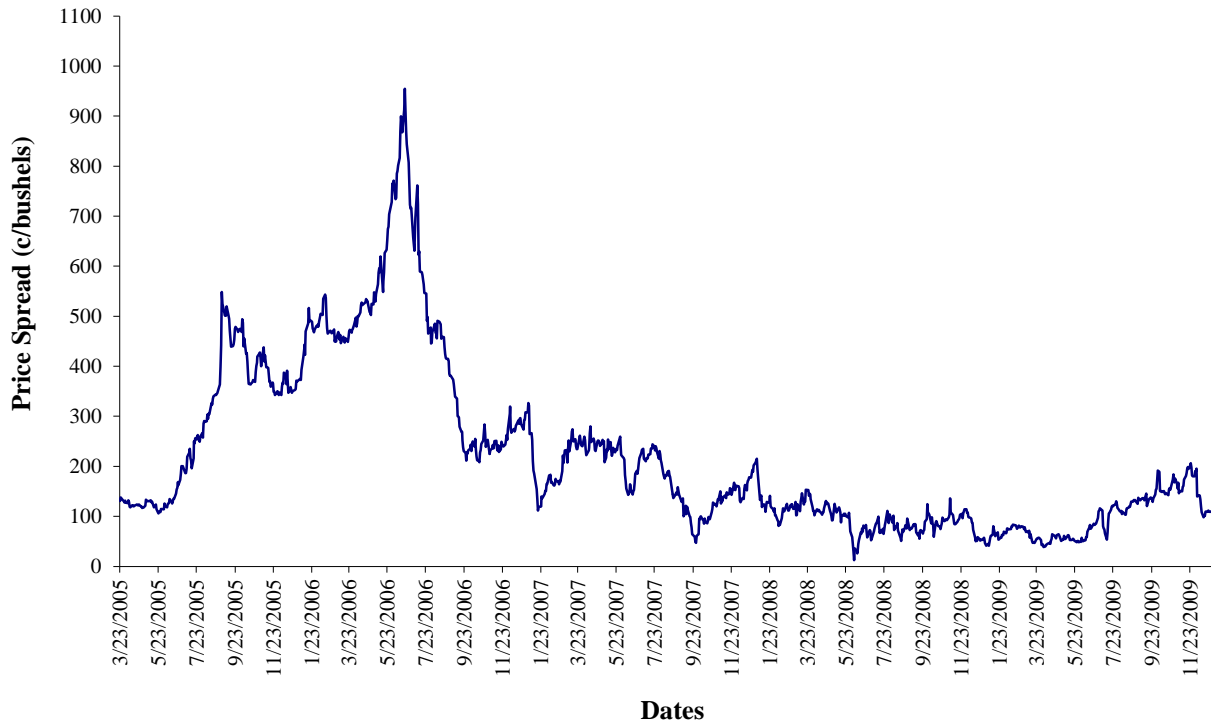


Figure 8. The corn-ethanol crush CBOT daily closing prices (23/03/2005 – 31/12/2009)

By observation of figure 8, it is apparent that post 2007 the spread displays mean reversion around 100 c/bushel. However, previous years are characterised by large deviations which are experienced in 2005 and 2006. These large deviations are as a result of shocks to both the demand for and supply of each commodity during the period of 2005 to 2006. Most notably, there are two major spikes in the spread occurring during the ‘in-sample’ period. The first of occurred on August 29, 2005 as a result of Hurricane Katrina which devastate the harvest in the South and Midwest of the United States. The second spike occurred as a result of a US government mandate which phased out MTBE (Methyl Tertiary Butyl Ether) oxygenates. Furthermore, this mandate called for ethanol refiners to increase their capacities in order to produce 4 billion gallons of ethanol in 2006, see McKay (2006).

The price behaviours of commodities have been observed and analysed by many over the past few decades. For one, Deaton and Laroque (1992) observe yearly prices for 13 of the most popular commodities (including corn) over a period spanning from 1900 to 1987. Findings identify a number of common pricing attributes associated with commodities. Most notably, even though commodity prices appear to be inherently

volatile they still remain mean reverting. Furthermore, the prices also display high degrees of autocorrelation even in normal times perhaps explained by seasonal patterns. In addition, Sorensen (2002) analyses the price behaviour associated with agricultural commodities between the periods of 1972 through to 1997. In this analysis the prices of soybean, corn and wheat are observed focusing primarily on permanent trend shifts, seasonality and mean reversion. Conclusions also reveal that commodity prices are generally mean reverting.

More recently however, Geman (2005) observes that on average commodity prices neither grow nor decline. As a result, Geman (2005) concludes that prices tend to mean-revert due to the marginal cost of production. Hence, mean-reversion is one of the main properties that have been systematically incorporated in the literature surrounding commodity price modelling.

3.2 Descriptive Statistics

Statistical analysis is based on the change in daily closing prices¹⁴ and from the histogram shown in figure 9 it can be observed that the corn/ethanol spread return series is non-normal (confirmed at a 99% confidence level by the Jarque-Bera test statistic), with a slight skewness and high kurtosis.

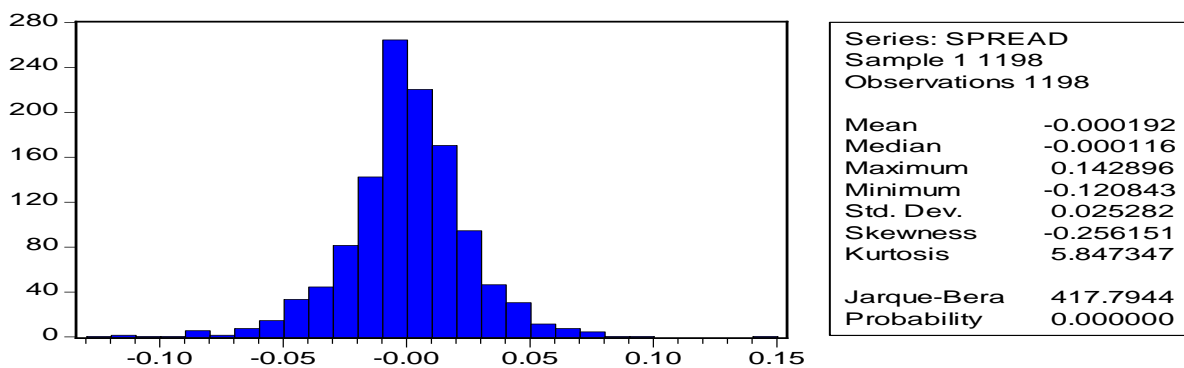


Figure 9. Histogram of corn/ethanol spread return series

¹⁴ In the analysis arithmetic returns are used rather than logarithmic returns due to the fact that the latter are not linearly additive across portfolio components. As a result, log returns can prove to be problematic and furthermore market participants have a tendency to look more at discrete returns in their daily trading activity. On this basis alone the use of arithmetic returns is deemed to be more realistic and more suitable for the purpose of this investigation.

The selection of suitable inputs for model estimation is extremely important when using Neural Networks. Without suitable explanatory variables it is difficult to generate accurate and satisfactory results. The selection of each input was decided based on correlation between returns of other commodity time series and each of the underlying legs. This analysis can be seen in table 46 of the appendix.

By observation of table 46 in the appendix, the CRB Index is strongly correlated with both corn and ethanol while crude brent oil, the S&P500 Energy, the MSCI Commodity and the AMEX Natural Gas indices all display moderate yet significant correlations during the in sample period. Using a combination of these explanatory variables as inputs the neural networks are tested for statistical accuracy and trading performance. Fundamentally, the strong correlation of these variables with each of the legs can be explained however natural gas is probably one of the most influential factors affecting the Crush. Natural gas is the main source of energy consumed during the ‘corn crushing’ process. Therefore, the more expensive natural gas becomes the more ethanol producers’ profit margins are potentially reduced. Funk *et al.* (2008) explain that with the price increase of natural gas, the price of ethanol should also increase as natural gas is a major component in the production of both corn and ethanol. Furthermore, natural gas also tends to trend with many other energy sources. Although this relationship is out of the scope of this paper it may be valuable to evaluate in subsequent investigations. A summary of the neural inputs can be seen in table 2.

<i>Number</i>	<i>Explanatory Variable</i>	<i>Lags (days)</i>
1	<i>Corn Crush spread returns</i>	1
2	<i>Corn Crush spread returns</i>	2
3	<i>Corn Crush spread returns</i>	3
4	<i>AMEX Natural Gas Index returns</i>	1
5	<i>Thomson Reuters/ Jefferies CRB Index returns</i>	1
6	<i>NYMEX Brent Crude Oil returns</i>	1
7	<i>1-Day RiskMetrics Volatility of spread returns</i>	1
8	<i>S&P 500 Energy Index returns</i>	1
9	<i>MSCI Commodity Index returns</i>	1
10	<i>CBOT Corn Returns</i>	1
11	<i>CBOT Ethanol Returns</i>	1
12	<i>Moving Average of the Corn Crush spread returns</i>	14
13	<i>Moving Average of the Corn Crush spread returns</i>	21

Table 2. Explanatory variables

It is also worth noting that all of the inputs are organised to take into account the hour time difference between those variables traded on CST (Central Standard Time) and EST (Eastern Standard Time) time zones. As a result, non-synchronous errors are avoided when estimating each of the networks. Although a full investigation into the determination of lag structures is beyond the scope of this paper the lag structure displayed in table 2 is retained as it produced the most satisfactory returns and forecasting accuracy during the training and test periods.

The observed data period spanning from 23/03/05 to 31/12/09 has been segregated into in- sample and out-of-sample datasets as used during the modelling process.

Period	Trading Days	Beginning	End
<i>Total dataset</i>	1,199	23 March 2005	31 December 2009
<i>Training dataset (in sample)</i>	664	23 March 2005	19 November 2007
<i>Test dataset (in sample)</i>	167	20 November 2007	21 July 2008
<i>Validation set (out of sample)</i>	368	22 July 2008	31 December 2009

Table 3. Data segregation for the full sample period

As an inference it should be noted that the reasoning behind further segmentation of the in-sample data set into sub samples is to avoid ‘over fitting’ when modelling the neural networks. This is discussed in more detail in Chapter 2 of this thesis.

3.3 Rolling Forward Procedure

A number of implications arise when applying analysis to a non-continuous time series as any valuable long term study of financial information requires scrutiny of continuous data. One of the biggest implications is the process of rolling a position forward from a contract that is nearing maturity to a new contract month in the future. As a result, a ‘rollover day’ is used by traders to start trading the new contract by switching, on this day, from the old contract before it reaches maturity to the new contract in order to maintain a continuous data series.

As some commodity contracts have longer lives than others the implications of creating a realistic, accurate and continuous spread series can be numerous. For example, grain contracts tend to be traded on

average for a year or two while financial markets can be traded up to as much as 5 to 10 years into the distant future. Therefore, agricultural traders and hedgers have to roll their positions more frequently.

For the purpose of this application all of these aspects were taken into consideration and it was decided to use the same rollover days for both the corn and ethanol contracts. As a result, the spread is simultaneously rolled forward for each of the underlying legs on the last Thursday of the month preceding maturity months. While it is accepted that this may not be the 'optimal' rollover procedure it is however recognised that the optimisation of rolling forward procedures might be another interesting topic to examine in future analysis. Despite this, the procedure is fairly pragmatic as it still enables the construction of an accurate and tradable time series which avoids both the risk of physical delivery and increasing volatility associated with illiquid periods.

3.4 Discounting the Existence of Seasonality

In the initial analysis of seasonality each of the legs are individually assessed. From this analysis it appears that both corn and ethanol are stationary when integrated of order 1. Regardless, it is apparent that in both the corn and ethanol series autocorrelation is present between lags 1-5. This is revealed by significant spikes indicating that both data series are not random. Results from a correlogram of the spread are displayed in more detail in the appendix. This observation is in line with those made by Klement (2005), Corsi (2003) and Anderson *et al.* (2003b) who have all found that seasonality does in fact exist in the daily returns of various agricultural commodities.

Analysis of the spread however reveals that the data series is actually random with no evidence of significant spikes. Furthermore, as there is nothing to support autocorrelation between any of the lags it can be assumed that by combining both corn and ethanol daily return series, in a spread trading system, the series is in fact random and free of seasonality. Pindyck and Rubinfeld (1998) highlight that a time series is only seasonal in circumstances when the autocorrelation function displays regular and frequent peaks. Ultimately, 'de-seasonalisation' techniques are not deemed necessary as autocorrelation is not identified between any of the lags.

4.0 Methodology

This section provides details for each of the different models, trading strategies and filters implemented to successfully establish parameters for modelling the corn/ethanol spread. The particulars with regard to the forecasting of each time series are discussed here however Chapter 2 discusses each of the benchmark and neural network models in more detail.

4.1 Benchmark Models

In this investigation three architecturally different neural network models are benchmarked against popular linear models. These benchmark models include trading signals produced by a Naive and a MACD (moving average convergence / divergence) trading strategies and a traditional ARMA (Autoregressive Moving Average) model. Co-integration was not deemed to be a suitable benchmark model due to the fact that the underlying legs (Corn and Ethanol) were not found to be co-integrated during the observed period. Co-integration between multiple non-stationary variables occurs when the linear combination of the variables results in a stationary series (Engle and Granger 1987). Taking this into consideration, the linear combination of ethanol and corn was not found to be stationary during the in-sample period. For this reason, cointegration has not been included as a benchmark model to forecast the Crush. For brevity, the I(1) test results and the trace statistics are not reported in the appendix¹⁵. However, all of the results and parameters for in-sample models can be found in the appendix.

4.2 MACD Model

During the backtest a 1 day moving average (hence the daily return of the Crush) for the shorter term and a 27 day moving average for the longer term were used. Therefore, a (1,27) combination was deemed to be the most profitable in terms of trading performance with $n = 1$ and 27 respectively. Trading signals are triggered when the two moving averages intersect. For instance, a long position is taken when the

¹⁵ To be provided on request.

short-term moving average intersects the long term moving average from below and a short position is adopted when the long-term moving average is intersected from above.

4.3 ARMA Model

Using the correlogram as a guide in the training and the test sub-periods a restricted ARMA (11,11) model is determine to be the most suitable. All of its coefficients are significant at the 99% confidence interval. The null hypothesis that all coefficients (except the constant) are not significantly different from zero is rejected at the 99% confidence interval (see Appendix A.2).

The model estimated during the in sample period was retained for out-of-sample trading. The performance of the strategy is evaluated in terms of forecasting accuracy and trading performance.

The specific ARMA model is presented in equation 22.

$$Y_t = -4.04 * 10^{-4} + 0.321Y_{t-1} + 0.288Y_{t-2} - 0.379Y_{t-8} + 0.548Y_{t-11} + 0.283\varepsilon_{t-1} + 0.261\varepsilon_{t-2} - 0.437\varepsilon_{t-8} + 0.585\varepsilon_{t-11} \quad (22)$$

5.0 Empirical Results

5.1 Statistical Performance

	Naive	ARMA	MLP	RNN	HONN
<i>MAE</i>	0.0200	0.0141	0.0142	0.0142	0.0140
<i>MAPE</i>	991.49%	205.67%	446.17%	272.42%	438.57%
<i>RMSE</i>	0.0282	0.0198	0.0200	0.0201	0.0198
<i>THEIL-U</i>	0.7164	0.8665	0.8638	0.8746	0.8801
<i>Correct Directional Change (CDC)</i>	51.36%	53.39%	52.72%	54.08%	54.89%

Table 4. Out-of-sample statistical performance

Table 4 reveals that the HONN model is the most accurate and statistically superior with leading statistics for three out of the five measures. For the most part, this is due to the fact that the sum of all squared differences between target and actual values for the HONN is lower than both the RNN and MLP neural models and the ARMA model. It can also be seen that the HONN's structural ability to predict the direction of change (54.89%) is the highest.

In summary, the lower the statistic for MAE, MAPE, RMSE and the THEIL – U, the better the forecasting accuracy. Results for the in-sample statistical performance can be seen in the appendix A.2.

5.2 Trading Performance

	Naive	MACD	ARMA	MLP	RNN	HONN
Annualised Return (excluding costs)	22.40%	10.67%	30.62%	36.59%	36.88%	37.02%
Annualised Volatility (excluding costs)	31.25%	31.30%	31.25%	31.23%	31.22%	31.22%
Maximum Drawdown (excluding costs)	-23.26%	-19.82%	-31.42%	-22.31%	-18.45%	-15.09%
Calmar Ratio (excluding costs)	0.96	0.54	0.97	1.64	2.00	2.45
Information Ratio (excluding costs)	0.72	0.34	0.98	1.17	1.18	1.19
# Transactions (annualised)	122	24	82	72	108	67
Trading Days	368	368	368	368	368	368
Transaction costs	17.8%	3.4%	8.13%	10.5%	15.7%	9.7%
Annualized Return (including costs) ¹⁶	4.6%	7.27%	22.49%	26.09%	21.18%	27.32%

Table 5. Out of sample trading performance results (unfiltered)

	Naive	MACD	ARMA	MLP	RNN	HONN
Annualised Return (excluding costs)	25.69%	20.96%	36.34%	41.26%	40.68%	42.76%
Annualised Volatility (excluding costs)	25.94%	27.44%	29.57%	27.24%	26.78%	27.81%
Maximum Drawdown (excluding costs)	-21.22%	-19.35%	-29.99%	-16.34%	-14.61%	-16.61%
Calmar Ratio	1.21	1.08	1.21	2.53	2.78	2.60
Information Ratio	1.15	0.76	1.23	1.51	1.52	1.54
# Transactions (annualised)	128	44	79	87	107	85
Trading Days	368	368	368	368	368	368
Transaction costs	18.6%	6.30%	7.82%	12.60%	15.60%	12.30%
Annualized Return (including costs)	7.09%	14.66%	28.52	28.66%	25.08%	30.46%

Table 6. Out of sample trading performance results (filtered)

Results obtained from the out-of-sample unfiltered trading simulation can be seen in table 5. The models used for out of sample trading were retained from the test sub-period. These were the models which produced attractive returns over the test sub-period. The same trading strategy was employed across all of the unfiltered models with the exception of the MACD model. In the MACD model trading signals were generated when the long term moving average either converged or diverged on the daily closing prices as discussed in section 4.1.2. The trading strategy that was used for the remaining models is to go long the spread when the forecasted returns are greater than zero and short when the forecast proves negative returns. In circumstances where consecutive upward or downward spread movements are experienced the previous day's position is held.

¹⁶ Calculated using five basis points per contract (round trip) as used by King and Zulauf (2010) for the electronic trading of agricultural futures. In this case every transaction consists of one corn contract and one ethanol contract.

Further observation of table 5 reveals that the best performing model is the HONN as it generates the highest annualised returns and the lowest maximum drawdowns. In addition, the HONN model performs marginally better than the other neural networks when considering its return/risk trade-off represented by the information ratio (1.19). As mentioned previously, the maximum drawdown is also at its lowest (-15.09%) which improves the return/maximum drawdown yield captured by the calmar ratio (2.45). For the remaining neural network models the trading performance for the MLP model, in terms of annualised returns, were slightly lower than the HONN model yet significantly higher when compared to the RNN model. However, there was little difference between the two when comparing information and calmar ratios although the RNN model did produce slightly lower maximum drawdowns.

As the crush spread is evidently volatile a market timing threshold filter similar to that used by Dunis and Miao (2006) is applied. The idea of this filter is to avoid entering the market during times of high volatility. A volatility time series is calculated using the RiskMetrics formula derived from JPMorgan (1997). The RiskMetrics formula used to calculate volatility is displayed in equation 23.

$$\sigma^2(t+1/t) = \mu * \sigma^2(t/t-1) + (1 - \mu) * r^2(t) \quad (23)$$

Where:

- σ^2 is the volatility forecast of spread returns,
- r^2 is the squared return of the spread,
- μ is 0.94 for daily data as computed in JP Morgan (1997).

The trading strategy is to stop trading the spread once an optimised level of volatility¹⁷ (derived from the above RiskMetrics formula) is breached. This strategy is depicted in equation 24.

$$\sigma^2(t+1/t) > T, \text{ then no trade} \quad (24)$$

¹⁷ All of the models were optimised in-sample. The optimal volatility thresholds were then selected for out-of-sample trading.

Where: σ^2 is the RiskMetrics volatility of the spread returns,
t is the dependent variable at time t ,
T is the optimised threshold of volatility.

From the results generated by the filtered models displayed in table 6 it is apparent that the market timing filter generates higher annualised returns while also decreasing each models' overall volatility. As a result, the RiskMetrics filter offers an improved risk / return trade off by increasing information ratios. Furthermore, the maximum drawdowns are also improved relative to the gross annualise returns. This can be seen in each of the calmar ratios.

6.0 Concluding Remarks

From the outset the aim was of this investigation was to model and forecast the corn/ethanol spread in a trading simulation from July 22, 2008 to December 31, 2009, the out-of-sample period. Results produced by each of the unfiltered models were for the most part satisfactory. The HONN generated the highest unfiltered set of results compared to the RNN and MLP neural network models as well as the Naive, MACD and ARMA linear methodologies. This is in line with current literature as Dunis *et al.* (2006b) who forecast the soybean crush also arrive at the same conclusion. It is also worth noting that the HONN model achieved the highest annualised returns and the most attractive risk return profile over both the training and validation sample periods. However, each model's volatility and maximum drawdown can be improved by applying a trading filter.

During a filtered trading simulation the HONN maintained its position as the most accurate and profitable trading model. On the whole, trading performance was enhanced by the application of the RiskMetrics market timing filter. This trading filter is similar to that used by Dunis and Miao (2006) who also achieve improved results when employing a RiskMetrics market timing strategy. During the out of

sample filtered trading returns are significantly improved, maximum drawdowns are reduced and overall model volatility is decreased.

With ethanol manufacturers improving processing efficiencies and expanding capacity, the future profitability of the Crush spread appears to be a prosperous one. The crushing process yielded 2.4 gallons per bushel a few years ago however due to technological improvements the current yield has been increased to 2.8 gallons per bushel. Furthermore, as demand for ethanol continues to grow this should, in theory, push supply/demand in favour of the stronger ethanol producers and perhaps lessen the fragmentation of the ethanol market. This may also encourage the acceleration of mergers and acquisitions improving 'economies of scale' and as a result lessening the cost of production further. Currently, large producers only occupy 40% of the market with the remaining 60% being filled by smaller family producers and farmers.

Limitations found within the research are numerous. For instance, future research could expand and provide more robust selection criteria for inputs. Different training algorithms could also be investigated to improve results. Lastly, the spread could include DDG in the computation however due to the current lack of historical data for DDG this is not currently possible in a long term study.

For the most part, this investigation offers an example of forecasting the spread between corn and ethanol futures providing ethanol plants, fund managers, grain elevators, processors, and other market participants with an insight into artificial intelligence as a methodology to capture and forecast non-linear relationships. All of the mentioned market participants are able to use alternative forecasting methods such as neural networks in order to manage price risk and profit margins of producing ethanol. Furthermore, some ethanol plants may opt to expand this spread trading strategy to include distiller's dried grains (DDG), as this is another by-product of 'corn crushing', as well as natural gas as this is the main energy consumed in the production process. Finally, it can be concluded that the Corn/Ethanol spread can also be traded successfully by speculators to benefit from arbitrage opportunities and diversified investment strategies.

CHAPTER 4: Trading and Hedging the Corn/Ethanol Crush Spread using Time Varying Leverage and Nonlinear Models

May 2011

Abstract

In contribution to Dunis *et al.* (2011b) this investigation endeavours to expand the selection of forecasting applications by delving further into the realm of artificial intelligence and non-linear modelling. The performances of a Multilayer Perceptron Neural Network (MLP) and Higher Order Neural Network (HONN) are gauged against a Genetic Programming Algorithm (GPA). Further to this, a time-varying volatility filter is applied by leveraging during lower volatility regimes in order to enhance the trading performance of the spread while avoiding trading completely during times of high volatility.

This paper models the Corn/Ethanol crush spread over a 6-year period commencing on March 23rd, 2005 (when the Ethanol futures contract was first traded on Chicago Board of Trade) through to December 31st, 2010. The spread acts as a good indicator of an ethanol producer's profit margin with corn being the principal raw ingredient used in a process called 'Corn Crushing' to produce Ethanol as a means for alternative energy.

Absent of leveraging, the GPA achieves the highest risk-adjusted returns followed by the HONN model. Once a time varying leverage strategy is introduced, the ranking is maintained as GPA continues to be the most profitable model with the HONN registering the second best risk-adjusted returns, followed by the MLP neural network. On that basis, and without the benefit of hindsight as in the real world, a fund manager would have selected the GPA model regardless of whether he decides to leverage or not. Furthermore, it is also observed that the time-varying leveraging strategy significantly improves annualised returns as well as reducing maximum drawdowns, two desirable outcomes for trading and hedging.

Keywords

Spread Trading, Corn Futures, Ethanol Futures, Time Varying Leverage, RiskMetrics, Leveraging, Multilayer Perceptron Neural Network, Higher Order Neural Network, Genetic Programming Algorithm.

1.0 Introduction

The motivation behind this paper derives from the recent global surge in commodities prices. In particular, this research is driven by the impact that this upward trend has had on bio fuels from a hedging perspective as well as the benefits available to speculators looking for alternative investment strategies. In recent times, commodities have been driven by a number of direct and indirect variables. For the most part, the rallying of commodity prices is a repercussion of varying political agendas, government policies, growing populations in China and India, and pressure imposed by global warming activists. More specifically, the supply and demand of agricultural commodities such as Corn and Ethanol are governed by but not limited to technological advances, government mandates for levels of production and funding, as well as weather conditions during harvest periods. Given these select few variables it is no wonder why commodity markets experience higher levels of volatility in comparison to other markets.

Rising and volatile commodity prices have also led to an increase in the number of market participants. For instance, farmers, commodity processors and grain elevators all use these financial markets to manage risk and hedge adverse price movements. On the other hand, speculators are also drawn to these markets to yield profits and take advantage of diversified investment strategies. Ultimately, the increase in demand for agricultural commodities coupled with an uncertainty of supply and ever increasing investment opportunities are all to blame for high volatility which is characteristic of the agricultural commodities market.

This investigation aims to rigorously evaluate the profitability of a Corn - Ethanol Spread. The profit margin created from a Corn - Ethanol spread is achieved from the process of converting corn into ethanol. The process involves extracting the profuse amounts of carbohydrates stored within corn to create simple sugars in order to produce ethanol as a valuable by-product. The Renewable Fuels Association (2011)

estimates that as much as 13 billion gallons of Ethanol was produced in 2010 compared to a mere 215 million gallons produced annually in the 1980's. As a result, the ethanol industry is one of the fastest growing industries making up as much as 10% of the United States' fuel supply. Moreover, the future outlook of the corn-ethanol market is an extremely prosperous one with the Energy Independence and Security Act (EISA) of 2007 being passed promoting an increase in the construction of ethanol plants to meet increasing demand. In particular, this act sets forth a target to increase capacity to accommodate for 15 billion gallons of ethanol to be produced in the United States by the year 2015. The underlying stimuli behind increasing the production of ethanol include the lessening U.S. dependence on foreign oil imports as well as efforts to satisfy demands from environmental activists who call for the use of alternative cleaner renewable energy. Furthermore, with U.S. crude oil prices reaching an all-time high in July 2008 at \$147.27 a barrel it became even more apparent that alternative cheaper bio fuels are essential. Most recently, political instability in the Middle East has seen oil prices rise significantly increasing the demand for cheaper alternative fuels. These price hikes, although high in relation to industry averages over the last few years, are yet to surpass the previous record high experienced in 2008.

In particular, growth and expansion of the Ethanol industry can be attributed to two major changes in fuel markets. As highlighted by Gallagher (2009), the ban of MTBE (Methyl Tertiary Butyl Ether) and the record high levels of petroleum prices have led to a significant increase in demand for and supply of Ethanol as an alternative bio fuel. The key players in the global ethanol market are the United States and Brazil who are the largest ethanol producing and exporting nations in the world. In comparison, the type of ethanol produced in each differs due to the raw ingredients used to produce the ethanol. Brazil uses sugarcane to produce ethanol whereas the United States uses corn however sugarcane based ethanol is more competitively priced due to lower sugarcane production costs. For the purpose of this investigation the focus is on the corn-ethanol market in the United States.

The main objective of this paper is to determine the potential profitability derived from speculatively trading a corn vs. ethanol spread as well as identifying hedging opportunities for producers of these commodities. This study covers a horizon of 6 years commencing when the Ethanol futures contract was

first traded on Chicago Board of Trade (CBOT) exchange (March 23, 2005). The relationship between the two commodities will be investigated by analysing spreads created from changes in their daily closing prices using sophisticated forecasting methodologies. This investigation also aims to build on earlier work carried out by Dunis *et al.* (2006), who investigate a soybean-oil crush spread and Dunis *et al.* (2011b) who initially observed the Corn Ethanol Crush spread. With the motivations for carrying out this research reviewed above, further investigation into forecasting the Corn ‘Crush’ spread is warranted.

The remainder of this paper is organised in the following manner. Section 2 provides a comprehensive review of all current literature regarding the trading of the Corn ‘Crush’ Spread. Section 3 discusses how the financial data was sourced and compiled for statistical analysis. Section 4 offers an introduction and explanation regarding each of the methodologies involved in this investigation. Subsequent sections of 5 and 6 present the results and final remarks respectively. The appendix A.3 presents relevant tables, figures and estimation parameters to conclude the research.

2.0 Literature Review

On review of past literature it becomes apparent that the virtues offered from hedging a Corn Crush spread over short term horizons are investigated by some, however there is limited literature regarding spread trading of agricultural commodity markets as a vehicle to hedge or speculate in the longer run. For instance, Dahlgran (2009) investigates the effectiveness of one-through eight-week hedges, for various commodities, over a three year horizon. More importantly, he examines the corn crush and the use of hedging as a risk management vehicle during the period of March 23rd, 2005 to December 31st, 2008. Over this period, Dahlgran (2009) concludes that the effectiveness of hedging a Corn Crush is comparable to returns yielded from trading a soybean crush as it offers ethanol producers similar ‘price risk reduction capabilities’ to those experienced by soybean processors who utilise the soybean crush hedge. In support of these findings, the CBOT (2007) also promotes the ‘corn crush’ hedge as analogous to the soybean crush hedge. The limited literature review regarding speculation of the corn crush spread can perhaps be attributed to the fact that ethanol has only been traded on the CBOT as a futures contract since early 2005. Franken and

Parcell (2003) explain that prior to the availability of ethanol futures contracts on the CBOT, ethanol price risk was cross-hedged with unleaded gasoline futures. However, with the recent creation of an ethanol specific futures contract, opportunities have arisen that enable direct hedging. Therefore, one can now hedge the price risk associated with holding ethanol stock as well as safeguard input costs of production and raw materials which are consumed during the processing of corn into ethanol.

Dunis *et al.* (2006) explore the soybean-oil crush spread as a means to test the forecasting ability of various forecasting methodologies. In this investigation the margin of trading a soybean-oil spread is analysed over a period of 11 years (01/01/1995 to 01/01/2005). The application of more sophisticated forecasting methodologies such as neural network architectures are employed and benchmarked against conventional forecasting techniques such as a fair value co-integration model. From this analysis it is concluded that one can profit from trading the soybean crush spread using neural networks. The Higher Order Neural Network (HONN) produced the highest annualised out-of-sample returns and the most superior statistical performance when compared to a Recurrent Neural Network (RNN), a Multilayer Perceptron (MLP) Neural Network, and Fair Value Co-integration model.

Dunis *et al.* (2011b) also investigate the predictive abilities of neural networks for hedging and trading purposes by evaluating the Corn Ethanol spread utilising MLP, RNN and HONN models. A period of 5 years commencing 23/03/2005 to 31/12/2009 is analysed. During this period the HONN is also found to possess superior forecasting abilities when benchmarked against an ARMA model as well as the two other neural networks. In particular, the HONN model (unfiltered) outperforms all other models generating a net annualised return of 27.32%. Each model was then filtered using a threshold filter. The intuition of this trading filter was to cease trading once volatility, measured using RiskMetrics as a market timing filter, is surpassed. As a result, trading performance was enhanced across all models with the HONN filtered model improving returns to 30.46%. Overall volatility and maximum drawdowns were also improved for each of the models as a result of market timing.

3.0 Descriptive Statistics and Related Financial Data

The daily closing prices (18:00 CST¹⁸) for each of the commodity contract months was obtained from Datastream for the period covering March 23, 2005 - December 31, 2010. Corn futures¹⁹ are the most heavily traded agricultural commodity and have been traded on the CBOT²⁰ exchange since the mid 1800's. On the other hand, ethanol futures²¹ are a more recent addition to the CBOT exchange having only been traded as a futures contract since March 23, 2005. As a result, ethanol is not traded as frequently as Corn and is therefore less liquid. Both contracts are traded from 09:30am to 13:15pm and 18:00pm to 7:15am CST. As a result, the issue of non-simultaneous pricing that plagues many other investigations does not present itself here with the closing times of each leg being at the same time.

The Corn 'Crush' spread is calculated taking into consideration the fact that both commodity futures contracts are priced and traded in different units. As corn is priced in cents per bushel and Ethanol is traded in dollars per gallon a conversion of prices into equal units is required. At the time of writing, one bushel of corn currently yields approximately 2.8 gallons of Ethanol (CME 2010). Therefore, to create a tradable spread between the two contracts the price of ethanol must be multiplied by 2.8 in order to convert it into dollars per bushel. Lastly to obtain the corn 'crush' price, the price of corn is then subtracted from the converted price of ethanol (in dollars per bushel). This calculation is mathematically depicted as follows:

$$C_t = [(2.8 * P_E) * 100] - P_C \quad (25)$$

where:

C_t = Price of the crush spread at time t (in cents per bushel)

P_E = Price of the ethanol contract at time t (in dollars per gallon)

P_C = Price of the corn contract at time t (in cents per bushel)

There are various other ways to construct the spread depending on what the market participant is attempting to achieve. Other Corn Crush spread combinations may include distillers dried grains (another by-product of corn) and natural gas as this is consumed during the 'crushing' process. However, for the purpose of this investigation it was decided to focus the analysis on the relationship between Corn and

¹⁸ Central Standard Time (CST).

¹⁹ Particulars with regards to CBOT Corn contract specifications can be found in appendix A.3.

²⁰ Chicago Board of Trade.

²¹ Particulars with regards to CBOT Ethanol contract specifications can be found in appendix A.3.

Ethanol as a longer history of data is available for these variables. In particular, the Distillers Dried Grains (DDGs) futures contract only commenced trading on the Chicago Mercantile Exchange as of April 26, 2010. Prior to the creation of a DDG futures contract, processors hedged price risk exposure of holding DDGs stock simply by trading corn futures. Equally as effective, a recent study has suggested that a cross hedge using corn and soybean meal futures contracts may offer more effective risk reduction (See, Brinker *et al.*, 2009).

The methodology applied throughout this investigation in order to calculate the returns of the corn crush spread can be seen below as provided by Butterworth and Holmes (2002) and more recently by Dunis *et al.* (2006) and Dunis *et al.* (2011b) :

$$\Delta S_t = \left[\frac{(P_{E(t)} - P_{E(t-1)})}{(P_{E(t-1)})} - \frac{(P_{C(t)} - P_{C(t-1)})}{(P_{C(t-1)})} \right] \quad (26)$$

where:

ΔS_t = Percentage returns of spread at time t

$P_{E(t)}$ = is the price of ethanol at time t (in cents per bushel)

$P_{E(t-1)}$ = is the price of ethanol at time $t-1$ (in cents per bushel)

$P_{C(t)}$ = is the price of corn at time t (in cents per bushel)

$P_{C(t-1)}$ = is the price of corn at time $t-1$ (in cents per bushel)

3.1 Statistical Behaviour of the Crush

The price time series for the full sample period (23/03/2005 – 31/12/2010) can be observed in the below:

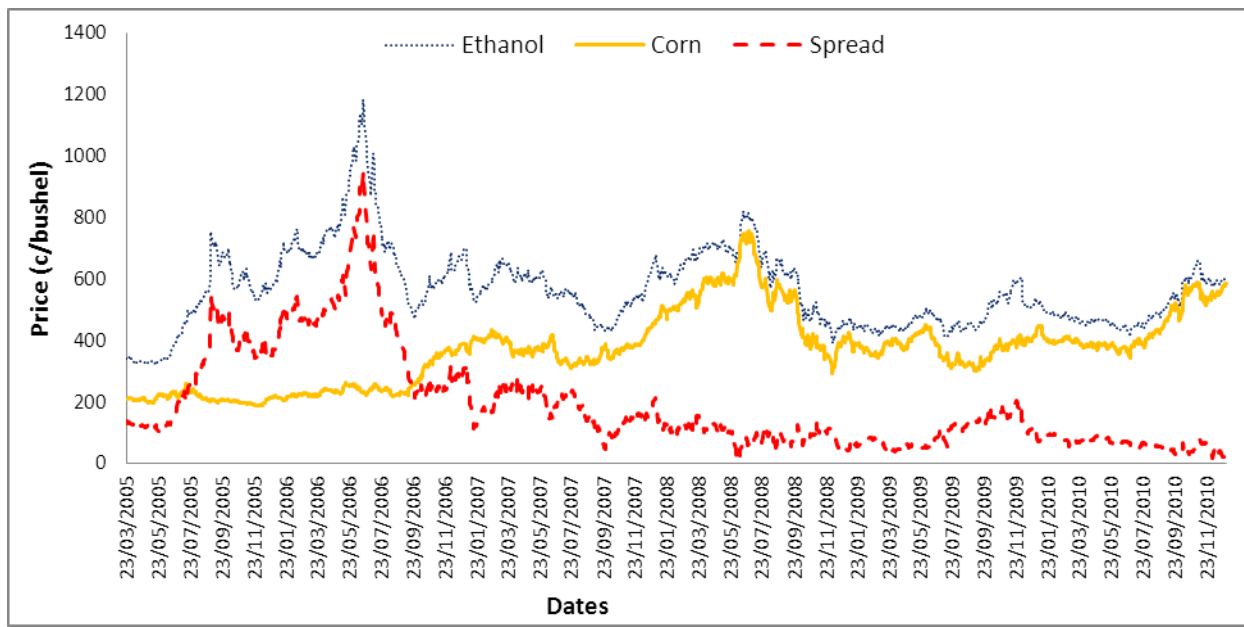


Figure 10. The corn-ethanol crush CBOT daily closing prices (23/03/2005 – 31/12/2010)

By observation of figure 10, it is apparent that the spread is mean reverting with large deviations occurring as a result of price rallies. Deviations away from the spread's mean are a consequence of shocks to the demand or supply of each of the underlying commodities experienced during the analysed time period. Most notably, there are two major spikes in the data occurring during our 'in-sample' period. The first of which occurred on August 29, 2005 as a result of Hurricane Katrina which devastated much of the South and Midwest of the United States. The second spike occurred as a result of the phasing out of federal MTBE (Methyl Tertiary Butyl Ether) and the phasing in of a requisite that refiners are to produce 4 billion gallons of ethanol in 2006, see McKay (2006). In reaction, the price of ethanol futures rallied to a record high of \$4.23/gallon up significantly from a record low of \$1.16/gallon experienced just 12 months earlier in May 2005.

More recently however, the CBOT ethanol and corn contracts have both been affected by volatility 'spill-overs' from other commodities such as those experienced in the Gasoline market, see Funk *et al.* (2008) for a detailed review. For one, the volatility experienced in the price of Crude Brent Oil during the summer of 2008 led to similar price spikes in ethanol and corn. This cannot be seen in the spread shown in

figure 10 due to offsetting positions created from the spread trade. Once converted into cents per bushel²² it can be seen that Ethanol averaged 561.75 c/bushel and corn averaged 372.2 c/bushel for the period of 2005-2010.

3.2 Descriptive Statistics and Explanatory Variables

Inferences are based on the change in daily closing prices²³ and from the below histogram it can be observed that the Corn/Ethanol spread return series is non-normal (confirmed at a 99% confidence level by the Jarque-Bera test statistic), with a slight skewness and high kurtosis.

