**Trading and Hedging the Corn/Ethanol Crush Spread**

**Using Time Varying Leverage and Nonlinear Models**

**by**

**Christian L. Dunis\***

**Jason Laws\*\***

**Peter W. Middleton\*\*\***

**Andreas Karathanasopoulos\*\*\*\***

***(Liverpool Business School,***

***CIBEF Liverpool John Moores University)***

***Abstract***

In contribution to Dunis *et al.* (2011b) we endeavour to expand the selection of forecasting applications by delving further into the realm of artificial intelligence and non-linear modelling. Therefore, 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 our spread while avoiding trading completely during times of high volatility.

In this paper we model 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 as corn is 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. Furthermore, once our 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 it reduces maximum drawdowns, two desirable outcomes for trading and hedging purposes.

***Keywords***

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

***\*Christian L Dunis*** *is Professor of Banking and Finance at Liverpool Business School and Director of the Centre for International Banking, Economics and Finance (CIBEF) at Liverpool John Moores University (E-mail:* *cdunis@tiscali.co.uk**).*

***\*\*Jason Laws*** *is Reader of Finance at Liverpool Business School and a member of CIBEF (E-mail:* *J.Laws@ljmu.ac.uk**).*

***\*\*\*Peter Middleton – Corresponding Author –*** *is an associate researcher with CIBEF and is currently a Reader of Finance at Liverpool Business School. (E-mail:* *peter.william.middleton@gmail.com**).*

***\*\*\*\*Andreas Karathanasopoulos*** *is a Senior Lecturer in Finance at London Metropolitan University (E-mail: karathanasopoulos.a@londonmet.ac.uk )*

## **1 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 recent rally of commodity prices can be associated with varying political agendas, government policies, growing populations in China and India, and the unrelenting pressures 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 lead 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 against adverse price movements. On the other hand, speculators are also drawn to these markets primarily to make profits and / or to 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 volatile price swings in 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 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 2007. The future outlook of the ethanol market appear to be extremely prosperous with the Energy Independence and Security Act (EISA) of 2007 Energy Act being passed 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 potential 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 $147.27 a barrel it became apparent that alternative cheaper bio fuels are essential. More recently, political instability in the Middle East has seen oil prices rise significantly affecting the demand for cheaper alternative fuels. These more recent prices rises although high in relation to industry averages over the last few years are yet to breach the previous record high mentioned above.

In particular, growth of the Ethanol industry can be attributed to two major changes in the fuel markets. As highlighted by Gallagher (2009), the ban of MTBE (Methyl Tertiary Butyl Ether) and the record high levels of petroleum prices have lead to a significant increase in demand and supply of Ethanol as an alternative bio fuel. The world ethanol market is dominated by the United States and Brazil who are the largest ethanol producing and exporting nations. In comparison the type of ethanol produced in each differ due to the raw ingredients used to produce the ethanol. Brazil uses sugarcane to produce ethanol whereas the United States uses corn as a basis. Sugarcane based ethanol is more competitively priced due to it being cheaper to produce whereas corn based ethanol is slightly more expensive. Our focus is however on the United States ethanol market which is extremely fragmented with only a handful of players occupying its market space. Among these few, Archer Daniels Midland (ADM) is the largest agricultural processor in the world generating a net turnover of $1.7 billion for the fiscal year ended June 30, 2009, see Kang (2010).

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 their daily closing prices with the application of 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 way. 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 5 and 6 present the results and final remarks respectively. An appendix presents relevant tables, figures and estimation parameters to conclude the research.

## **2 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 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. In his research 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 reduction capabilities’ akin to those experienced by soybean processors who utilise the soybean crush hedge. 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. Therefore, 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) explore another grain spread known as the soybean-oil crush spread. In this investigation they analyse the profitability of trading a soybean-oil spread over a duration of 11 years spanning from 01/01/1995 to 01/01/2005. Additionally, they test the effectiveness of various neural network architectures against more conventional forecasting techniques such as the fair value co-integration model. From their analysis they conclude that profitability is in fact 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 forecasting and trading the soybean-oil spread.

Dunis *et al.* (2011b) also extol the virtues of using neural networks for hedging and trading purposes by evaluating the Corn Ethanol spread utilising MLP, RNN and HONN models. They observe a period of 5 years commencing 23/03/2005 until 31/12/09. 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) beats all other models as it generates an annualised return of 27.32%. This unfiltered model was then improved to generate 30.46% annualised returns when subjected to a RiskMetrics volatility market timing filter. Using this filter, trading is ceased once an optimal level of RiskMetrics volatility is breached. More importantly, overall model volatilities and maximum drawdowns were also improved as a result of the volatility trading filter.

## **3 THE CORN/ETHANOL SPREAD AND RELATED FINANCIAL DATA**

The daily closing prices (13:15 CST[[1]](#footnote-1)) for each of the commodity contract months were obtained from Datastream for the period covering March 23, 2005 - December 31st, 2010. Corn futures[[2]](#footnote-2) are the most heavily traded agricultural commodity and have been traded on the CBOT[[3]](#footnote-3) exchange since the mid 1800’s. On the other hand, ethanol futures[[4]](#footnote-4) 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.

The Corn ‘Crush’ spread is calculated taking into consideration the fact that both commodity futures contracts are priced and traded in different units. 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. Thus, one bushel of corn yields approximately 2.8 gallons of Ethanol (CME 2010). Hence, 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 price/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:

Ct = PC – [(2.8 \* PE)/100] (1)

Where: Ct = Priceof the crush spread at time t (in cents per bushel)

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

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

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 we have decided to focus our analysis on the relationship between Corn and Ethanol.

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):



 (2)

Where: = Percentage returns of spread at time t.

PC*(t)* = is the price of corn at time t (in cents per bushel)

PC*(t-1)* = is the price of corn at time t -1 (in cents per bushel)

PE (t) = is the price of ethanol at time t (in cents per bushel)

PE (t-1) = is the price of ethanol 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 1. The Corn-Ethanol Crush CBOT Daily Closing Prices (23/03/2005 – 31/12/2010)

By observation, it is apparent that the spread is mean reverting with large deviations reflected as a consequence of prices rallies. These large deviations are associated with shocks experienced during this period that resulted in drastic price increases. 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 devastating the South and Midwest of the United States disrupting the harvest season for that summer. 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). As a result, 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 previously 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 lead to similar price rises in ethanol and corn. This cannot be seen in the above graph due to the risk mitigation synonymous with spreads as rises in both legs offset each other. Once converted into cents per bushel[[5]](#footnote-5) 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 prices[[6]](#footnote-6) 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 2. 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*** | ***10*** |
| ***11*** | ***CBOT Ethanol Returns*** | ***12*** |
| ***12*** | ***Moving Average of the Corn Crush spread returns*** | ***14*** |
| ***13*** | ***Moving Average of the Corn Crush spread returns*** | ***21*** |

Table 1. 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 1 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 in Dunis *et al.* (2011b).

The observed data period spanning from 23/03/05-31/12/10 has been segregated into in- sample and out-of-sample data as used during the modelling process.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name of period** | **Trading days** | **Beginning** | **End** |
| ***Total dataset*** | **1,460** | **23 March 2005** | **31 December 2010** |
| ***Training dataset (in-sample)*** | **707** | **23 March 2005** | **23 January 2008** |
| ***Test dataset (in-sample)*** | **291** | **24 January 2008** | **19 March 2009** |
| ***Validation set (out-of-sample)*** | **462** | **20 March 2009** | **31 December 2010** |

Table 2. 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 ‘overfitting’ 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 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 our 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).

**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 appendix[[7]](#footnote-7). However, all of the results and parameters for in-sample models can be found in Appendices A.3 and A.4 respectively.