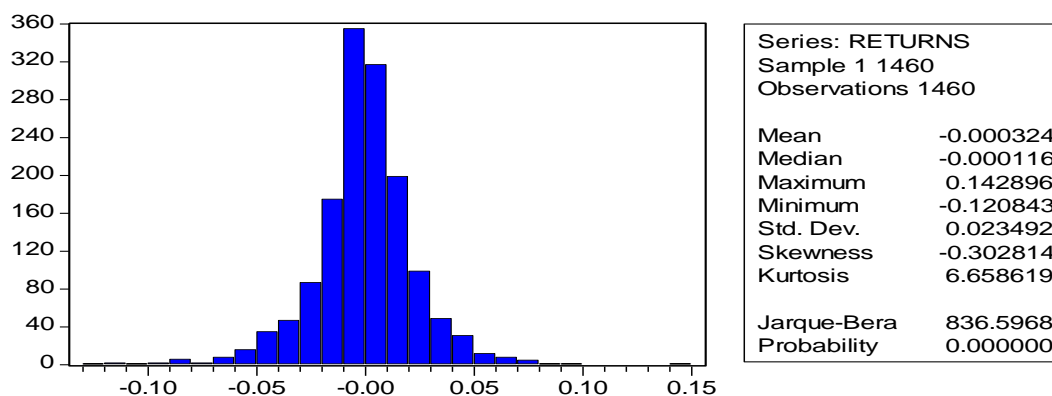


Figure 11. Histogram of corn/ethanol spread return series

Number	Variable	Lags (days)
1	Corn Crush spread returns	1
2	Corn Crush spread returns	2
3	Corn Crush spread returns	3
4	AMEX Natural Gas Index returns	1
5	Thomson Reuters/ Jefferies CRB Index returns	1
6	NYMEX Brent Crude Oil returns	1
7	1-Day RiskMetrics Volatility of the Crush spread returns	1
8	S&P 500 Energy Index returns	1
9	MSCI Commodity Index returns	1
10	CBOT Corn Returns	1
11	CBOT Ethanol Returns	1
12	Moving Average of the Corn Crush spread returns	14

²² This conversion is explained in more detail in section 3.0

²³ In the analysis arithmetic returns are used instead of logarithmic returns due to the fact that the latter are not linearly additive across portfolio components. Hence, this can prove to be problematic and furthermore market participants have a tendency to look more at discrete returns in their daily trading activity. On this basis alone the use of arithmetic returns are deemed to be more realistic and suitable for the purpose of the application.

Table 7. Explanatory variables for the neural networks

All inputs are organised to take into account the hour time difference between those variables traded on CST (Central Standard Time) and EST (Eastern Standard Time) time zones. Hence, non-synchronous errors are avoided in the estimation of the networks. Although a full investigation into the determination of lag structures is beyond the scope of this paper the lag structure displayed in table 7 was retained as it produced the best forecasting accuracy during the training period. It is also worth noting that the same explanatory variables and lag structure are adopted as those used in earlier work produced by Dunis *et al.* (2011b).

The observed data period spanning from 23/03/2005-31/12/2010 has been segregated into in- sample and out-of-sample data as used during the modelling process.

Period	Trading Days	Beginning	End
<i>Total dataset</i>	1,460	23 March 2005	31 December 2010
<i>Training dataset (in-sample)</i>	707	23 March 2005	23 January 2008
<i>Test dataset (in-sample)</i>	291	24 January 2008	19 March 2009
<i>Validation set (out-of-sample)</i>	462	20 March 2009	31 December 2010

Table 8. Data segregation for the full sample period

In the functionality of neural networks the above in-sample period was segregated once more into two sub-periods to compensate for the training and testing of each network to avoid ‘over fitting’ or over-familiarisation with the data set. Ultimately, this could prove to be detrimental to future forecasts.

3.3 Rolling Forward Procedure

A number of implications arise when applying analysis to a non-continuous time series as any valuable long-term study of financial information requires scrutiny of continuous data. One of the biggest implications is the process of rolling a position forward from a contract that is nearing maturity to a new contract month in the future. As a result, a ‘rollover day’ is used by traders to start trading the new contract

by switching, on this day, from the old contract before it reaches maturity to the new contract in order to maintain a continuous data series.

As it is understood that some commodity contracts have longer lives than others the implications of creating a realistic, accurate and continuous spread series can become overwhelming. For example, grain contracts tend to be traded on average for a year or two whilst financial markets can be traded up to as much as 5 to 10 years into the distant future. Therefore, agricultural traders and hedgers have to roll their positions more frequently.

For the purpose of the investigation these aspects were taken into consideration and it was decided to use the same rollover days for both the corn and ethanol contracts. As a result, the spread is simultaneously rolled forward for each of the underlying legs on the last Thursday of the month preceding maturity months. While it is accepted that this may not be the ‘optimal’ rollover procedure it is however recognised that the optimisation of rolling forward procedures may be an interesting aspect to examine in future investigations. Despite this, the rolling procedure is fairly pragmatic as it enables the construction of an accurate and tradable time series. Both the risk of physical delivery and increased volatility associated with illiquid periods are avoided.

4.0 Methodology

This section details the different models, trading strategies and filters implemented in order to successfully establish parameters for modelling the corn/ethanol spread. The particulars with regard to the forecasting of the time series are also discussed encompassing various benchmark models, two different neural network architectures and a genetic programming algorithm (GPA). De-seasonalisation techniques were not deemed necessary as it was found that the spread series was in fact random with no significant spikes being present to indicate seasonality. For further justification please refer to Dunis *et al.* (2011b). Please note that Chapter 2 introduces and explains each of the proposed models in more detail.

4.1 Benchmark Models

In this investigation two neural network models are benchmarked against other traditional models as well as a GPA model. These benchmark models include Naive and MACD (moving average convergence/divergence) trading strategies and a traditional ARMA model. Co-integration was not deemed to be a suitable benchmark model due to the fact that the underlying legs (Corn and Ethanol) were not found to be co-integrated during the in-sample period. Co-integration between multiple non-stationary variables occurs when a linear combination of the variables results in a stationary series (Engle and Granger, 1987). Taking this into consideration, the linear combination of ethanol and corn was not stationary during the in-sample period. For brevity, the I(1) test results and the trace statistics are not reported in the appendix²⁴. However, all of the results and parameters for in-sample models can be found in appendix A.3.

4.2 MACD Model

In this study a 1-day moving average (hence the daily returns of the spread) and a 60-day moving average is retained for out-of-sample evaluation. Therefore a (1, 60) combination was deemed to be the most profitable in terms of trading performance with $n = 1$ and $n = 60$ respectively. Trading signals are triggered when the two moving averages intersect. For instance, a long position is taken when the short-term moving average intersects the long term moving average from below and a short position is adopted when the long-term moving average is intersected from above.

4.3 ARMA Model

Using a correlogram as a guide in the training and the test sub-periods a restricted ARMA (11,11) model was selected. Furthermore, it is also worth noting that this is the same ARMA model as used by Dunis *et al.* (2011b²⁵) to ensure consistency. All of its coefficients are significant at the 99% confidence interval. The null hypothesis that all coefficients (except the constant) are not significantly different from zero is rejected at the 99% confidence interval (see table 57 in Appendix A.3).

The model selected during in-sample optimisation was retained for the out-of-sample trading simulation. This is mathematically represented as:

²⁴ These can however be provided on request.

²⁵ The dataset used by Dunis *et al.* (2011-b) covered a period from 23/03/2005 until 31/12/2009 (chapter 3).

$$Y_t = -3.56 * 10^{-4} + 0.288Y_{t-1} + 0.301Y_{t-2} - 0.369Y_{t-8} + 0.517Y_{t-11} + 0.254\varepsilon_{t-1} + 0.267\varepsilon_{t-2} - 0.426\varepsilon_{t-8} + 0.561\varepsilon_{t-11} \quad (27)$$

5.0 Empirical Results

5.1 Statistical Performance

By observation, table 9 statistically reveals that both the GPA and MLP models are slightly more accurate compared to the HONN model while noticeably superior to the Naive and ARMA models. However, the GPA model does in fact produce better results beating the MLP model on three out of the five statistical measures while matching its mean absolute error (MAE). It can also be deduced that the GPA's structural ability to predict the direction of change (54.74%) during our in-sample is the third best. When taking both the in- and out-of-sample (table 10) periods into consideration it appears that our GPA model is the most 'robust' as it maintains strong predictability throughout the entire sample. In summary, the lower the statistic for MAE, MAPE, RMSE and the THEIL-U, the better the forecasting accuracy a model produces. Hence, it can be concluded that the nonlinear 'artificially intelligent' models are more accurately able to capture significant movements and trends experienced within the corn/ethanol spread data when benchmarked against more traditional linear models.

	Naive	ARMA	MLP	HONN	GPA
<i>MAE</i>	0.0272	0.0191	0.0187	0.0189	0.0187
<i>MAPE</i>	552.52%	155.71%	122.81%	153.85%	140.76%
<i>RMSE</i>	0.0363	0.0259	0.0254	0.0257	0.0251
<i>THEIL-U</i>	0.6932	0.8620	0.8026	0.8364	0.7639
<i>Correct Directional Change (CDC)</i>	48.60%	52.91%	54.42%	55.38%	54.74%

Table 9. In-sample statistical performance

	Naive	ARMA	MLP	HONN	GPA
<i>MAE</i>	0.0152	0.0108	0.0123	0.0115	0.0108
<i>MAPE</i>	675.73%	193.09%	254.34%	405.56%	351.65%
<i>RMSE</i>	0.0228	0.0164	0.0201	0.0170	0.0164
<i>THEIL-U</i>	0.6989	0.8649	0.7095	0.7560	0.8254
<i>Correct Directional Change (CDC)</i>	51.30%	53.03%	53.25%	52.38%	54.11%

Table 10. Out-of- sample statistical performance

5.2 Trading Performance

The in-sample trading performance results are given in appendix A.3. They clearly show that, without leverage, the GPA achieves the highest risk-adjusted returns followed by the HONN model. This performance ranking is also maintained under the supervision of a time varying leverage strategy as explained below. Therefore, the GPA continues to register the best risk-adjusted returns, followed by the HONN. On this basis, and without the benefit of hindsight as in the real world, a fund manager would select the GPA model regardless of whether or not he decides to leverage his returns.

Table 11 exhibits the out-of-sample trading results from all of the models. These are obviously the most important results as they are achieved on data not ‘seen’ by the models and therefore represent an acid test of robustness for each of the models as they reproduce what would happen in a true trading environment.

From this it is evident that, without leverage, the GPA model marginally beats the HONN model with higher annualised return. It is also worth noting that the GPA model also manages to produce the best information and calmar ratios²⁶. Each of these models was retained from in sample training with the results being provided in table 58 of appendix A.3. Formulas for each of the performance measurements can be found in table 44 of appendix A.1.

	Naive	MACD	ARMA	MLP	HONN	GPA
Annualised Return (excluding costs)	20.00%	12.26%	20.59%	32.70%	36.06%	37.43%
Annualised Volatility (excluding costs)	25.76%	25.86%	25.84%	25.77%	25.77%	25.77%
Maximum Drawdown (excluding costs)	-23.26%	-20.82%	-36.84%	-19.72%	-25.62%	-18.01%
Calmar Ratio (excluding costs)	0.86	0.59	0.56	1.66	1.41	2.08
Information Ratio (excluding costs)	0.78	0.47	0.80	1.27	1.40	1.45
# Transactions (annualised)	118	12	89	125	113	110
Trading Days	462	462	462	462	462	462
Transaction costs	11.73%	1.19%	8.84%	12.49%	11.29%	10.96%
Annualized Return (including costs) ²⁷	8.27%	11.07%	11.75%	20.21%	24.77%	26.46%

Table 11. Out-of-sample (unleveraged trading performance)

²⁶ Both ratios measure risk-adjusted returns: the information ratio divides annualised return by annualised return volatility, while the Calmar ratio divides annualised return by the maximum drawdown so that its inverse gives a measure of the time necessary to recoup the largest loss ever recorded by a given portfolio.

²⁷ Calculated using five basis points per contract (round trip) as used by King and Zulauf (2010) for the electronic trading of agricultural futures. In this case every transaction consists of one corn contract and one ethanol contract.

In contribution to Dunis *et al.* (2011b), this investigation develops a more refined volatility filter in order to offer a more sophisticated insight into market timing. The intuition of the strategy is to avoid trading when volatility is very high while at the same time exploiting days when the volatility is relatively low. The significant difference between market timing techniques as used in Dunis *et al.* (2011b) and time varying leverage as used here is that leverage can be easily achieved by scaling position sizes inversely to computed risk measures.

There are a number of different measurement techniques available to analysts when calculating time varying volatility. The most common method is the calculation of a moving average as used by Dunis and Miao (2006) who estimate volatilities using a fixed window of time (number of days, weeks, months). The same volatility regime classification technique was employed for the purpose of this research. Following JP Morgan (1997), the estimation of volatility regimes is therefore based on a rolling historical average of RiskMetrics volatility (μ_{AVG}) as well as the standard deviation of this volatility (σ). The latter is in essence ‘the volatility of the volatility’. For the purpose of this investigation both historical parameters μ_{AVG} and σ are calculated on a 3- month rolling window²⁸. The average of both μ_{AVG} and σ are then calculated based on these 3- month historic periods. For instance, the historical volatilities for our in-sample data are calculated over a total 16 periods with each spanning 3 months in duration. Thus, volatility regimes are then classified based on both the average totals for μ and σ over this time period.

Volatility regimes are categorised as ‘Lower High’ (between μ_{AVG} and $\mu_{AVG} + 2\sigma$), ‘Medium High’ (between $\mu_{AVG} + 2\sigma$ and $\mu_{AVG} + 4\sigma$), and ‘Extremely High’ (greater than $\mu_{AVG} + 4\sigma$) volatility periods. Similarly, periods of lower volatility are categorised as ‘Higher Low’ (between μ_{AVG} and $\mu_{AVG} - 2\sigma$), ‘Medium Low’ (between $\mu_{AVG} - 2\sigma$ and $\mu_{AVG} - 4\sigma$) and ‘Extremely Low’ (less than $\mu_{AVG} - 4\sigma$).

As the crush spread is evidently volatile the trading strategy is two-fold: leverage during lower volatility regimes and avoid trading altogether during volatility regimes classified as ‘Extremely High’. The RiskMetrics formula used to calculate volatility is:

²⁸ During the in-sample period different rolling windows were tested and it was found that a 3-month rolling window produces the best results when subjected to the leveraging strategy.

$$\mu^2_{(t+1/t)} = \beta * \mu^2_{(t/t-1)} + (1 - \beta) * r^2_{(t)}$$

(28)

where: μ^2 is the volatility forecast of the spread returns,
 r^2 is the squared return of the spread,
 β 0.94 for daily data as computed in JP Morgan (1997).

To elaborate further, the ‘no trade’ trading strategy can be best explained by the following formula:

$$\mu^2_{(t+1/t)} > T, \text{ then no trade}$$

(29)

where: μ^2 is the RiskMetrics volatility of the spread returns,
 T is the ‘extremely high’ volatility regime (greater than $\mu_{AVG} + 4\sigma$).

Different levels of leveraging are incrementally employed during each of the regimes. Leveraging structures can be seen in the table 12.

	Extremely Low Vol.	Low Medium Vol.	Low Higher Vol.	Lower High Vol.	Medium Vol.	High	Extremely High Vol.
Leverage	2.5	2	1.5	1	0.5	0	

Table 12. Leverage structure

When utilising this leveraging trading strategy, trading profits are amplified under the supervision of a market timing filter. As explained by Ang *et al.* (2010), sophisticated use of leveraging is vital to the performance of hedge funds. Therefore, gearing with this time varying filter enables the maximisation of profits when volatility is anticipated to be low or moderate. Trading and leveraging is ceased during times of high volatility to avoid the possibility of experiencing catastrophic losses. In particular, this strategy is employed by many hedge funds and professional traders who run ‘market neutral’ arbitrage strategies. The leveraging structure

provided in table 12 was chosen as on average, successful hedge funds tend to leverage between 1.5 and 2.5 times their returns²⁹. Interestingly, investment banks tend to leverage at much higher levels with some leveraging well above 5 and even 10 times. In principle, leveraging is trading on credit in order to boost returns of a trading model and in this case the leveraging strategy is applied when the spread is experiencing lower volatility based on time varying parameters.

By observation, table 13 displays trading results produced by each of the models when subjected to a time varying volatility leverage strategy. In comparison to table 11 it is apparent that the GPA remains the most profitable model producing annualised returns of 33.92% while the MLP leveraged model experiences the largest improvement with an increase of 12.45% in annualised returns. The HONN model however, records the lowest maximum drawdowns, followed closely by the GPA. On the whole, the leveraging structure offers improved annualised returns for 4 out of 6 of the models. The exceptions being the Naïve and MACD models of which both experience eroded annualised returns. Due to the nature of leveraging, the volatilities associated with all of the leveraged models are increased slightly when compared to unleveraged models. More importantly, by avoiding trading during times of extreme volatility leveraged models produce more acceptable maximum drawdowns. As a consequence, maximum drawdowns are reduced for all six of the models. Most notably, the leveraged market timing strategy reduces the ARMA model's maximum drawdown from -39.44% to -14.92%.

	Naive	MACD	ARMA	MLP	HONN	GPA
Annualised Return (excluding costs)	18.27%	8.85%	31.43%	43.96%	42.67%	44.01%
Annualised Volatility (excluding costs)	29.04%	25.13%	29.16%	29.02%	29.07%	29.10%
Maximum Drawdown (excluding costs)	-18.41%	-18.14%	-14.92%	-15.78%	-14.24%	-14.89%
Calmar Ratio	0.99	0.49	2.11	2.79	3.00	2.96
Information Ratio	0.63	0.35	1.08	1.51	1.47	1.51
# Transactions (annualised)	84	6	60	96	79	84
Trading Days	462	462	462	462	462	462
Transaction and Leverage costs ³⁰	10.15%	2.95%	7.75%	11.30%	9.66%	10.10%
Annualized Return (including costs)	8.12%	5.90%	23.68%	32.66%	33.01%	33.92%
Leverage Improvement	-0.15%	-5.17%	11.93%	12.45%	8.24%	7.44%
Drawdown Improvement	4.85%	2.68%	21.92%	3.94%	11.38%	3.12%

Table 13. Out-of-sample (leveraged trading performance)

²⁹ See Lan *et al.* (2011) for a more detailed account.

³⁰ The cost of leverage (interest payments for the additional capital) is calculated at 1.75% p.a. (0.0069% per trading day assuming 252 trading days in a year).

6.0 Concluding Remarks

From the outset the aim of this investigation was to model and forecast the corn/ethanol spread in a trading simulation expanding from 20/03/2009 to 31/12/2010, the out-of-sample trading period. Results produced from each of the unleveraged models were for the most part satisfactory; with the GPA proving superior. In comparison to earlier work carried out by Dunis *et al.* (2011b) it can be concluded that the ARMA, MLP and HONN models are all robust due to the fact that they maintained profitability throughout both sample periods (using the same estimation parameters and inputs) when applied to an expanded data set to include 2010. In all cases the annualised returns for each remained fairly constant. These observations are also in line with those drawn by Dunis *et al.* (2011a) who forecast the EUR/USD relationship finding that a GPA model also outperforms a generic MLP neural network model. In particular, they find that a GPA returned 3.75% more in annualised returns compared to an MLP neural network model.

When considering leveraged results, the most profitable model was the GPA with the HONN being the second best. The MLP model however is most improved with annualised returns being enhanced from 20.21% (unleveraged) to 32.66% (leveraged). The HONN model also records the lowest maximum drawdown, closely followed by the GPA. In terms of maximum drawdown the ARMA model experiences the largest reduction down from a staggering -39.44% to -14.92%. Ultimately, leveraged models experience slightly higher volatility but reduced maximum drawdowns as a result of the 'no trade' threshold filter.

On the whole, the application of non-linear methodologies and time varying volatility leverage filters has proven to be profitable however their application will vary depending on market participants. For instance, hedgers, grain elevators and grain processors are generally less inclined to leverage as their main strategy is to avert risk and protect margins. On the other hand, fund managers and speculators have a greater risk appetite and they are primarily driven to maximise profit. In addition to this, hedgers may look to manage risk over longer term horizons in line with processing time frames. For example, hedgers could model the spread using weekly prices as ethanol producers generally hedge from 2 to 8 week periods as explained by

Dahlgran (2009). Furthermore, neural networks also enable hedgers to use an expansive universe of inputs in order to capture the non-linear relationships between explanatory variables and the spread.

Ultimately, this investigation offers an example of forecasting and trading of the spread between corn and ethanol futures. This provides an insight into non-linear forecasting methodologies offering a profitable application of their uses for the benefit of fund managers, grain elevators, processors, and other market participants.

CHAPTER 5: Non-Linear Forecasting of the Gold Miner Spread: An Application of Correlation Filters

January 2012

Abstract

This paper models and forecasts the Gold Miner Spread from 23/05/06 to 30/06/11. The Gold Miner spread acts as a suitable performance indicator for the relationship between Physical Gold and US Gold Equity.

The contribution of this investigation is twofold. Firstly, the accuracy of each model is evaluated from a statistical perspective. Various forecasting methodologies are then applied to trade the spread. Trading models include a ARMA (12,12) model, a Co-integration model, a Multi-layer Perceptron Neural Network (NN), a Particle Swarm Optimisation Radial Basis Function NN, and a Genetic Programming Algorithm.

Results obtained from an out of sample trading simulation validate the in sample back test as the GPA model produced the highest risk-adjusted returns. Correlation filters are also applied to enhance performance and as a consequence volatility is reduced by 5%, on average, while returns are improved between 2.54% and 8.11% across 5 of the 6 models.

Keywords

Spread Trading, Multi-Layer Perceptron Neural Network, Particle Swarm Optimisation, Radial Basis Function Neural Network, Genetic Programming Algorithm, Correlation Filter.

1.0 Introduction

This investigation evaluates the relationship between gold bullion (physical gold) and US gold mining equity. Historically, the statistical relationship between the two has displayed strong correlation as the values of gold mining stocks are predominately determined by the price performance of gold. However, despite this long term relationship the spread frequently experiences short term irregularities. When this occurs an opportunity arises for investors to profit from trading disparities found in the spread between US gold equity and physical gold.

Due to market manipulation in the form Western government's monetary policies, better known as quantitative easing, hedgers and investors are turning to alternative and more lucrative investments in order to offset the effects of inflation and seek profitable trading strategies respectively. At the same time they are looking to profit in a volatile and highly unpredictable economic climate. In particular, some investors are investing in gold to protect their purchasing power against rising inflation however with gold rising to record highs in 2011 many are now speculating whether this upward trend is losing momentum. Nevertheless, the 'Gold Miner Spread' may present investors with another long term alternative investment opportunity as it also offers exposure to gold mining stocks. The market offers two gold miner exchange traded funds (ETFs). The first is the GDX Market Vectors Gold Miners ETF which provides exposure to large cap³¹ gold miners while the second is the GDXJ Market Vectors Junior Gold Miners. The latter has exposure to small-cap³² gold mining companies. Here, the larger cap exposure is evaluated by trading the spread between GLD (physical gold) and GDX.

In particular, gold miners use this spread to offset inflation as physical gold is extracted and then exchanged for 'paper'. To hedge against inflation, gold miners 'long' gold and 'short' US gold equity to protect purchasing power of the dollar. Among other variables, inflation is just one of the

³¹Market capitalisation greater than \$100 million with an average trading volume of 50,000 shares during a six month trading period.

³²Includes small to medium sized mining companies with the majority producing revenues from either gold and/or silver mining.

factors affecting Gold Miner's ability to match the price performance of Gold as a physical asset and hedging on the financial markets becomes vital to offset systematic risk and maintain profitability.

GDX is known as Market Vectors Gold Miners ETF and its underlying value is derived from stocks and American Depository Receipts of gold mining companies. Following its inception in 2006 the objective of this ETF has been to replicate the price and yield performance of the NYSE ARCA Gold Miners Index (GDM). The net asset valuation of GDX is currently calculated from 31 underlying mining stocks and thus is a general indicator for the performance of large cap US Gold Equity stocks. This ETF is weighted based on market capitalisation with its larger holdings being in Barrick Gold Corporation (16.51%), Gold Corp Inc. (13.19%), Newmont Mining Corporation (11.18%), Kinross Gold Corporation (5.76%), AngloGold Ashanti Ltd. (5.60%), Agnico-Eagle Mines Ltd. (4.40%), Randgold Resources Ltd. (4.39%), and Yamana Gold Inc. (4.35%).

To gain physical gold exposure many investors are drawn to investing in the SPDR Gold Trust GLD ETF as this is the most liquid commodity ETF offered on the financial markets. GLD is from the State Street Global Advisor fund family and was first introduced in 2004 as a means to replicate the price of gold bullion. As a financial instrument it has provided market participants with exposure to price fluctuations in the commodity while removing risks associated with the delivery of the physical commodity as well as avoiding storage costs.

Pressures to grow and expand production capacity in the mining industry have been challenged by a shortage of technical expertise and increasing production costs. Many gold miners have undertaken expensive and wasteful projects which have eroded profitability. As a result, recent years have seen little value being returned to shareholders with much of the capital being spent on project related activities. Such wasteful spending has created a drag on gold miners' stock

performance relative to the physical commodity that they produce. In addition, the price of gold has also been affected by monetary policies. Ultimately, fluctuations in the prices of US Gold Mining stocks as well as Gold create short term disparities in the spread from which one can profit by trading effective market timing strategies. With high inflation and the diminishing purchasing power of the US dollar as a world currency further analysis of the Gold Miner Spread is warranted.

2.0 Literature Review

On review of past literature it becomes apparent that few have analysed the advantages of trading the Gold Miner Spread. Furthermore, the use of non-linear modelling as a vehicle to forecast next day returns is virtually non-existent within the confines of current academic spread trading literature.

Existing literature such as Triantafyllopoulos and Montana (2009) analyses the benefits of using cointegration modelling to trade the Gold Miner Spread. In this analysis the co-integrating relationship between the GLD and GDX ETFs for the period of May 23, 2006 to August 06, 2008 is traded. Findings conclude that the pair displays a strong cointegrating relationship during the period of 19/07/2006 to 17/12/07. Chan (2009) also explores the GLD GDX spread using a cointegration trading model. In his investigation, Chan (2009) finds that there are numerous instances of mean reversion. Furthermore, he also discovers that the spread is most profitable when 'shorted' during highs experienced in mid-July and 'longed' during seasonal lows experienced in early September. A short position consists of shorting gold and longing gold mining stock while a long position is the opposite with a long position in gold and a short position in gold equity.

Vidyamurthy (2004) uses the Engle and Granger (1987) two step methodology of cointegration to identify stationarity between pairs of stocks. The difference between stock A and stock B is essentially the spread. Once spreads are identified and the residuals are deemed

stationary using the Augmented Dickey Fuller (ADF) test then trading strategies are developed. Vidyamurthy (2004) opens a long position when the portfolio of these cointegrated stocks is below its long-run equilibrium and he ‘short sells’ the portfolio when it is above its long-run equilibrium. Profit is generated by this strategy when the portfolio mean reverts and positions are closed off.

On the topic of spread trading, Dunis *et al.* (2011c) utilise both Neural Networks and a Genetic Programming Algorithm (GPA) in the evaluation of the ‘Corn Crush’ spread. In this analysis a time varying leverage filter is also employed to improve annualised returns. More importantly, the GPA model is found to display superior forecasting abilities when compared to both linear models and neural networks. The time varying leverage filter was also found to significantly improve annualised returns while reducing maximum drawdowns. These findings are of particular interest as the results produced by the MLP and GPA models can be compared directly to results produced from an application of similar methodologies to the Gold Miner Spread. Although both spreads are different in nature they are back tested over similar time periods with both spreads widening during the onset of the crisis and, at times, both can also be extremely volatile. Results reveal that non-linear methodologies produce both statistically intuitive and economically profitable forecasts.

3.0 Descriptive Statistics and Related Financial Data

Daily closing prices for each commodity ETF were obtained from DataStream for the period extending from May 23, 2006 to June 30th, 2011. GLD is the most liquid commodity ETF and has been traded on the NYSE Arca platform since 18/11/04. The GDX ETF is a more recent addition to NYSE Arca platform and has only been traded as an ETF since the 23/05/2006. Both ETFs are traded on the NYSE ARCA exchange from market open to close and as a result the issue of non-synchronous pricing that plagues many other investigations does not present itself here. In

order to provide a continuous time series it was decided that ETF data would be favoured to model the commodity spread, as ETFs in particular, offer numerous advantages over trading futures contracts and equities. In general, ETFs provide diversification, lower transaction costs equivalent to trading equities, and healthy volumes of liquidity while escaping company specific risks. During the full sample period a daily average of 10,336,353 shares of GLD were traded compared to a daily average of 5,317,258.83 shares for GDX. For this study, the GLD and GDX ETFs are traded to better understand and evaluate the ‘Gold Miner Spread’.

During the in sample period (23/05/2006 – 16/12/2009) a correlation coefficient of 0.75 is calculated for the relationship between GLD and GDX. This suggests that an increase in the price of Gold Bullion results in an increase in share prices of gold mining companies (assuming that all other costs including operating costs remain constant).

$$\Delta S_t = \left[\frac{(P_{GLD}(t) - P_{GLD}(t-1))}{(P_{GLD}(t-1))} - \frac{(P_{GDX}(t) - P_{GDX}(t-1))}{(P_{GDX}(t-1))} \right] \quad (30)$$

where:

ΔS_t = percentage returns of spread at time t .

$P_{GLD(t)}$ = is the price of GLD at time t (in cents per \$)

$P_{GLD(t-1)}$ = is the price of GLD at time $t-1$ (in cents per \$)

$P_{GDX(t)}$ = is the price of GDX at time t (in cents per \$)

$P_{GDX(t-1)}$ = is the price of GDX at time $t-1$ (in cents per \$).

The methodology applied to calculate the returns of the Gold Miner spread can be seen in equation 30. This calculation has also been used by Butterworth and Holmes (2002) and more recently by Dunis *et al.* (2011c) when computing spread returns. Using discrete returns enables

investors to easily measure how fluctuations in the spread affect profitability on a daily basis when considering daily profit and loss accounts irrespective of how much is invested. Furthermore, discrete returns such as Equation 1 are used in time series analysis because they often result in stationary time series which is important when optimising models. For instance, stationary time series converge much quicker when training neural networks compared to price time series.

3.1 Statistical Behaviour of the Spread

The price time series for the full sample period (23/05/2006 – 30/06/2011) can be observed in figure 12.

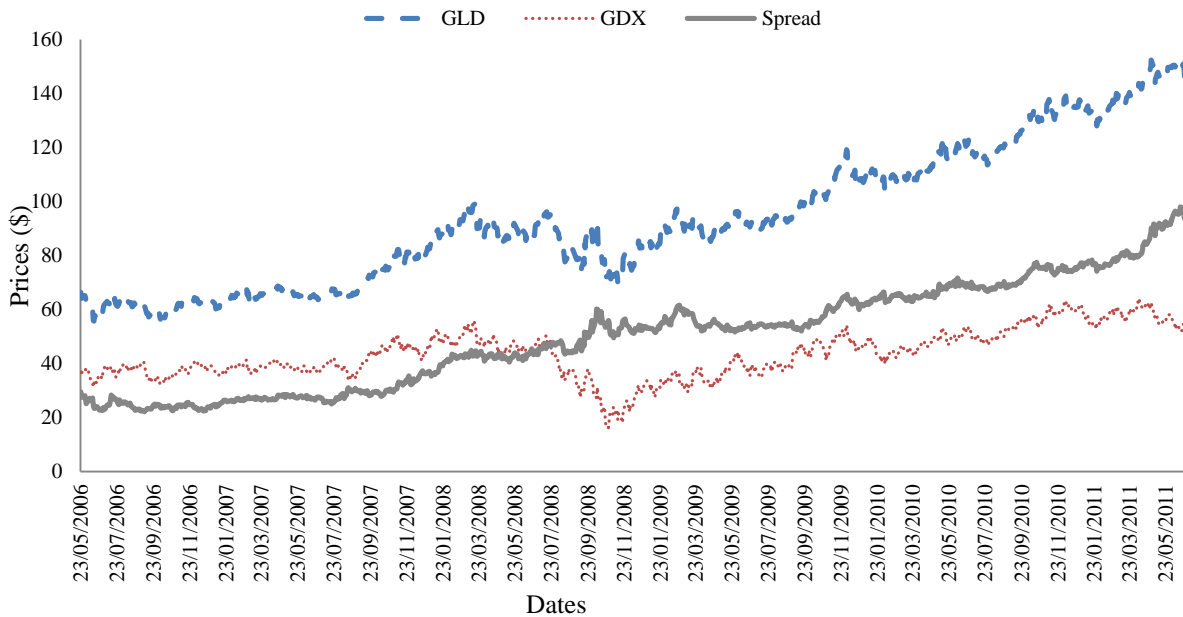


Figure 12. The gold miners spread ETF daily closing prices (23/05/2006 – 30/06/2011)

3.2 Descriptive Statistics and Explanatory Variables

Inferences are based on the daily change in closing prices³³ and from the histogram in figure 13 it can be observed that the Gold Miners spread return series is non-normal (confirmed at a 99% confidence level by the Jarque-Bera test statistic), with a slight skewness and high kurtosis. Negative skewness is widespread in time series returns because, in general, market participants react more dramatically to negative news than they do to positive news which creates an asymmetric distribution of returns as seen in figure 13.

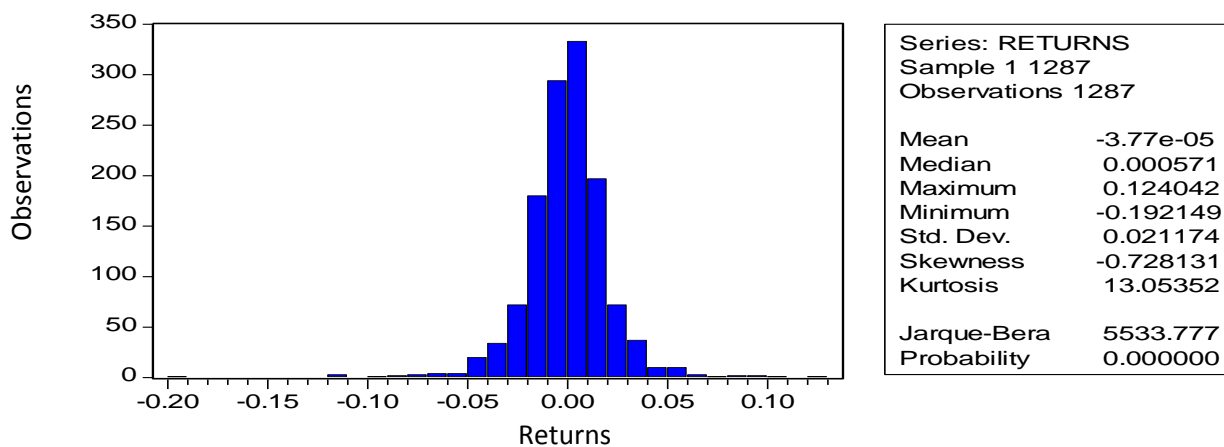


Figure 13. Histogram of GLD/GDX spread return series (full sample)

The observed data period has been segregated into sample periods as conducted during the modelling process.

Name of period	Trading Days	Beginning	End
Total dataset	1,287	23 May 2006	30 June 2011
Training dataset (in-sample)	640	23 May 2006	04 December 2008
Test dataset (in-sample)	260	05 December 2008	16 December 2009
Validation set (out-of-sample)	387	17 December 2009	30 June 2011

Table 14. Neural data segregation for the full sample period

³³In the analysis, arithmetic returns were used as opposed to logarithmic returns due to the fact that the latter are not linearly additive across portfolio components. This can prove to be problematic and furthermore market participants have a tendency to look more at discrete returns in their daily trading activity. On this basis the use of arithmetic returns is deemed to be more realistic and more suited to the study.

Many have visited and reviewed the topic of ‘over fitting’ when using neural networks. For one, White (1989) analyses the different learning procedures used to train artificial neural networks in attempt to determine appropriate neural architectures. He concludes that a networks complexity should be determined by the number of input nodes. Furthermore, White (1989) also finds that learning techniques used to train neural networks are inherently statistical techniques. In particular he explains that “as with any random variable, the behaviour of the weights (w) is completely described by its probability law.” Therefore, the law of large numbers and the central limit theorem fully apply during this process. The greater the training sample size ‘ n ’, the more statistically accurate a neural network model will be. However, even with large samples a control needs to be in place to allow a network to ‘grow’ at an appropriate learning rate relative to the size of the available training set. For this reason the MLP neural network here is restricted by a learning rate of 0.001 as explained in the appendix while the learning rate for the PSO RBF is optimised by the PSO algorithm. Nevertheless, many have found that stopping the learning ‘early’ helps to generate more consistent performance during the out of sample period.

Due to the nature of neural networks and the risk of ‘over fitting,’ the in sample period is divided into two subsets as displayed table 14. As a result, the training and testing of each network is separated to reduce the risk of ‘over fitting’ of the dataset during the learning phase. Unfortunately this is common drawback when using neural networks so dividing the dataset in such a way is necessary to preserve the integrity of each model. In comparison, GPA models are also affected by this issue however not in the same way as they differ architecturally. In a GPA model this is controlled by setting a pre-determined tree structure. In this case it was found that a tree length of 6 was the most appropriate³⁴. As a result, data segregation for the GPA model only requires data to be separated into two sample periods as shown in table 15. Using more data to

³⁴ Further justification for this can be seen in section 4.3.

estimate the GPA while limiting the dataset by imposing an ‘early stop’ when training neural networks could prove to be beneficial in some cases while problematic in others. For instance, a GPA model may outperform the MLP NN simply because it is using more data however there is also a higher possibility that the GPA model may ‘over fit’ the data. When using a variety of different forecasting methodologies to model the same time series this is something that a practitioner must be aware of.

Name of period	Trading Days	Beginning	End
<i>Total dataset</i>	1,287	23 May 2006	30 June 2011
<i>Training dataset (in-sample)</i>	900	23 May 2006	16 December 2009
<i>Validation set (out-of-sample)</i>	387	17 December 2009	30 June 2011

Table 15.GPA data segregation for the full sample period

It is important for practitioners to put controls in place and set parameters during the learning phase in order to reduce the risk of over familiarisation with the dataset as this could prove detrimental to the accuracy of forecasts.

Number	Input Variable	Lag
1	Gold Miner Spread Returns	1
2	Gold Miner Spread Returns	2
3	Gold Miner Spread Returns	3
4	Gold Miner Spread Returns	4
5	Gold Miner Spread Returns	5
6	Gold Miner Spread Returns	6
7	Gold Miner Spread Returns	7

Table 16.Explanatory variables for the neural networks and the GPA

To forecast the future direction of the spread, the above past lagged observations of spread returns are used for the neural networks. Each of the algorithms are run and back tested to identify patterns found within each of the lagged time series displayed in table 16. For the purpose of this investigation 7 inputs were initially selected and tested during the in sample period. On the whole, although a limited universe of inputs is used these lagged spread returns were found to produce

adequate annualised returns and attractive trading performances. Future studies will place more of an emphasis on input selection and significance however for the purpose of this forecasting exercise the 7 chosen explanatory variables are justified.

4.0 Methodology

This section provides details about the different models, trading strategies and trading filters used during the simulation. All of which were implemented in order to establish parameters for modelling the Gold Miner Spread. The particulars with regards to forecasting the time series are also discussed to include various benchmark models, two different types of neural network architectures and a genetic programming algorithm (GPA). For the neural network models the widely used Multi-Layer Perceptron (MLP) model is applied as well as a novel meta-heuristic Particle Swarm Optimisation Radial Basis Function (PSO RBF) network. Please note that Chapter 2 introduces and explains each of the proposed models in more detail.

4.1 Benchmark Strategies and Models

Benchmark strategies include a naive 1 day a head strategy and a buy and hold strategy. For the Buy and Hold strategy the fundamental view is that gold as a physical asset will outperform gold mining equity in the medium to long term. With this as a justification, the spread is ‘bought’ at the beginning of the sample period and held until the end of the sample period at which point it is sold in order to realise profit / loss. Additional, benchmark models were estimated and optimised to include an Autoregressive Moving Average (ARMA) model and a cointegration model.

4.1.1 ARMA

Using a correlogram as a guide in the training and the test sub-periods a restricted ARMA (12,12) model was selected. All coefficients are significant at the 99% confidence interval. The null hypothesis that all coefficients (except the constant) are not significantly different from zero is rejected at the 99% confidence interval (please refer to table 62 in Appendix A.4).