**4.1.1 Naive Trading Strategy**

This strategy is known as ‘naive’ due to its simplistic nature: the forecasted returns for today are simply the returns generated from the previous day. In other words, we generate a forecasted time series based on a one day regressive linear method. The naive model is mathematically depicted as follows:

  (3)

Where:  is the actual rate of return at period *t*

  is the forecast rate of return for the next period

**4.1.2 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, we have decided to include its signals as a benchmark model. The MACD model can mathematically be defined as follows:

|  |
| --- |
|   |

(4)

where: M*t* is the moving average at time *t*

*n* is the number of terms in the moving average

*Yt* is the actual rate of return at period *t*

The MACD strategy used is also fairly straightforward 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 determined based on which combination performed best over the in-sample period in terms of annualised returns. This combination is then retained for out-of-sample evaluation. In this study a 1-day moving average (hence the daily returns of our spread) for the shorter term and a 60-day moving average for the longer term. Therefore a (1, 60) combination was deemed to be the most profitable in terms of trading performance with n = 1 and 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.

**4.1.3 ARMA Model**

Autoregressive moving average (ARMA) models 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:



 (5)

Where:  is the dependent variable at time *t*

, , and  are the lagged dependent variable

, , , and  are regression coefficients

 is the residual term

, , and  are previous values of the residual

 , , and  are weights.

Using a correlogram as a guide in the training and the test sub-periods a restricted ARMA (17,17) model was selected. Furthermore, the same ARMA model as used by Dunis *et al.* (2011b[[8]](#footnote-8)) was retained 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 figure 8 in Appendix A.4).

The model selected was also retained for out-of-sample trading simulation. Therefore our specific ARMA model is presented in the following manner:

***Yt = -3.42 \* 10-4 + 0.903Yt-2 – -0.417Yt-6 – -0.126Yt-17  + -0.902ε t-2 - 0.419ε t-6 -0.182ε t-17*** (6)

**4.2 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 variables[[9]](#footnote-9). 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:

1. 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.
2. The input data are then processed by the input nodes which contain a value of explanatory variables.
3. 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.
4. The nodes of the hidden layer then pass the processed data through a nonlinear activation function.
5. This is then processed by the output layer providing the calculated value is above the threshold (determined by the backpropagation of errors algorithm).
6. Finally, the processed outputs are then validated to measure whether the network needs to be retrained in order to better fit the data series.

## **4.2.1 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 sine[[10]](#footnote-10). 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 below:

MLP











Figure 3. A single output, inter-connected MLP model

where:

  model inputs (including the input bias node) at time t

  hidden node outputs (including the hidden bias node)

 MLP model output

 and  network weights

 transfer sigmoid function: , (7)

 linear function:  (8) The error function to be minimised is  (9)

with  being the target value .

The 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.

**4.2.2 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*. (2009) 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 recall[[11]](#footnote-11), HONNs have not yet been widely used in finance. The architecture of a three input second order HONN is shown below:

 

Figure 4. Left, MLP with three inputs and two hidden nodes; right, second order HONN with three inputs.

where:

  model inputs (including the input bias node) at time t

 HONNs model outputs

 network weights

 model inputs.

 transfer sigmoid function:  (10)

 a linear function:  (11)

The error function to be minimised is:  (12)

with  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. Additionally, another benefit of reducing free weights is that issues of ‘overfitting’ and local optima which are inherent in most 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.* (2009).

The methodology for HONNs was devised in line with that utilised for MLP networks in that the training process was stopped once optimal annualised returns were reached during the in-sample simulation. A summary of findings can be seen in the following section with the parameters for the HONN network also being included in appendix A.3.

**4.3 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) have 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).

For the purpose of this research, the GP application is coded and implemented to evolve tree based structures that present models (sub-trees) of input-output (see figure 5 below). Fundamentally the GP application builds algebraic expressions in order to calculate next day returns from a variety of inputs. Once the GP application arrives at a suitable expression during the in sample period this is then used during a validation period to produce results. 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 ‘overfitting’ 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’ in that a single member of the population is replaced at a time. The reasoning behind the decision to use a steady state algorithm is justified as they hold a greater selection strength and genetic drift over other algorithms such as typical generational GAs. Additionally 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’.