The (12,12) model was also retained for an out-of-sample trading simulation. The specific ARMA model used to forecast the Gold Miner Spread is mathematically represented as:

$$Y_t = -3.56 * 10^{-5} - 0.541Y_{t-1} - 0.287Y_{t-2} + 0.176Y_{t-8} + 0.267Y_{t-11} + 0.581Y_{t-12} - 0.490\varepsilon_{t-1} - 0.205\varepsilon_{t-2} + 0.171\varepsilon_{t-8} + 0.401\varepsilon_{t-11} + 0.611\varepsilon_{t-12} \quad (31)$$

4.1.2 Cointegration Model – The Long Run Relationship

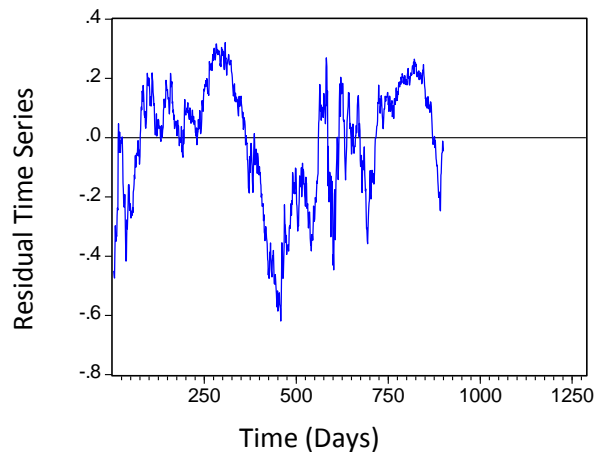


Figure 14. In Sample residuals of the long-run cointegrating relationship (μ)

Figure 14 displays characteristics of a mean reverting spread throughout the in sample dataset and as a result it does not follow a random geometric walk as is often the case when modelling other spreads. In support of this, the residual series is subjected to the Augmented Dickey-Fuller (ADF) test to determine whether or not the series has a unit root. Unit-roots are key features associated with random walk models which are subsequently deemed non-stationary.

Therefore, it is beneficial to the modelling process for the test to reject the null hypothesis that the time series has a unit root. Results from the ADF test can be seen in table 17 which confirms stationarity throughout the in sample period and rejects the presence of a unit root at a 95% confidence level. From this analysis, a cointegration model of the pair is deemed to be a suitable benchmark when trading of the Gold Miner Spread.

Null Hypothesis: CointEq01 has a unit root		
Exogenous: Constant, Linear Trend		
Lag Length: 0 (Automatic based on Modified SIC, MAXLAG=20)		
Augmented Dickey-Fuller test statistic	t-Statistic	Prob.*
	-3.443952	0.0463
Test critical values:	1% level	-3.968325
	5% level	-3.414838
	10% level	-3.129588

*MacKinnon (1996) one-sided p-values.

Table 17. Augmented dickey fuller unit root test (in-sample)

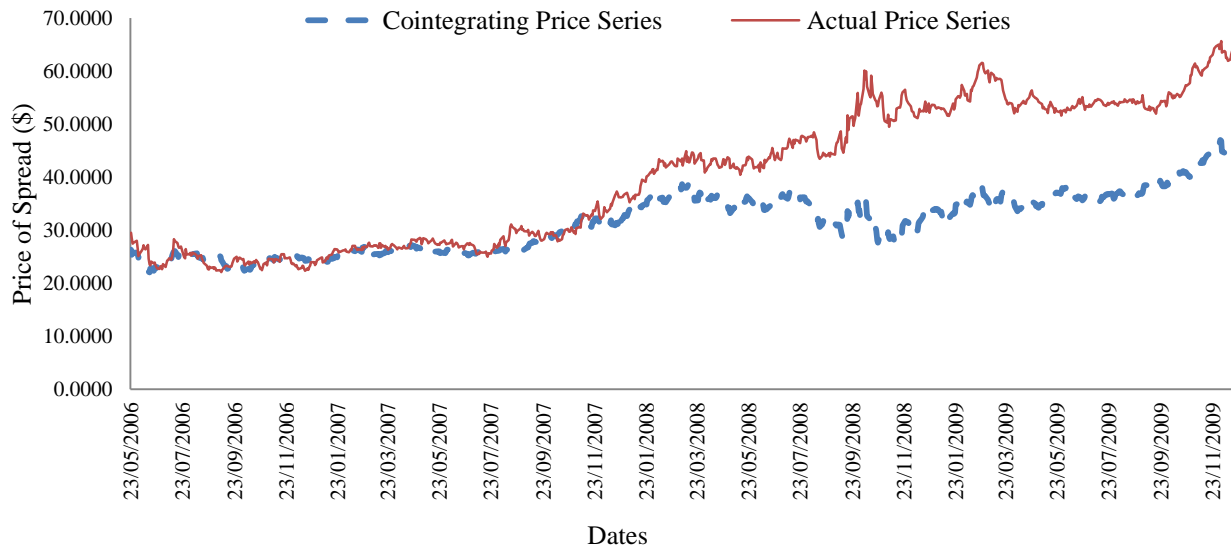


Figure 15. In sample price series

The cointegration model is estimated in accordance with Dunis *et al.* (2006) who use the cointegrating vector as calculated by the Johansen (1988) cointegration test. Both variables

LGDXPICESER (log of GDX price series) and LGLDPRICESER (log of GLD price series) are I(1). Hence they are stationary in First difference or integrated of order one. By observation of figure 15, it can be seen that there is a structural break in the relationship as a result of a ‘black swan’ event caused by the ‘Credit Crisis’ experienced in September / October 2008. Despite a structural break in the spread between GDX and GLD, the trading model successfully captures the long term relationship between GLD and GDX for the period of 23/06/2006 to 23/05/2008.

$$\text{Forecasted Spread Price Series}_{t+1} = \text{GDX}_{t-1} * 0.015615 + \text{GLD}_{t-1} * 0.009882 / (0.015615 + 0.009882) \quad (32)$$

This is also validated by Triantafyllopoulos and Montana (2009) who carry out similar research. The cointegrating price series of the Gold Miner Spread is calculated as shown in equation 32 using the coefficients derived from the Johansen (1988) cointegration test when estimating the Vector Error Correction Model (VECM).

A return series is then calculated from the resulting price series as:

$$R_t = (P_t / P_{t-1}) - 1 \quad (33)$$

5.0 Empirical Results

As a generic unfiltered trading rule, the spread is traded based on the signals produced by each of the models with the exception of the buy and hold strategy. For the buy and hold strategy, the spread is traded only at the beginning and end of each sample period. For instance, during the in-sample period the spread is bought on 23/05/2006 and sold on 16/12/2009. The spread is then bought back on 17/12/2009 and sold on 30/06/2011, completing the out of sample period. Transaction costs are calculated using 10 basis points as commission per trade with each leg costing 5 basis points. This is quite a conservative figure and is in line with the commission

schedule offered by interactive brokers³⁵ with stocks costing 5 basis points per transaction. One of the benefits of trading ETFs is that transaction costs are the similar as those charged for trading equities.

Statistical analysis and trading results are displayed in sections 5.1 and 5.2 respectively. Formulas for the performance measures used in the analysis can be found in A.1 of the appendix.

5.1 Statistical Measures

	Naive	Cointegration	ARMA	MLP	RBF	GPA
<i>MAE</i>	0.0235	0.0196	0.0161	0.0162	0.0163	0.0156
<i>MAPE</i>	2113.51%	952.86%	294.71%	179.30%	249.69%	157.04%
<i>RMSE</i>	0.0346	0.0283	0.0234	0.0235	0.0235	0.0215
<i>THEIL-U</i>	0.7228	0.7190	0.8116	0.8270	0.7725	0.6393
<i>Correct Directional Change (CDC)</i>	47.44%	53.44%	51.33%	52.11%	53.56%	54.22%

Table 18. In sample statistical accuracy

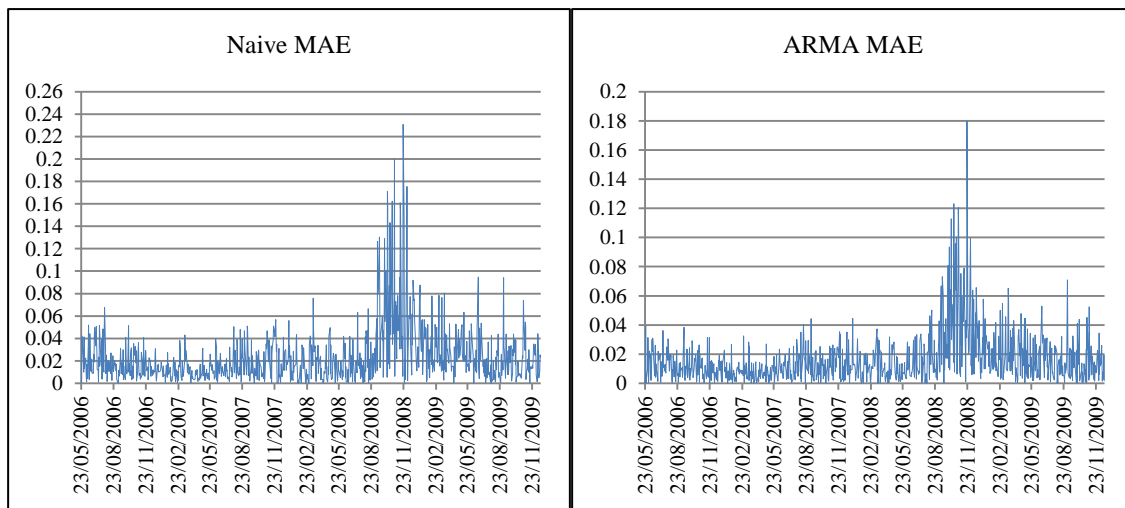
	Naive	Cointegration	ARMA	MLP	RBF	GPA
<i>MAE</i>	0.0142	0.0125	0.0099	0.0099	0.0101	0.0098
<i>MAPE</i>	484.05%	485.45%	176.50%	143.71%	128.65%	108.46%
<i>RMSE</i>	0.0185	0.0164	0.0127	0.0128	0.0131	0.0127
<i>THEIL-U</i>	0.7304	0.7151	0.8205	0.8490	0.7939	0.8603
<i>Correct Directional Change (CDC)</i>	48.84%	53.23%	53.49%	53.23%	53.49%	54.01%

Table 19. Out-of-sample statistical accuracy

By observation of tables 18 and 19 the differences between some of the models are marginal however the GPA model maintains the best performance with the lowest errors over both sample periods. The PSO RBF network and ARMA model marginally beat other linear and non-linear models to rank joint second during the in sample period. Over both periods the PSO RBF model is closely challenged by statistics produced by both the MLP and ARMA models. Notably, the overall MAE and RSME statistics are slightly higher during the in sample dataset due to the fact that this time period experienced the shock of the financial crisis in 2008.

³⁵ <http://www.interactivebrokers.co.uk/en/index.php?f=commission&p=stocks2>

With reference to the correct directional change (CDC henceforth) statistic, the rankings remain the same as the GPA model ranked highest producing 54.22% during the in sample period and 54.01% during the validation period. PSO RBF however was ranked second in its signal prediction accuracy with 53.56% CDC in sample and 53.49% CDC during the out of sample dataset. This indicator is widely used as a measurement to gauge a models ability to correctly forecast change of direction as opposed to magnitudes of change. None of these statistics are statistically significant however scores of greater than 50% are more desirable. Ultimately, it is apparent that the non-linear models provide slightly more accurate forecasts when compared to linear forecasting methodologies.



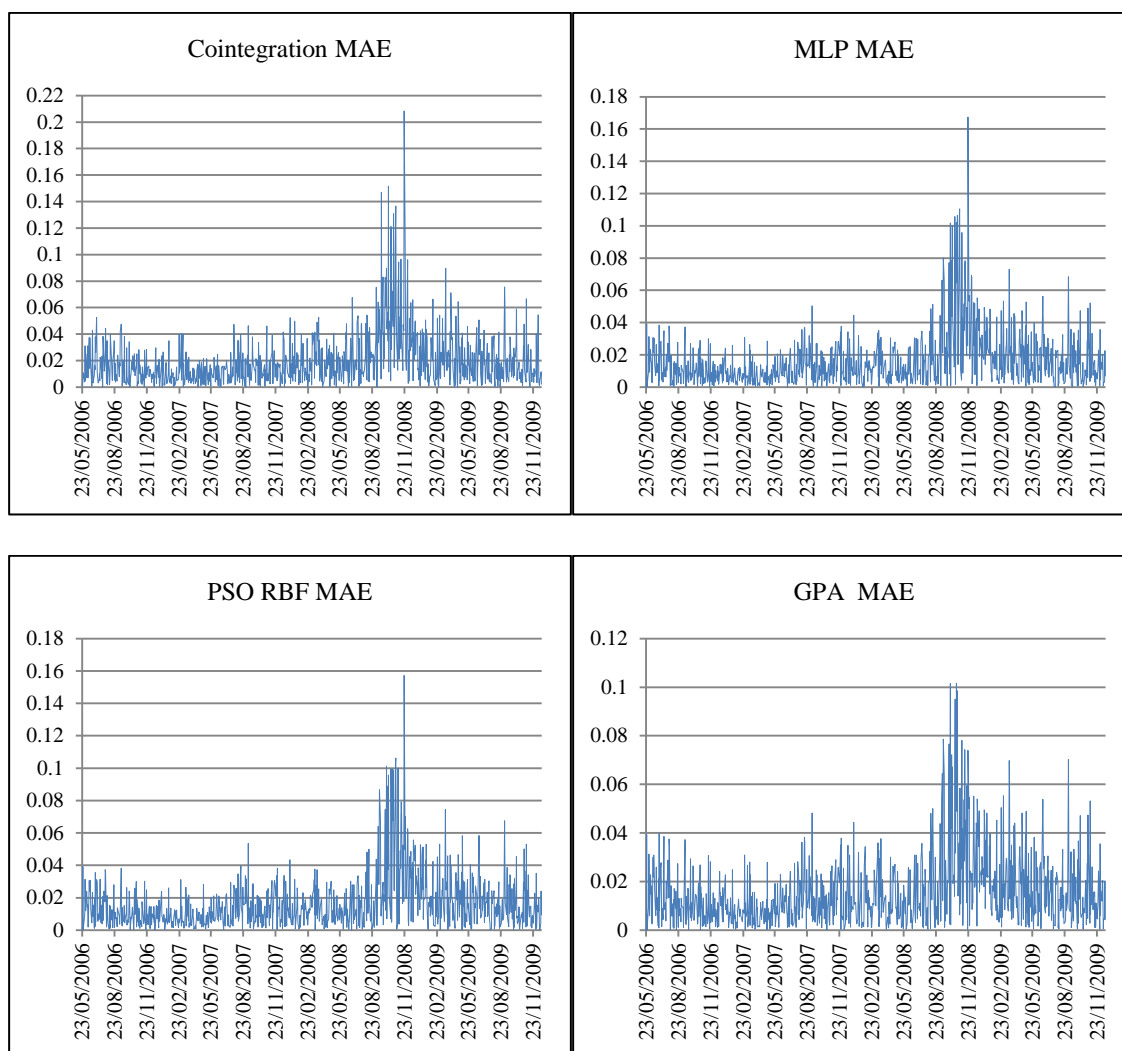


Figure 16.MAE in-sample statistics

By observation of figure 16 it is apparent that all of the models are incapable of forecasting next day returns during the period of September 2008 to November 2008. This is the start of the financial crisis sending financial markets plummeting around the world. Although these ‘black swan’ events are very difficult to foresee, perhaps nearer the impossible, the non-linear methodologies do in fact display lower standard errors during this period.

5.2 Summary Trading Performance

Inherently, neural network and evolutionary models are stochastic which, as a result, creates large variances between model outputs. In attempt to minimise these variances an average of the 10 best performing models is used to produce non-linear forecasts during the in sample back test. Aranha and Iba (2008) also adopt a similar approach when calculating forecasted stock market returns. They use an average of returns derived from 30 models generated by the Genetic Programming algorithm. Performance is measured in terms of annualised returns during both sample periods.

Only the training set is used for model selection and parameter optimisation. These models were then selected and applied to a validation period in order to test performance and robustness. Trading models are ranked according to annualised returns during the in sample training (as provided in the appendix). Results from the out of sample trading simulation are displayed in table 20.

	Buy & Hold	Naive	Cointegration	ARMA	MLP	PSO RBF	GPA
Gross annualised return (excl. costs)	5.66%	-0.41%	20.00%	22.93%	22.70%	28.30%	28.73%
Annualised volatility (excl. costs)	20.16%	20.06%	16.15%	20.11%	20.01%	19.99%	19.98%
Maximum drawdown (excl. costs)	-	-34.62%	-8.35%	-13.26%	-12.30%	-13.95%	-11.86%
Calmar ratio	-	-0.01	2.39	1.73	1.85	2.03	2.42
Information ratio	0.28	-0.02	1.24	1.14	1.13	1.42	1.44
# Transactions (annualised)	1.30	129	136	132	91	169	135
Total trading days	387	387	387	387	387	387	387
Transaction costs (annualised)	1.06%	13.82%	14.47%	13.15%	9.98%	17.80%	14.34%
Net annualized returns (incl. costs) ³⁶	5.53%	-13.30%	6.46%	9.78%	13.65%	11.43%	15.31%
Ranking	6	7	5	4	2	3	1

Table 20. Out of sample trading performance (unfiltered)

During both sample periods results reveal that model rankings remain unchanged at the bottom however the PSO RBF model is now the third best with MLP moving up to second. At the top the GPA model remains consistent in generating superior trading performance over both sample periods. Interestingly, the PSO RBF model is ranked second without transaction costs however these returns are eroded by 17.80% when costs are taken into account. The use of a trading filter is

³⁶ Transaction costs amount to a total of ten basis points (5 basis points per leg).

also justified as high volatility and poor maximum drawdowns present themselves as unattractive and unsustainable to the unseasoned investor. For this reason a correlation trading filter was applied. Results from this filter can be seen in table 22. Furthermore, this also allows for the exploitation of the correlation relationship between GLD and GDX. Volatility in terms of annualised standard deviation and RiskMetrics experienced during ‘unfiltered’ trading scenarios is also high as seen in table 21 and figure 17 respectively.

Volatility of returns	GLD	GDX	Spread
2006	22.51%	37.20%	24.48%
2007	17.61%	33.35%	21.59%
2008	32.62%	75.08%	55.87%
2009	20.68%	50.20%	34.29%
2010	16.56%	30.30%	20.04%
2011	13.57%	27.43%	19.92%
In sample	24.16%		
Out of sample	15.95%		

Table 21. Volatility of spread returns (annualised)

By observation, table 21 displays a higher historical volatility in the GDX time series compared to GLD as it was nearly twice as volatile during the in sample period. The reasoning behind this difference is because GDX is exposed to both the risk of physical gold fluctuations as well as equity and stock market risks. Holding the GDX ETF provides a leveraged exposure to gold as a physical asset. Analysis also finds that GLD and GDX are highly correlated with a correlation coefficient of 0.75 during the in sample period. Hence, they both exhibit similar upward and downward moves with the magnitude of GDX’s movements being double that of GLD.

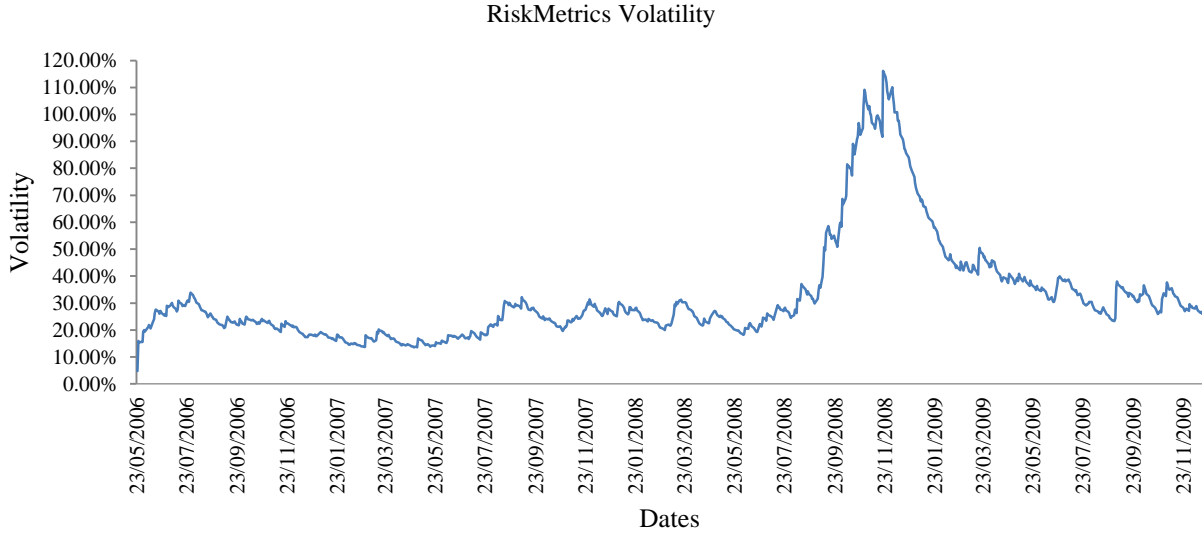


Figure 17. RiskMetrics of returns (in-sample)

The RiskMetrics formula used to calculate volatility is:

$$\mu^2_{(t+1/t)} = \beta * \mu^2_{(t/t-1)} + (1 - \beta) * r^2_{(t)} \quad (34)$$

where: μ^2 is the volatility forecast of the spread returns,
 r^2 is the squared return of the spread,
 β 0.94 for daily data as computed in JP Morgan (1997).

Parameters for a rolling correlation filter are optimised and estimated during the in sample period. The adopted methodology is similar to that utilised by Dunis *et al.* (2006). Firstly, the rolling correlation window is optimised and once the correlation is calculated for this chosen period the daily change in correlation is then calculated as: $\Delta Correl = (Correl_t / Correl_{t-1}) - 1$. This provides an indication of daily changes in correlation over the selected period. A trading filter is then derived from the daily changes in correlation to filter out periods of positive correlation. The

filter produces a ‘1’ for negative change and ‘0’ for positive change. It is important to note that the filtered model only trades during periods of negative changes in correlation. Finally, the trading rule is established by trading on the signals produced by each of the models only when the filter signifies a ‘1’. When ‘0’ is signalled by the correlation filter then no transactions occur.

The use of a correlation trading filter has proven to be lucrative in previous spread trading literature as it enhances annualised returns while at the same time reducing overall volatility. For instance, Dunis *et al.* (2006) and Dunis *et al.* (2010) also develop profitable trading models using rolling correlation filters.

Figure 18 displays the in sample correlation of the spread which, as mentioned above, is only traded when the change in correlation over an 18-day rolling period is negative. This presents opportunities to profit from disparity in the spread as each leg moves in opposite directions causing the spread to widen.

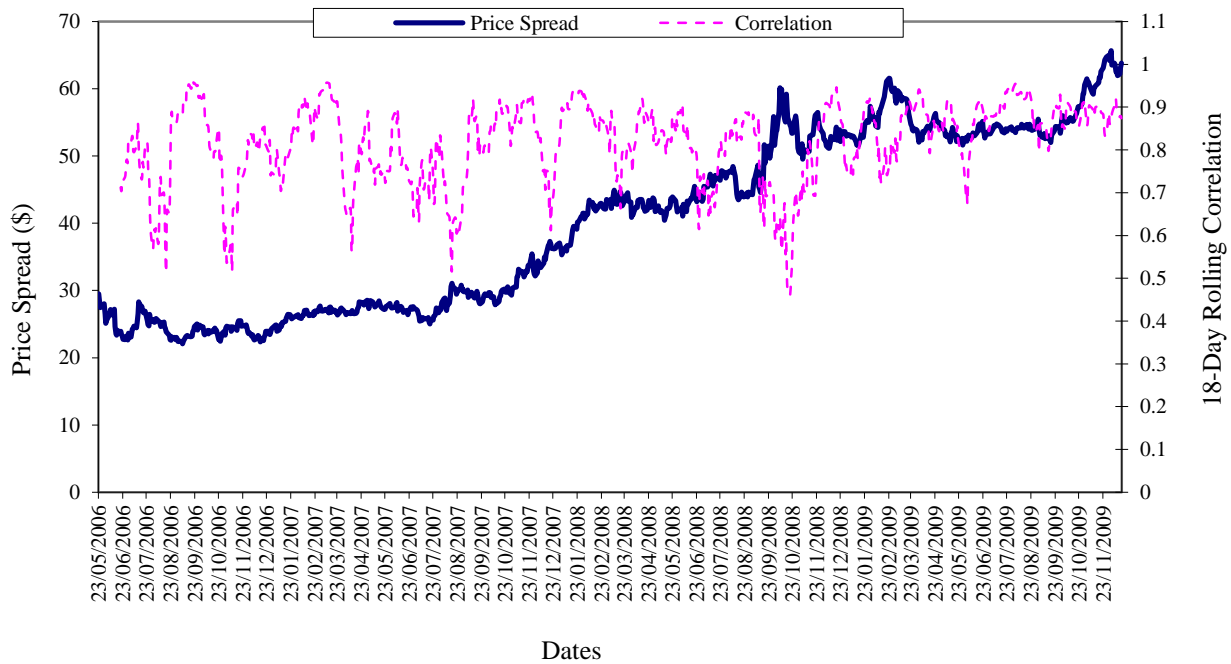


Figure 18. In sample correlation

By observation of figure 18, it can be seen that the correlation coefficients between GLD and GDX falls below 0.50 during the onset of the financial crisis. This resulted in widening of the spread from just over \$40 to almost \$60. At this point the trading filter is triggered and each of the models is traded based on their individual trading signals.

	Naive	Cointegration	ARMA	MLP	PSO RBF	GPA
Gross annualised return (excl. costs)	-3.46	23.23%	22.34%	26.78%	29.96%	33.61%
Annualised volatility (excl. costs)	15.06%	14.99%	13.69%	14.97%	14.95%	14.91%
Maximum drawdown (excl. costs)	-33.09%	-11.53%	-12.36%	-12.55%	-11.88%	-9.04%
Calmar ratio	-0.10	2.02	1.81	2.13	2.52	3.72
Information ratio	-0.23	1.55	1.63	1.79	2.00	2.25
# Transactions (annualised)	99.63	95.72	101	94.42	104.19	107.44
Total trading days	387	387	387	387	387	387
Transaction costs (annualised)	11.35%	10.50%	10.03%	10.37%	11.35%	11.67%
Net annualized return (incl. costs)	-10.19%	13.66%	12.32%	17.34%	19.54%	22.86%
Correlation filter excess return	-0.13%	7.20%	2.54%	3.69%	8.11%	7.55%
Maximum drawdown reduction	-1.53%	-3.18%	0.90%	-0.25%	2.07%	2.82%
Volatility reduction	5.00%	1.16%	6.42%	5.04%	5.04%	5.07%
Ranking	6	4	5	3	2	1

Table 22. Out of sample trading performance (filtered)

On a review of the empirical results in table 22 it is apparent that the correlation filters enhance trading performance. Remarkably, the correlation filter improves annualised returns for the PSO RBF model by 8.11%. Under the supervision of a trading filter the GPA model maintains its ranking as the best performing model. The impact of the correlation filter resulted in 7.55% more in returns for the GPA model. While the MLP model is relegated to third, based on annualised returns, the correlation filter generated excess returns of 3.69% when compared its unfiltered performance in table 20. Furthermore, the filter also reduces the annualised volatility for all 6 of the models. Maximum drawdowns, on the other hand, are only improved in half of the models with the cointegration model, in particular, experiencing 3.18% worse drawdowns. Lastly, with the exception of the MLP model, transaction costs are reduced when the correlation filter is implemented as fewer transactions are undertaken.

On the whole, the non-linear methodologies prove to generate higher returns while also achieving the best risk/return profiles.

6.0 Concluding Remarks

From the outset the aim was to model and forecast the Gold Miner spread in a trading simulation expanding from 17/12/2009 to 30/06/2011, the out-of-sample trading period. Results produced from each of the unfiltered models were for the most part satisfactory; with the GPA proving superior. The use of a PSO algorithm during training, as an alternative to a traditional back propagation algorithm, proved less time consuming while only slightly improving the accuracy of the RBF neural network. Perhaps a direct comparison between MLP models trained using the PSO algorithm and MLP models trained through the use of a back propagation algorithm would produce a more distinguished set of results. This is however a limitation of the investigation and further research will be provided to provide such comparisons. Another limitation of this research is the unadventurous universe of explanatory variables used as inputs for the Neural Network and GPA models. A more exhaustive universe of non-linear inputs will also be investigated in subsequent analysis of the Gold Miner spread however by only using autoregressive spread returns as inputs it can be argued that the quality of each non-linear model is tested.

Results from this analysis are corroborated by earlier work carried out by Dunis *et al.* (2011a) who forecast the EUR/USD relationship and find that their GPA model also outperforms a generic MLP neural network model by contributing an additional 3.75% to annualised returns. When considering filtered results, the most profitable model was the GPA with the PSO RBF ranking second best. In particular, the PSO model however is most improved under the supervision of a correlation trading filter with annualised returns being enhanced by 8.11%. This is closely followed by the GPA model which registered the second most improved returns with 7.55% more. Filtered models experience reduced annualised volatility with most reducing volatility on average by 5%.

Ultimately, this investigation offers the reader an example of forecasting and trading a spread between Physical Gold and US Gold Equity while providing a valuable insight into the use of nonlinear modelling for trading and hedging purposes. The benefits of using non-linear forecasting methodologies are not exclusive to commodity spreads. They can also be applied to pairs trading of equities, spread trading of futures contracts, straight equities and various other asset classes. While correlation filters are of particular use when trading commodity spreads they can also be applied to pairs trading of other financial instruments. The use of different optimisation methods during the training of these non-linear models also offers a comparative analysis for both academics and market participants.

CHAPTER 6: An Analysis of ETF Commodity Spread Portfolios – A Case of Mean Reversion

August 2012

Abstract

This paper models portfolios of ETF (Exchange Traded Funds) spreads over a 5 year and 7 month period. Daily closing prices are observed for the period of May 2006 to December 2011. In total, 3 portfolios are constructed to trade spreads between baskets of US Mining / Refining stocks and the physical commodities that they produce.

The contribution of this investigation is threefold. Firstly, the research aims to identify and construct stationary spreads during an in-sample period using time varying betas derived from the non-conditional covariance of each spread. Secondly, the work endeavours to model each of the spreads using an Autoregressive Moving Average (ARMA) model, a cointegration model and a Multi-layer Perceptron (MLP) Neural Network (NN) model. Models with the same methodologies are then grouped together and traded as equally weighted unfiltered portfolios. Finally, in order to enhance performance a mean reversion trading filter is employed.

Results reveal that the MLP NN model display superior statistical accuracy and the most profitable trading performance as a portfolio when tasked with forecasting next day returns. Furthermore, trading a portfolio of ETF spreads using a mean reversion strategy offers greater risk-return trade-offs than portfolios of unfiltered models.

Keywords

Spread Trading, Mean-Reversion, Time Varying Betas, Exchange Traded Funds, ARMA, Cointegration, Multi-Layer Perceptron Neural Network.

1.0 Introduction

The motivation behind this paper is to investigate price relationships between commodity mining/refining companies and the physical commodities that they produce. Fundamentally, there is a relationship between the two and as a result there is a good reason to believe that mining and refining companies depend heavily on the spot and futures prices of commodities such as gold, oil, and silver. In the short term however, these relationships have a tendency to deviate from long term equilibriums and it is from these deviations that profitable mean reversion strategies can be developed. Deviations from long term relationships may be due to a number of factors. For instance, the CFTC (Commodity Futures Trading Commission) indicate that speculation was one of the driving forces behind the crude oil spike experienced in 2008. As speculators are not interested in physical delivery of commodities it is believed that large quantities of oil contracts were speculatively purchased creating a shortage of physical oil and as a result prices of crude were temporarily increased. During this time the price performance of US oil stocks failed to match the price performance of oil as a physical commodity.

Various academic papers have analysed spread trading opportunities between futures markets and ETFs (Exchange Traded Funds) however in practice this presents a number of issues. For instance, futures contracts require market participants to 'roll' expiring contracts, non-synchronous closing prices may also occur when trading different asset classes on different exchanges, and futures contracts are also more expensive to trade than ETF shares. In order to avoid such implications only ETF data is analysed here. In particular, the GLD (SPDR Gold Trust), the SLV (iShares Silver Trust), the USO (United States Oil Fund), the GDX (Market Vectors ETF Trust) and the OIH (Market Vectors Oil Services) ETFs are all included in this study. The GLD and SLV ETFs are based on the physical commodity spot price movements of gold and silver

respectively whereas the USO contract is based on futures contracts that track the future spot price of oil.

The composition of each ETF contract ultimately determines its price performance. For instance the GDX contract is influenced by the performance of large³⁷ cap mining companies. The GDX ETF is weighted according to market capitalisation with top holdings including the Barrick Gold Corporation, Goldcorp, Inc., Newmont Mining Corporation, AngloGold Ashanti Limited ADR, and the Buenaventura Mining Company Inc. American Depository Receipt (ADR). The USO contract however is weighted to reflect the price performance of light, sweet and crude oil. The value of this ETF is derived from numerous underlying futures contracts which occasionally suffer from contango and it is during these times that the price of the USO contract fails to match the spot price performance of Crude Oil as well as the OIH ETF. The OIH ETF is another market cap weighted ETF based on a basket of large cap publicly traded oil companies in the United States. Holdings include Schlumberger NV, Halliburton Company, National Oilwell Varco, Inc., Baker Hughes, and Transocean Ltd to mention a few. Contango is caused by the cost and timing associated with ‘rolling’ underlying futures. In order to provide a continuous time series this process of ‘rolling’ is necessary as older contracts expire and new futures contracts begin. In general, the nearer the expiration date the more expensive the process of rolling futures contracts becomes as volumes from old contracts decline and shift to newer futures contracts. Resulting price differences in the USO contract creates opportunities for arbitrage when trading a USO OIH spread. Unlike the USO EFT, both GLD and SLV contracts do not suffer from contango as they are not futures based.

³⁷ Market Capitalisation greater than \$100 million with an average trading volume of 50,000 shares during a six month trading period.

Figures 19 and 21 both reveal that physical commodity ETFs GLD and SLV outperform the GDX ETF from early 2008 until the end of the in-sample period 20/05/2010³⁸. In part, this is due to the fact that GDX is exposed to both market and company specific risks and when market turmoil is experienced then large cap miners generally fail to match the price performance of physical commodities such as Gold and Silver. This is also apparent in the USO vs. OIH spread as shown in figure 20. USO outperforms OIH during the summer of 2008 when the price of oil surged.

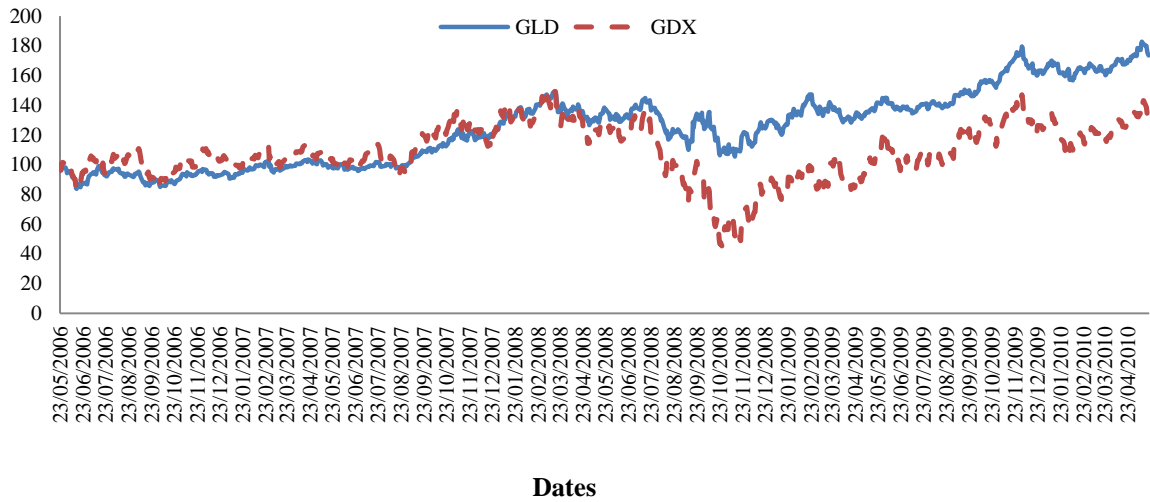
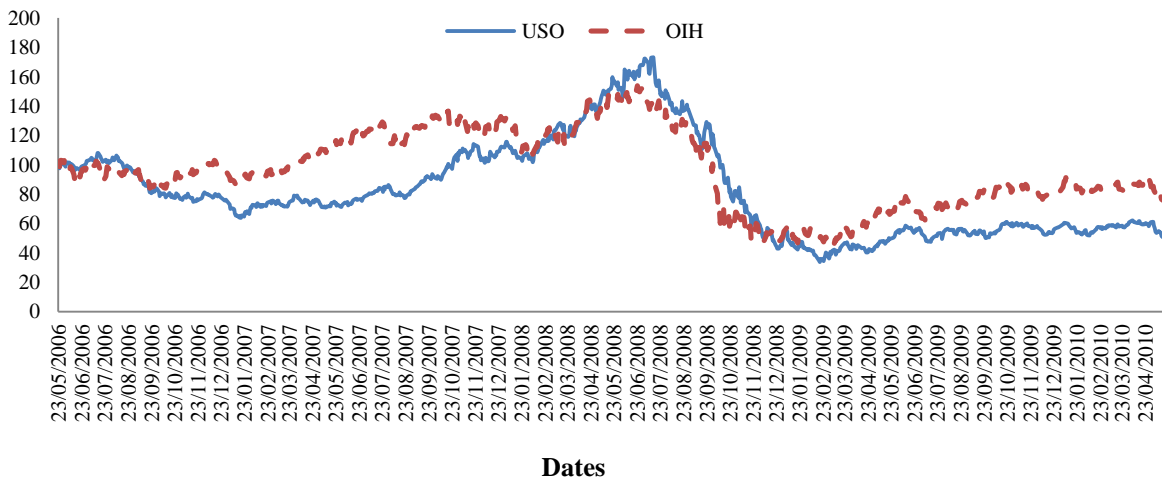


Figure 19. In sample GLD vs. GDX relative price performance (23/05/2006 = 100)



³⁸ Each time series has been rebased to 100 commencing 23/05/2006 to measure relative price performance.

Figure 20. In sample USO vs. OIH relative price performance (23/05/2006 = 100)

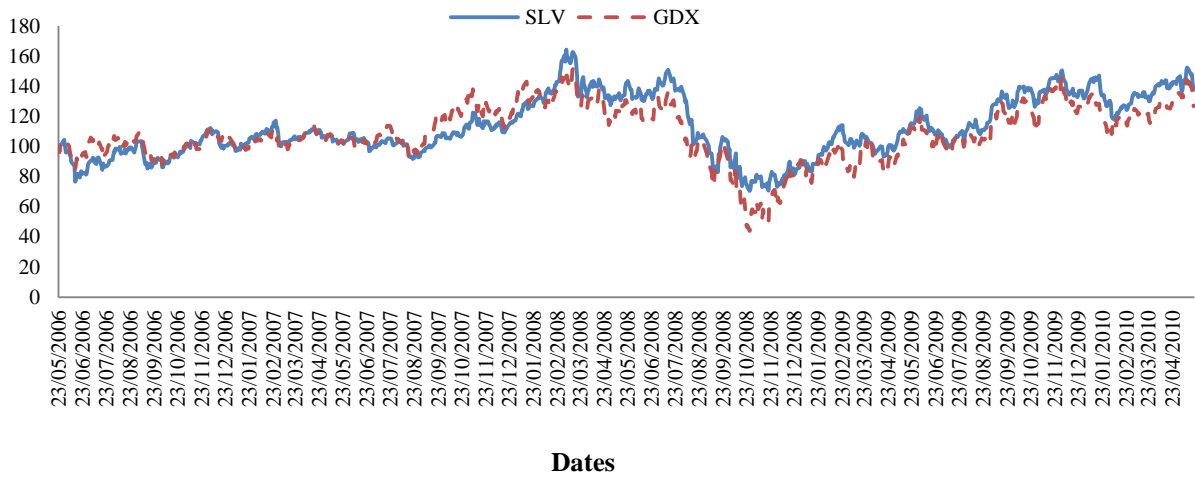


Figure 21. In sample SLV vs. GDX relative price performance (23/05/2006 = 100)

Each ETF has been rebased using equation 35 with 23/05/2006 as the base date.

$$\text{Rebased Time Series} = (1 + \text{Return}_t / 100) * \text{Return}_{t-1} \tag{35}$$

The remaining structure of this paper is presented as follows. Section 2 provides a brief literature review identifying various applications for both non-linear models and mean reversion trading strategies. Section 3 analyses the statistical behaviour of each spread. Section 4 presents the methodologies and estimation parameters. Section 5 describes trading rules and strategies as well as evaluating empirical results. The final section offers concluding remarks and observations and limitations.

2.0 Literature Review

Spread trading, referred to by some as pairs trading, was first documented in earlier literature by Gatev *et al.* (1999) and Gatev (2000). More recent studies such as Vidyamurthy (2004) and Elliott *et al.* (2005) also analyse relationships between pairs of assets. Collectively, these

papers successfully model mean reversion and develop profitable trading strategies. In particular, Elliot *et al.* (2005) use mean reversion as a market timing device to generate trading signals from which positions are held until the spread of each pair reverts back to a mean in order to generate profit. In support of this, Nicholas (2004) also finds evidence of profitability when trading a pair of related securities / assets which experience momentary deviations from an equilibrium. Vidyamurthy (2004) examine the long term relationships of pairs by applying cointegration to model and trade financial time series. Ultimately, as observed by Jin *et al.* (2008), pairs / spread trading strategies are devised in order to exploit markets and assets when they are at disequilibrium.

Triantafyllopoulos and Montana (2009) analyse ETF commodity spreads and in particular they focus on mean reversion. They evaluate the gold miner spread from May 23, 2006 to August 06, 2008. As discussed by Chan (2009), a stationary series is an ideal candidate for a mean reversion strategy. With this as an objective Triantafyllopoulos and Montana (2009) test for stationarity and their findings reveal that the spread is in fact stationary from July 19th, 2006 to December 17th, 2007. During this time period traders would have been able to profit using mean reversion as a trading strategy. Furthermore, this is confirmed statistically as a p-value of 0.01 is found using the Augmented Dickey Fuller (1979) test. In comparison, Jin *et al.* (2008) also test the Gold Miner Spread for stationarity, producing p-values of less than 0.01 using both the ADF and PP (Phillips and Perron, 1988) tests over an extended period from June 14, 2006 until July 31st, 2008.

Herlemont (2003) investigates the trading of market and sector neural spreads based on mean reverting hedge ratios rather than using mean reversion of a spread time series. In his analysis the identification of mean reversion in spread ratios is conducted using the Dickey Fuller (1979) test and once stationarity is established then pairs are selected for trading. Trading signals are triggered when the ratio of two equities' share prices drifts 2 standard deviations away from the 130

day moving average. Positions are then held until the spread fully reverts back to the 130 day mean. Herlemont (2003) finds that his results are in line with other existing literature and that mean reverting trading filters offer profitability when the ratio of a spread is stationary.

Kim *et al.* (1999) focus more on non-linear forecasting by applying neural networks to the task of predicting the spread between the spot Canadian Dollar / US Dollar foreign exchange rate and a short term interest rate spread. In this analysis, neural network models are found to have an ability to capture 'corrective mean reversion' during times when the Canadian dollar is either under or over valued in the market. Furthermore, Kim *et al.* (1999) find that neural networks provide between 2% - 5% more in correct directional change (CDC) when compared to an ARMA (5,5) model.

Dunis *et al.* (2005) model the Gasoline Crack spread using both linear and non-linear methodologies. These models are then traded under the supervision of threshold (% change in the spread), asymmetric (moving above or below the fair value of the spread) and correlation (change in correlation of underlying legs) trading filters. Results confirm that the spread is best traded using a threshold filter as the nature of the spread is such that, on average, movements below the fair value tend to be larger than those above the fair value. These significant moves are more adequately captured by the threshold filter and as a result profitability is most improved using the threshold filter than any of the other trading filters.

Dunis *et al.* (2012) develop a number of models in order to trade the spread between physical gold and US gold equity. In this investigation the spread was found to be mean reverting at times with non-linear models proving to be the most accurate as they produced the highest correct directional change statistics. In particular, Dunis *et al.* (2012) finds that a Genetic Programming Algorithm outperforms next day forecasted returns produced by both a cointegration model and a (12,12) Autoregressive Moving Average Model (ARMA).

3.0 Descriptive Statistics and Related Financial Data

In search of suitable candidates for the purpose of spread trading it was decided to select pairs of ETFs that are fundamentally and economically related. Spreads are created using time varying hedge ratios derived from the analysis of historical data. In particular, each spread makes economic sense as the values of commodity based equities are generally governed by the directional change in the value of underlying physical commodities that they produce. Table 23 provides a summary of variables selected for this investigation.

ETF	TICKER	EXCHANGE	DATE INCEPTION	OF	Daily Volumes Traded ³⁹	PURPOSE
United States Oil Fund	USO	NYSE ARCA	10/04/2006		9,570,902	To reflect changes in the spot price of Light, Sweet, Crude Oil.
Market Vectors Oil Services	OIH	NYSE ARCA	09/02/2005		27,485,934	To reflect the general performance of US Oil Equity.
SPDR Gold Trust	GLD	NYSE ARCA	18/11/2004		11,618,322	To reflect changes in the spot price of Gold Bullion.
Market Vectors ETF Trust (US Mining Equity)	GDX	NYSE ARCA	23/05/2006		5,440,281	To reflect the general performance of US Mining Equity.
iShares Silver Trust	SLV	NYSE ARCA	21/04/2006		7,373,173	To reflect changes in the spot price of Silver

Table 23. ETF contract specifications

Each spread was constructed using a time varying hedge ratio as mathematically depicted in equations 36 and 37. This essentially provides an estimate of ‘sensitivity’ on the price of commodity equity prices to changes in the prices of physical commodities. From this ratio, the optimal number of shares of the second ETF versus one share of the first ETF can be decided. Fixed betas are also included in figures 22-24 in order to draw comparisons.

Time-varying betas are derived from a non-conditional variance and covariance of the in-sample period (23 May 2006 - 20 May 2010) and then estimated based on the time varying ratio of covariance and variance using the RiskMetrics daily factor of 0.94. While it is accepted that the

³⁹ Average daily volumes traded during the in-sample period 23/05/2006 – 20/05/2010.

optimal lambda varies depending on asset class an overall optimal daily factor of 0.94 is used by JP Morgan (1997) across all asset classes⁴⁰.

$$\text{Time Varying Beta} = \rho_t = \frac{\text{COV}(r_A, r_B)_t}{\sigma_{A^t} \sigma_{B^t}} \quad (36)$$

$$\text{COV}(r_A, r_B)_t = \lambda \text{COV}(r_A, r_B)_{t-1} + (1 - \lambda) r_A r_B \quad (37)$$

The ‘fixed’ beta is otherwise known as a non-conditional hedge ratio and is calculated simply by looking a regression fit of the covariance between each leg over the variance of each leg. Spreads using both time varying and fixed betas are computed in equations 38, 39 and 40. Figures 22 - 24 also provide graphical representations of each spread as they plot spreads calculated using both time varying betas as well as spreads constructed using fixed betas.

Gold vs.US Miner’s equity

$$\text{Spread 1} = \text{GLD} - (\beta * \text{GDX}) \quad (38)$$

where:

$$\text{Fixed } \beta = 2.05$$

$$\text{Time Varying } \beta = 2.36$$

⁴⁰ The time-varying beta varies on a daily basis however for out-of-sample trading the final time-varying beta estimated on 20/05/2010 was used to construct each of the spreads.

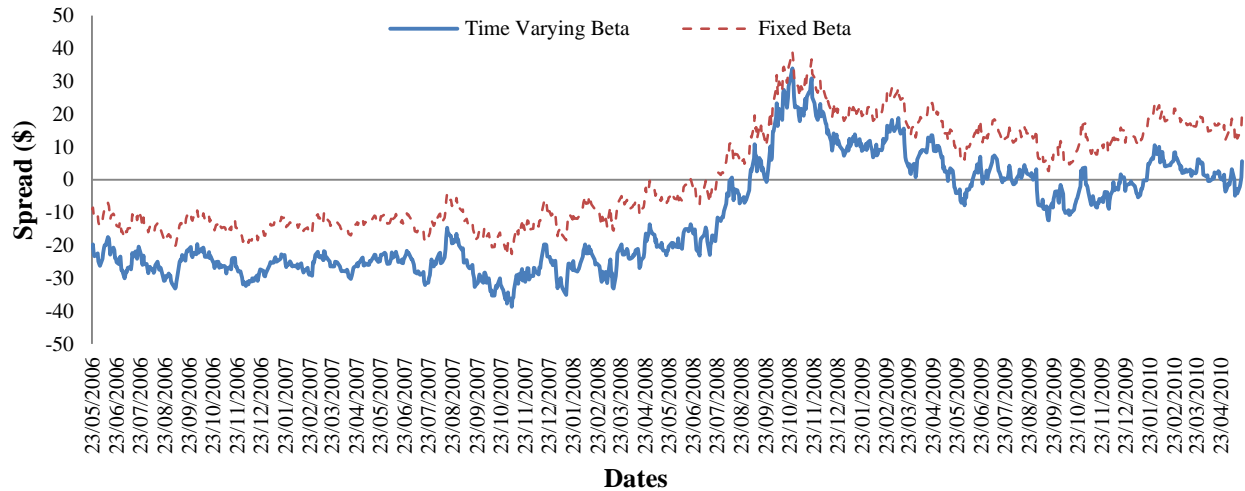


Figure 22. GLD vs. GDX spread (in-sample)

As the exact number of shares of GLD and GDX can be calculated by a regression fit of the two component time series the time varying beta is used to determine this. For instance, in this case a trader would buy 100 GLD contracts while selling 236 contracts of GDX. Alternatively, a trader could sell 100 GLD contracts and buy 236 GDX contracts. The spread is ‘long’ when GLD is bought and GDX is sold. Alternatively, the spread is ‘shorted’ when the inverse is true (sell GLD and buy GDX).

Oil vs. US Oil equity

$$\text{Spread 1} = \text{USO} - (\beta * \text{OIH}) \tag{39}$$

where:

$$\text{Fixed } \beta = 0.79$$

$$\text{Time Varying } \beta = 1.06$$

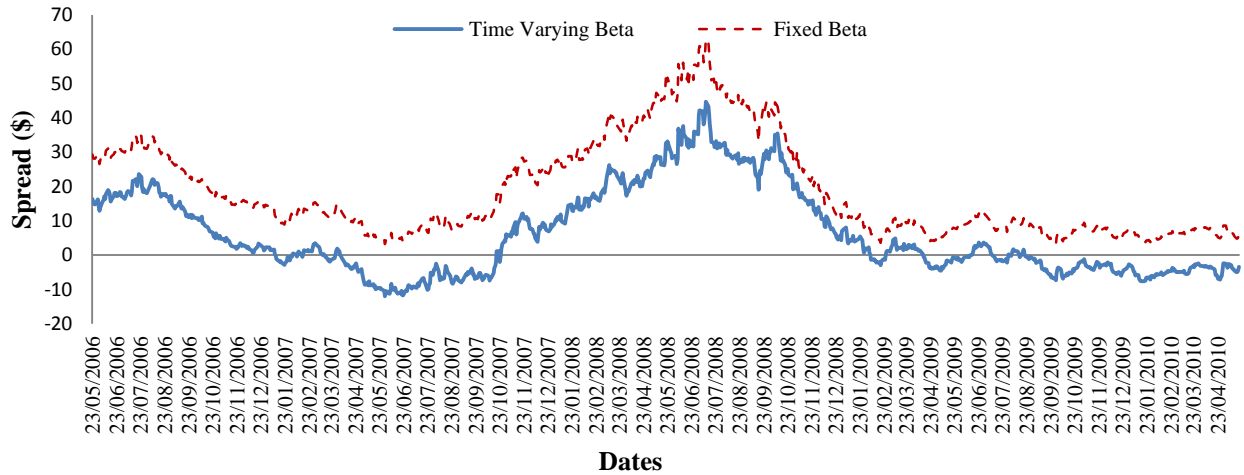


Figure 23. USO vs. OIH spread (in-sample)

Using the time varying beta as an indicator a trader would buy 100 USO contracts while selling 106 contracts of OIH. Similar to the gold miner spread the reverse combination also applies as the ratio of the two remains unchanged.

Silver vs. US Miner's equity

$$\text{Spread 3} = \text{SLV} - (\beta * \text{GDX})$$

(40)

where:

$$\text{Fixed } \beta = 0.35$$

$$\text{Time Varying } \beta = 0.37$$

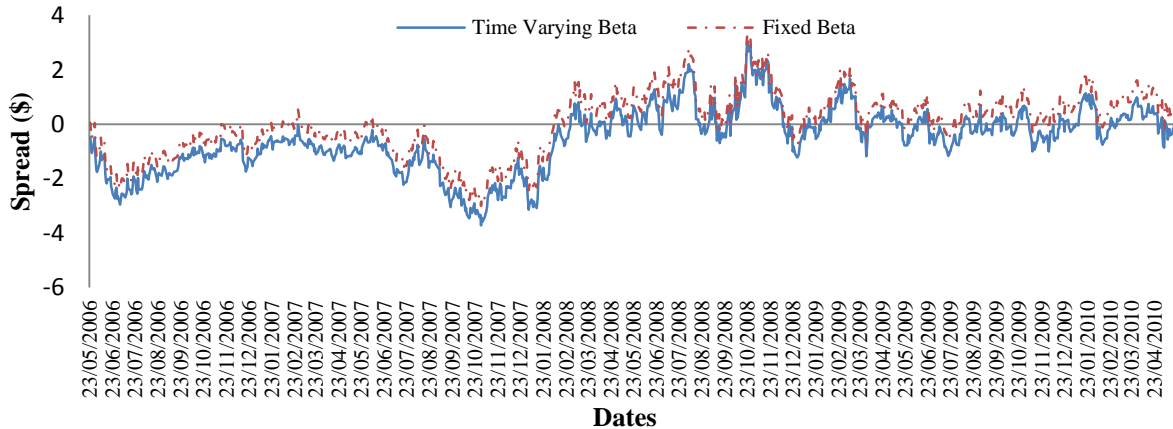


Figure 24. SLV vs. GDX spread (in-sample)

Using the time varying beta as an indicator, 100 SLV contracts are bought while 37 contracts of GDX are sold. The opposite is also true as is the case with the other two spreads. Once each spread is calculated and optimal hedge ratios are obtained then observations are drawn from the Augmented Dickey-Fuller (ADF) methodology to test for stationarity. H0: If the p-value is > 0.05 the time series is not stationary and a unit root is present. Accept H0. H1: If the p-value is < 0.05 the H0 can be rejected as the time series is stationary. Hence, 5 times out of 100 it can be assumed that the spread will be significantly different from the mean and present itself as an outlier or tail statistic when considering the distribution of the spread around the 'norm' or mean. For the sake of brevity only a summary of the testing results are included in this paper. Full extracts from the testing may be requested directly from the corresponding author.

Figures 25 and 26 plot the Gold Miner spread during times of stationarity as observed throughout the in sample period. Figure 27 however, plots the entire in sample period which is not found to be stationary as a result of the financial crisis.

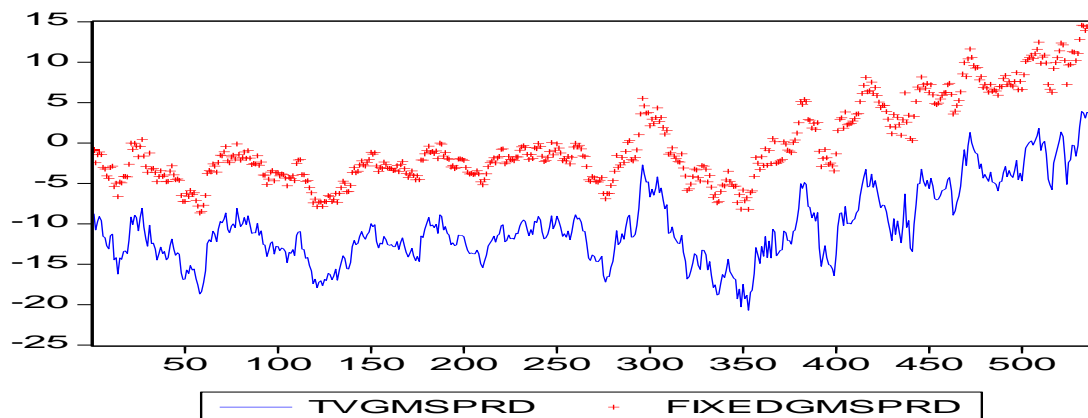


Figure 25. GM spread (14/06/06-31/07/08)

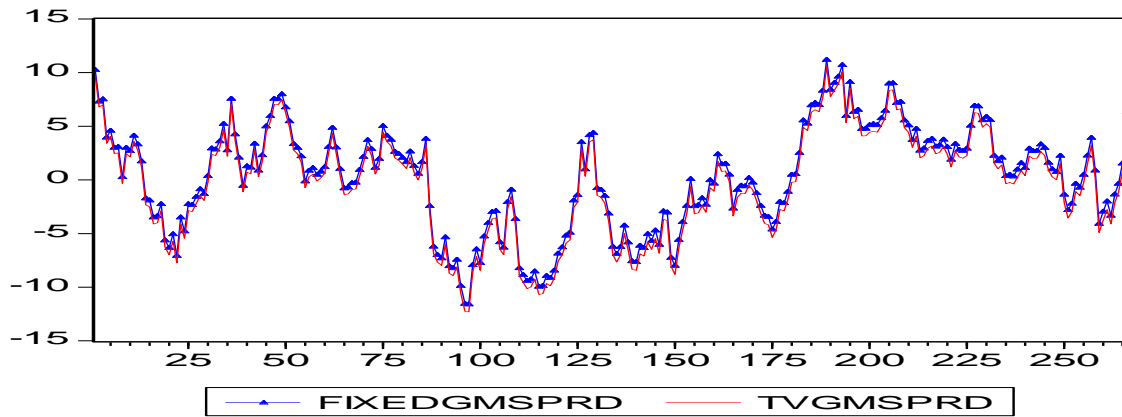


Figure 26. GM spread (01/05/09-20/05/10)

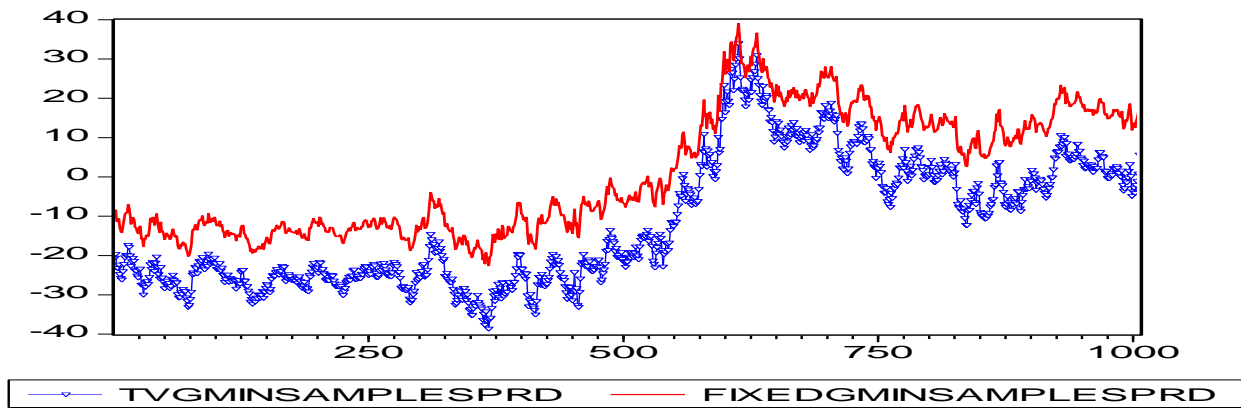


Figure 27. GM spread in-sample (23/05/06-20/05/10)

Spread	Period	P-values	Hypothesis	Betas
Fixed GM	14/06/06 – 31/07/08	0.0285	Reject H0	1.77
Time Varying GM	14/06/06 – 31/07/08	0.0151	Reject H0	2.01
Fixed GM	01/05/09 – 20/05/10	0.0407	Reject H0	2.34
Time Varying GM	01/05/09 – 20/05/10	0.0411	Reject H0	2.36
Fixed GM (In-Sample)	23/05/06 – 20/05/10	0.1978	Accept H0	2.05
Time Varying GM (In-Sample)	23/05/06 – 20/05/10	0.1829	Accept H0	2.36

Table 24. Augmented dickey-fuller test statistics

By observation of table 24, statistically the spread is stationary for the period from 14/06/06 to 31/07/08 as the p-value is 0.0151. Stationarity disappears for almost a year as a result of the financial crisis and then re-emerges from 01/05/2009 to the end of the data set.

Following ADF testing of the full in-sample period results in table 24 indicate that the spread is non-stationary with a p-value of 0.1829 using a time varying ratio of 2.36. In part, the reason for this erratic behaviour is that gold miners were directly affected by the 2008 market turmoil and many investors flocked to gold as a safe haven. As a result, the share prices of gold miners decreased while the price of gold increased significantly. Therefore, share prices lagged the spot / futures price performance of gold as a physical commodity.

A similar pattern emerges from the analysis of the oil spread over the same time period as it also displays non-stationary characteristics. This is as a result of surging oil prices during the summer of 2008 and various instances of contango. At these points, the price performance of oil refiners' share prices underperformed crude oil. Figures 28 and 29 plot periods of stationarity and figure 30 displays a non-stationary time series over the entire in sample dataset.

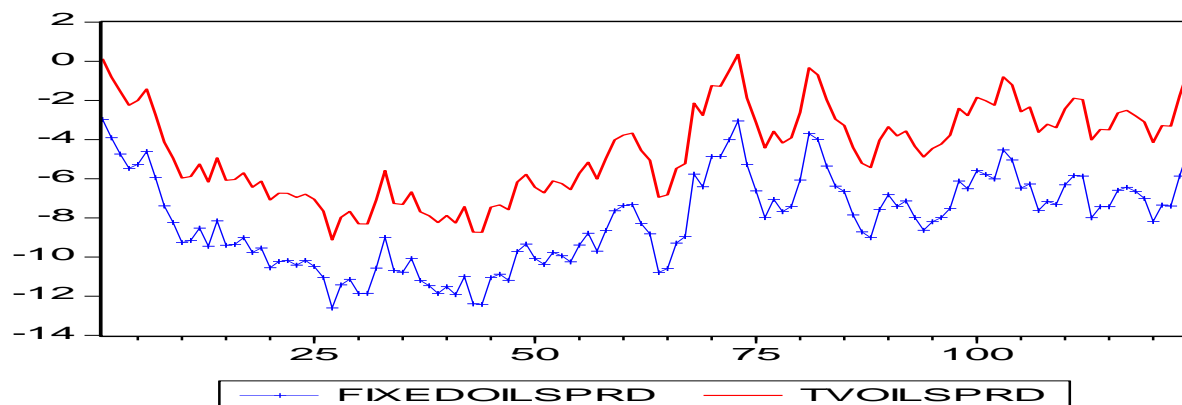


Figure 28. Oil spread (23/04/07-16/10/07)

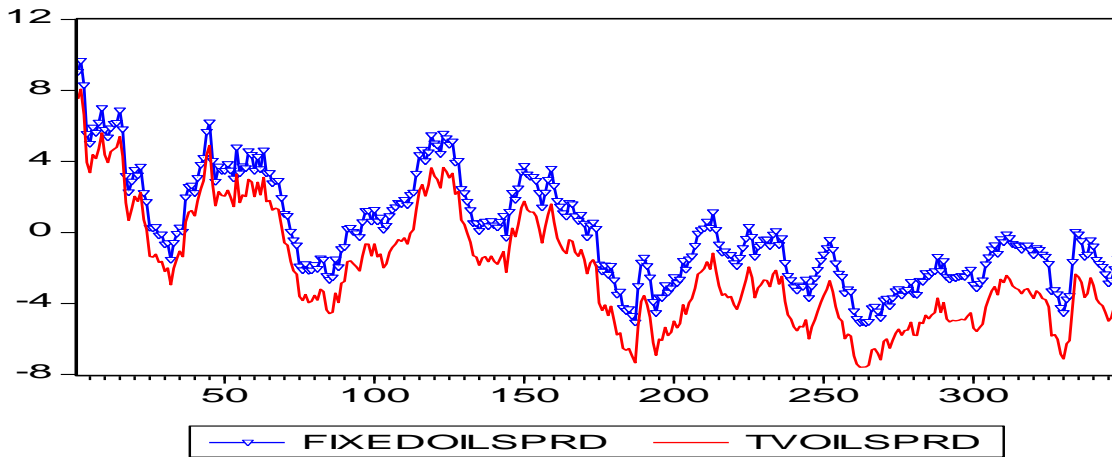


Figure 29. Oil spread (02/01/09-20/05/10)

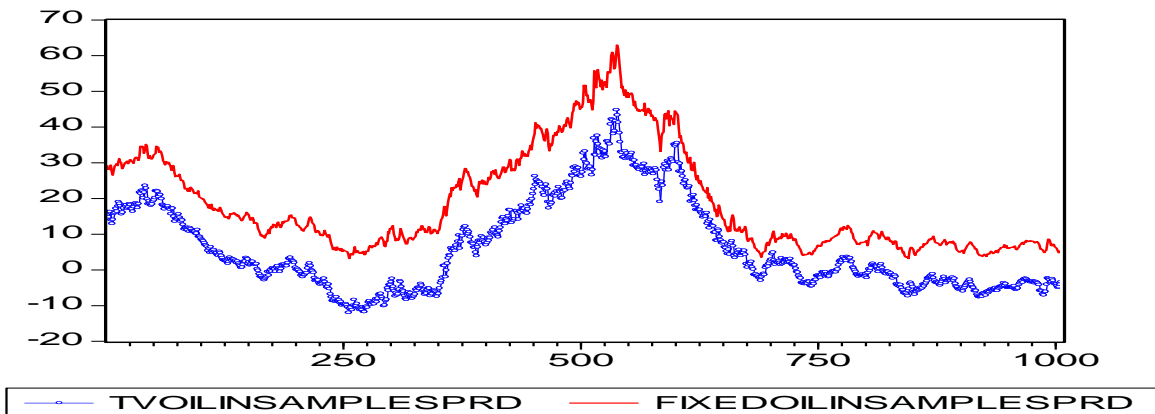


Figure 30. Oil spread in-sample (23/05/06-20/05/10)

Spread	Period	P-value	Hypothesis	Beta
Fixed Oil	23/04/07 – 16/10/07	0.0273	Reject H0	1.07
Time Varying Oil	23/04/07 – 16/10/07	0.0274	Reject H0	1.01
Fixed Oil	02/01/09 – 20/05/10	0.0185	Reject H0	1.00
Time Varying Oil	02/01/09 – 20/05/10	0.0443	Reject H0	1.06
Fixed Oil (In-Sample)	23/05/06 – 20/05/10	0.8973	Accept H0	0.79
Time Varying Oil (In-Sample)	23/05/06 – 20/05/10	0.8651	Accept H0	1.06

Table 25. Augmented dickey-fuller test statistics

By observation of table 25 statistics prove that the oil refiner spread is periodically stationary from 23/04/2007 – 16/10/2007 using a fixed beta of 1.07 and a time varying beta of 1.01.

Stationarity is then lost between USO and OIH in 2008 however the spread becomes stationary again in 2009 until the end of the in-sample period on 20/05/2010. For this period a fixed beta of 1.00 and a time varying beta of 1.06 are calculated.

The Silver Miner Spread is found to be stationary for the entire in-sample period as seen in table 26 and plotted in figure 31. A fixed beta of 0.35 and a time varying beta of 0.37 were estimated for this period.

Spread	Period	P-value	Hypothesis	Beta
Fixed Silver (In Sample)	23/05/06 – 20/05/10	0.0245	Reject H0	0.35
Time Varying Silver (In Sample)	23/05/06 – 20/05/10	0.0257	Reject H0	0.37

Table 26. Augmented dickey-fuller test statistics

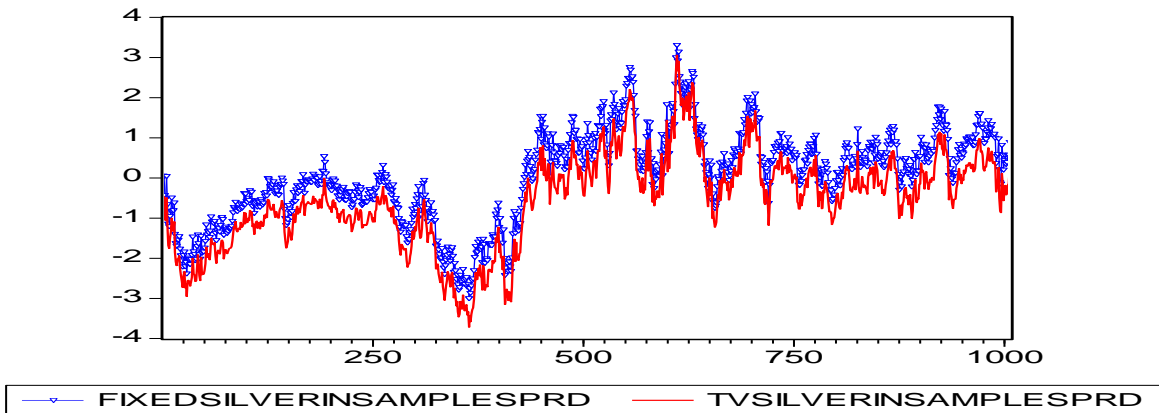


Figure 31. Silver spread in-sample spread (23/05/06 – 20/05/10)

Chan (2009) suggests that if a price series (of a stock, a pair of stocks, or even a portfolio of stocks) is stationary, then a mean reversion strategy is guaranteed to be profitable, as long as the stationarity persists into the future.

In summary, stationarity is periodically lost in both the gold and oil spreads as the price of oil surged in the summer of 2008 while both Gold and Silver also experienced the beginning of commodity bull runs during the height of the financial crisis. All of which created structural breaks in the data and as a result a ‘break’ in the mean and trend was experienced. However, there is

strong evidence of mean reversion throughout all in-sample periods. As discussed by Perron (1989), it is well known that in the case of unknown regime changes, unit root tests are biased towards the null hypothesis of non-stationarity. Under such conditions, each of the unit root tests display low powers of rejecting the null hypothesis of non-stationarity.

Modelling ETF Spreads

In order to construct viable trading models for each spread, return series are generated using equation 41:

$$\Delta S_t = \left(P_{LEG1(t)} / P_{LEG1(t-1)} \right) - \left(P_{LEG2(t)} / P_{LEG2(t-1)} \right) \tag{41}$$

where:

ΔS_t = Percentage returns of spread at time t.

$P_{LEG1(t)}$ = the price of the Commodity based ETF at time t (\$)

$P_{LEG1(t-1)}$ = the price of Commodity based ETF at time t-1 (\$)

$P_{LEG2(t)}$ = the price of Equity based ETF at time t (\$)

$P_{LEG2(t-1)}$ = the price of Equity based ETF at time t-1 (\$).

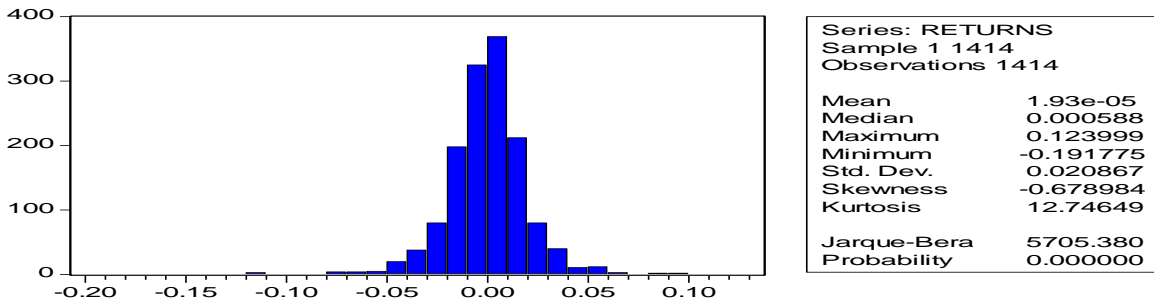


Figure 32. Gold miner - full sample distribution of spread returns

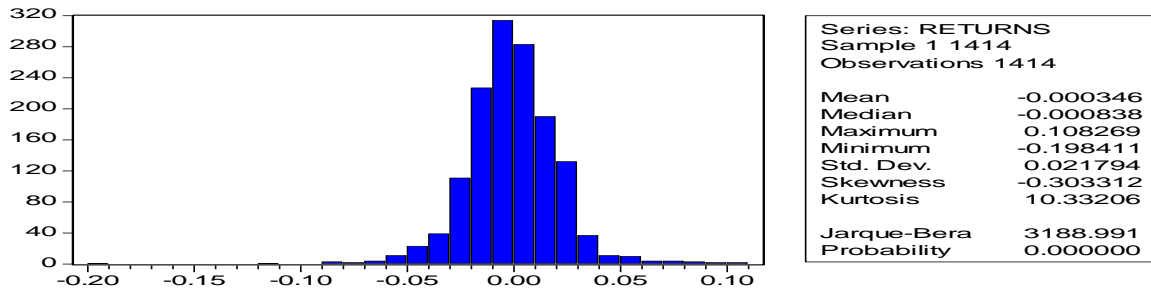


Figure 33. Oil - full sample distribution of spread returns

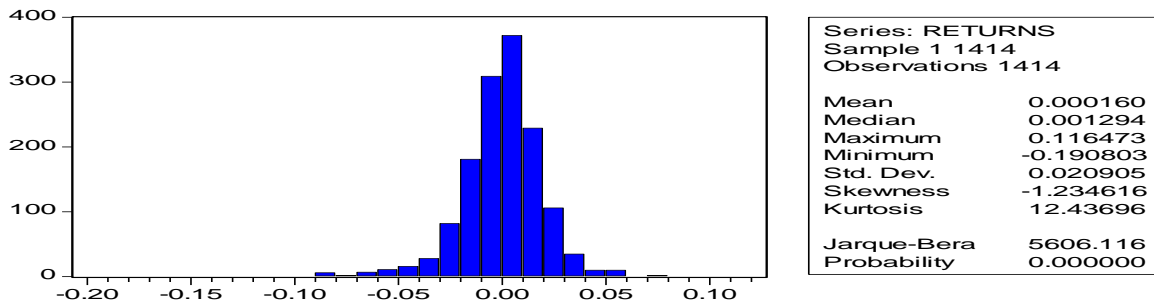


Figure 34. Silver - full sample distribution of spread returns

Figures 32, 33 and 34 reveal that all of the return series are leptokurtic distributions with each displaying positively high kurtosis. This however is quiet common when observing normal distributions of return series as data points tend to be highly concentrated around the mean. Furthermore, all of the spreads are confirmed to be non-normal (confirmed at a 99% confidence level by the Jarque-Bera test statistic).

3.1 The Commodity Spreads and Related Financial Data

During the construction of each dataset the closing times for each ETF were taken into consideration in order to avoid the issue of non-synchronous time series modelling. For this reason, only ETFs traded on the NYSE are used with closing prices being recorded at 16:00 EST (Eastern Standard Time). AMEX ETFs were not used as their closing prices are priced at 16:15 EST.

Furthermore, all closing prices for ETFs based on a basket of underlying equities are adjusted for dividends and stock splits to allow for a more accurate representation of trading when back testing the mean reversion strategy. The dataset ends on 30/12/2011 as shown in table 27 and it should also be noted that ETF contracts are highly liquid.

Period	Trading Days	As a % of Total Sample	Beginning	End
<i>Total dataset</i>	1,414	100%	23/05/2006	20/12/2011
<i>Training dataset (in-sample)</i>	714	50.49%	23/05/2006	24/03/2009
<i>Test dataset (in-sample)</i>	292	20.65%	25/03/2009	20/05/2010
<i>Validation set (out-of-sample)</i>	408	28.86%	21/05/2010	30/12/2011

Table 27. Full dataset

Following the segregation of data into training, test and validation sets the below inputs detailed in table 28 were used to estimate the MLP neural networks. Autoregressive returns series from 1 to 21 lags were used as initial inputs and then 3 moving average time series were added in order to provide more informational content during the training process. These parameters were found to generate suitable annualised returns during each of the in-sample periods.

Inputs	Explanatory Variables	Lays (Days)
1-21	Autoregressive Spread Returns	1-21
22	21 Day Moving Average of Spread Returns	21
23	50 Day Moving Average of Spread Returns	50
24	100 Day Moving Average of Spread Returns	100

Table 28. Explanatory variables for the MLP neural network

4.0 Methodology

Please note that Chapter 2 introduces and explains each of the proposed models in more detail. However, the subsequent sections provide details regarding estimations of coefficients for the selected ARMA and Cointegration models.

4.1 ARMA

Using a correlogram as a guide in the training and the test sub-periods the below restricted ARMA models were selected to trade each spread. All coefficients were found to be significant at the 99% confidence interval. Therefore, the null hypothesis that all coefficients (except the constant) are not significantly different from zero is rejected at the 99% confidence interval (see, tables 65 -67 in appendix A.5).

The models are also retained for out-of-sample trading simulations and are mathematically represented in equations 42, 43, and 44.

ARMA (9,9)

$$GM\ Spread\ Y_t = -1.81 * 10^{-5} - 0.552Y_{t-1} - 0.271Y_{t-4} + 0.204Y_{t-6} + 0.391Y_{t-9} - 0.599\varepsilon_{t-1} - 0.361\varepsilon_{t-4} + 0.174\varepsilon_{t-6} + 0.463\varepsilon_{t-9} \quad (42)$$

ARMA (8,8)

$$Oil\ Spread\ Y_t = -5.64 * 10^{-4} + 0.706Y_{t-1} + 0.609Y_{t-3} - 0.761Y_{t-5} + 0.332Y_{t-8} + 0.763\varepsilon_{t-1} + 0.591\varepsilon_{t-3} - 0.812\varepsilon_{t-5} + 0.359\varepsilon_{t-8} \quad (43)$$

ARMA (7,7)

$$Silver\ Spread\ Y_t = -5.74 * 10^{-6} - 0.946Y_{t-1} + 0.647Y_{t-6} + 0.609Y_{t-7} - 0.873\varepsilon_{t-1} + 0.618\varepsilon_{t-6} + 0.525\varepsilon_{t-7} \quad (44)$$

4.2 Cointegration Models

Cointegration between two variables occurs when a linear combination of the variables results in a stationary time series (Engle and Granger, 1987). Taking this into consideration, the in-sample linear combinations of each spread are analysed.

Cointegration models are devised in a similar manner to that used by Dunis *et al.* (2006) with the cointegrating vector being estimated using the Johansen (1988) cointegration test. Speed adjustment coefficients were taken from the VECM test statistics⁴¹ and can be seen in equations 45, 46, and 47. All variables are found to be I(1) and as a result are deemed stationary in 1st difference

⁴¹ For conciseness VECM estimates are not included in this paper. These can however be supplied on request from the corresponding author.

or integrated of order one. In this case, each of the null hypotheses that a time series has a unit root is rejected. By observation, the gold miner and oil refiner spreads periodically experience ‘structural breaks’ as a result of ‘black swan’ events such as the 2008 surge in oil prices and the onset of the ‘Credit Crisis’ experienced in September / October 2008. Irrespective of this, over the in sample period each trading model successfully captures long term relationships from 23/05/2006 to 20/05/2010.

GM Coefficients

$$\text{Spread Price Series}_{t+1} = GDX_t * -0.04953 + GLD_t * -0.036578 / (-0.04953 - 0.036578) \quad (45)$$

OIL Coefficients

$$\text{Spread Price Series}_{t+1} = USO_t * 0.919344 + OIH_t * -0.017471 / (0.919344 - 0.017471) \quad (46)$$

SILVER Coefficients

$$\text{Spread Price Series}_{t+1} = SLV_t * -0.008286 + GDX_t * 0.013923 / (-0.008286 + 0.013923) \quad (47)$$

Return series for each of the spreads are then calculated from the resulting price series as shown in equation 48.

$$R_t = (P_t / P_{t-1}) - 1 \quad (48)$$

Results obtained from the Johansen (1988) test for the gold miner spread reveal that during the in-sample period of 23/05/2006 –20/05/2010 GLD cointegrates with GDX with an 89% probability.