As mentioned above, the genetic programming algorithm utilises formulas to evolve algebraic expressions to enable the analysis and optimisation of results in ‘tree-like structures’. This genetic tree structure consists of nodes (depicted as circles in the diagram below) which represent functions that exist to perform actions within this structure. The purpose of functions is to generate output signals whereas the square-like symbols are terminal functions representing the end of a function once the most superior sub-tree (model) is reached. For example, the tree structure below is characterised by the algebraic expression 4 / x1 (t-1) + ln(x2(t-2)). In this example there are 3 terminal nodes which are expressed by x1(t-1), x2(t-2) and 4. The non terminal nodes represent the functions, in this case /, ln and + (see figure 5). Notably, 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’ as discussed above. For this application a tree size of 6 is decided based on trial and error optimisation. Furthermore, this is in line with current literature as Iba (1999) also use a maximum tree depth of 6 to forecast Japanese stock market prices.

A Generic Genetic Tree Structure

4

X1(t-1)

X2(t-2)

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.[[12]](#footnote-12)
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.
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.

1. ***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.

AFTER MUTATION

(Offspring 2)

BEFORE MUTATION

(Offspring 1)

Figure 6*.* Mutation of a Tree Structure

where: denotes the original sub-tree (model)

 denotes the new sub-tree (model)

 denotes the mutation point

1. ***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.

OFFSPRING 2

PARENT 2

 PARENT 1

OFFSPRING 1

\*

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

1. 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 in experienced in nature.
2. 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.
3. 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. Estimation parameters used for the GP in this paper are presented in Table 11 of Appendix A.3.

## **5. EMPIRICAL RESULTS**

**5.1 Statistical Performance**

By observation, Table 3 statistically reveals that both of the GPA and MLP models are marginally stronger than 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 however when taking both the in- and out-of-sample (Table 4) 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 |
| *MAE* | 0.0272 | 0.0205 | 0.0187 | 0.0189 | 0.0187 |
| MAPE | 552.52% | 148.51% | 122.81% | 153.85% | 140.76% |
| *RMSE* | 0.0363 | 0.0269 | 0.0254 | 0.0257 | 0.0251 |
| *THEIL-U* | 0.6932 | 0.8793 | 0.8026 | 0.8364 | 0.7639 |
| *Correct Directional Change (CDC)* | 48.60% | 55.05% | 54.42% | 55.38% | 54.74% |

Table 3. In-Sample Statistical Performance

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Naive** | **ARMA** | MLP | HONN | GPA |
| *MAE* | 0.0152 | 0.0108 | 0.0123 | 0.0115 | 0.0108 |
| MAPE | 675.73% | 188.93% | 254.34% | 405.56% | 351.65% |
| *RMSE* | 0.0228 | 0.0163 | 0.0201 | 0.0170 | 0.0164 |
| *THEIL-U* | 0.6989 | 0.8659 | 0.7095 | 0.7560 | 0.8254 |
| *Correct Directional Change (CDC)* | 51.30% | 51.73% | 53.25% | 52.38% | 54.11% |

Table 4. Out-of- Sample Statistical Performance

**5.2 Trading Performances**

The in-sample trading performance results are given in Appendix A.5. 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 conditions 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.