Date: 04/08/12 Time: 15:24
Sample (adjusted): 3 1006
Included observations: 1004 after adjustments
Trend assumption: Linear deterministic trend (restricted)
Series: LGLDINSAM LGDXINSAM
Lags interval (in first differences): 1 to 1

Unrestricted Cointegration Rank Test (Trace)

Hypothesized		Trace	0.05	
No. of CE(s)	Eigenvalue	Statistic	Critical Value	Prob.**
None	0.014038	22.81042	25.87211	0.1148
At most 1	0.008545	8.616110	12.51798	0.2054

Trace test indicates no cointegration at the 0.05 level

* denotes rejection of the hypothesis at the 0.05 level

**MacKinnon-Haug-Michelis (1999) p-values

Table 29. Johansen (1988) test results GLD/GDX

In-sample (23/05/2006 – 20/05/2010) results obtained from the Johansen (1988) test for the oil spread show a stronger case for cointegration as USO cointegrates with OIH with a 99% probability.

Date: 04/08/12 Time: 18:34

Sample (adjusted): 3 1006

Included observations: 1004 after adjustments

Trend assumption: No deterministic trend (restricted constant)

Series: LUSOINSAM LOIHINSAM

Lags interval (in first differences): 1 to 1

Unrestricted Cointegration Rank Test (Trace)

Hypothesized		Trace	0.05	
No. of CE(s)	Eigenvalue	Statistic	Critical Value	Prob.**
None *	0.025621	30.97671	20.26184	0.0011
At most 1	0.004886	4.917656	9.164546	0.2926

Trace test indicates 1 cointegrating eqn(s) at the 0.05 level

* denotes rejection of the hypothesis at the 0.05 level

**MacKinnon-Haug-Michelis (1999) p-values

Table 30. Johansen (1988) test results USO/OIH

Results from the Johansen (1988) test also reveal that during the in-sample period of 23/05/06 – 20/05/10 SLV cointegrates with GDX at a 96% probability.

Date: 04/08/12 Time: 19:27

Sample (adjusted): 3 1006
 Included observations: 1004 after adjustments
 Trend assumption: No deterministic trend (restricted constant)
 Series: LSLVINSAM LGDXINSAM
 Lags interval (in first differences): 1 to 1

Unrestricted Cointegration Rank Test (Trace)

Hypothesized		Trace	0.05	
No. of CE(s)	Eigenvalue	Statistic	Critical Value	Prob.**
None *	0.015075	20.96842	20.26184	0.0399
At most 1	0.005679	5.717648	9.164546	0.2137

Trace test indicates 1 cointegratingeqn(s) at the 0.05 level

* denotes rejection of the hypothesis at the 0.05 level

**MacKinnon-Haug-Michelis (1999) p-values

Table 31. Johansen (1988) test results SLV/GDX

For the most part, Johansen (1988) testing of each spread estimates high probabilities of being cointegrated during the in-sample dataset and as a result it can be assumed that each spread does not display random geometric walks as may be the case when modelling other spreads. Hence, it is determined that Cointegration models provide suitable benchmarks for the trading each of the spreads.

5.0 Empirical Results

5.1 Trading Rules and Strategies

Models are tasked with forecasting next day returns for each of the three selected spreads. Trading signals that are used for the unfiltered models are derived directly from forecasts of each of the models with no additional rules to be considered. In this case, trading signals are produced by the directional forecast of each model. For example, if a model predicts a downward move in the spread then a short position is generated. Alternatively, when the model predicts an upward move a

long position is executed. For consecutive negative or positive trading days positions are held until forecasts indicate otherwise.

A mean reversion strategy is employed as a filter for the ‘filtered’ models. Optimisation of this filter is carried out during in-sample trading and with parameters being selected based on the maximisation of annualised returns. Parameters for the filtered models can be seen in table 32.

The trading filter is such that a trade signal is produced once the standard deviation of the spread is larger than ‘x’ from a moving average ‘z’ and the position is held until the spread either reverts back to within ‘y’ standard deviations of ‘z’ or the position is open for 20 consecutive days. In most cases that spread reverts back to within ‘y’ standard deviations of ‘z’ before hitting the 20 day maximum holding threshold. This maximum holding period is a precautionary condition in order to limit the amount of time one is exposed to fluctuations in the spread.

The ‘x’, ‘y’ and ‘z’ parameters vary from model to model depending on in-sample optimisation. However, the ‘x’ parameter is generally found to be around 2 standard deviations while the ‘y’ parameter is approximately 0.5 standard deviations (stds). In some cases, both the optimal ‘x’ and ‘y’ estimations are found to be the same.

Rules	Estimation Parameters
<i>Volatility Filter (Diversion)</i>	x stds.
<i>Volatility Filter (Reversion)</i>	y stds.
<i>Maximum Holding Period</i>	20 days
<i>Moving Average (Spread)</i>	z days

Table 32. Trading parameters

While the maximum drawdowns and maximum drawdown durations are important to monitor, rules for these are set on a portfolio level rather than for each individual model. This is discussed in more detail in section 5.2. Regardless of half-life calculations, the most profitable maximum holding period was universally found to be 20 days for each model.

The half-life formula provides an estimation of the time it takes for the spread to revert to half its initial deviation from the mean of the spread. Hence, half the time one should expect to hold a spread before realising a profit. For full reversion each half life is to be multiplied by two. Using equation 49 each of the half-life calculations are provided in table33.

$$HL = \frac{\ln(2)}{\theta} \tag{49}$$

where:

θ = time varying beta which is calculated for each of the spreads in equation 49. This is essentially the daily change in the spread.

Spread	Half Life (days)
Gold Miner Spread	13
Oil Spread	29
Silver Spread	83

Table 33. Half-life days

Many practitioners do however use this formula to determine the maximum holdings period when trading spreads over consecutive days. Ideal candidates for a mean reversion strategy are spreads which produce a lower number of half-life days because this means that a trader has to wait less time to achieve profitability. In this case and as shown in table 33 the Gold Miner Spread would be the most attractive.

5.2 Results

A total of 18 individual models are created and traded of which 9 are unfiltered while the other 9 are traded under the supervision of a trading filter. In other words, each of the trading methodologies discussed in section 4 are traded in both an unfiltered and filtered simulation. Of the

total 18 models, six of the models are estimated using the ARMA methodology of which 3 are unfiltered and 3 are filtered. Another six models are based on cointegrating coefficients with 3 being unfiltered and 3 filtered. The final six models, also equally divided into unfiltered and filtered trading scenarios, are all MLP neural network models.

Each of these individual models is evaluated statistically in order to measure trading accuracy and forecasting ability on a model to model basis. Results from both sample periods can be seen in tables 34 and 35.

MODELS	MAE	MAPE	RMSE	THEIL-U	CDC
GM ARMA	0.0155	142.94%	0.0226	0.8088	53.18%
OIL ARMA	0.0164	134.01%***	0.0228	0.8738	52.19%
SILVER ARMA	0.0149**	150.03%	0.0216**	0.7884	51.29%
GM Cointegration	0.0190	405.01%	0.0274	0.7182	53.48%***
OIL Cointegration	0.0246	496.42%	0.0338	0.7009***	49.30%
SILVER Cointegration	0.0232	871.24%	0.0322	0.6928*	52.29%
GM MLP Neural Network	0.0150***	143.21%	0.0218***	0.7257	53.25%
OIL MLP Neural Network	0.0159	134.01%**	0.0213*	0.7005**	53.96%*
SILVER MLP Neural Network	0.0145*	123.12%*	0.0217**	0.8905	53.93**

Table 34. Unfiltered trading statistics (in-sample) (rankings: * = 1st, ** = 2nd, *** = 3rd)

MODELS	MAE	MAPE	RMSE	THEIL-U	CDC
GM ARMA	0.0107**	146.12%	0.0138**	0.8074	51.96%
OIL ARMA	0.0141	137.74%	0.0185	0.8773	50.00%
SILVER ARMA	0.0127	205.02%	0.0177	0.8687	53.92%
GM Cointegration	0.0138	347.48%	0.0183	0.7187***	53.92%
OIL Cointegration	0.0207	581.23%	0.0264	0.6963*	48.28%
SILVER Cointegration	0.0225	559.43%	0.0309	0.7058**	51.23%
GM MLP Neural Network	0.0104*	127.32%**	0.0136*	0.8669	54.41%***
OIL MLP Neural Network	0.0139	130.51%***	0.0185	0.8733	54.66%**
SILVER MLP Neural Network	0.0126***	120.96%*	0.0173***	0.8874	55.39%*

Table 35. Unfiltered trading statistics (out-of-sample) (rankings: * = 1st, ** = 2nd, *** = 3rd)

By observation of tables 34 and 35 it can be seen that the neural network models produce the most accurate forecasts with the lowest errors when predicting next day returns for each of the spreads. This observation is consistent throughout both the in- and out-of-sample datasets. Statistically, the second most accurate models were the ARMA models. The MLP models also produced the first and second highest ability to forecast directional change during the in-sample

dataset with 53.96% and 53.93% CDC (Correct Directional Change) respectively. This is followed closely by the gold miner cointegration model which ranked third with 53.48%. Rankings were altered slightly during the out-of-sample period with the MLP models producing the first (55.39%), second (54.66%) and third (54.41%) best CDC statistics across all three spreads. It is also worth noting that all Theil-u statistics were less than one during both datasets. This indicates that each model offers greater forecasting accuracy in comparison to a naïve model. Essentially, the Theil-u statistic is a relative performance measure comparing each model’s forecast with a naïve one day ahead forecast.

Tables 36 and 37 provide a collective summary of trading performance for each methodology. As mentioned previously, these methodologies are grouped into three separate portfolios with each portfolio comprising of three equally weighted individual models for each of the spreads. For instance, the first portfolio is based on an ARMA strategy, the second portfolio is based on a Cointegration strategy and the third is based on a non-linear MLP neural network strategy. All in-sample results can be found in appendix A.5. Formulas for each of the performance measures can be seen in appendix A.1.

	ARMA (3 models)	Cointegration (3 models)	MLP (3 models)
Annualised Return (excl. costs) ⁴²	18.09%	4.64%	23.81%
Annualised Volatility (excl. costs)	15.22%	16.47%	14.15%
Maximum Drawdown (excl. costs)	-14.32%	-15.30%	-11.16%
Average Drawdown	-3.70%	-5.98%	-2.10%
Maximum Drawdown Duration	71	419	87
Calmar Ratio	1.26	0.30	2.13
Information Ratio	1.19	0.28	1.68
# Transactions (annualised)	357	293	282
Total Trading Days	408	408	408
Transaction costs (annualised)	17.85%	14.64%	14.05%
Weightings (Capital Allocation)	33.33	33.33%	33.33%
Annualized Return (incl. costs) ⁴³	0.24%	-10.00%	9.76%
RANKING	2	3	1

Table 36. Unfiltered out-of-sample portfolio performance

⁴³ Transaction costs are calculated to be 5 basis points per transaction irrespective of which ETF is traded.

	ARMA (3 models)	Cointegration (3 models)	MLP (3 models)
Annualised Return (excl. costs)	8.67%	12.09%	25.28%
Annualised Volatility (excl. costs)	11.32%	10.27%	9.62%
Maximum Drawdown (excl. costs)	-9.56%	-9.52%	-5.90%
Average Drawdown	-2.72%	-1.98%	-1.48%
Maximum Drawdown Duration	127	92	75
Calmar Ratio	0.91	1.27	4.29
Information Ratio	0.77	1.18	2.63
# Transactions (annualised)	38	55	47
Total Trading Days	408	408	408
Transaction costs (annualised)	1.88%	2.72%	2.32%
Weightings (Capital Allocation)	33.33%	33.33%	33.33%
Annualized Return (incl. costs)	6.79%	9.38%	22.97%
Change in Annualised Returns	6.55%	19.38%	13.21%
Maximum Drawdown Reduction	4.76	5.78%	5.26%
Volatility Reduction	3.90	6.20%	4.53%
RANKING	3	2	1

Table 37. Filtered out-of-sample portfolio performance

By observation, the trading filter offers improved results in a number of areas. In particular, the mean reversion trading filter significantly reduces transaction costs. The filter also improves overall risk/return and downside risk /return profiles across all portfolios. This is due to lower overall volatility, reduced maximum drawdowns and higher annualised returns. Annualised returns are enhanced for all 3 portfolios with the cointegration portfolio remarkably generating 19.38% more in profit. As a result, this portfolio now ranks second behind the portfolio of MLP models.

For each of the portfolios (unfiltered and filtered), a comparative analysis of in-sample volatility is presented in figures 35, 36 and 37. Volatility is measured in terms of RiskMetrics volatility which is calculated using equation 50.

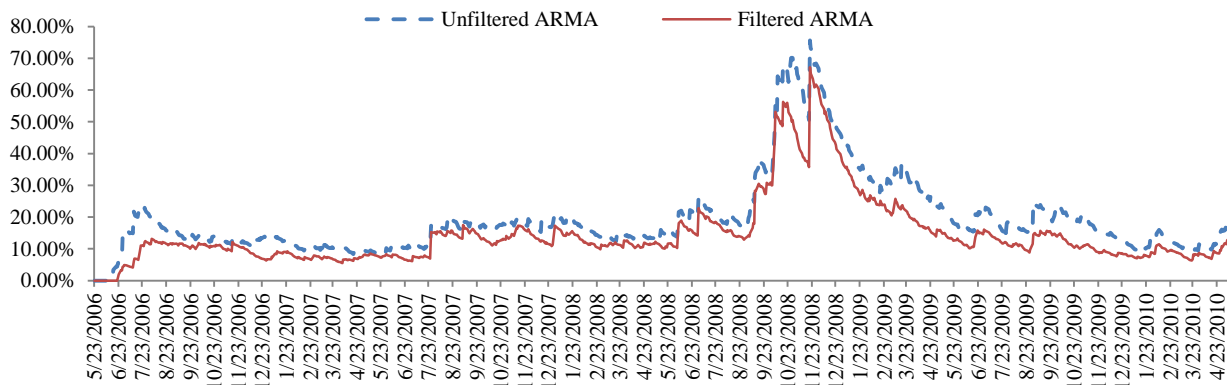


Figure 35. ARMA riskmetrics volatility (in-sample)

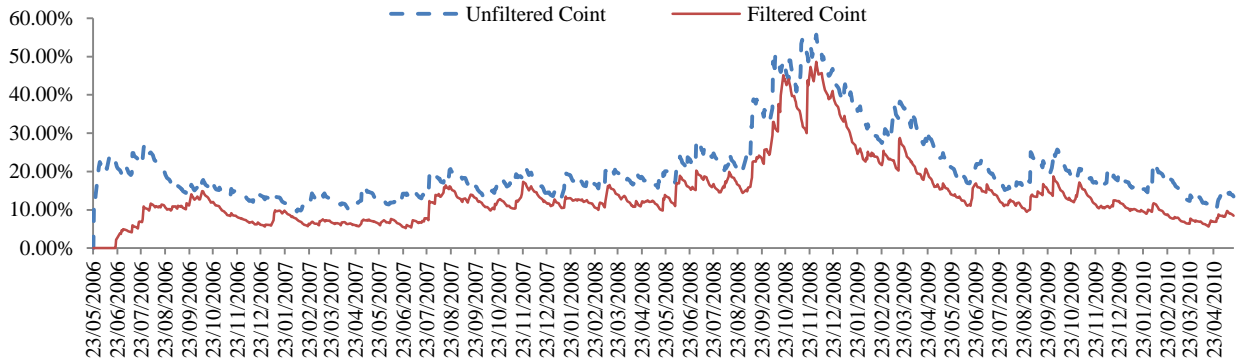


Figure 36. Cointegration riskmetrics volatility (in-sample)

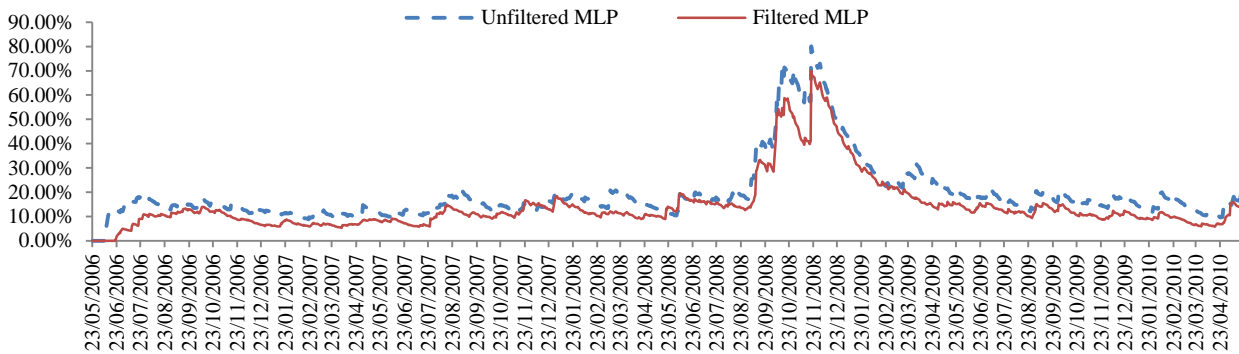


Figure 37. MLP riskmetrics volatility (in-sample)

RiskMetrics volatility was first used by JPMorgan (1997) based on the following formula:

$$\mu^2_{(t+1/t)} = \beta * \mu^2_{(t/t-1)} + (1 - \beta) * r^2_{(t)} \quad (50)$$

- where:
- μ^2 is the volatility forecast of our spread returns,
 - r^2 is the squared return of the spread,
 - β 0.94 for daily data as computed in JP Morgan (1997).

It is also worth noting that average drawdowns during the filtered simulations are less than those experienced with the unfiltered strategy. Furthermore, maximum drawdown experienced in

the filtered simulation are more acceptable and can be used to gauge whether or not a strategy should continue or cease to be traded. The maximum drawdown and maximum drawdown duration indicators quantify the level and duration of losses from which practitioners and clients can base their investment decisions. The maximum drawdown duration is the time from which cumulative returns last reached its previous high mark to when it next hits and surpasses this high mark. In all cases losses were not endured for more than 127 days for each of the filtered portfolios. As a result of the mean reversion trading filter, the maximum drawdown duration is improved dramatically from 419 days to 92 days for the portfolio of cointegration models. Furthermore, given the time period in which these models were backtested the maximum drawdowns for each of the portfolios are deemed acceptable as they are less than 25%. During the out of sample unfiltered trading, maximum drawdowns did not surpass 16% and during the filtered simulation they did not pass 10%.

6.0 Concluding Remarks

The purpose of this paper was to identify stationarity in the linear relationship between the prices of physical commodities and common stock prices for commodity companies, build models in which to forecast next day returns for these spreads, and then trade each model using a mean reversion trading filter.

Stationarity is found throughout much of in-sample dataset for each of the spreads with the silver spread providing the strongest evidence of stationarity. Periodically, both the gold miner and oil refiner spreads accept the hypothesis that the spread is not mean reverting due to shocks experienced during the in sample period. For instance, the 2008 summer surge in oil prices and the onset of the financial crisis at the end of 2008 both resulted in widening of the oil refiner and gold

miner spreads respectively. Evidence from each of the trading scenarios supports current literature as mean reversion is found to be profitable when spreads are stationary.

The mean reversion strategy used here attempts to exploit deviations from a spread's moving average which is otherwise historically stable given that the co-movements of two fundamentally related ETFs display stationarity. Using mean reversion to generate next day trading signals, an equally-weighted portfolio of three 'mean reverting spreads' is traded. In the first instance each model is traded in an unfiltered scenario using trading signals produced by each of the individual models. For example, when a trading model forecasts a negative next day return then a trading signal will be produced to 'short' the spread. The opposite is also true and when a model forecasts a positive next day movement a trading signal is generated to 'long' the spread.

The unfiltered out of sample performance of each model produces positive returns after transaction costs for two of the three portfolios. The ranking of portfolios remained the same as the performance registered during the backtest with the portfolio consisting of three MLP models generating the highest returns. The portfolio of cointegration models ranked third and produced negative annualised returns of -10.00%. Information ratios however were all positive with the portfolio of MLP models producing 1.68, the ARMA models 1.19 and the aggregation of cointegration models generating the lowest risk / return trade off with just 0.28. Maximum drawdowns were moderate with the cointegration fund producing -15.30% and a maximum drawdown duration of 419 days. Average drawdowns however were slightly more acceptable with the exception of the Cointegration portfolio. The MLP portfolio produced -2.10%, the ARMA portfolio a -3.70 and the Cointegration models a -5.98% in average drawdowns.

Following the initial unfiltered trading simulation it was deemed necessary to introduce a trading filter which would trade less frequently while capitalising on significant movements in each spread. The aim of the filter was to not only increase profitability of each model but to do so while

improving the risk return profiles. On the whole, the mean reverting trading filter produces superior results when compared to the unfiltered portfolios. Ranking of the models by annualised returns alters slightly with the portfolio of cointegration models moving up to second as the group of ARMA models moves down to third. In summary, the filter improves annualised returns, reduces maximum drawdowns, maximum drawdown durations and it also decreases overall volatility.

CHAPTER 7: Modelling, Forecasting and Trading the Crack – A Sliding Window Approach to Training Neural Networks

August 2013

Abstract

The aim of this analysis is to expand on earlier work carried out by Dunis *et al.* (2005) who model the Crack Spread from 01/01/1995 to 01/01/2005. This paper however provides a more sophisticated approach to non-linear modelling of the ‘Crack’. The selected trading period covers 777 trading days starting on 09/04/2010 and ending on 28/03/2013. The proposed model is a combined PSO (Particle Swarm Optimiser) and a RBF (Radial Basis Function) Neural Network (NN) which is trained using sliding windows of 380 and 500 days. This is benchmarked against a Multi-Layer Perceptron (MLP) NN using the same training protocol. Outputs from the neural networks provide forecasts for 1 day ahead trading simulations with each network being retrained every 5 days. To model the spread an expansive universe of 59 inputs across different asset classes are also used. Included in the input dataset are 5 Autoregressive Moving Average (ARMA) models and 2 GARCH (Generalised Autoregressive Conditional Heteroscedasticity) volatility models.

Results reveal that the sliding window approach to modelling the Crack Spread is effective when using 380 and 500 day training periods. Sliding windows of less than 380 days were found to produce unsatisfactory trading performance and reduced statistical accuracy. The PSO RBF model which was trained over 380 is superior in both trading performance and statistical accuracy when compared to its peers. As each of the unfiltered models’ volatility and maximum drawdown were unattractive a threshold confirmation filter is employed.

The threshold confirmation filter only trades when the forecasted returns are greater than an optimised threshold of forecasted returns. As a consequence, only forecasted returns of stronger conviction produce trading signals. This filter attempts to reduce maximum drawdowns and volatility by trading less frequently and only during times of greater predicted change. Ultimately, the confirmation filter improves risk return profiles for each model and transaction costs were also significantly reduced.

Keywords:

Spread Trading, PSO RBF Neural Network, MLP Neural Network, Sliding Window Training, ARMA, GARCH, Threshold Confirmation Filters.

1.0 Introduction

Petroleum refiners are exposed to price fluctuations on both sides of the refining process which may reduce profit margins. Refiner's primary risk is that posed by an increase in input (raw materials) prices while output prices such as RBOB (Reformulated Gasoline Blendstock for Oxygen Blending) gas and heating oil remain static or simultaneously decrease. This would result in narrowing of the spread and perhaps momentarily result in a negative spread as the price of crude becomes greater than the sum of output prices. In order to hedge this risk the 'Crack Spread' is traded to safeguard profit margins. The process of converting Crude oil into 'refined' outputs which include petroleum gas, gasoline, kerosene, diesel, industrial fuel oil (heating oil), lubricating oil, paraffin wax and asphalt is known as 'cracking' because crude oil is cracked to produce each by-product. In the refining industry there are two widely used crack ratios as the hedge traded by each refiner varies based on variables such as capacity and operational configuration. Furthermore, both the inputs (grades of crude oil) and outputs vary from region to region depending on requirements for delivery and demand for finished products. The RBOB unleaded gasoline contract traded here is

relatively new to the NYMEX exchange as the grade of gasoline changed in 2005 to include ethanol in the mix.

The first hedge is based on the 3:2:1 ratio which means that 3 barrels of crude oil are required to ‘crack’ 2 barrels of gasoline and 1 barrel of distillate heating oil fuel. The other ratio which refiners may trade is known as the 5:3:2 ratio. In this case, 5 barrels of crude are ‘cracked’ into 3 barrels of gasoline and 2 barrels of heating oil. Refiners that crack crude with a lower yield of gasoline relative to distillate are more likely to trade using the latter of the two combinations.

The spread is positive and hence profitable when the sum of by-products is greater than the cost to procure crude oil. As the hedge is executed based on the output side of the spread refiners would generally purchase crude oil futures to hedge rising crude prices and sell both the gasoline and heating oil futures to hedge decreasing output prices. This would be considered ‘shorting’ or selling the spread. Furthermore, these counteracting positions allow the market participant to ‘lock into’ a predetermined margin. For the purpose of this investigation, a spread between crude oil, gasoline and heating oil is formed by trading 3 futures contracts of crude oil, 2 futures contracts of RBOB unleaded gasoline and 1 futures contract of heating oil. This spread most closely represents the WTI Cushing / NYH RBOB 3:2:1 Crack as displayed figure 38.

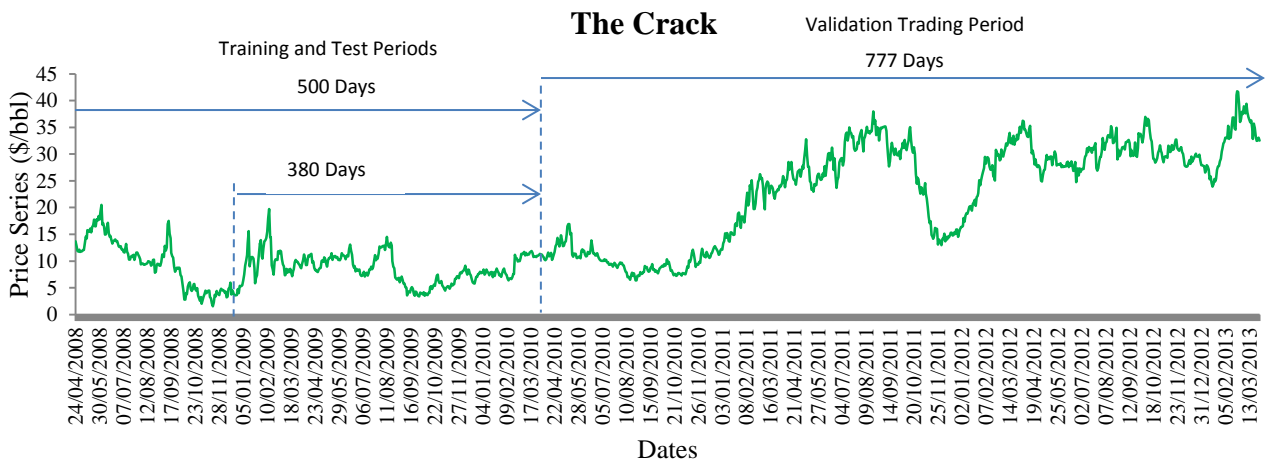


Figure 38. Full sample price series of the 3:2:1 crack spread

Motivation for this investigation derives from the initial analysis carried out by Dunis *et al.* (2005) who model the Crack Spread between NYMEX West Texas Intermediate (WTI) for crude oil and NYMEX Unleaded Gasoline (GAS). Conclusions reveal that neural networks offer interesting results and the aim here is to offer more insight into the benefits of using non-linear modelling by expanding the universe of explanatory variables, to train the network over different sliding windows using both a PSO algorithm and a traditional back propagation algorithm. In addition, each model is filtered using a threshold confirmation filter to enhance performance. Furthermore, the Spread which is investigated here also includes heating oil as an output. In general, the Crack Spread is calculated using three variables and not just crude oil and gasoline as traded in Dunis *et al.* (2005). Therefore, a more in depth application of neural networks is investigated to more accurately predict next day returns for the Crack Spread.

The Crack Spread is calculated using three variables. The input variable is crude oil (CL) which is denominated in US dollars per barrel while the outputs consist of gasoline (RBOB) and heating oil (HO) of which both are denominated in US cents per gallon. In order to create the spread a conversion of units is required. As the quantity of a crude contract is 1,000 barrels per contract and both the gasoline and heating oil amount to 42,000 gallons per contract then the latter two are multiplied by 0.42. This is based on the calculation that there are 42 gallons of oil per barrel. Using this conversion of units the outputs are converted into US dollars per barrel as mathematically depicted in equation 51.

$$3: 2: 1 \text{ CRACK SPREAD } St = \frac{(((2 \times \text{RBOB} \times 0.42) + (1 \times \text{HO} \times 0.42)) - (3 \times \text{CL}))}{3} \quad (51)$$

The methodology applied throughout this investigation in order to calculate the returns of the Crack spread can be seen below as provided by Butterworth and Holmes (2002) and more recently by Dunis *et al.* (2006) and Dunis *et al.* (2011b):

$$\Delta S_t = \left[\left(\frac{(P_{RBOB(t)} - P_{RBOB(t-1)})}{(P_{RBOB(t-1)})} + \frac{(P_{HO(t)} - P_{HO(t-1)})}{(P_{HO(t-1)})} \right) - \left(\frac{(P_{CL(t)} - P_{CL(t-1)})}{(P_{CL(t-1)})} \right) \right] \quad (52)$$

where: ΔS_t = percentage change in returns of the Crack spread at time t

$P_{RBOB(t)}$ = the price of RBOB at time t (in dollars per barrel)

$P_{RBOB(t-1)}$ = the price of RBOB at time $t-1$ (in dollars per barrel)

$P_{HO(t)}$ = the price of Heating Oil at time t (in dollars per barrel)

$P_{HO(t-1)}$ = the price of Heating Oil at time $t-1$ (in dollars per barrel)

$P_{CL(t)}$ = the price of Crude Oil at time t (in dollars per barrel)

$P_{CL(t-1)}$ = is the price of Crude Oil at time $t-1$ (in dollar per barrel)

The larger cap refiners include Exxon Mobil Corp., Total S.A., Royal Dutch Shell Plc., Chevron Corp., ConocoPhillips and BP Plc. as displayed in figure 39. Figure 40 on the other hand, focuses on small to medium sized refiners such as Western Refining Inc., Alon USA Energy Inc., Hess Corp., Tesoro Corp., and Valero Energy Corp. Both figures 39 and 40 display price performance (rebased to 100) of each company compared to the crack spread traded over the period from 09/04/2010 to 28/03/2013.

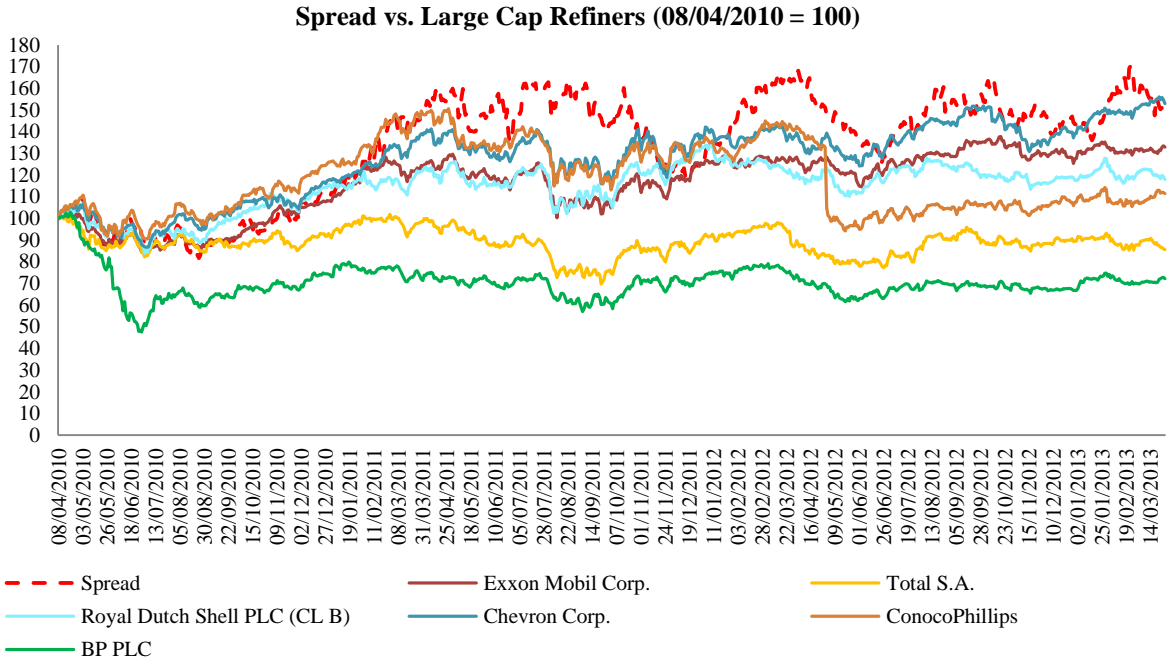


Figure 39. Trading dataset price performance. The ‘crack’ vs. large cap refiners equity

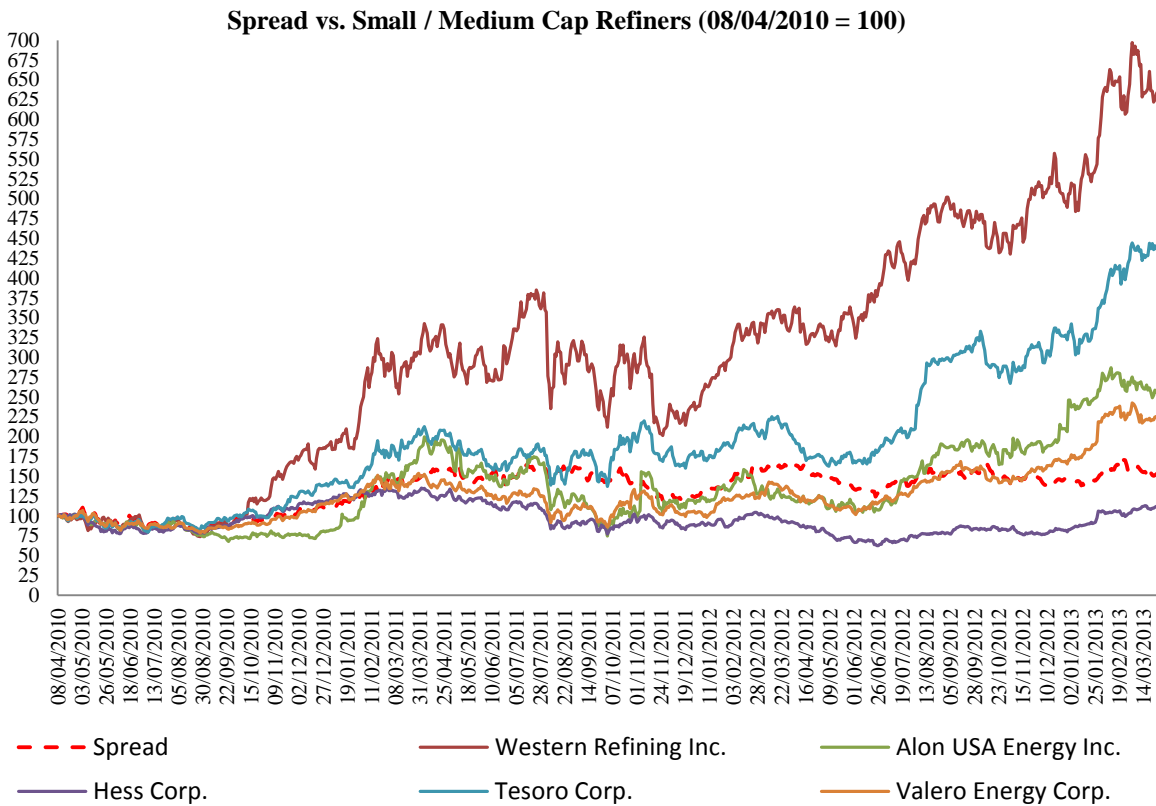


Figure 40. Trading dataset price performance (rebased to 100). The 'crack' vs. small / medium cap refiners equity

By observation, figure 39 displays a clear and strong relationship with each of the refining companies' equity. Refiner's equity increases as the spread widens and decreases as it narrows. The one exception or break in this relationship is in the summer of 2010 when BP Plc's stock price declined as a result of the oil spill in the Gulf of Mexico. This however, shows how many other additional factors besides endogenous factors such as operational efficiency also affect profit margins. Refining margins are also eroded by fixed costs and generally refiners aim to operate at their determined break even points to avoid inefficiency associated with excess capacity. Furthermore, as explained by Dunis *et al.* (2005) the magnitude of adjustments on the upside tend to be greater and more favourable than the losses endured on the downside. This could indicate that larger refiners have more influence on the crack spread and some may even manipulate margins to enhance their earnings.

Instead of benchmarking the proposed PSO RBF and MLP models against linear models, which is frequently criticised, this investigation utilises informational content from traditional models. Traditional models are included in the universe of inputs to produce a mixed model approach in attempt to improve the accuracy and trading performance of each neural network. In particular, the inclusion of a GARCH volatility time series was justified as it enhanced performance by reducing volatility and maximum drawdowns.

Preliminary research has led to a number of unanswered questions when using neural networks as a methodology for forecasting commodity spread time series. For instance, how large should the training window be? Should the inputs be pre-processed (i.e. normalisation of inputs or the removal of outliers)? What network configuration (e.g. number of hidden neurons, number of layers, etc.) should be selected? What algorithm should be used to train the data? In attempt to

answer these questions the remaining structure of this paper is presented as follows. Section 2 provides a review of all current literature relevant to modelling the Crack Spread and other Gasoline spreads. A review of literature which uses sliding windows to train networks is also included in section 2.2. Section 3 presents descriptive statistics of the data used to model the spread. Section 4 presents the methodologies and estimation parameters for the Particle Swarm Optimiser (PSO) Radial Basis Function (RBF) Neural Network (NN) and the Multi-Layer Perceptron (MLP) NN. Section 5 offers an evaluation of empirical results and trading performance. This is then followed by concluding remarks and research limitations.

2.0 Literature Review

2.1 Modelling the Crack

Numerous linear methodologies have been applied to the task of modelling and trading various combinations of gasoline spreads as well as the Crack Spread investigated here. For instance, Al-Gudhea *et al.* (2006) use threshold cointegration models to capture the relationships between crude, spot wholesale and retail gasoline price adjustments during the period of December 1998 to January 2004. In total four spreads are analysed. The first is a spread between crude oil prices and retail gasoline prices, the second is between crude oil prices and spot gasoline prices, the third spot gasoline prices and wholesale gasoline prices, and the fourth spread is that of wholesale gasoline prices and retail gasoline prices. Test statistics from each of these spreads confirm that they are all cointegrated with evidence of asymmetric adjustments toward long-run equilibrium.

Chen *et al.* (2005) also utilise threshold cointegration models when examining price adjustments for the spread between Crude Oil and Gasoline Prices. They find evidence of asymmetry in both the short and long run adjustments using both futures data and spot prices. In

particular, conclusions reveal that retail gasoline prices respond asymmetrically to crude oil price changes.

In a similar approach to modelling the Crack Spread, Dunis *et al.* (2005) use both the aforementioned Enders and Granger (1998) threshold cointegration technique and numerous neural network architectures. They apply a Higher Order Neural Network (HONN), a Recurrent Neural Network (RNN) and a Multilayer Perceptron Neural Network (MLP) to the task of predicting next day spread returns. A fixed training period is used to train each of the networks with the training set being divided into training and test datasets in order to avoid ‘over-fitting’. Over-fitting in this application is largely avoided due to the fact that the training window slides ‘x’ amount of days splitting each period into training and test datasets each time a forecast is produced. This is however discussed in more detail in section 3.0. Results from Dunis *et al.* (2005) reveal that the spread does in fact exhibit asymmetric adjustment. It is also observed that movements away from fair value are almost three times larger on the downside than on the upside. Overall the fair value cointegration model produces the most profitable trading performance. Out of three neural networks the HONN produces profits in excess of those achieved by the RNN and MLP neural networks.