The table below 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 our 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 in that its annualised return is greater. It is also worth noting that the GPA model also manages to produce the best information and Calmar ratios[[13]](#footnote-13). Each of these models was retained from our in sample training with the results being provided in Table 12 of appendix A.5. Details of statistical measures used to calculate these results can be found in appendix Table 14 of Appendix A.6.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Naive**  | **MACD** | **ARMA** | MLP | HONN | GPA |
| Annualised Return (excluding costs) | 20.00% | 12.26% | 23.56% | 32.70% | 36.06% | 37.43% |
| Annualised Volatility (excluding costs) | 25.76% | 25.86% | 25.83% | 25.77% | 25.77% | 25.77% |
| Maximum Drawdown (excluding costs) | -23.26% | -20.82% | -39.44% | -19.72% | -25.62% | -18.01% |
| Calmar Ratio (excluding costs) | 0.86 | 0.59 | 0.60 | 1.66 | 1.41 | 2.08 |
| Information Ratio (excluding costs) | 0.78 | 0.47 | 0.91 | 1.27 | 1.40 | 1.45 |
| # Transactions (annualised) | 118 | 12 | 169 | 125 | 113 | 110 |
| Trading Days | 462 | 462 | 462 | 462 | 462 | 462 |
| Transaction costs  | 11.73% | 1.19% | 16.85% | 12.49% | 11.29% | 10.96% |
| Annualized Return (including costs)[[14]](#footnote-14) | 8.27% | 11.07% | 6.70% | 20.21% | 24.77% | 26.46% |

Table 5. Out-of-sample (unleveraged trading performance)

In contribution to Dunis *et al.* (2011b) we develop our volatility filter further 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. In reality, the opposition between market timing techniques as used in Dunis *et al.* (2011b) and time varying leverage is only apparent as time varying leverage can be easily achieved by scaling position sizes inversely to computed risk measures.

There are a number of different measurement techniques available the 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 (1994), the estimation of volatility regimes is therefore based on a rolling historical average of RiskMetrics volatility (μAVG) 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 μAVG and σ are calculated on a 3- month rolling window[[15]](#footnote-15). The average of both μAVG 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.

Hence, volatility regimes are categorised as ‘Lower High’ (between μAVG and μAVG + 2σ), ‘Medium High’ (between μAVG + 2σ and μAVG +4σ), and ‘Extremely High’ (greater than μAVG + 4σ) volatility periods. Similarly, periods of lower volatility are categorised as ‘Higher Low’ (between μAVG and μAVG - 2σ), ‘Medium Low’ (between μAVG - 2σ μAVG -4σ) and ‘Extremely Low’ (less than μAVG - 4σ).

As the crush spread is evidently volatile our trading strategy is two-fold in that while we decide to leverage during times of lower volatility we also decide to avoid trading altogether during volatility regimes classified as ‘Extremely High’. The RiskMetrics formula used to calculate volatility is:

 (13)

where: μ2 is the volatility forecast of our spread returns,

 r2  is the squared return of the spread,

β 0.94 for daily data as computed in JP Morgan (1994).

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

 

 (14)

where: μ2 is the RiskMetrics volatility of our spread returns,

T is the ‘extremely high’ volatility regime (greater than μAVG + 4σ).

We employ different levels of leveraging during each of these regimes. Leveraging structures can be seen in the below table.

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

Table 6. 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. During times of high volatility we avoid leveraging and even trading altogether as this could result in catastrophic losses. In particular, this strategy is employed by many hedge funds and professional traders who run ‘market neutral’ arbitrage strategies however in our case we base our leveraging on different volatility regimes determined by the RiskMetrics forecasted volatility. We have selected the above leveraging structure as on average, successful hedge funds tend to leverage between 1.5 and 2.5 times their returns[[16]](#footnote-16). 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 our case this strategy is applied when our spread is experiencing lower volatility based on the time varying parameters discussed above.