2.2 Training of Neural Networks

Different approaches to training neural networks have been explored by many over the years and even more so in recent years. Kaastra and Boyd (1996) discuss these various techniques used to train neural networks. The most popular and widely used approach is one where the practitioner elects fixed training and validation datasets. For example, this training approach was adopted by Dunis *et al.* (2005) who also model the Crack spread. Using a fixed training and test dataset they train the network using 80% of the data and then validate the neural parameters over

the remaining 20% of the dataset (out of sample trading). Training datasets usually account for 70% to 90% of the in sample period while the validation dataset covers anywhere from 30% to as little as 10%. Another approach is one where the practitioner randomly selects the validation data set which is usually within the training dataset. This however may bias the test and reduce the accuracy when validating the training using larger out of sample datasets. For this reason the first approach is usually favoured by practitioners. In addition to this the first approach of selecting simultaneous in sample and out of sample datasets allows for practitioners to test the parameters of the ‘trained’ neural network on more recent data which is usually more relevant than historical data. The final approach Kaastra and Boyd (1996) propose is a ‘sliding window’ approach as used in this investigation when training both the PSO RBF and MLP neural networks. Kaastra and Boyd (1996) call this a ‘walk-forward’ testing routine which is commonly adopted by commodity trading systems to model and trade data in dynamic and changing market conditions. In order to adapt to these changing conditions a sliding window is utilised to provide a more robust and time varying approach. This technique continuously updates the training dataset and as a result it provides a more practical and realistic approach to trading financial assets.

More recently, Tsai and Wang (2009) use an average of different sliding windows to obtain an ensemble forecast when predicting next day returns for Taiwanese electronic stocks. They run 4 different sliding windows and take an average of these 4 training sets to produce a forecast. Chang *et al.* (2004) find that performance of neural networks is enhanced when using ensemble and hybrid techniques such as combining multiple forecasts of varying sliding windows. Thawornwong and Enke (2004) use a sliding window training technique to forecast an S&P500 monthly time series using a total of 31 inputs from 24 years of data. In particular they use four different sliding windows to capture different trends while also registering the significance of inputs during each of these windows. Over the four training periods, Thanwornwong and Enke (2004) find that six inputs

were consistently selected. These include the consumer price index for all urban consumers (CP), the money supply (M1), the 3-month T-bill rate (T3), the one-month certificate of deposit rate (CD1), the default spread between BAA and AAA credit ratings, the default spread between BAA and T120 (DE2) and the default spread between BAA and T3 (DE5). Therefore, it can be assumed that these variables were ‘reasonably’ significant as explanatory variables for the prediction of a monthly S&P500 time series.

In a comparative analysis of Artificial Neural Networks (ANN) and Genetic Evolutionary Algorithms (GEAs) Cortez *et al.* (2001) discuss the implications that may arise when selecting the duration of sliding windows. For instance, a large sliding window may increase the complexity of the neural network which could ultimately reduce the learning capabilities of the model. On the other hand, smaller windows may not contain a sufficient amount of information for the neural network to be able to train the data and produce ‘informationally’ significant forecasts.

3.0 Descriptive Statistics and Related Financial Data

All data was sourced from Bloomberg for the period of 24/04/2008 to 28/03/2013 for WTI Crude, RBOB Unleaded Gasoline and Heating Oil futures contracts. The RBOB Unleaded Gasoline contract is fairly new to the exchange as it replaced the old Unleaded Gasoline contract when Methyl Tertiary-Butyl Ether (MTBE) was phased out in 2005. This was seen to be less environmentally friendly than its alternative ethanol. As a result, this new blend now comprises of 10% ethanol. Segregation of the dataset is displayed in table 38.

Period	In Sample Training / Test Days	Trading Days	Beginning	End
<i>Total Dataset</i>	500* 380**	777	24/04/2008* 10/10/2008**	28/03/2013

<i>380 Day Training (Initial Window)</i>	380	380	10/10/2008	08/04/2010
<i>500 Day Training (Initial Window)</i>	500	500	24/04/2008	08/04/2010
<i>Validation set (out-of-sample)</i>	0	777	09/04/2010	28/03/2013

Table 38. Segregation of Dataset

As presented in table 38 the modelling and trading of the PSO RBF and the MLP neural networks is based on two sliding training windows using 380 for the shortest period and 500 days for the longer period. The first represents 1.5 years of working days and the second covers 2 full years of working days. Anything less than 380 days was found to produce unsatisfactory results therefore it is assumed that the training period did not include enough data points to accurately capture patterns within the data.

For the proposed PSO RBF and MLP models, over-fitting is dealt with using a two pronged approach. Firstly, each of the sliding windows is separated into training and test datasets. Training sets account for 66.66% of the sliding window while the remaining 33.33% is allocated for testing. The second control that has been tested during the in-sample backtest and implemented for the validation period is to use a fixed and constant amount of neurons in the hidden layer. For instance, for the PSO RBF model a total of 10 neurons were found to produce adequate results during the in-sample backtest while avoiding over-fitting. In the absence of a ‘feature selection method’ all inputs are selected during the training process for the MLP model. The complexity of the network is calculated based on the number of inputs as displayed in equation 53.⁴⁴

$$h = (n + 1) / 2 \tag{53}$$

where: h = number of hidden neurons
 n = number of inputs

⁴⁴ For this application a total of 30 hidden neurons were used for the MLP training process.

A PSO algorithm is used to calculate the number of hidden neurons for the RBF neural network. This algorithm is programmed to adapt, search for and identify the ‘optimal’ number of neurons. Results from these experiments produced an average of 25 to 30 neurons in the hidden layer. In this case, the complexity of a network with as many as 30 neurons was found to ‘over fit’ the dataset. Therefore, it was decided to use fewer (10 neurons) neurons in order to reduce the risk of over fitting with a less complex network topology.

Each of these training periods produces 1-day ahead forecasts. In a similar approach, Von Mettenheim and Breitner (2012) use a sliding window of 128 days to produce forecasts for 1 day ahead retaining the network every 10 days ($t, t_{+1}, t_{+2}, \dots, t_{+10}$) when modelling various stocks and ETFs. As the training process is rolling so too are the forecasts. For instance, the model which is trained over 380 days uses 380 data points in addition to the 10 autoregressive spread return series commencing on 10/10/2008 which is 390 days before the beginning of the validation period 09/04/2010. This would provide a forecast for t_{+1} . In order to obtain the predicted t_{+2} output the window moves forward by one day to include the actual return produced at t_{+1} in the training period which is used to estimate t_{+2} . Then t_{+1} and t_{+2} are used in the training window to produce t_{+3} and so on. Therefore, the sliding window approach is where the PSO-RBF and MLP networks are trained to use the last k values of a series ($t_{n-k} \dots t_n$) to predict the next value at t_{n+1} . In practice, this means that the model only needs to be trained every ‘x’ day(s) depending on the forecast horizon. In this case, the neural network is retrained every 5 days to produce a forecast as traded during the out-of sample validation period. More frequent retraining of a sliding window is not problematic as it takes a matter of minutes to retrain and generate forecasts. This can be done over a weekend or outside trading hours such as in the morning before market open.

The two models used here are trained to forecast the next day change in the Crack Spread (S_{t+1}) using historical returns from 59 different explanatory variables. S_t is essentially the daily

change in the spread as calculated in equation 52. Simple returns are used as inputs due to the fact that they enable neural networks to converge much quicker than price series data. Furthermore, many simple return time series are found to be stationary which is the main reason for quicker convergence. This is however not always the case as some time series display unit roots.

The selection of input variables is a modelling decision that can greatly affect a model's performance. Dunis *et al.* (2005), who also model and trade the Crack Spread, only use autoregressive returns of the spread to produce non-linear forecasts however for the purpose of this application a more comprehensive and significant set of inputs are considered. The aim is more accurately capture and forecast the directional change of the spread by introducing more informational content in the input series. A larger universe of inputs was initially evaluated over the duration of each training window. Following numerous backtests a total of 59 inputs were retained for out of sample trading. Included in these 59 inputs are various moving average time series based on 21, 50, 100, 150, 200 and 250 days, changes in daily implied volatility was also included by using the CBOE VIX index, five ARMA and two GARCH models are also incorporated into the training process. Research conducted by Dunis *et al.* (2011) find that the inclusion of the ARMA models as inputs to a 'mixed neural network' improves both statistical accuracy and trading performance as the training of the neural network is enhanced. Therefore, the inclusion of linear models as inputs for neural network training is justified. For the most part, in this application the inclusion of volatility models is found to effectively reduce overall volatility while also improving maximum drawdowns during the training period.

The majority of existing neural network literature uses fixed training windows during in-sample datasets which is not realistic especially during times when the dataset is continuous or when it experiences various regime changes. Furthermore, for the proposed RBF neural network the application of a PSO algorithm in input selection also provides more insight into the

significance of each input as the percentage that each is selected during the sliding windows is also recorded as displayed in table 39. This enables practitioners to see which explanatory variables are more influential during the period of 09/04/2010 – 28/03/2013 (777 trading days). The difference of results between the 380 and 500 day sliding windows may indicate that each sliding window identifies different trends in the data with different inputs becoming more significant at times than others.

The percentage of time each input is selected over all of the training periods is estimated based on:

$$\text{Input Selection Percentage} = N / R^* \quad \text{*with } R = (S - X) / S_{t+n} \quad (54)$$

Where:

- N = number of sliding window repetitions an input was selected
- R = repetitions
- S = total sample dataset
- X = days of sliding window
- S_{t+n} = spread forecast horizon

NEURAL INPUTS = 59 Total	SELECTION AS A % OF THE TRAINING WINDOW		Lags
	PSO RBF 380 Day Sliding Window 5 Day Forecast	PSO RBF 500 Day Sliding Window 5 Day Forecast	
Autoregressive Returns	46.26%	47.01%	1
Autoregressive Returns	68.06%	65.92%	2
Autoregressive Returns	47.91%	49.15%	3
Autoregressive Returns	45.19%	47.01%	4
Autoregressive Returns	46.37%	48.83%	5
Autoregressive Returns	44.45%	45.19%	6
Autoregressive Returns	48.82%	48.72%	7
Autoregressive Returns	50.54%	46.26%	8
Autoregressive Returns	50.43%	53.10%	9
Autoregressive Returns	46.69%	39.74%	10
Exxon Mobil Corp. Stock Price Returns	43.70%	43.91%	1
Total S.A. Stock Price Returns	47.65%	51.50%	1
Royal Dutch Shell Stock Price Returns	52.35%	50.85%	1
Chevron Corp. Stock Price Returns	46.15%	52.03%	1
ConocoPhillips Stock Price Returns	51.39%	44.44%	1
BP PLC Stock Price Returns	52.46%	52.78%	1
Western Refining Inc. Stock Price Returns	48.61%	54.28%	1
Alon USA Energy Inc. Stock Price Returns	53.10%	52.89%	1
Hess Corp. Stock Price Returns	50.43%	51.28%	1
Tesoro Corp. Stock Price Returns	49.57%	45.94%	1
Valero Energy Corp. Stock Price Returns	50.43%	55.02%	1
Crude Oil (NYM \$/bbl) Returns	50.97%	50.96%	1
Brent Crude (ICE \$/bbl) Returns	34.08%	26.71%	1
NY Harb RBOB (NYM \$/gal) Returns	47.97%	49.25%	1
Heating Oil (NYM \$/gal) Returns	47.12%	45.41%	1
Natural Gas (NYM \$/btu) Returns	48.82%	51.28%	1
CBOE Market VIX Return Series	48.18%	51.71%	1
Gold (NYM \$/ozt) Return Series	47.76%	47.54%	1
Silver (NYM \$/ozt) Return Series	48.51%	52.67%	1
British Pound (CME) Return Series	46.15%	46.90%	1
U.S. Dollar per Euro Return Series	48.18%	45.09%	1
USD / JPY Return Series	48.83%	46.15%	1
USD / CHF Return Series	47.65%	47.33%	1
USD / CAD Return Series	47.86%	46.90%	1
USD / AU D Return Series	46.69%	46.15%	1
USD / GBP Return Series	50.00%	47.22%	1
Euro STOXX 50 Return Series	50.75%	48.08%	1
S&P 500 Return Series	48.29%	50.64%	1
FTSE 100 Return Series	49.79%	47.44%	1
MSCI EAFE Return Series	48.83%	49.47%	1
MSCI The World Index Return Series	49.36%	49.47%	1
MSCI AC World Return Series	50.54%	50.43%	1
US TREASURY Bond 2 yr. Return Series	51.39%	48.93%	1
US TREASURY Bond 5 yr. Return Series	51.82%	51.07%	1
US TREASURY Bond 10 yr. Return Series	54.81%	49.89%	1
US TREASURY Bond 30 yr. Return Series	53.31%	48.18%	1
21 Day MA Return Series	53.74%	49.57%	21
50 Day MA Return Series	51.39%	48.72%	50
100 Day MA Return Series	53.95%	48.61%	100
150 Day MA Return Series	50.32%	47.65%	150
200 Day MA Return Series	50.11%	51.49%	200
250 Day MA Return Series	54.38%	54.06%	250
ARMA 1 Returns	43.80%	48.29%	(10,10)
ARMA 2 Returns	45.62%	44.55%	(8,8)
ARMA 3 Returns	55.99%	63.57%	(13,13)
ARMA 4 Returns	47.65%	47.76%	(4,4)
ARMA 5 Returns	55.63%	52.99%	(12,12)
GARCH 1 Returns	59.61%	59.40%	(16,16)
GARCH 2 Returns	59.08%	68.27%	(15,15)

Table 39. PSO RBF input selection during the training windows

By observation, table 40 provides a summary of the most significant PSO RBF neural inputs. In particular, the ARMA and GARCH inputs prove to be among the most valuable as explanatory variables.

Explanatory Variable	Lags (days)	380 Day Sliding Window	500 Day Sliding Window
Spread Return Series	2	68.06%	65.92%
BP PLC Stock Price Returns	1	52.46%	52.78%
Western Refining Inc. Stock Price Returns	1	48.61%	54.28%
Alon USA Energy Inc. Stock Price Returns	1	53.10%	52.89%
Valero Energy Corp. Stock Price Returns	1	50.43%	55.02%
US TREASURY Bond 10 yr. Return Series	1	54.81%	49.89%
US TREASURY Bond 30 yr. Return Series	1	53.31%	48.18%
250 Day MA Return Series	250	54.38%	54.06%
ARMA (13,13)	13	55.99%	63.57%
ARMA (12,12)	12	55.63%	52.99%
GARCH (16,16)	16	59.61%	59.40%
GARCH (15,15)	15	59.08%	68.27%

Table 40. Most significant explanatory variables

By including ARMA and GARCH time series the trading performance and statistical accuracy of the models was increased substantially. In addition, autoregressive time series of spread returns were also included in the modelling of the Crack Spread. The most significant input of the lagged spread returns from lags of 1-10 days was the 2-day lag with this being selected as often as 68.06% of the time during the 380 days sliding window period and 65.92% during the 500 day sliding window. Other more influential inputs included the daily changes in some of the refiners' share prices. For instance, BP Plc., Valero Energy Corp., Alon Energy Inc. and Western Refining Inc. were all seen as more significant relative to the other refiners. Each of these inputs is lagged by 1 day. Furthermore, of the daily changes in US treasury rates the 10 and 30 year rates were selected as much 54.81% and 53.31% respectively. Interestingly, Brent was the least selected input as it was only included 26.71% of the time during the 500 day sliding window.

A histogram of the spread's return series over the entire sample period is displayed in figure 41. This is found to display a leptokurtic distribution with positively high kurtosis. This however is quiet common when observing normal distributions of return series as data points tend to be highly

concentrated around the mean. Furthermore, all of the spreads are confirmed to be non-normal (confirmed at a 99% confidence level by the Jarque-Bera test statistic).

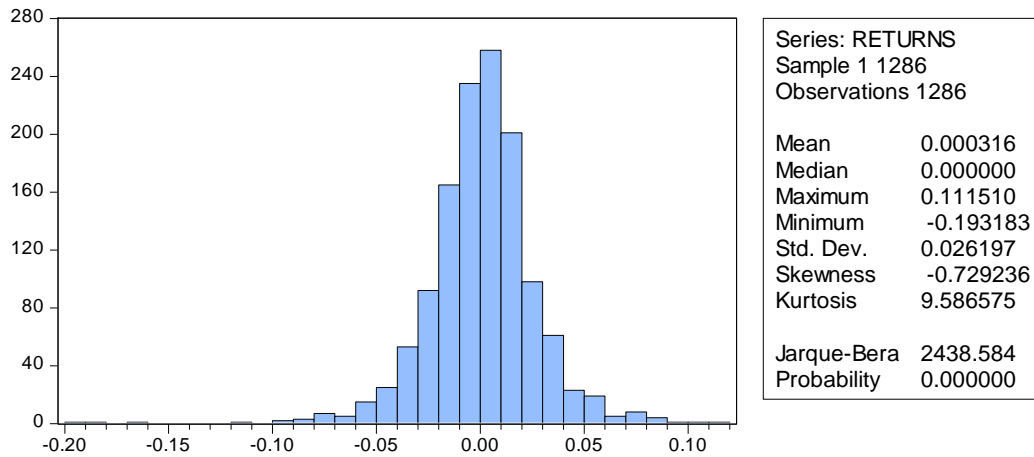


Figure 41. Spread returns (out of sample)

Equations and estimation output for each of the ARMA and GARCH models have been included in appendix. All ARMA models were found to be significant at a 95% confidence level as their p-values were less than 0.05 for each of the estimated (p,q) terms. The GARCH models were deemed stable and terms for both models were also significant at 95% confidence level. Residuals were tested for serial correlation using the squared residual test revealing that serial correlation is not present in either of the models. Therefore, the estimated models are deemed adequate and have been used to estimate two of the explanatory variables which are included during the training sliding window process of the neural network.

4.0 Methodology

The proposed models have been introduced and explain in Chapter 2 of the thesis. Estimation parameters and output from each of the ARMA and GARCH models can be seen in the appendix. The two proposed nonlinear methodologies are the MLP Neural Network and the Particle Swarm Optimiser Radial Basis Function Neural Network.

5.0 Empirical Results

The general trading rule is to long the spread on a positive forecast and short the spread when a negative forecast is indicated. When consecutive positive or negative signals are generated then the position is held from the previous signal. “Longing” the spread or buying the spread is when WTI Crude oil is sold and both Heating Oil and RBOB Gasoline are bought. “Shorting” the spread or selling the spread occurs when WTI Crude is bought and both Heating Oil and RBOB gasoline are sold.

5.1 Statistical Accuracy

Statistics are computed by taking the average of 10 executions in order to reduce the variance of each forecast. As neural networks are stochastic by nature it is in the best interest of a practitioner to use an average derived from numerous models. Computationally this is not too time consuming as forecasts are generated by numerous computers.⁴⁵

STATISTICAL PERFORMANCE	PSO RBF MODEL		MLP MODEL	
Sliding Training Windows	380	500	380	500
Forecast	1 day a head	1 day a head	1 day a head	1 day a head
MAE	0.0147	0.0148	0.205	0.0203
MAPE	166.87%	158.46%	420.05%	442.12%
RMSE	0.0194	0.0196	0.0263	0.0260
THEIL-U	0.8369	0.8349	0.6877	0.6974
Correct Directional Change (CDC)	52.38%	51.87%	50.32%	50.84%

Table 41. Out-of-sample trading statistics

From a statistical perspective the PSO-RBF model which is trained over 380 days is the most accurate when predicting $t+1$ returns. In particular the Correct Directional Change (CDC) statistic is more than 50%. A CDC of greater than 50% is more desirable. Both of the MLP sliding window models are also found to be less accurate in comparison to the PSO RBF models. For all

⁴⁵ Intel core i5 processors were used during both the back testing and forecasting phases. Furthermore, in order to reduce the estimation time 4 out of the 5 cores are utilised by executing the Parallel Toolbox function in Matlab 2011.

other statistics the lower they are the more accurate a model is considered to be. As explained by Dunis *et al.* (1996) the Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) statistics are ‘scale-dependent’ measures. These provide a modeller with statistics to compare each of the models with actual Crack Spread returns. The Theil-U statistic is one which falls between 0 and 1 with a model producing 0 being considered a ‘perfect’ model. Despite the significance of statistical accuracy the ultimate test is for a model to produce profit at acceptable levels of risk. Therefore, many traders will be more interested in how a model trades as discussed in section 5.2.

5.2 Trading Performance

During the training process the best weights for each of the PSO RBF models were registered. These have been included in appendix A.6. In total there are ten sets of best weights as each model is based on the average of 5 underlying models.

Numerous sliding windows were backtested and then traded for the purpose of forecasting the Crack Spread. As mentioned previously, any windows with less than 380 days of observations were found to produce unsatisfactory results.

Trading Performance	PSO RBF MODELS		MLP MODELS	
	380	500	380	500
Sliding Training Windows	380	500	380	500
Forecast Horizon	1 day a head	1 day a head	1 day a head	1 day a head
Gross Annualised Return	32.99%	28.84%	22.02%	18.86%
Annualised Volatility	23.92%	23.94%	23.94%	23.95%
Maximum Cumulative Drawdown	-44.90%	-29.10%	-46.31%	-30.09%
Average Daily Drawdown	-6.13%	-6.50%	-7.07%	-5.22%
Maximum Drawdown Duration (days)	248	234	318	191
Average Drawdown Duration (days)	29	44	50	30
Calmar Ratio	0.73	0.99	0.48	0.63
Information Ratio	1.38	1.21	0.92	0.79
# Transactions (annualised)	109	94	104	113
Total Trading Days	777	777	777	777
Transaction costs (annualised)	10.83%	9.37%	10.38%	11.29%
Net Annualized Return ⁴⁶	22.16%	19.47%	11.64%	7.31%
RANKING	1	2	3	4

Table 42. Out-of-sample unfiltered trading performance

⁴⁶Using a 10 basis point (bps) round trip transaction cost as offered by interactive brokers.

Unfiltered results from both 380 and 500 day sliding training windows are presented in table 42. By observation of table 42, the PSO RBF which was trained using a 380 day sliding window achieved the highest annualised returns and the best risk return trade off. This is challenged closely by the PSO RBF which is trained over 500 days. The MLP models which were also trained over 380 and 500 days ranked 3rd and 4th consecutively. Another interesting observation is that both the PSO RBF and the MLP models which were trained using a sliding window of 380 days had much worse maximum drawdowns in comparison to their respective 500 day models. However, in order to minimise maximum drawdowns a threshold filter which was optimised during the training period is used to filter each model.

Trading Performance	PSO RBF MODELS		MLP MODELS	
	380	500	380	500
Sliding Training Windows	380	500	380	500
Forecast Horizon	1 day a head	1 day a head	1 day a head	1 day a head
Gross Annualised Return	33.26%	28.60%	19.84%	21.08%
Annualised Volatility	18.19%	17.77%	13.28%	14.72%
Maximum Cumulative Drawdown	-19.70%	-19.35%	-16.68%	-16.72%
Average Daily Drawdown	-2.75%	-3.16%	2.63%	-2.78%
Maximum Drawdown Duration (days)	102	162	194	148
Average Drawdown Duration (days)	19	22	32	22
Calmar Ratio	1.69	1.48	1.19	1.26
Information Ratio	1.83	1.61	1.49	1.43
# Transactions (annualised)	90	88	55	71
Total Trading Days	777	777	777	777
Transaction costs (annualised)	8.98%	8.79%	5.48%	7.04%
Net Annualized Return (incl. costs)	24.28%	19.81%	14.36%	14.05%
Annualised Returns Filter Effect	2.12%	0.34%	2.72%	6.74%
Volatility Reduction	5.73%	6.17%	10.66%	9.23%
Drawdown Reduction	25.20%	9.75%	29.63%	13.37%
RANKING	1	2	3	4

Table 43. Out-of-sample filtered trading performance

Results from a filtered trading simulation are presented in table 43. With this threshold filter the model only trades when the PSO RBF and MLP NN models produce forecasts greater than ‘x’ or less than ‘x’. These ‘x’ parameters are optimised during the in sample period as a threshold for trading each of the models.⁴⁷ When comparing each of the forecasted return series it is clear that the MLP models are more erratic as they did not include the additional fitness function defined in equation 16

⁴⁷ For the RBF 380 and 500 day models the ‘x’ parameter = 0.20%. For the MLP 380 day model the ‘x’ parameter = 1.90% and for the MLP 500 day model ‘x’ = 1.45%.

(chapter 2) which maximises annualised returns. Using this filter, only larger more significant forecasts are traded while smaller less significant changes in the spread are filtered out. This minimises maximum drawdowns and reduces volatility while also increasing annualised returns. Model rankings remain constant with the PSO-RBF model, which is trained over a 380 day sliding window, producing the highest annualised returns and best risk/return profile.

When trading futures contracts a trader has to be aware of margins. At present, margins are around 9% for each of the contracts however most brokers calculate margins on an aggregate level and in this case margins would be calculated based on the spread performance of WTI crude, RBOB gasoline and heating oil. With this in mind traders could consider Calmar ratios as an indicator of how much return a model produces for 1 unit of drawdown as part of their criteria for selecting suitable models. Therefore, similar to the information ratio, a model which produces more than 1 would be considered a ‘good’ model. The formula used to calculate the Calmar ratio is displayed in the appendix A1. In this case, a trader would trade a filtered PSO RBF model using 380 days to train the network in order to forecast 1 day ahead. The Calmar ratio provides an indication of risk-adjusted performance and for the proposed PSO RBF filtered (380 day sliding window) model it is 1.69. Therefore it can be assumed that for one unit of drawdown 1.69 unit of return is produced. This is more than double the unfiltered performance which only produces 0.73 as maximum drawdowns are substantially higher with the filter. In terms of volatility the PSO RBF (380 day sliding window) also produces the most attractive risk/return profile as it trades with a 1.83 information ratio. As a result of the filter the model trades less frequently which reduces the impact of transaction costs. High transaction costs is one of the main drawbacks highlighted by Dunis *et al.* (2005) with an annualised average of 17.03% in transaction costs and between 93 and 106 trades per year being triggered for a Multilayer Perceptron (MLP), a Recurrent Neural Network (RNN) and a Higher Order Neural Network (HONN) model.

Figures 42 and 43 display the best two unfiltered trading performances over the out-of-sample trading periods for each of the sliding windows. By observation, both PSO RBF models experience periods of long drawdowns particularly from 18/04/2012 to 28/03/2013. However, the sliding window of 500 days recovers slightly and hits a new high watermark on 27/02/2013.

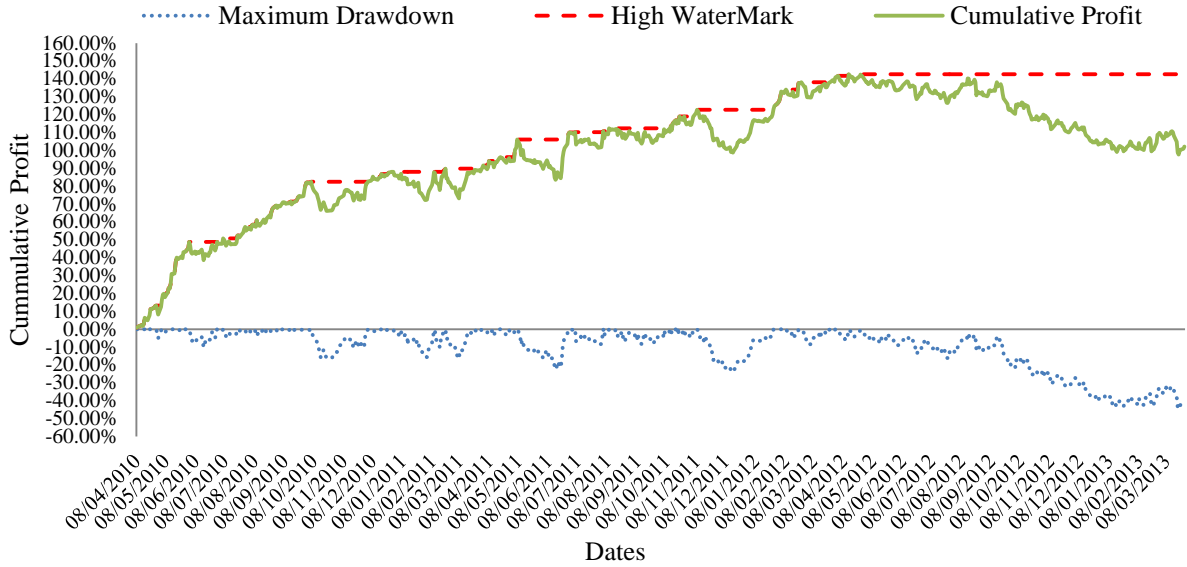


Figure 42. PSO RBF unfiltered trading performance (380 days sliding window).

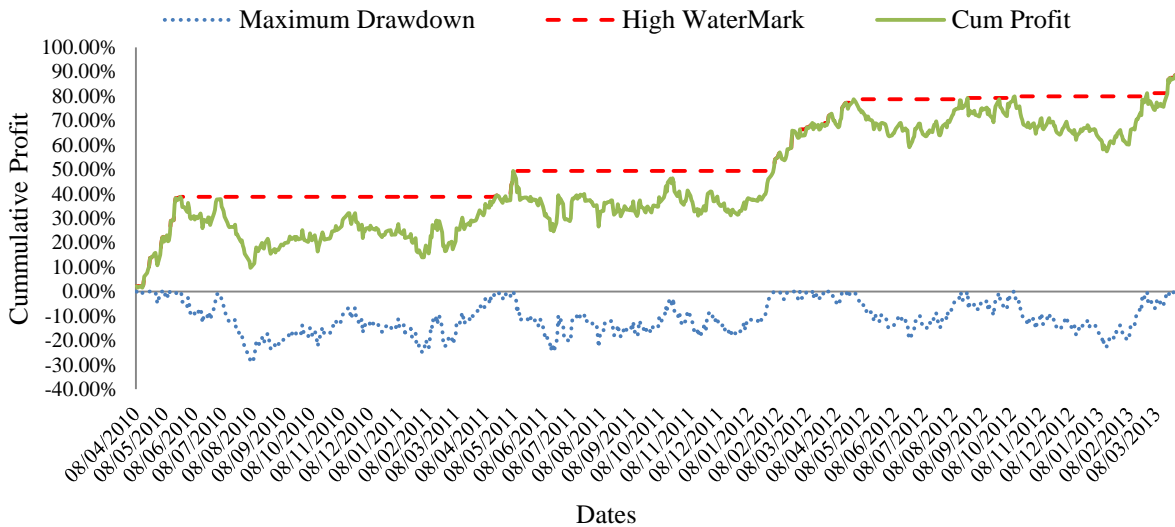


Figure 43. PSO RBF unfiltered trading performance (500 days sliding window).

Figures 42 and 43 display the best two filtered trading performances. Notably, the period of prolonged drawdowns mentioned during the unfiltered simulation is reduced as new high watermarks are more frequently achieved. Each of these models is also less erratic which reduces volatility by between 1.51% and 5.82%.

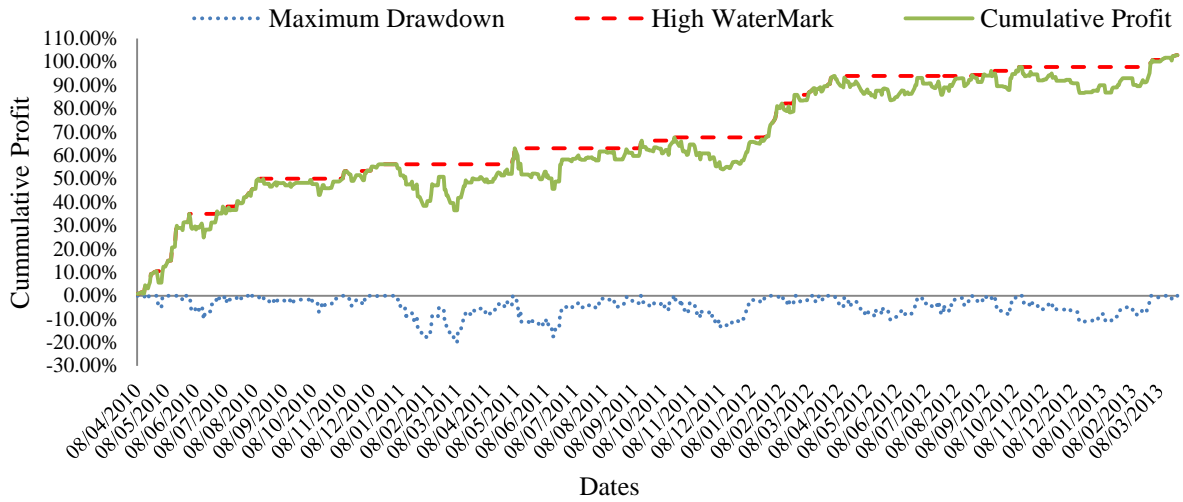


Figure 44. PSO RBF filtered trading performance (380 days sliding window)

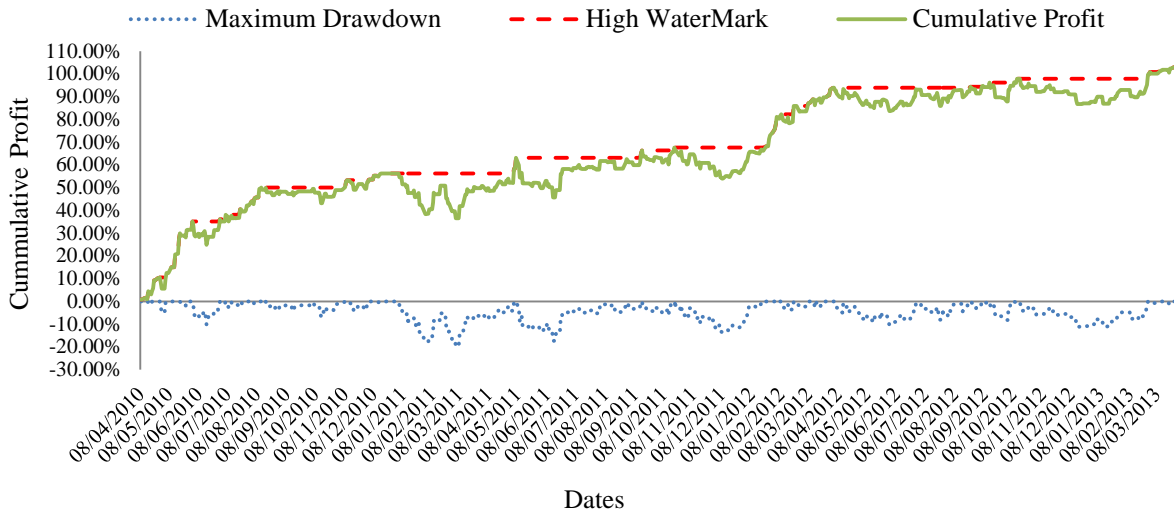


Figure 45. PSO RBF filtered trading performance (500 days sliding window)

A threshold filter is applied to reduce the frequency of trading while lessening volatility and maximum drawdowns. In Dunis *et al.* (2005) high transaction costs were found to significantly reduce profitability of each neural network.

6.0 Concluding Remarks

Results from empirical analysis clearly show that the sliding window technique for training the proposed PSO RBF neural network offers a mixture of positive results. The same is also true for the MLP neural network. Furthermore the inclusion of linear models as inputs also assists in enhancing the performance of both the PSO RBF and MLP models. This is corroborated by Newbold *et al.* (1974), Makridakis (1989), Clemen (1989), and Palm *et al.* (1992) who all establish that forecasts are improved by combining different linear forecasting methodologies when compared to individual forecasts. For the PSO RBF models a feature selection method is explored by using the PSO algorithm to optimise the inputs. During both sliding windows only the more significant inputs are selected to train the PSO RBF NN. Each time an input is selected the algorithm produces a '1' and when an input is not selected then a '0' is generated. At the end of the trading period the algorithm then calculates a total for each input as a percentage of time each were selected. Over the 380 and 500 day sliding windows a handful of more significant explanatory variables emerged. Table 40 in the descriptive statistics section summarises the more significant inputs over these periods. In summary, the longer term moving average inputs along with the ARMA and GARCH inputs ranked among the most significant. On the other hand, the MLP used all of the inputs for its training as no optimisation algorithms were employed during the input selection phase.

Unfiltered trading simulations are generated from sliding training windows of 380 and 500 day. Each of these models forecast 1 day ahead using a total of 59 explanatory variables to train both the PSO RBF and MLP models. Empirical results for the RBF NN produced 22.16% and 19.47% in annualised returns respectively. Information ratios for the RBF models were 1.38 for the 380 day window and 1.21 for the 500-day sliding window. Calmar ratios were slightly lower with 0.73 and 0.99 respectively. MLP models generated 11.64% and 19.47% in annualised returns, 0.92 and 0.79 as information ratios, 0.48 and 0.63 as Calmar ratios. Transaction costs for each scenario were extremely high as the models were frequently trading even during times of little change. This was also found to be the case by Dunis *et al.* (2005) who initially model the Crack Spread. For this reason a threshold confirmation filter was imposed.

The threshold confirmation filter only generates a trading signal once each of the forecasts is greater than $x\%$ or less than $-x\%$. This way each of the models only trades when the forecasts indicate more significant movements in the spread. As a result Information and Calmar ratios are significantly increased. The RBF model which is trained by a 380 day sliding window now trades with an information ratio of 1.83 and a Calmar ratio of 1.69. The other RBF model which is trained using a 500-day sliding window produces 1.61 as an information ratio and 1.48 for its Calmar ratio. Filtered returns for the 380 and 500 day MLP sliding window models were also improved with 14.36% and 14.05% respectively. Similar to the RBF models both the Information and Calmar ratios are also enhanced considerably. In summary, the risk/return and maximum drawdown/return profiles for each of the simulations are improved. All models return more than one unit of return (annualised return) for every one unit of risk (annualised volatility). As spread trading of futures contracts routinely requires market participants to meet margin calls a trader has to be aware of adverse movements in the spread. With this in mind, a trader would aim to select a model which

produces the highest return relative to drawdowns. In this case, a trader would select the PSO RBF 380-day sliding window model as it trades with a superior Calmar ratio of 1.69.

There are a few limitations found within this research. For one, only a few sliding windows are analysed and traded with two of the most suitable periods being displayed in the empirical findings. Results taken from combined sliding windows may enhance performance and will be researched in future applications. Further research could also be conducted to produce forecasts from an ensemble of many models as proposed by Mettenheim and Breitner (2012) who use a Historically Consistent Neural Network (HCNN) to provide forecasts. Finally the proposed PSO RBF could also be applied to other asset classes such as equities, foreign exchange, derivatives and fixed income in order to test its robustness.

CHAPTER 8: Final Remarks and Future Research

The main motivation of this thesis is to provide empirical evidence to support the use of non-linear methodologies when modelling, forecasting and trading commodity spreads. The research focuses on various commodity spreads such as the Corn/ Ethanol ‘Crush’ spread, the Gold Miner spread, spreads between US Mining / Refining stocks and the physical commodities that they produce and the last chapter trades the ‘Crack’ spread. The proposed non-linear models include a Recurrent, a Higher Order, a Multilayer Perceptron, and a Radial Basis Function Neural Network which are all tasked with forecasting daily changes in spreads. In addition to this, a Genetic Programming Algorithm was also used to forecast daily changes in the Corn/Ethanol spread. All trading models take transaction costs into consideration which are in line with commission rates charged by interactive brokers. Models are also evaluated using statistical measurements and trading metrics which are widely used in the finance industry. These measurements provide a thorough analysis of performance and put each of the model’s results into perspective. In order to test effectiveness and value added, each of the aforementioned non-linear models is benchmarked against more conventional methodologies. Furthermore, all models are traded in an unfiltered and filtered simulation. Numerous filters were examined however the most profitable strategies include a volatility threshold filter, a time varying leveraged volatility filter, a correlation filter, a mean reversion filter and a forecasted return threshold filter.

Each of the non-linear methodologies was trained using a mixture of inputs with the exception of Chapter 6. In Chapter 6 only autoregressive inputs from daily changes in the Gold Miner spread were used in order to test the MLP model using the same data as each of the benchmark models. Therefore, no additional ‘knowledge’ other than a dataset of autoregressive returns was used to train the MLP NN. For all of the other chapters however, a universe of

Multivariate explanatory variables are included to compute daily returns from equities, market indices, fixed income, commodities, outputs produced by linear models and volatility time series. As a result, the research provides an in depth analysis of the nonlinearities that exist between input datasets and desired outputs.

In Chapter 3 the Higher Order Neural Network (HONN) produced the best risk adjusted returns for both the unfiltered and filtered trading simulations. The threshold trading filter also proved to enhance profitability by improving risk adjusted returns. In Chapter 4 the HONN was challenged by the Genetic Programming Algorithm (GPA) and results reveal that the GPA model produced the highest statistical accuracy and profit. The time varying leveraged volatility filter generated returns in excess of the unfiltered trading scenario. By leveraging during lower volatility regimes and avoiding trading during higher levels of volatility information and calmar ratios are enhanced. Maximum drawdowns are also substantially reduced. In Chapter 5 the GPA model once again displayed superior forecasting ability when benchmarked against Multilayer Perceptron (MLP) and Radial Basis Function (RBF) NNs. A correlation filter was also applied which enabled the model to capitalise on changes in the correlation between Gold Miners Equity and Gold. In Chapter 6 only a MLP NN is used to trade a portfolio of commodity spreads. This is benchmarked against a portfolio of Cointegration models and a portfolio of ARMA models. Empirical results reveal that the MLP NN produces the most accurate statistical performance and the highest annualised returns over both sample periods. The mean reversion trading filter managed to exploit mean reversion during times of stationarity which appeared to be present during 'normal' market conditions. Chapter 7 concludes the research by offering a more in depth review of neural network modelling, forecasting and trading. An additional fitness function is also utilised to offer a multi-objective approach to forecasting. This additional fitness function is programmed to optimise the forecasted annualised returns. As a result, a threshold trading filter is employed to trade only during

times when the forecast is significant and is above a threshold. A further contribution of this chapter is the optimisation of inputs. Using a Particle Swarm Optimisation (PSO) algorithm, inputs are optimised and selected at each time step. This enables the network to only use the most ‘relevant’ inputs during the training process. Over the trading period the selection of inputs is registered to provide an indication of which explanatory variables are more relevant and influential. To conclude the research this final chapter also focuses on different training periods in order to test the stability and robustness of MLP and RBF NNs. This is of particular interest because in the real world market practitioners would need to retrain networks as new information becomes available. In the case of the MLP and RBF NNs, various sliding windows are evaluated however for the ‘Crack’ spread windows of 380 and 500 days are found to be the most profitable. In addition to this, the network is retrained every 5 days over these ‘sliding’ window periods to generate $t+1$ forecasts.

The collective empirical evidence provided by each of these chapters should hopefully provide market participants with enough compelling evidence to support the use of non-linear models as they provide accurate, profitable and robust forecasts. Furthermore, non-linear models are of particular use when an individual is presented with a large number of explanatory variables which need to be rationalised. Future research could be to improve more traditional models using various optimisation techniques such as PSO and Genetic Algorithms. This would not only improve existing models but it would also provide a fairer comparison to the aforementioned proposed artificially intelligent models. In particular, the researcher would like to apply PSO and GA algorithms to optimise ARMA and Cointegration models while also applying a sliding window/time varying estimation technique. In general, the results should go some way towards convincing quantitative fund managers and hedgers to use alternative non-linear techniques such as Neural Networks and Genetic Programming Algorithms to generate alpha.

CHAPTER 9: Appendix

A.1 Performance Measures

Root Mean Squared Error (RMSE)	$RMSE = \sqrt{(1/N) * \sum_{t=1}^{t+N} (\bar{\sigma}_t - \sigma_t)^2}$
Mean Absolute Error (MAE)	$MAE = (1/N) * \sum_{t=1}^{t+N} \bar{\sigma}_t - \sigma_t $
Mean Absolute Percentage Error (MAPE)	$MAPE = (1/N) * \sum_{t=1}^{t+N} \left \frac{\bar{\sigma}_t - \sigma_t}{\sigma_t} \right $
Theils-U Statistic	$THEIL-U = \sqrt{(1/N) * \sum_{t=1}^{t+N} (\bar{\sigma}_t - \sigma_t)^2} / \left[\sqrt{(1/N) * \sum_{t=1}^{t+N} (\bar{\sigma}_t)^2} + \sqrt{(1/N) * \sum_{t=1}^{t+N} (\sigma_t)^2} \right]$
Correct Directional Change (CDC)	$CDC = (100/N) * \sum_{t=1}^{t+N} D_t$ <p>Where $D_t = 1$ if $(\sigma_t - \sigma_{t-1}) * (\bar{\sigma}_t - \bar{\sigma}_{t-1}) > 0$, Else $D_t = 0$.</p>
Annualised Return	$R^A = 252 * \frac{1}{N} \sum_{t=1}^N R_t$ with R_t being the daily return
Cumulative Return	$R^C = \sum_{t=1}^N R_t$ with R_t being the daily return
Annualised Volatility	$\sigma^A = \sqrt{252} * \sqrt{\frac{1}{N-1} * \sum_{t=1}^N (R_t - \bar{R})^2}$
Information Ratio	$IR = \frac{R^A}{\sigma^A}$
Maximum Drawdown	Maximum negative value of $\sum (R^C_t)$ over the period $MaxDD = \min \left[R_t - \max \left(\sum_{t=1}^N R_t \right) \right]$
Calmar Ratio	$CR = \frac{R^A}{ MaxDD }$

Table 44. Statistical and trading performance measures

A.2. CHAPTER 3

A.2.1 Contract specifications

Contract Specifics	Corn	Ethanol
Product Code (Ticker)	ZC	EH

Contract Size	5,000 bushels	29,000 gallons
Contract Months	March, May, July, September, December.	All
Trading Venue	CME Globex	CME Globex
Last Trading Day	The business day prior to the 15 th calendar day of the contract month.	3 rd business day of delivery month.
Tick Size	¼ of 1 cent per bushel (\$12.50 per contract).	\$0.001 per gallon (\$29 per contract)
Trading Times	6:00pm – 7:15am and 9:30am – 1:15pm (CST)	6:00pm – 7:15am and 9:30am – 1:15pm (CST)

Table 45. Contract specifications

A.2.2 Network input criteria and selection

	CORN	AMEX Gas	Natural Gas	CRB Index	Crude Oil	Brent	ETHANOL	MSCI Commodity	S&P500 Energy IG
CORN	1.00	0.43		0.86	0.46		0.74	0.55	0.31
AMEX Natural Gas	0.43	1.00		0.35	0.35		0.53	0.93	0.96
CRB Index	0.86	0.35		1.00	0.44		0.71	0.37	0.33
Crude Brent Oil	0.46	0.35		0.44	1.00		0.56	0.38	0.34
ETHANOL	0.74	0.53		0.71	0.56		1.00	0.51	0.51
MSCI Commodity	0.55	0.93		0.37	0.38		0.51	1.00	0.94
S&P500 Energy IG	0.31	0.96		0.33	0.34		0.51	0.94	1.00

Table 46. Correlation matrix of neural inputs (in-sample correlation of returns)

Where the following correlation criteria were retained:

- 0.0 to 0.2: Very weak to negligible correlation
- 0.2 to 0.4: Weak, low correlation (not very significant)
- 0.4 to 0.7: Moderate correlation
- 0.7 to 0.9: Strong, high correlation
- 0.9 to 1.0: Very strong correlation

A.2.3 Correlogram of spread returns

Date: 05/22/13 Time: 15:47					
Sample: 1 1199					
Included observations: 1199					
Autocorrelation	Partial Correlation	AC	PAC	Q-Stat	Prob

				1	0.041	0.041	2.0432	0.153
				2	0.033	0.032	3.3753	0.185
				3	0.037	0.035	5.0633	0.167
				4	0.010	0.006	5.1823	0.269
				5	-0.010	-0.013	5.2935	0.381
				6	0.018	0.017	5.6870	0.459
				7	0.020	0.019	6.1779	0.519
				8	0.008	0.006	6.2564	0.619
				9	-0.050	-0.053	9.2544	0.414
				10	-0.019	-0.017	9.6731	0.470
				11	-0.008	-0.004	9.7529	0.553
				12	-0.003	0.002	9.7662	0.636
				13	0.030	0.032	10.858	0.623
				14	0.020	0.017	11.358	0.658
				15	0.019	0.017	11.791	0.695
				16	0.002	-0.000	11.797	0.758
				17	0.035	0.034	13.270	0.718
				18	-0.033	-0.039	14.602	0.689

Table 47. Correlogram of crush returns

Observation:

Serial correlation is not present as none of the terms are considered to be significant. All are greater than 0.05.

A.2.4 Networks characteristics

Parameters for each of the neural networks are displayed in table 48.

<i>Parameters</i>	<i>MLP</i>	<i>Re-current</i>	<i>HONNs</i>
<i>Learning algorithm</i>	<i>Gradient descent</i>	<i>Gradient descent</i>	<i>Gradient descent</i>
<i>Learning rate</i>	<i>0.001</i>	<i>0.001</i>	<i>0.001</i>
<i>Momentum</i>	<i>0.003</i>	<i>0.003</i>	<i>0.003</i>
<i>Iteration steps</i>	<i>1500</i>	<i>1500</i>	<i>1000</i>
<i>Initialisation of weights</i>	<i>N(0,1)</i>	<i>N(0,1)</i>	<i>N(0,1)</i>
<i>Input nodes</i>	<i>13</i>	<i>13</i>	<i>13</i>
<i>Hidden nodes (1layer)</i>	<i>6</i>	<i>5</i>	<i>0</i>
<i>Output node</i>	<i>1</i>	<i>1</i>	<i>1</i>

Table 48. Network characteristics

A.2.5 ARMA modelling

Estimation output from the ARMA modelling is shown in table 49

Dependent Variable: RETURNS
Method: Least Squares

Sample (adjusted): 12 831
 Included observations: 820 after adjustments
 Convergence achieved after 27 iterations
 MA Backcast: 1 11

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	-0.000404	0.001344	-0.300557	0.7638
AR(1)	0.321385	0.062300	5.158709	0.0000
AR(2)	0.287610	0.064725	4.443539	0.0000
AR(8)	-0.379210	0.044733	-8.477207	0.0000
AR(11)	0.547661	0.029768	18.39757	0.0000
MA(1)	-0.282525	0.050096	-5.639677	0.0000
MA(2)	-0.260749	0.054168	-4.813708	0.0000
MA(8)	0.436688	0.041548	10.51045	0.0000
MA(11)	-0.584663	0.023774	-24.59253	0.0000
R-squared	0.025820	Mean dependent var		-0.000365
Adjusted R-squared	0.016211	S.D. dependent var		0.027505
S.E. of regression	0.027281	Akaike info criterion		-4.354347
Sum squared resid	0.603583	Schwarz criterion		-4.302660
Log likelihood	1794.282	Hannan-Quinn criter.		-4.334514
F-statistic	2.686931	Durbin-Watson stat		1.992264
Prob(F-statistic)	0.006407			
Inverted AR Roots	.96 .44-.87i -.56+.68i	.89-.44i -.16+.89i -.93+.30i	.89+.44i -.16-.89i -.93-.30i	.44+.87i -.56-.68i
Inverted MA Roots	.94 .44-.89i -.56+.69i	.89+.44i -.17+.90i -.94+.30i	.89-.44i -.17-.90i -.94-.30i	.44+.89i -.56-.69i

Table 49. ARMA output

A.2.6 Empirical results in the training and test sub-periods

	Naive	ARMA	MLP	RNN	HONN
<i>MAE</i>	0.0284	0.0200	0.0201	0.0203	0.0201
<i>MAPE</i>	500.35%	178.31%	164.81%	163.56%	141.96%
<i>RMSE</i>	0.0376	0.0270	0.0274	0.0274	0.0273
<i>THEIL-U</i>	0.6867	0.8444	0.8864	0.8571	0.8705
<i>Correct Directional Change (CDC)</i>	48.80%	52.28%	54.03%	51.74%	54.15%

Table 50. In sample statistical accuracy

	Naive	MACD	ARMA	MLP	RNN	HONN
Annualised Return (excluding costs)	30.06%	34.87%	55.41%	67.32%	66.04%	68.82%
Annualised Volatility (excluding costs)	43.35%	43.31%	43.23%	43.15%	43.16%	43.14%
Maximum Drawdown (excluding costs)	-39.47%	-33.24%	-36.76	-28.08%	-34.75%	-27.43%
Calmar Ratio	0.76	1.05	1.51	2.40	1.90	2.51
Information Ratio	0.69	0.81	1.28	1.56	1.53	1.60
# Transactions (annualised)	131	19	87	93	103	98
Trading Days	831	831	831	831	831	831

Table 51. In sample trading performance (unfiltered)

	Naive	MACD	ARMA	MLP	RNN	HONN
Annualised Return (excluding costs)	33.56%	41.39%	58.08%	71.72%	72.70%	75.69%
Annualised Volatility (excluding costs)	33.19%	32.43%	35.39%	33.05%	34.80%	34.41%
Maximum Drawdown (excluding costs)	-31.46%	-25.56%	-32.44%	-24.19%	-28.80%	-24.39%
Calmar Ratio	1.07	1.62	1.79	2.96	2.52	3.10
Information Ratio	1.01	1.28	1.64	2.17	2.09	2.20
# Transactions (annualised)	150	42	58	90	85	83
Trading Days	831	831	831	831	831	831

Table 52. In sample trading performance results (filtered)

A.3 CHAPTER 4

A.3.1 Contract specifications

Contract Specifics	Corn	Ethanol
Product Code (Ticker)	ZC	EH
Contract Size	5,000 bushels	29,000 gallons
Contract Months	March, May, July, September, and December.	All
Trading Venue	CME Globex	CME Globex
Last Trading Day	The business day prior to the 15 th calendar day of the contract month.	3 rd business day of delivery month.
Tick Size	¼ of 1 cent per bushel (\$12.50 per contract).	\$0.001 per gallon (\$29 per contract)
Trading Times	6:00pm – 7:15am and 9:30am – 1:15pm (CST)	6:00pm – 7:15am and 9:30am – 1:15pm (CST)

Table 53. Contract specifications

A.3.2 Network input criteria and selection

	CORN	AMEX Natural Gas	CRB Index	Crude Brent Oil	ETHANOL	MSCI Commodity	S&P500 Energy IG
CORN	1.00	0.43	0.86	0.46	0.74	0.55	0.31
AMEX Natural Gas	0.43	1.00	0.35	0.35	0.53	0.93	0.96
CRB Index	0.86	0.35	1.00	0.44	0.71	0.37	0.33
Crude Brent Oil	0.46	0.35	0.44	1.00	0.56	0.38	0.34
ETHANOL	0.74	0.53	0.71	0.56	1.00	0.51	0.51
MSCI Commodity	0.55	0.93	0.37	0.38	0.51	1.00	0.94
S&P500 Energy IG	0.31	0.96	0.33	0.34	0.51	0.94	1.00

Table 54. Correlation matrix of neural inputs (in-sample return correlations)

Where the following correlation criteria were retained:

- 0.0 to 0.2: Very weak to negligible correlation
- 0.2 to 0.4: Weak, low correlation (not very significant)

- 0.4 to 0.7: Moderate correlation
- 0.7 to 0.9: Strong, high correlation
- 0.9 to 1.0: Very strong correlation

A.3.3 Model parameters

The below presents parameters that were used for the neural networks and the genetic programming algorithm. These were determined as they produced the best trading performance during the test sub-period.

<i>Parameters</i>	MLP	HONN
<i>Learning algorithm</i>	<i>Gradient descent</i>	<i>Gradient descent</i>
<i>Learning rate</i>	<i>0.001</i>	<i>0.5</i>
<i>Momentum</i>	<i>0.003</i>	<i>0.5</i>
<i>Iteration steps</i>	<i>10000</i>	<i>10000</i>
<i>Initialisation of weights</i>	<i>N(0,1)</i>	<i>N(0,1)</i>
<i>Input nodes</i>	<i>13</i>	<i>13</i>
<i>Hidden nodes (1layer)</i>	<i>7</i>	<i>7</i>
<i>Output node</i>	<i>1</i>	<i>1</i>

Table 55. Neural network characteristics

<i>Parameters</i>	GP
<i>Population Size</i>	<i>1,000</i>
<i>Tournament Size</i>	<i>20</i>
<i>Mutation Probability</i>	<i>0.75</i>
<i>Maximum Generations</i>	<i>100,000</i>

Table 56. GPA characteristics

A.3.4 ARMA modelling

The ARMA model used for this paper is as follows:

Dependent Variable: RETURNS

Method: Least Squares

Sample (adjusted): 12 998

Included observations: 987 after adjustments

Convergence achieved after 14 iterations

MA Backcast: 1 11

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	-0.000356	0.001098	-0.324182	0.7459
AR(1)	0.288226	0.063097	4.568006	0.0000
AR(2)	0.301169	0.063973	4.707719	0.0000
AR(8)	-0.368642	0.054960	-6.707406	0.0000

AR(11)	0.516903	0.044364	11.65138	0.0000
MA(1)	-0.254498	0.053703	-4.738988	0.0000
MA(2)	-0.267279	0.054945	-4.864501	0.0000
MA(8)	0.425671	0.050719	8.392696	0.0000
MA(11)	-0.561063	0.039067	-14.36173	0.0000
R-squared	0.021407	Mean dependent var		-0.000328
Adjusted R-squared	0.013402	S.D. dependent var		0.026263
S.E. of regression	0.026087	Akaike info criterion		-4.445693
Sum squared resid	0.665553	Schwarz criterion		-4.401061
Log likelihood	2202.950	Hannan-Quinn criter.		-4.428719
F-statistic	2.674272	Durbin-Watson stat		2.006461
Prob(F-statistic)	0.006565			
Inverted AR Roots	.95	.88+.43i	.88-.43i	.43-.87i
	.43+.87i	-.16-.88i	-.16+.88i	-.56+.68i
	-.56-.68i	-.93-.30i	-.93+.30i	
Inverted MA Roots	.94	.89+.44i	.89-.44i	.43+.88i
	.43-.88i	-.17+.89i	-.17-.89i	-.56-.69i
	-.56+.69i	-.94+.30i	-.94-.30i	

Table 57. ARMA results

A.3.5 Empirical results in the training and test sub-periods

	Naive	MACD	ARMA	MLP	HONN	GPA
Annualised Return (excluding costs)	23.96%	8.77%	62.87%	71.77%	72.54%	75.36%
Annualised Volatility (excluding costs)	41.43%	40.66%	41.27%	40.69%	40.68%	40.66%
Maximum Drawdown (excluding costs)	-39.47%	-39.77%	-39.77%	-39.47%	-39.77%	-32.31%
Calmar Ratio	0.61	0.22	1.58	1.82	1.82	2.33
Information Ratio	0.58	0.22	1.52	1.76	1.78	1.85
# Transactions (annualised)	128	14	98	101	100	126
Trading Days	998	998	998	998	998	998

Table 58. In sample (unleveraged trading performance)

	Naive	MACD	ARMA	MLP	HONN	GPA
Annualised Return (excluding costs)	1.18%	0.98%	49.22%	79.53%	88.63%	84.61%
Annualised Volatility (excluding costs)	43.21%	41.07%	43.10%	41.04%	41.10%	41.40%
Maximum Drawdown (excluding costs)	-25.40%	-25.09%	-24.70%	-24.79%	-24.31%	-23.40%
Calmar Ratio	0.05	0.04	1.99	3.21	3.65	3.62
Information Ratio	0.03	0.02	1.14	1.94	2.16	2.04
# Transactions (annualised)	122	5	93	92	90	118
Trading Days	998	998	998	998	998	998

Table 59. In sample (leveraged trading performance)

A.4 CHAPTER 5

A.4.1 Non-linear network parameters

Table 60 detail the parameters used for the neural networks and table 61 sets out the genetic programming specifications. These were used because they produced the most superior trading and statistical performance during the test sub-period.

<i>Parameters</i>	<i>MLP</i>	<i>PSO RBF</i>
<i>Learning algorithm</i>	<i>Gradient descent</i>	<i>PSO</i>
<i>Learning rate</i>	<i>0.001</i>	<i>-</i>
<i>Momentum</i>	<i>0.003</i>	<i>-</i>
<i>Iteration steps</i>	<i>10000</i>	<i>100</i>
<i>Initialisation of weights</i>	<i>N(0,1)</i>	<i>Adaptable</i>
<i>PSO Agents</i>	<i>-</i>	<i>30</i>
<i>Input nodes</i>	<i>7</i>	<i>7</i>
<i>Hidden nodes (1layer)</i>	<i>4</i>	<i>10</i>
<i>Output node</i>	<i>1</i>	<i>1</i>

Table 60. Neural network parameters

<i>Parameters</i>	<i>GP</i>
<i>Population size</i>	<i>1000</i>
<i>Tournament size</i>	<i>20</i>
<i>Mutation probability</i>	<i>0.75</i>
<i>Maximum generations</i>	<i>100000</i>

Table 61. Genetic programming parameters

A.4.2 ARMA modelling

The ARMA model used for this paper is as follows:

Dependent Variable: RETURNS

Method: Least Squares

Sample (adjusted): 12 998

Included observations: 987 after adjustments

Convergence achieved after 14 iterations

MA Backcast: 1 11

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	-0.000356	0.001098	-0.324182	0.7459
AR(1)	0.288226	0.063097	4.568006	0.0000
AR(2)	0.301169	0.063973	4.707719	0.0000
AR(8)	-0.368642	0.054960	-6.707406	0.0000
AR(11)	0.516903	0.044364	11.65138	0.0000
MA(1)	-0.254498	0.053703	-4.738988	0.0000
MA(2)	-0.267279	0.054945	-4.864501	0.0000
MA(8)	0.425671	0.050719	8.392696	0.0000
MA(11)	-0.561063	0.039067	-14.36173	0.0000
R-squared	0.021407	Mean dependent var		-0.000328
Adjusted R-squared	0.013402	S.D. dependent var		0.026263
S.E. of regression	0.026087	Akaike info criterion		-4.445693
Sum squared resid	0.665553	Schwarz criterion		-4.401061
Log likelihood	2202.950	Hannan-Quinn criter.		-4.428719
F-statistic	2.674272	Durbin-Watson stat		2.006461
Prob(F-statistic)	0.006565			
Inverted AR Roots	.95	.88+.43i	.88-.43i	.43-.87i
	.43+.87i	-.16-.88i	-.16+.88i	-.56+.68i
	-.56-.68i	-.93-.30i	-.93+.30i	
Inverted MA Roots	.94	.89+.44i	.89-.44i	.43+.88i
	.43-.88i	-.17+.89i	-.17-.89i	-.56-.69i
	-.56+.69i	-.94+.30i	-.94-.30i	

Table 62. ARMA output

A.4.3 In-sample trading performance

	Buy & Hold	Naive	Cointegration	ARMA	MLP	PSO RBF	GPA
Annualised return (excl. costs)	-3.80%	-28.82%	17.06%	37.54%	43.12%	44.40%	47.94%
Annualised volatility (excl. costs)	37.97%	37.92%	24.52%	37.74%	37.76%	37.76%	37.74%
Maximum drawdown (excl. costs)	-	-117.57%	-31.82%	-52.21%	-34.19%	-32.20%	-27.04%
Calmar ratio	-	-0.25	0.54	0.72	1.26	1.38	1.77

Information ratio	-0.10	-0.76	0.70	0.99	1.14	1.18	1.27
# Transactions (annualised)	0.56	132	125	132	151	160	155
Total trading days	900	900	900	900	900	900	900

Table 63. In-sample trading performance (unfiltered)

	Naive	Cointegration	ARMA	MLP	PSO RBF	GPA
Annualised return (excl. costs)	-10.98%	6.70%	42.77%	45.96%	53.76%	52.90%
Annualised volatility (excl. costs)	28.79%	28.79%	29.40%	28.65%	28.59%	28.60%
Maximum drawdown (excl. costs)	-45.66%	-77.37%	-36.89%	-20.24%	-22.39%	-15.44%
Calmar ratio	-0.24%	0.09	1.16	2.27	2.40	3.43
Information ratio	-0.38%	0.23	1.45	1.60	1.88	1.85
# Transactions (annualised)	97	95	102	104	103	105
Total trading days	900	900	900	900	900	900
Correlation filter excess returns	17.84%	-10.36%	5.23%	2.84%	9.36%	4.96%
Volatility effect	9.13%	-4.27%	8.34%	9.11%	9.17%	9.14%
Maximum drawdown effect	71.91%	-45.55%	15.32%	13.95%	9.81%	11.60%

Table 64. In-sample trading performance (filtered)

A.5 CHAPTER 6

A.5.1 ARMA results

GM spread

Dependent Variable: GOLD MINER SPREAD RETURNS

Method: Least Squares

Sample (adjusted): 10 1006

Included observations: 997 after adjustments

Convergence achieved after 23 iterations

MA Backcast: 1 9

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	-1.81E-05	0.000779	-0.023209	0.9815
AR(1)	-0.551951	0.047657	-11.58186	0.0000
AR(4)	-0.270664	0.048921	-5.532681	0.0000
AR(6)	0.203737	0.042578	4.784984	0.0000
AR(9)	0.391021	0.043414	9.006880	0.0000
MA(1)	0.598786	0.037774	15.85185	0.0000
MA(4)	0.361253	0.038841	9.300940	0.0000
MA(6)	-0.174209	0.032821	-5.307795	0.0000
MA(9)	-0.463197	0.032857	-14.09732	0.0000
R-squared	0.044658	Mean dependent var		-4.24E-05
Adjusted R-squared	0.036923	S.D. dependent var		0.023193
S.E. of regression	0.022761	Akaike info criterion		-4.718544
Sum squared resid	0.511849	Schwarz criterion		-4.674269
Log likelihood	2361.194	Hannan-Quinn criter.		-4.701714
F-statistic	5.773094	Durbin-Watson stat		2.105465
Prob(F-statistic)	0.000000			

Inverted AR Roots	.85 .16+.83i -.89+.30i	.62+.61i -.59-.78i	.62-.61i -.59+.78i	.16-.83i -.89-.30i
Inverted MA Roots	.85 .16+.84i -.91+.33i	.64-.63i -.61-.77i	.64+.63i -.61+.77i	.16-.84i -.91-.33i

Table 65. GM spread ARMA results

OIL spread

Dependent Variable: RETURNS				
Method: Least Squares				
Date: 05/31/13 Time: 16:35				
Sample (adjusted): 9 1006				
Included observations: 998 after adjustments				
Convergence achieved after 34 iterations				
MA Backcast: 1 8				
Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	-0.000564	0.000648	-0.870074	0.3845
AR(1)	0.706466	0.078865	8.957906	0.0000
AR(3)	0.608617	0.085284	7.136320	0.0000
AR(5)	-0.761154	0.072415	-10.51099	0.0000
AR(8)	0.332068	0.094558	3.511778	0.0005
MA(1)	-0.762511	0.071512	-10.66277	0.0000
MA(3)	-0.591170	0.084390	-7.005224	0.0000
MA(5)	0.812198	0.070675	11.49199	0.0000
MA(8)	-0.358755	0.089035	-4.029375	0.0001
R-squared	0.017853	Mean dependent var		-0.000452
Adjusted R-squared	0.009908	S.D. dependent var		0.023082
S.E. of regression	0.022967	Akaike info criterion		-4.700535
Sum squared resid	0.521683	Schwarz criterion		-4.656294
Log likelihood	2354.567	Hannan-Quinn criter.		-4.683719
F-statistic	2.247202	Durbin-Watson stat		2.023479
Prob(F-statistic)	0.022222			
Inverted AR Roots	.94 -.17-.95i	.85-.41i -.36+.61i	.85+.41i -.36-.61i	-.17+.95i -.86
Inverted MA Roots	.94 -.16-.95i	.87-.41i -.37+.61i	.87+.41i -.37-.61i	-.16+.95i -.87

Table 66. Oil spread ARMA results

SILVER spread

Dependent Variable: RETURNS				
Method: Least Squares				
Date: 06/01/13 Time: 00:03				
Sample (adjusted): 8 1006				
Included observations: 999 after adjustments				
Convergence achieved after 144 iterations				
MA Backcast: 1 7				

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	-5.74E-06	0.000728	-0.007894	0.9937
AR(1)	-0.946493	0.065984	-14.34434	0.0000
AR(6)	0.646965	0.152173	4.251516	0.0000
AR(7)	0.608717	0.138243	4.403243	0.0000
MA(1)	0.873074	0.078495	11.12263	0.0000
MA(6)	-0.618301	0.155440	-3.977746	0.0001
MA(7)	-0.525271	0.136862	-3.837968	0.0001
R-squared	0.051619	Mean dependent var		-8.32E-05
Adjusted R-squared	0.045883	S.D. dependent var		0.022039
S.E. of regression	0.021528	Akaike info criterion		-4.831980
Sum squared resid	0.459730	Schwarz criterion		-4.797599
Log likelihood	2420.574	Hannan-Quinn criter.		-4.818912
F-statistic	8.998906	Durbin-Watson stat		2.138209
Prob(F-statistic)	0.000000			
Inverted AR Roots	.93	.46+.81i	.46-.81i	-.47+.80i
	-.47-.80i	-.91	-.97	
Inverted MA Roots	.92	.46-.80i	.46+.80i	-.46-.80i
	-.46+.80i	-.83	-.96	

Table 67. Silver spread ARMA results

A.5.2 MLP training parameters

Parameters	GM MLP	OIL MLP	SILVER MLP
Learning algorithm	Gradient descent	Gradient descent	Gradient descent
Learning rate	0.001	0.001	0.001
Momentum	0.003	0.003	0.003
Iteration steps	10000	10000	10000
Initialisation of weights	N(0,1)	N(0,1)	N(0,1)
Input nodes	24	24	24
Hidden nodes (1layer)	5	5	5
Output node	1	1	1

Table 68. Training parameters

A.5.3 Portfolio results

	ARMA	Cointegration	MLP
Annualised Return (excl. costs)	52.67%	17.46%	57.11%
Annualised Volatility (excl. costs)	22.46%	22.80%	22.85%
Maximum Drawdown (excl. costs)	-18.16%	-24.67%	-17.73%
Average Drawdown	-5.21%	-6.15%	-3.70%
Maximum Drawdown Duration	366	181	102
Calmar Ratio	2.90	0.71	3.22
Information Ratio	2.35	0.77	2.50
# Transactions (annualised)	473	293	282
Total Trading Days	1006	1006	1006
RANKING	2	3	1

Table 69. Unfiltered in-sample portfolio trading performance

	ARMA	Cointegration	MLP
Annualised Return (excl. costs)	47.81%	22.85%	59.10%
Annualised Volatility (excl. costs)	18.99%	16.27%	17.98%
Maximum Drawdown (excl. costs)	-12.37%	-15.06%	-5.98%
Average Drawdown	-2.77%	-3.08%	-1.18%
Maximum Drawdown Duration	179	201	66
Calmar Ratio	3.87%	1.52	8.00
Information Ratio	2.52	1.40	3.29
# Transactions (annualised)	252	198	226
Total Trading Days	1006	1006	1006
RANKING	2	3	1

Table 70. Filtered in-sample portfolio trading performance

A.6 CHAPTER 7

A.6.1 Supplementary information

Refiner	Market Capitalisation (m\$)	As at:	Source
Exxon Mobil Corp.	384,819	03/05/2013	FactSet (2013)
Chevron Corp.	222,559	03/05/2013	FactSet (2013)
Royal Dutch Shell PLC (CL B)	134,994	03/05/2013	FactSet (2013)
BP PLC	84,283	03/05/2013	FactSet (2013)
Total S.A.	83,399	03/05/2013	FactSet (2013)
ConocoPhillips	69,458	03/05/2013	FactSet (2013)
Hess Corp.	22,959	03/05/2013	FactSet (2013)
Valero Energy Corp.	21,190	03/05/2013	FactSet (2013)
Tesoro Corp.	6,737	03/05/2013	FactSet (2013)
Western Refining Inc.	2,629	03/05/2013	FactSet (2013)
Alon USA Energy Inc	1,056	03/05/2013	FactSet (2013)

Table 71. The refiners market capitalisation

A.6.2 ARMA equations and estimations

Autoregressive moving average (ARMA) models assume that the future value of a time series is governed by its historical values (the autoregressive component) and on previous residual values (the moving average component). A typical ARMA model takes the form of equation 55.

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \dots + \phi_p Y_{t-p} + \varepsilon_t - w_1 \varepsilon_{t-1} - w_2 \varepsilon_{t-2} - \dots - w_q \varepsilon_{t-q} \quad (55)$$

Where:

Y_t is the dependent variable at time t

Y_{t-1} , Y_{t-2} , and Y_{t-p} are the lagged dependent variable

ϕ_0 , ϕ_1 , ϕ_2 , and ϕ_p are regression coefficients

ε_t is the residual term

ε_{t-1} , ε_{t-2} , and ε_{t-p} are previous values of the residual

w_1 , w_2 , and w_q are weights.

Using a correlogram as a guide in the training and the test sub-periods the below restricted ARMA models were selected to trade each spread. All coefficients were found to be significant at a 95% confidence interval. Therefore, the null hypothesis that all coefficients (except the constant) are not significantly different from zero is rejected at the 95% confidence interval.

ARMA Models	Equations
(10,10)	$Y_t = 5.22 \cdot 10^{-4} - 0.583Y_{t-1} + 0.458Y_{t-6} - 0.481Y_{t-9} - 0.568Y_{t-10} - 0.575\varepsilon_{t-1} + 0.450\varepsilon_{t-6} - 0.516\varepsilon_{t-9} - 0.595\varepsilon_{t-10}$ (56)
(8,8)	$Y_t = 4.79 \cdot 10^{-4} - 1.161Y_{t-1} - 0.208Y_{t-4} + 0.122Y_{t-8} - 1.067\varepsilon_{t-1} - 0.257\varepsilon_{t-4} + 0.159\varepsilon_{t-8}$ (57)
(13,13)	$Y_t = 3.91 \cdot 10^{-4} - 0.605Y_{t-1} + 0.503Y_{t-5} + 0.220Y_{t-13} - 0.603\varepsilon_{t-1} + 0.551\varepsilon_{t-5} + 0.222\varepsilon_{t-13}$ (58)
(4,4)	$Y_t = 4.33 \cdot 10^{-4} - 0.510Y_{t-1} + 0.109Y_{t-2} - 0.558Y_{t-3} - 0.891Y_{t-4} - 0.519\varepsilon_{t-1} + 0.113\varepsilon_{t-2} - 0.569\varepsilon_{t-3} - 0.953\varepsilon_{t-4}$ (59)
(12,12)	$Y_t = 4.80 \cdot 10^{-4} + 0.551Y_{t-1} - 0.699Y_{t-3} - 0.345Y_{t-7} - 0.177Y_{t-12} + 0.554\varepsilon_{t-1} - 0.709\varepsilon_{t-3} - 0.283\varepsilon_{t-7} - 0.171\varepsilon_{t-12}$ (60)

Table 72. ARMA equations

A.6.3 GARCH equations and estimations

Each of the GARCH models (16,16) and (15,15) are deemed stable and significant at a 95% confidence level. Following the initial estimation of significant terms a squared residuals test, JarqueBera test and an ARCH test are all conducted to test the reliability of the residuals. For the sake of brevity outputs from these tests are not included. These can be obtained on request from the

corresponding author. Autocorrelation is absent from both models and as a result returns derived from each model were used as inputs during the training of the proposed PSO-RBF Neural Network.

GARCH MODEL # 1 (16,16)

Dependent Variable: RETURNS
 Method: ML – ARCH
 Sample (adjusted): 4/22/2008 3/29/2012
 Included observations: 1028 after adjustments

Variable	Coefficient	Std. Error	z-Statistic	Prob.
C	0.001335	0.000660	2.023991	0.0430
AR(1)	-0.114106	0.016479	-6.924442	0.0000
AR(2)	-0.069719	0.013631	-5.114936	0.0000
AR(10)	-0.019143	0.007719	-2.480175	0.0131
AR(16)	-0.879046	0.013540	-64.92089	0.0000
MA(1)	0.120446	0.016908	7.123546	0.0000
MA(2)	0.065817	0.012455	5.284208	0.0000
MA(10)	-0.018657	0.008031	-2.323231	0.0202
MA(16)	0.897817	0.012306	72.95477	0.0000
Variance Equation				
C	1.42E-05	4.07E-06	3.478155	0.0005
RESID(-1)^2	0.083038	0.012924	6.425038	0.0000
GARCH(-1)	0.896013	0.014846	60.35321	0.0000
R-squared	0.046906	Mean dependent var		0.000433
Adjusted R-squared	0.039424	S.D. dependent var		0.027925
S.E. of regression	0.027369	Akaike info criterion		-4.622048
Sum squared resid	0.763277	Schwarz criterion		-4.564436
Log likelihood	2387.733	Hannan-Quinn criter.		-4.600181
Durbin-Watson stat	1.987245			

Table 73. GARCH model # 1

OBSERVATION:

The AR(1), AR(2), AR(10), AR(16), MA(1), MA(2), MA(10) and MA(16) terms are all deemed significant at a 95% confidence level. The model is also deemed stable due to the fact that the sum of GARCH(-1) and RESID(-1)^2 is less than 1. In this case it is, $0.896013 + 0.083038 = 0.979$.

GARCH MODEL # 2 (15,15)

Dependent Variable: RETURNS

Method: ML – ARCH
Sample (adjusted): 4/21/2008 3/29/2012
Included observations: 1029 after adjustments

Variable	Coefficient	Std. Error	z-Statistic	Prob.
C	0.001178	0.000697	1.691729	0.0907
AR(1)	-0.994472	0.026416	-37.64675	0.0000
AR(4)	-0.289867	0.024123	-12.01638	0.0000
AR(15)	0.153101	0.020739	7.382238	0.0000
MA(1)	1.016204	0.017594	57.75922	0.0000
MA(4)	0.322259	0.016612	19.39862	0.0000
MA(15)	-0.170083	0.014273	-11.91625	0.0000
Variance Equation				
C	1.75E-05	5.09E-06	3.437281	0.0006
RESID(-1)^2	0.091396	0.016350	5.589899	0.0000
GARCH(-1)	0.883565	0.020421	43.26752	0.0000
R-squared	0.005203	Mean dependent var		0.000431
Adjusted R-squared	-0.000638	S.D. dependent var		0.027911
S.E. of regression	0.027920	Akaike info criterion		-4.594528
Sum squared resid	0.796679	Schwarz criterion		-4.546556
Log likelihood	2373.885	Hannan-Quinn criter.		-4.576321
Durbin-Watson stat	2.051188			

Table 74. GARCH model # 2

OBSERVATION:

The AR(1), AR(4), AR(15), MA(1), MA(4), and MA(15) terms are all deemed significant at a 95% confidence level. The model is also deemed stationary due to the fact that the sum of GARCH(-1) and RESID(-1)^2 is less than 1. In this case it is, $0.883565 + 0.091396 = 0.9749$.