By observation, table 7 displays trading results produced by our models under the leveraging strategy and must be compared to those of Table 5 above. From this table, it is apparent that the GPA model continues to be the most profitable 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 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 our leveraged models are increased slightly when compared to our unleveraged models. More significantly, by avoiding trading during times of extreme volatility our leveraged models produce more acceptable maximum drawdowns. Thus, maximum drawdowns are reduced for all six of our models. Most notably, the leveraged market timing strategy reduces our ARMA model’s maximum drawdown from -39.44% to -15.25%.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Naive**  | **MACD** | **ARMA** | MLP | HONN | GPA |
| Annualised Return (excluding costs) | 18.27% | 8.85% | 28.15% | 43.96% | 42.67% | 44.01% |
| Annualised Volatility (excluding costs) | 29.04% | 25.13% | 29.17% | 29.02% | 29.07% | 29.10% |
| Maximum Drawdown (excluding costs) | -18.41% | -18.14% | -15.25% | -15.78% | -14.24% | -14.89% |
| Calmar Ratio | 0.99 | 0.49 | 1.85 | 2.79 | 3.00 | 2.96 |
| Information Ratio | 0.63 | 0.35 | 0.96 | 1.51 | 1.47 | 1.51 |
| # Transactions (annualised) | 84 | 6 | 141 | 96 | 79 | 84 |
| Trading Days | 462 | 462 | 462 | 462 | 462 | 462 |
| Transaction and Leverage costs[[17]](#footnote-17)  | 10.15% | 2.95% | 15.77% | 11.30% | 9.66% | 10.10% |
| Annualized Return (including costs) | 8.12% | 5.90% | 12.38% | 32.66% | 33.01% | 33.92% |
| Leverage Improvement | -0.15% | -5.17% | 5.67% | 12.45% | 8.24% | 7.44% |
| Drawdown Improvement | 4.85% | 2.68% | 24.19% | 3.94% | 11.38% | 3.12% |

Table 7. Out-of-sample (leveraged trading performance)

## **6. CONCLUDING REMARKS**

From the outset our aim 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 continued to generate attractive results (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 similar. These, observations are also in line with those drawn by Dunis *et al.* (2011a) who forecast the EUR/USD relationship in that their GPA model also outperforms a generic MLP neural network model. In particular, they find that a GPA returned 3.75% more in annualised returns than the MLP 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 biggest reduction down from a staggering -39.44% to -15.25%. For the most part, leveraged models experience slightly higher volatility and reduced maximum drawdowns as a result of the ‘no trade’ threshold filter.

Ultimately, this investigation offers an example of forecasting and trading the spread between corn and ethanol futures providing ethanol plants, fund managers, grain elevators, processors, and other market participants with a valuable insight into the use of nonlinear modelling for trading or hedging purposes.

## **Appendix**

**A.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 15th calendar day of the contract month. | 3rd 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 8. Contract Specifications

**A.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 9. 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 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.

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

Table 10. Neural Network Characteristics

|  |  |
| --- | --- |
|  Parameters | **GP** |
|  *Population Size* | *1000* |
|  *Tournament Size* | *20* |
|  *Mutation Probability* | *0.75* |
|  *Maximum Generations* | *100000* |

Table 11. GPA Characteristics

**A.4 ARMA Modelling**

The ARMA model used for this paper is as follows:

|  |  |  |
| --- | --- | --- |
| Dependent Variable: RETURNS |  |  |
| Method: Least Squares |  |  |
| Date: 04/02/11 Time: 15:40 |  |  |
| Sample (adjusted): 18 998 |  |  |
| Included observations: 981 after adjustments |  |
| Convergence achieved after 43 iterations |  |
| Backcast: 1 17 |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| Variable | Coefficient | Std. Error | t-Statistic | Prob.   |
|  |  |  |  |  |
|  |  |  |  |  |
| C | -0.000342 | 0.000913 | -0.374330 | 0.7082 |
| AR(2) | 0.902881 | 0.032716 | 27.59762 | 0.0000 |
| AR(6) | -0.416831 | 0.073591 | -5.664143 | 0.0000 |
| AR(17) | -0.125887 | 0.062711 | -2.007419 | 0.0450 |
| MA(2) | -0.901931 | 0.027539 | -32.75085 | 0.0000 |
| MA(6) | 0.419448 | 0.069568 | 6.029301 | 0.0000 |
| MA(17) | 0.182188 | 0.065001 | 2.802834 | 0.0052 |
|  |  |  |  |  |
|  |  |  |  |  |
| R-squared | 0.017201 |     Mean dependent var | -0.000335 |
| Adjusted R-squared | 0.011147 |     S.D. dependent var | 0.026339 |
| S.E. of regression | 0.026192 |     Akaike info criterion | -4.439624 |
| Sum squared resid | 0.668179 |     Schwarz criterion | -4.404741 |
| Log likelihood | 2184.636 |     F-statistic | 2.841150 |
| Durbin-Watson stat | 1.919907 |     Prob(F-statistic) | 0.009529 |
|  |  |  |  |  |
|  |  |  |  |  |
| Inverted AR Roots |  .93-.20i |      .93+.20i |    .84+.38i |  .84-.38i |
|  |  .53-.63i |      .53+.63i |    .22-.82i |  .22+.82i |
|  | -.07+.87i |     -.07-.87i |   -.38+.74i | -.38-.74i |
|  | -.68+.51i |     -.68-.51i |        -.91 | -.94-.31i |
|  | -.94+.31i |  |  |
| Inverted MA Roots |  .95-.20i |      .95+.20i |    .85-.40i |  .85+.40i |
|  |  .55-.65i |      .55+.65i |    .23-.84i |  .23+.84i |
|  | -.07+.88i |     -.07-.88i |   -.39+.76i | -.39-.76i |
|  | -.70+.53i |     -.70-.53i |        -.93 | -.95+.31i |
|  | -.95-.31i |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Figure 8. ARMA Results

**A.5 Empirical Results in the Training and Test Sub-Periods**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Naive** | **MACD** | **ARMA** | MLP | HONN | GPA |
| Annualised Return (excluding costs) | 23.96% | 8.77% | 57.82% | 71.77% | 72.54% | 75.36% |
| Annualised Volatility (excluding costs) | 41.43% | 40.66% | 41.32% | 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 | 1.45 | 1.82 | 1.82 | 2.33 |
| Information Ratio | 0.58 | 0.22 | 1.40 | 1.76 | 1.78 | 1.85 |
| # Transactions (annualised) | 128 | 14 | 153 | 101 | 100 | 126 |
| Trading Days | 998 | 998 | 998 | 998 | 998 | 998 |

Table 12. In-Sample (unleveraged trading performance)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Naive**  | **MACD** | **ARMA** | MLP | HONN | GPA |
| Annualised Return (excluding costs) | 1.18% | 0.98% | 49.49% | 79.53% | 88.63% | 84.61% |
| Annualised Volatility (excluding costs) | 43.21% | 41.07% | 43.00% | 41.04% | 41.10% | 41.40% |
| Maximum Drawdown (excluding costs) | -25.40% | -25.09% | -24.42% | -24.79% | -24.31% | -23.40% |
| Calmar Ratio | 0.05 | 0.04 | 2.03 | 3.21 | 3.65 | 3.62 |
| Information Ratio | 0.03 | 0.02 | 1.15 | 1.94 | 2.16 | 2.04 |
| # Transactions (annualised) | 122 | 5 | 145 | 92 | 90 | 118 |
| Trading Days | 998 | 998 | 998 | 998 | 998 | 998 |

Table 13. In-Sample (leveraged trading performance)

A.6 Performance Measures

The performance measures are calculated as follows:

|  |  |
| --- | --- |
| *Annualised Return* |  *with being the daily return* |
| *Cumulative Return* |  |
| *Annualised Volatility* |  |
| *Information Ratio* |  |
| *Calmar Ratio* | *R A**⎯⎯⎯*⎯ ⎜*MaxDD* ⎪ |
| *Maximum Drawdown* | *Maximum negative value of  over the period* |

Table 14. Trading simulation performance measures

**references**

Ang, A., Gorovyy. S, 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 (Atlanta, GA, USA, July 12 - 16, 2008). M. Keijzer, Ed. GECCO* *'08. ACM, New York, NY*, pp. 873-880.