A.6.4 PSO parameters

Characteristics	380-day Sliding Window	500-day Sliding Window
Iterations	100	100
Number of Particles	30	30
Inertia Constant (w)	Adaptive	Adaptive
Cognitive Acceleration Constant (C ₁)	Adaptive	Adaptive
Social Acceleration Constant (C ₂)	Adaptive	Adaptive
Maximum Velocity	2/Number of Particles	2/Number of Particles
Number of Neurons (1 hidden layer)	10	10
Constant Size of Hidden Layer	Yes	Yes
Input Nodes	59	59
Output Nodes	1	1

Table 75. PSO RBF parameters

Parameters	RBF	MLP
------------	-----	-----

Learning algorithm	PSO	Gradient descent (Levenberg Marquardt variation)
Learning rate	Not Applicable	0.001
Momentum	Not Applicable	0.003
Iteration steps	100	5000
Initialisation of weights	No Initialization Required. Deterministic method for finding them.	N(0,1)
Input nodes	59	59
Hidden nodes (1 layer)	10	30
Output node	1	1

Table 76. Neural characteristics

A.6.5 Best weights over the training windows

MODEL 1		MODEL 2		MODEL 3		MODEL 4		MODEL 5	
Input Weights (59)	Neuron Weights (10)	Input Weights (59)	Neuron Weights (10)	Input Weights (59)	Neuron Weights (10)	Input Weights (59)	Neuron Weights (10)	Input Weights (59)	Neuron Weights (10)
1.9767	-0.0080	1.4888	0.8273	1.0584	-0.9852	0.8129	-0.0870	1.3663	0.0686
1.2275	-0.5489	1.5138	0.0075	1.2021	-0.1254	1.2912	0.1183	1.4555	0.0094
1.2147	0.2786	0.5293	0.2896	1.0792	-0.3226	0.8654	-0.0422	0.9676	-0.0908
1.2037	-0.9810	1.4406	-0.1087	1.4536	-0.1381	1.0240	-0.2030	1.4662	0.4919
0.5465	0.7071	1.5420	-0.4135	0.6496	1.1087	1.1924	-0.1213	1.1912	-0.4026
1.7877	-0.6872	1.4698	-1.0310	1.6624	0.3610	0.3727	-0.1485	0.8524	0.2388
1.2847	0.1213	0.2456	-0.5791	0.0439	0.6602	1.1430	-0.3494	0.6324	0.2551
0.9546	0.8774	1.1454	0.5787	1.7873	0.1058	1.2559	0.1910	0.4705	0.1141
1.3211	0.8775	0.5560	0.6667	0.9773	0.0443	0.9835	-0.1413	1.3414	-0.4315
1.6836	-0.7672	0.9511	-0.1383	0.7939	-0.5638	0.7293	0.8557	1.4916	-0.2457
0.7538	#N/A	1.2515	#N/A	1.2831	#N/A	0.9263	#N/A	0.2890	#N/A
1.1182	#N/A	0.8281	#N/A	0.3945	#N/A	1.4340	#N/A	0.8577	#N/A
1.3411	#N/A	0.6596	#N/A	1.6227	#N/A	0.6905	#N/A	0.6781	#N/A
1.0868	#N/A	1.1913	#N/A	1.6811	#N/A	1.0471	#N/A	0.8857	#N/A
0.1301	#N/A	0.8083	#N/A	0.3740	#N/A	1.7155	#N/A	0.3531	#N/A
1.3963	#N/A	1.5124	#N/A	0.4884	#N/A	1.2705	#N/A	0.9914	#N/A
1.5567	#N/A	1.6089	#N/A	0.7847	#N/A	0.6418	#N/A	0.7139	#N/A
1.5600	#N/A	1.1842	#N/A	1.3060	#N/A	0.6828	#N/A	1.0258	#N/A
1.4989	#N/A	1.3692	#N/A	0.6556	#N/A	0.9276	#N/A	0.9841	#N/A
0.6817	#N/A	0.3603	#N/A	1.3789	#N/A	0.9137	#N/A	0.5429	#N/A
1.1011	#N/A	1.6180	#N/A	1.0049	#N/A	0.9241	#N/A	1.5708	#N/A
0.2932	#N/A	0.4895	#N/A	1.4075	#N/A	1.0258	#N/A	0.7604	#N/A
1.2537	#N/A	0.6291	#N/A	0.0129	#N/A	1.0547	#N/A	0.5180	#N/A
1.4219	#N/A	1.4403	#N/A	0.2804	#N/A	1.4634	#N/A	1.2458	#N/A
1.4897	#N/A	0.8672	#N/A	1.3861	#N/A	0.9816	#N/A	1.0990	#N/A
1.6812	#N/A	0.6399	#N/A	1.6276	#N/A	1.1977	#N/A	0.9839	#N/A
1.4353	#N/A	1.2362	#N/A	0.9870	#N/A	0.8731	#N/A	1.4542	#N/A
1.9373	#N/A	1.2822	#N/A	0.4746	#N/A	1.2274	#N/A	0.5208	#N/A
0.7310	#N/A	0.9915	#N/A	1.6292	#N/A	0.9713	#N/A	1.2539	#N/A
1.0500	#N/A	1.7610	#N/A	0.9946	#N/A	0.7712	#N/A	0.4589	#N/A
0.5196	#N/A	1.1448	#N/A	0.8270	#N/A	1.1352	#N/A	0.8601	#N/A
1.1944	#N/A	0.3551	#N/A	1.0830	#N/A	1.6924	#N/A	0.4033	#N/A
1.7218	#N/A	1.5575	#N/A	1.2248	#N/A	0.5942	#N/A	1.5243	#N/A
1.4142	#N/A	0.3726	#N/A	0.4434	#N/A	0.9198	#N/A	1.1583	#N/A
1.0300	#N/A	0.4051	#N/A	0.3318	#N/A	1.7501	#N/A	0.5996	#N/A
1.3445	#N/A	0.4431	#N/A	0.4533	#N/A	0.9295	#N/A	0.9293	#N/A
1.2052	#N/A	0.5780	#N/A	0.8925	#N/A	1.3986	#N/A	0.9627	#N/A
1.5575	#N/A	0.4983	#N/A	0.8876	#N/A	1.2384	#N/A	0.5914	#N/A

1.5098	#N/A	0.6500	#N/A	1.4789	#N/A	1.0788	#N/A	0.3870	#N/A
0.5978	#N/A	1.2786	#N/A	0.5971	#N/A	0.7477	#N/A	1.0267	#N/A
0.6767	#N/A	1.5096	#N/A	1.5781	#N/A	1.1139	#N/A	1.0592	#N/A
1.3888	#N/A	0.8247	#N/A	0.8714	#N/A	1.3540	#N/A	0.9540	#N/A
0.9817	#N/A	1.3836	#N/A	0.9597	#N/A	1.8071	#N/A	0.9369	#N/A
0.8911	#N/A	0.4707	#N/A	1.4192	#N/A	0.4874	#N/A	1.1501	#N/A
1.2582	#N/A	1.0670	#N/A	1.3436	#N/A	0.5496	#N/A	1.1824	#N/A
1.2932	#N/A	1.3856	#N/A	0.6297	#N/A	1.1938	#N/A	1.1921	#N/A
0.2583	#N/A	0.7562	#N/A	0.6838	#N/A	1.5778	#N/A	1.1152	#N/A
1.0916	#N/A	0.5513	#N/A	0.8909	#N/A	1.0532	#N/A	0.9206	#N/A
2.0000	#N/A	1.3917	#N/A	0.2423	#N/A	1.2421	#N/A	0.5978	#N/A
1.3646	#N/A	1.0909	#N/A	1.1447	#N/A	0.9372	#N/A	0.9002	#N/A
0.5081	#N/A	1.2711	#N/A	0.6872	#N/A	1.2271	#N/A	0.7934	#N/A
0.3709	#N/A	1.2379	#N/A	1.1902	#N/A	0.8882	#N/A	1.1771	#N/A
0.5136	#N/A	1.5178	#N/A	1.5556	#N/A	1.7429	#N/A	0.6308	#N/A
1.0695	#N/A	0.5873	#N/A	1.0279	#N/A	0.9654	#N/A	1.0861	#N/A
1.1609	#N/A	0.5277	#N/A	0.6760	#N/A	0.7039	#N/A	1.0185	#N/A
0.6120	#N/A	1.5059	#N/A	1.6096	#N/A	1.1453	#N/A	1.4657	#N/A
1.5132	#N/A	1.3238	#N/A	0.9222	#N/A	0.7755	#N/A	0.6311	#N/A
1.3349	#N/A	1.3526	#N/A	1.4649	#N/A	1.2904	#N/A	0.1833	#N/A
0.6911	#N/A	0.4935	#N/A	0.3462	#N/A	0.7502	#N/A	0.6867	#N/A
1.8144	#N/A	0.6191	#N/A	1.0211	#N/A	0.9639	#N/A	1.5926	#N/A

Table 77. Best weights obtained from the 380-day training window

MODEL 1		MODEL 2		MODEL 3		MODEL 4		MODEL 5	
Input Weights (59)	Neuron Weights (10)	Input Weights (59)	Neuron Weights (10)	Input Weights (59)	Neuron Weights (10)	Input Weights (59)	Neuron Weights (10)	Input Weights (59)	Neuron Weights (10)
0.9070	0.4015	0.6819	-0.0703	0.9070	0.4015	1.7253	0.4609	0.8720	-0.1244
1.4054	-0.1953	1.6728	0.2570	1.4054	-0.1953	1.3730	0.1288	1.2579	-0.0285
0.7730	-0.1110	0.7241	-0.2934	0.7730	-0.1110	0.7039	0.1699	0.8418	1.0970
1.3742	-0.4231	1.2879	-0.1102	1.3742	-0.4231	0.5764	0.2313	0.8797	0.1393
0.9698	-0.0118	1.3060	-0.3200	0.9698	-0.0118	1.0941	-0.1848	1.4859	-0.7727
1.1820	-0.4267	0.7503	0.6308	1.1820	-0.4267	0.9311	-0.5983	0.6381	0.2812
0.9609	0.0749	0.8368	0.8377	0.9609	0.0749	1.0947	-0.1012	1.0521	-0.3009
1.1526	-0.1345	1.5200	-0.8396	1.1526	-0.1345	1.2402	-0.5187	0.7102	-0.3916
0.8540	-0.0502	0.2606	-0.0575	0.8540	-0.0502	0.9163	0.3221	0.9099	-0.6510
0.7939	0.8447	1.2288	0.0940	0.7939	0.8447	1.0475	0.0367	0.4763	0.5374
0.7716	#N/A	0.3574	#N/A	0.7716	#N/A	1.2223	#N/A	1.0691	#N/A
0.6642	#N/A	0.9626	#N/A	0.6642	#N/A	1.4063	#N/A	0.5522	#N/A
0.9845	#N/A	0.9560	#N/A	0.9845	#N/A	0.8503	#N/A	0.9638	#N/A
0.8801	#N/A	0.6218	#N/A	0.8801	#N/A	0.5076	#N/A	1.0708	#N/A
1.3758	#N/A	0.5021	#N/A	1.3758	#N/A	1.3177	#N/A	0.7855	#N/A
0.9750	#N/A	0.5827	#N/A	0.9750	#N/A	0.7341	#N/A	0.9721	#N/A
0.5542	#N/A	1.5309	#N/A	0.5542	#N/A	0.6713	#N/A	0.5005	#N/A
0.8604	#N/A	1.6899	#N/A	0.8604	#N/A	0.9526	#N/A	1.2341	#N/A
1.3104	#N/A	0.8265	#N/A	1.3104	#N/A	1.4329	#N/A	0.8003	#N/A
1.3563	#N/A	1.2091	#N/A	1.3563	#N/A	0.3215	#N/A	0.4206	#N/A
0.8165	#N/A	0.2437	#N/A	0.8165	#N/A	1.3214	#N/A	1.3452	#N/A
1.7920	#N/A	1.2195	#N/A	1.7920	#N/A	1.2343	#N/A	1.0293	#N/A
1.3503	#N/A	0.6166	#N/A	1.3503	#N/A	0.7596	#N/A	0.7727	#N/A
1.3684	#N/A	1.0831	#N/A	1.3684	#N/A	0.8708	#N/A	0.6222	#N/A
1.0686	#N/A	1.4177	#N/A	1.0686	#N/A	0.7813	#N/A	0.7203	#N/A
1.6583	#N/A	0.9189	#N/A	1.6583	#N/A	1.5476	#N/A	1.3443	#N/A
0.7123	#N/A	1.5404	#N/A	0.7123	#N/A	1.4377	#N/A	1.3139	#N/A
0.6076	#N/A	0.9581	#N/A	0.6076	#N/A	0.8825	#N/A	1.5451	#N/A

0.8656	#N/A	1.5559	#N/A	0.8656	#N/A	1.9318	#N/A	0.9862	#N/A
0.2273	#N/A	0.6753	#N/A	0.2273	#N/A	1.0459	#N/A	1.1460	#N/A
1.0345	#N/A	0.2068	#N/A	1.0345	#N/A	0.7238	#N/A	1.2063	#N/A
0.7623	#N/A	0.4673	#N/A	0.7623	#N/A	0.9009	#N/A	0.9245	#N/A
0.7128	#N/A	0.6491	#N/A	0.7128	#N/A	0.8554	#N/A	1.7615	#N/A
1.2197	#N/A	1.1203	#N/A	1.2197	#N/A	0.8877	#N/A	0.8165	#N/A
0.7839	#N/A	1.5416	#N/A	0.7839	#N/A	0.9554	#N/A	0.6599	#N/A
0.5538	#N/A	1.3938	#N/A	0.5538	#N/A	1.0397	#N/A	1.3623	#N/A
1.5902	#N/A	0.6421	#N/A	1.5902	#N/A	1.1515	#N/A	1.3136	#N/A
0.9899	#N/A	0.6421	#N/A	0.9899	#N/A	0.0879	#N/A	0.9088	#N/A
0.8756	#N/A	1.7203	#N/A	0.8756	#N/A	0.4917	#N/A	0.5814	#N/A
0.3445	#N/A	0.5204	#N/A	0.3445	#N/A	1.3025	#N/A	1.1367	#N/A
0.9331	#N/A	1.7534	#N/A	0.9331	#N/A	0.5988	#N/A	0.7868	#N/A
0.9731	#N/A	0.6174	#N/A	0.9731	#N/A	0.9355	#N/A	1.2207	#N/A
0.9116	#N/A	0.8339	#N/A	0.9116	#N/A	0.2439	#N/A	1.1726	#N/A
0.7557	#N/A	0.9006	#N/A	0.7557	#N/A	1.3068	#N/A	1.0262	#N/A
1.5539	#N/A	0.8249	#N/A	1.5539	#N/A	1.5354	#N/A	1.0560	#N/A
0.3683	#N/A	0.3886	#N/A	0.3683	#N/A	1.0518	#N/A	0.8511	#N/A
0.5836	#N/A	1.0118	#N/A	0.5836	#N/A	0.8430	#N/A	1.1025	#N/A
1.6273	#N/A	0.4865	#N/A	1.6273	#N/A	0.8351	#N/A	0.9436	#N/A
1.4291	#N/A	0.6186	#N/A	1.4291	#N/A	0.7076	#N/A	0.3921	#N/A
1.1770	#N/A	0.2630	#N/A	1.1770	#N/A	0.5120	#N/A	1.3862	#N/A
0.6836	#N/A	1.2072	#N/A	0.6836	#N/A	0.7497	#N/A	1.0211	#N/A
1.4776	#N/A	1.4199	#N/A	1.4776	#N/A	1.4651	#N/A	1.2946	#N/A
0.3409	#N/A	1.0153	#N/A	0.3409	#N/A	1.1771	#N/A	1.2813	#N/A
1.1485	#N/A	1.2068	#N/A	1.1485	#N/A	0.3835	#N/A	1.5403	#N/A
0.9180	#N/A	0.3319	#N/A	0.9180	#N/A	0.5203	#N/A	0.8887	#N/A
1.2806	#N/A	0.6448	#N/A	1.2806	#N/A	0.8930	#N/A	0.5769	#N/A
0.8480	#N/A	1.5232	#N/A	0.8480	#N/A	0.6923	#N/A	1.4762	#N/A
1.3163	#N/A	1.0349	#N/A	1.3163	#N/A	1.2372	#N/A	0.2311	#N/A
0.6303	#N/A	1.2965	#N/A	0.6303	#N/A	1.2530	#N/A	1.3228	#N/A
0.7381	#N/A	0.5941	#N/A	0.7381	#N/A	1.0344	#N/A	1.3092	#N/A

Table 78. Best weights obtained from the 500-day training window

CHAPTER 10: References

- Ang, A., Gorovyy, S., and Van Inwegen, G.B. (2010). *Hedge Fund Leverage, Working Paper*. Columbia University.
- Appel, G. (1979). *The Moving Average Convergence-Divergence Method*. Great Neck, NY: Signalert.
- Aranha, C.C., and Iba, H. (2008). "A Tree-Based GA Representation for The Portfolio Optimization Problem." *In Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation, GECCO '08, Atlanta, GA, USA, July 12 - 16, 2008*, edited by M. Keijzer, pp. 873-880. New York, NY: ACM.
- Andersen, T. G., Bollerslev, T., and Diebold, F. X. (2003b) *Some Like It Smooth, and Some Like It Rough: Untangling Continuous and Jump Components in Measuring, Modelling and Forecasting Asset Return Volatility, Manuscript*, Northwestern University, Duke University and University of Pennsylvania.
- Bessimber, H., and Chan, K. (1995) "The Profitability of Trading Rules in the Asian Stock Markets", *Pacific Basin Finance Journal*, 3, pp.257-284.
- Beja, A., and Goldman, M. (1980) "On the Dynamic Behaviour of Prices in Disequilibrium", *Journal of Finance*, 34, pp.235-247.
- Brinker, A.J., Parcell, J.L., Dhuyvetter, K.C. and Franken, J. (2009). "Cross-Hedging Distillers Dried Grains: Corn and Soybean Meal Futures Contracts." *Journal of Agribusiness*, 27(1/2).
- Butterworth, D., and Holmes, P. (2002) "Inter-Market Spread Trading: Evidence from UK Index Futures Markets." *Applied Financial Economics*. 12(11), pp.783-791.

- CBOT (2007) "Ethanol Futures – Corn Crush Reference Guide." Accessed April, 2010.
<http://www.cbot.com/cbot/docs/73511.pdf>.
- Chan, E. (2009) *Quantitative Trading: How to Build Your Own Algorithmic Trading Business*. New Jersey: John Wiley & Sons, Inc.
- Chang, T.J., Meade, N., Beasley, J., and Sharaiha, Y. (2000) "Heuristics for Cardinality Constrained Portfolio Optimisation." *Computers and Operations Research*. 27, pp. 1271-1302.
- Chang, P.C., Wang, Y.W., and Yang, W.N. (2004) "An Investigation of the Hybrid Forecasting Models for Stock Price Variation in Taiwan." *Journal of the Chinese Institute of Industrial Engineering*. 21(4), pp. 358–368.
- Chen, L.H.C., Finney, M., and Lai, K.S. (2005) "A Threshold Cointegration Analysis of Asymmetric Price Transmission from Crude Oil to Gasoline Prices." *Economics Letters*, 89, pp. 233-239.
- Chen, Z., and Qian, P. (2009) "Application of PSO-RBF Neural Network in Network Intrusion Detection." *Intelligent Information Technology Application*. pp. 362- 364.
- Clemen, R.T. (1989) "Combining Forecasts: A Review and Annotated Bibliography." *International Journal of Forecasting*, 5, pp. 559-583.
- CME (2010) "Commodity Products - Trading The Corn for Ethanol Crush." Accessed April, 2010.
http://www.cmegroup.com/trading/agricultural/files/AC-406_DDG_CornCrush_042010.pdf.
- Corsi, F. (2003) *A Simple Long Memory Model of Realized Volatility, Manuscript*, University of Southern Switzerland.
- Cortez, P., Rocha, M., and Neves, J. (2001) "Evolving Time Series Forecasting Neural Network Models." *In Proceedings of the 3rd Int. Symposium on Adaptive Systems: Evolutionary Computation and Probabilistic Graphical Models (ISAS 2001)*, pp. 84–91

- Dahlgran, R. A. (2009) "Inventory and Transformation Hedging Effectiveness in Corn Crushing" *Journal of Agricultural and Resource Economics*. 34(1), pp.154-171.
- Deaton, A., and Laroque, G. (1992) "On the Behaviour of Commodity Prices." *Review of Economic Studies*, 59, pp.1-23.
- Dickey, D., and Fuller, W. (1979) "Distribution of the Estimators for Autoregressive Time Series with a Unit Root," *Journal of the American Statistical Association*, 74, pp. 427-431.
- Donaldson, R.G., and Kamstra, M. (1997) "An Artificial Neural Network—GARCH Model for International Stock Return Volatility." *Journal of Empirical Evidence*. 4, pp. 17–46.
- Draye, J.S., Pavisic, D.A., Cheron, G.A., and Libert, G.A. (1996) "Dynamic Recurrent Neural Networks: A Dynamic Analysis." *IEEE Transactions SMC – Part B*. 26(5), pp. 692-706.
- Dunis, C.L. (1989) *Computerised Technical Systems and Exchange Rate Movements*, in C. Dunis and M. Feeny [eds.]. *Exchange Rate Forecasting*. Probus Publishing Company, Cambridge, UK, pp. 165 – 205.
- Dunis, C.L. (1996) "The Economic Value of Neural Network Systems for Exchange Rate Forecasting." *Neural Network World*, 1, pp. 43-55.
- Dunis, C. L., Laws, J., and Evans, B. (2005) "Modelling and Trading the Gasoline Crack Spread: A Non-Linear Story", *Derivatives Use, Trading & Regulation*, 12(1-2), pp. 126-145.
- Dunis, C. L., and Miao, J. (2006) "Volatility Filters for Asset Management: An Application to Managed Futures", *Journal of Asset Management*, 7(3-4), pp.179-189.
- Dunis, C. L., Laws, J., and Evans, B. (2006-a) "Trading Futures Spreads: An Application of Correlation and Threshold Filters", *Applied Financial Economics*, 16(12), pp.903-914.
- Dunis, C. L., Laws, J., and Evans, B. (2006-b) "Modelling and Trading the Soybean-Oil Crush Spread with Recurrent and Higher Order Networks: A Comparative Analysis", *Neural Network World*, 13(3/6), pp.193-213.

- Dunis, C., Laws, J., and Evans, B. (2010) "Trading and Filtering Futures Spread Portfolios: Further Applications of Threshold and Correlation Filters." *Journal of Derivatives and Hedge Funds*, (15), pp. 274-287.
- Dunis, C.L., Laws, J., and Karathanassopoulos, A. (2011) "Modelling and Trading the Greek Stock Market with Mixed Neural Network Models." *Applied Financial Economics*. 21(23), pp. 1793 – 1808.
- Dunis, C.L., Sermpinis, G., and Karathanasopoulos, A. (2011a). "Forecasting and Trading the EUR/USD Exchange Rate with Gene Expression and Psi Sigma Neural Networks", *CIBEF Working Paper*, Liverpool Business School, available at www.cibef.com.
- Dunis, C. L., Laws, J. and Middleton, P.W. (2011b). "Modelling and Trading the Corn/Ethanol Crush Spread with Neural Networks", *CIBEF Working Paper*, Liverpool Business School, available at www.cibef.com.
- Dunis, C. L., Laws, J., and Middleton, P.W. (2012) "Non-Linear Forecasting of the Gold Miner Spread: An Application of Correlation Filters", *CIBEF Working Paper*, Liverpool Business School, available at www.cibef.com.
- Eberhart, R., Simpson, P., and Dobbins, R. (1996) *Computational Intelligence PC Tools*. San Diego, CA: Academic Press Professional.
- Elliott, R., Van der Hoek, J., and Malcolm, W. (2005) "Pairs Trading", *Quantitative Finance*, 5(3), pp. 271-276.
- Elman, J. L. (1990) "Finding Structure in Time", *Cognitive Science*, 14, pp.179-211.
- Enders, W., and Granger, C. W. J. (1998) "Unit-Root Tests and Asymmetric Adjustment with An Example Using the Term Structure of Interest Rates." *Journal of Business and Economic Statistics*, 16(3), pp. 304-311.

- Engle, R. F., and Granger, C.W.J. (1987) “Co-integration and Error-Correction: Representation, Estimation and Testing.” *Econometrica*, 55, pp. 251-276.
- Fama, E., and French, K. (1986) “Permanent and Temporary Components of Stock Prices”. *Journal of Political Economy*, 98, pp.246-274.
- Faraway, J., and Chatfield, C. (1998) “Time Series Forecasting with Neural Networks: A Case Study”. *Applied Statistics*, 47, pp.231-250.
- Ferreira, C. (2006). *Gene Expression Programming: Mathematical Modelling by an Artificial Intelligence*, Springer, San Francisco.
- Franken, J. R. V., and Parcell, J. L. (2003) “Cash Ethanol Cross-Hedging Opportunities”. *Journal of Agricultural and Applied Economics*, 35(3), pp.1-19.
- Freeman, J. A. (1994) *Simulating Neural Networks with Mathematica*. Addison-Wesley: Reading, MA.
- Fulcher, J., Zhang, M., and Xu, S. (2006) *The Application of Higher-Order Neural Networks to Financial Time Series, Artificial Neural Networks in Finance and Manufacturing*, Hershey, PA: Idea Group, London.
- Funk, S. M., Zook, J. E., and Featherstone, A. M. (2008) “Chicago Board of Trade Ethanol Contract Efficiency”. *Selected Paper prepared for a presentation at the Southern Agricultural Economics Association Annual Meeting*, Dallas, TX, February 2-6, 2008. Accessed April, 2010. <http://ageconsearch.umn/bitstream/6811/2/sp08fu01.pdf>
- Gallagher, P.W. (2009). “Roles for Evolving Markets, Policies, and Technology, Improvements in U.S. Corn Ethanol Industry Development”. *Federal Reserve Bank of St. Louis Regional Economic Development*. 5(1) pp. 12-33.

- Gatev, E., Goetzmann, W. N., and Rouwenhorst, K. G. (2000) *Pairs Trading: Performance of a Relative Value Arbitrage Rule*. Yale School of Management Working Paper. New Haven, Connecticut: Yale University.
- Geman, H. (2005) *Commodities and Commodity Derivatives*. Chichester: John Wiley & Sons.
- Giles, L., and Maxwell, T. (1987) "Learning, Invariance and Generalization in Higher Order Neural Networks". *Applied Optics*, 26, pp. 4972-4978.
- Haschke, R., and Steil, J. J. (2005) "Input Space Bifurcation Manifolds of Recurrent Neural Networks." *Neurocomputing*, 64C, pp.25-38.
- Herlemont, D. (2003) *Pairs Trading, Convergence Trading, Cointegration*, YATS Finance and Technologies.
- Hill, T., O'Conner, M., and Remus, W. (1996) "Neural Network Models for Time Series Forecasts", *Management Science*. 42(7), pp.1082-1092.
- Holland, J. (1975). *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: University of Michigan Press.
- Hornik, K., Stinchcombe, M., and White, H., (1989) "Multilayer Feedforward Networks are Universal Approximators," *Neural Networks*. 2, pp.359-366.
- Hutchinson, J.M. (1994) *A Radial Basis Function Approach to Financial Time Series Analysis*. PhD dissertation, Massachusetts Institute of Technology, Cambridge, MA.
- 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), pp. 851-889.
- Iba, H. (1999) "Bagging, Boosting and Bloating in Genetic Programming," In *Proceedings of the Genetic and Evolutionary Computation Conference, (GECCO'99)*. Banzhaf W, Daida J, Eiben AE, Garzon MH, Honavar V, Jakiela M, Smith RE (eds). Morgan Kaufmann: San Francisco, CA, pp. 1053-1060.

- Jegadeesh, N., and Titman, S. (1993) "Returns to Buying Winners and Selling Losers: Implications for Stock Market Efficiency". *Journal of Finance*, 48(1).
- Jayawardena, A.W., and Fernando, D.A.K. (1995) *Artificial Neural Networks in Hydrometeorological Modelling*. In *Proceedings. Fourth Int. Conf. on the Application of Artificial Intelligence to Civil and Structural Engineering*, B. H. V. Topping [eds.], Civil Comp Press: Cambridge, UK, pp.115-120.
- Jayawardena, A. W., Fernando, D. A. K., and Zhou, M. C. (1996). *Comparison of Multilayer Perceptron and Radial Basis Function Networks as Tools for Flood Forecasting*, in: *Destructive Water: Water-Caused Natural Disaster, their Abatement and Control*. International Association of Hydrological Sciences Press: Oxfordshire; pp.173–182.
- Jayawardena, A.W., and Fernando, D. A. K. (1998) "Use of Radial Basis Function Type Artificial Neural Networks for Runoff Simulation." *Computer-Aided Civil and Infrastructure Engineering*. 13(2), pp. 91–99.
- Jin, X., Yoon, C.Y., and Yan, J. (2008) "State-Space Modeling of Mean-Reverting Spreads For PairsTrading". *Working Paper*, Accessed March, 2010. <http://www.seekalpha.net/pdf/mcmc.pdf>.
- Johansen, S. (1988) "Statistical Analysis of Cointegration Vectors," *Journal of Economic Dynamics and Control*, 12(2–3), pp. 231–254.
- J.P. Morgan (1997) *RiskMetrics Technical Document*. Morgan Guaranty Trust Company, New York.
- Kaastra, I., and Boyd, M. S. (1995) "Forecasting Futures Trading Volume Using Neural Networks." *The Journal of Futures Markets*. 16(8), pp.953 – 970.
- Kaastra, I., and Boyd, M. (1996) "Designing a Neural Network for Forecasting Financial and Economic Time Series." *Neurocomputing*.10(3), pp.215.

- Karayiannis, N., and Venetsanopoulos, A. (1994) "On the Training and Performance of High-Order Neural Networks", *Mathematical Biosciences*, 129, pp.143-168.
- Kennedy, J and Eberhart, R. (1995) "Particle Swarm Optimization". *In Proceedings of the IEEE International Conference on Neural Networks*, 4, pp.1942-1948.
- Kim, P., Pan, L., and Wirjanto, T. S. (1999): *Neural Network Models of the Spot Canadian/U.S. Exchange Rate*, Mimeo, University of Waterloo, Waterloo, Ontario, Canada.
- King, K., and Zulauf, C. (2010) "Are New Crop Futures and Option Prices for Corn and Soybeans Biased? An Updated Appraisal." Paper presented at the NCCC-134 Conference on Applied Commodity Price Analysis, Forecasting, and Market Risk Management, St. Louis, MO, pp. 19–20.
- Klement, J. (2005) "Riding the Waves of Investment Returns", *UBS Wealth Management Research Working Paper* 01-2005.
- Knowles, A.A., Hussein, W., Deredy, P., Lisboa, and Dunis.C.L. (2005). *Higher-Order Neural Networks with Bayesian Confidence Measure for Prediction of EUR/USD Exchange Rate*. Artificial Higher Order Neural Networks for Economic and Business, 1, pp. 48–59. *Working Paper*. Available at www.cibef.com
- Konstantinos, T., Parsopolous, E., and Vrahatis, M.N. (2010) "Particle Swarm Optimization and Intelligence: Advances and Applications." *IGI Global*, pp. 149 – 164.
- Koza, J.R. (1992), *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT Press, Cambridge, Massachusetts.
- Koza, J.R. (1998) *Genetic Programming*. In *Encyclopaedia of Computer Science and Technology*, Williams J.G, Kent, A (eds). Marcel-Dekker: New York, NY; pp. 29–43.
- Lachtermacher, G., and Fuller, J. D. (1995) "Backpropagation in Time Series Forecasting". *Journal of Forecasting*, 14(26), pp.381-393.

- Lan, Y., Wang, N., and Yang, J. (2011) *The Economics of Hedge Funds: Alpha, Fees, Leverage, and Valuation. Working Paper.* Columbia University and Hunan University.
- Lawrenz, C., and Westerhoff, F. (2003) "Modeling Exchange Rate Behavior with a Genetic Algorithm." *Computational Economics*. 21, pp. 209-229.
- Li, J., and Tsang, E.P.K. (1999). "Improving Technical Analysis Predictions: An Application of Genetic Programming." *In Proceedings of The 12th International Florida AI Research Society Conference*, pp. 108 -112.
- Li, J., and Xiao, X (2008) "Multi-Swarm and Multi-Best Particle Swarm Optimization Algorithm". *In Proceedings of the 7th World Congress on Intelligent Control and Automation, China.* pp. 6281-6286.
- Lim, G., and V. Martin (1995) "Regression-based Cointegration Estimators". *Journal of Economic Studies*, 22 (1), pp. 3-22.
- Lin, D., Wang, S., and Yan, H. (2001) "A Multi-objective Genetic Algorithm for Portfolio Selection Problem." *In Proceedings of ICOTA 2001.* Hong Kong, December pp.15-17.
- Lippman, R. (1987) "An Introduction to Computing with Neural Nets."s *In Proceedings of IEEE ASSP Magazine*, pp. 4-22.
- Lisi, F., and Schiavo, R. A. (1999) "A Comparison between Neural Networks and Chaotic Models for Exchange Rate Predictions". *Computational Statistics and Data Analysis*. 30, pp.87-201.
- MacKinnon, J. G. (1996), "Numerical Distribution Functions for Unit Root and Cointegration Tests," *Journal of Applied Econometrics*, 11, pp. 601–618.
- MacKinnon, J. G., Haug, M. A., and Michelis, L. (1999) "Numerical Distribution Functions of Likelihood Ratio Tests for Cointegration" *Journal of Applied Econometrics*, 14, pp. 563-577
- Makridakis, S. (1989) "Why Combining Works?" *International Journal of Forecasting*. 5, pp. 601-603.

- McKay, P.A. (2006) “Demand for Ethanol Aggravates Pain at the Pump.” *Wall Street Journal*, Eastern Edition, June 19, C4.
- Medsker, L., Turban, E., and Trippi, R. (1993) *Neural Network Fundamentals for Financial Analysts*. In: Trippi R, Turban E. editors. *Neural Networks in Finance and Investing*, Chicago: Probus Publishing.
- Meland, L. (1981) “Futures Market Liquidity and the Technique of Spreading”. *Journal of Futures Markets*. 1, pp.405-411
- Mettenheim, V.H.J., and Breitner, M.H. (2012) “Forecasting Daily Highs and Lows of Liquid Assets with Neural Networks.” *Operations Research Proceedings, Selected Papers of the Annual International Conference of the German Operations Research Society*, Hannover.
- Mohaghegi, S., Valle, Y., Venayagamoorthy, G., and Harley, R. (2005) “A Comparison of PSO and Backpropagation for Training RBF Neural Networks for Identification of a Power System with Statcom.” *In Proceedings of IEEE Swarm Intelligence Symposium*, pp. 381-384.
- Neely, C.J., Weller, P.A., and Dittmar, R. (1997). “Is Technical Analysis Profitable in the Foreign Exchange Market? A Genetic Programming Approach.” *Journal of Financial and Quantitative Analysis*. 32 (4): pp. 405–436.
- Newbold, P., and Granger, C.W.J. (1974) “Experience with Forecasting Univariate Time Series and the Combination of Forecasts (with discussion).” *Journal of Statistics*, 137, pp. 131-164.
- Niaki S.T.A, and Hoseinzade, S. (2013) “Forecasting S&P 500 Index Using Artificial Neural Networks and Design of Experiments.” *Journal of Industrial Engineering International*. 9(1): article 1.
- Nicholas, J.G. (2000) *Market Neutral Investing— Long/Short Hedge Fund Strategies*. Bloomberg Professional Library, Bloomberg Press: Princeton, NJ, USA.

- Ormoneit D., and Neuneier R. (1996) “Experiments in Predicting the German Stock Index DAX with Density Estimating Neural Networks. “ In *Proceedings of the 1996 Conference on Computational Intelligence Finance Engineering (CIFER)*. pp. 66–71.
- Palm, F.C., and Zellner, A. (1992) “To Combine or Not to Combine? Issues of Combining Forecasts.” *Journal of Forecasting*. 11, pp. 687-701.
- Park, J.W., Harley, R.G., and Venayagamoorthy, G.K. (2002) “Comparison of MLP and RBF Neural Networks Using Deviation Signals For On-line Identification of a Synchronous Generator,” *In Proceedings of Power Engineering Society Winter Meeting, IEEE*, 1 (27–31), pp. 274–279.
- Perron, P (1989) “The Great Crash, the Oil Price Shock, and The Unit Root Hypothesis”. *Econometrica*. 57, pp.1361-1401.
- Peterson, R. (1977) “Investor Preferences for Futures Straddles”. *Journal of Financial and Quantitative Analysis*, 12, pp.105-120.
- Phillips, P.C.B., and P. Perron (1988). “Testing for Unit Roots in Time Series Regression,” *Biometrika*, 75, pp.335-346.
- Pindyck, R., and Rubinfeld, D. (1998) *Econometric Models and Economic Forecasts*, 4th Edition, McGraw-Hill, New York.
- RFA (Renewable Fuels Association). (2011). *Building Bridges to a More Sustainable Future: 2011 Ethanol Industry Outlook*. Washington, DC: RFA, p. 26.
- Rumelhart, D. E., Hinton, G.E., and Williams, R.J. (1986) *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, in: *Rumelbart, D.E and McClelland, J.L, Eds., Parallel Distributed Processing*, MIT Press, Cambridge, Massachusetts. 1, pp.318-362.
- Santini, M., and Tattamanzi, A. (2001) “Genetic Programming for Financial Time Series Prediction.” *In Proceedings of Genetic Programming, Euro GP’2001, LNCS*, 2038, pp. 361-370.

- Sermpinis, G., Theofilatos, K., Karathanasopoulos, A., Georgopoulos, E., and Dunis, C.L. (2013) "Forecasting Foreign Exchange Rates with Adaptive Neural Networks Using Radial-Basis Functions and Particle Swarm Optimization." *European Journal of Operational Research*, 225(3), pp. 528-540.
- Shapouri, H., Duffield, J. A., and Graboski, M. S. (1995) "Estimating the Net Energy Balance of Corn Ethanol." Agr. Econ. Rep. No.721, USDA/Economic Research Service, Office of Energy, Washington, DC. Accessed February, 20 <http://www.ers.usda.gov/publication/aer721/AER721>
- Shapouri, H., Duffield, J. A., and Wang, M. Q. (2002) "The Energy Balance of Corn Ethanol: An Update", *U.S. Department of Agriculture, Economic Research Service*, AER-813.
- Sopena, J.M., Romero, E., and Alquezar, R. (1999) "Neural Networks with Periodic and Monotonic Activation Functions: A Comparative Study in Classification Problems," *In Proceedings of 9th International Conference in Artificial Neural Networks*, 1, pp.323-328.
- Sorensen, C. (2002) "Modelling Seasonality in Agricultural Commodity Futures". *Journal of Futures Markets*, 22(5), pp.393-426.
- Swales, G., and Yoon, Y. (1992). "Applying Artificial Neural Networks to Investment Analysis." *Financial Analysts Journal*, pp. 78-80.
- Sweeney, R. J. (1988) "Some New Filter Rule Tests Methods and Results". *Journal of Financial and Quantitative Analysis*. 23, pp.285-300.
- Tam, K. Y., and Kiang, M. (1992) "Managerial Applications of Neural Networks: The Case of Bank Failure Predictions." *Management Science*, 38(7), pp. 926-947.
- Tenti, P. (1996) "Forecasting Foreign Exchange Rates Using Recurrent Neural Networks", *Applied Artificial Intelligence*, 10, pp.567-581.
- Thawornwong, S., and Enke, D. (2004) "The Adaptive Selection of Financial and Economic Variables for use with Artificial Neural Networks." *Neurocomputing*. 56, pp.205–232.

- Till, H., and Eagleeye, J. (2004) *How to Design a Commodity Futures Trading Program*, in: Greg Gregoriou et al [eds.], *Commodity Trading Advisors: Risk, Performance Analysis and Selection*, Hoboken, NJ: Wiley Finance.
- Tino, P., Horne, B.G., and Giles, C.L. (2001) "Attractive Periodic Sets in Discrete-Time Recurrent Neural Networks (with Emphasis on Fixed-Point Stability and Bifurcations in Two-Neuron Networks)." *Neural Computation*. 13, pp.1379-1414.
- Triantafyllopoulos, K and Montana, G. (2009) "Dynamic Modelling of Mean-Reverting Spreads for Statistical Arbitrage." *Computational Management Science*. 8(1-2), pp. 23-49.
- Trippi, R., and DeSieno, D. (1992) "Trading Equity Index Futures with a Neural Network". *Journal of Portfolio Management*. 19, pp.27-34.
- Tsai, C.F., and Wang, S.P. (2009) "Stock Price Forecasting by Hybrid Machine Learning Techniques." In *Proceedings of the International Multi-Conference of Engineers and Computer Scientists*. 1, pp.755-760.
- Vanstone, B., Finnie, G., and Hahn, T. (2010) "Stock Market Trading Using Fundamental Variables and Neural Networks," In *Proceedings of ICONIP 2010: 17th International Conference on Neural Information Processing*.
- Vidyamurthy, G. (2004) *Pairs Trading, Quantitative Methods and Analysis*, John Wiley & Sons, Canada.
- White, H. (1989) "Learning in Artificial Neural Networks: A Statistical Perspective," *Neural Computation*, 1, 425-464.
- Working, H. (1949) "The Theory of Price of Storage". *American Economic Review*. 39, pp.1254-1262.
- Working, H. (1953) "Futures Trading and Hedging". *American Economic Review*. 43, pp.314-343.

Working, H. (1954) "Whose Markets? Evidence on Some Aspects of Futures Trading". *Journal of Marketing*, 29, pp.1-11.

Working, H. (1960) "Speculation on Hedging Markets." *Food Research Institute Studies*,1, pp.185-220.

Working, H. (1962) "New Concepts Concerning Futures Markets and Prices." *American Economic Review*. 62, pp.432-459.

Zhang, G. P., and Qi, M. (2005) "Neural Network Forecasting for Seasonal and Trend Time Series", *European Journal of Operational Research*, 160 (2), pp.501-514.