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.” Online. *Available at http://www.cbot.com/cbot/docs/73511.pdf*. [Accessed April 2010].

CME (2010). “Commodity Products - Trading The Corn For Ethanol Crush.” Online. Available at http://www.cmegroup.com/trading/agricultural/files/AC-406\_DDG\_CornCrush\_042010.pdf. [Accessed April 2010].

Dahlgran, R. A. (2009). “Inventory and Transformation Hedging Effectiveness in Corn Crushing”, *Journal of Agricultural and Resource Economics*. 34(1), pp.154-171.

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). “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.L, Sermpinis, G, 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.

Engle, R. F., and Granger, C. W. (1987). “Co-integration and Error-Correction: Representation, Estimation and Testing.” *Econometrica,* 55, pp. 251-276.

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*](http://ideas.repec.org/s/ags/joaaec.html), 35(3), pp. 1-19.

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 online at: <http://ageconsearch.umm/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.

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

Holland, J. (1975). *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: University of Michigan Press.

H, Iba. (1999). “Bagging, Boosting and Bloating in Genetic Programming”, *In Proceedings of the Genetic and Evolutionary Computation Conference, (GECCO’99). Banzhaf W., et al. (Eds), Morgan Kaufmann*, pp. 1053-1060.

JP Morgan (1994). *RiskMetrics Technical Document*. Morgan Guaranty Trust Company*,* New York.

Kang, H.B. (2010). “A Case Study On The Archer Daniels Midland (ADM) Company’s Financial Statement Analysis: Strengths And Weaknesses”. *Journal of Business Case Studies.* 6(3), pp. 1-7

Karayiannis, N. and Venetsanopoulos, A. (1994). “On the Training and Performance of High-Order Neural Networks”, *Mathematical Biosciences*, 129, pp. 143-168.

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, Missouri, pp. 19-20.

Knowles, A., Hussein, A., Deredy, W., Lisboa, P. and Dunis, C. L. (2009). “Higher-Order Neural Networks with Bayesian Confidence Measure for Prediction of EUR/USD Exchange Rate”, pp. 48-59 in: *Artificial Higher Order Neural Networks for Economics and Business*, M. Zhang [eds.], Information Science Reference.

Koza, J.R. (1992), *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT Press, Cambridge, Mass.

Koza J.R. (1998), *Genetic Programming*, 29-43, in Williams, J. G. and Kent, A., [eds.] Encyclopedia of Computer Science and Technology, NY: Marcel-Dekker, New York.

Lan, Y., Wang, N.,Yang, J. (2011). “The Economics of Hedge Funds: Alpha, Fees, Leverage, and Valuation”, *NBER Working Paper No. 16842*. Issued in March 2011

Li, J., and Tsang, E.P.K. (1999). “Improving Technical Analysis Predictions: An Application Of Genetic Programming.” *In Proc. Of The 12th International Florida AI Research Society Conference*, pp. 108 -112.

McKay, P. A. (2006). “Demand for Ethanol Aggravates Pain at the Pump”. *Wall Street Journal*, Eastern Edition (19 June 2006).

Neely, C., Weller, P. And Ditmar, R. (1997). *Is Technical Analysis in Foreign Exchange Market Profitable? A Genetic Programming Approach*, in Dunis, C. and Rustem, B. (ed.), Proceedings, Forecasting Financial Markets: Advances for Exchange Rates, Interest Rates and Asset Management, London.

Sopena, J.M., Romero, E., Alquezar, R. (1999). “Neural Networks With Periodic and Monotonic Activation Functions: A Comparative Study In Classification Problems,” in Proc. 9th Int. Conf. Artif. Neural Netw, 1, 323-328.

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.

1. Central Standard Time (CST). [↑](#footnote-ref-1)
2. Particulars with regards to CBOT Corn contract specifications can be found in appendix A.1 (Table 8). [↑](#footnote-ref-2)
3. Chicago Board of Trade. [↑](#footnote-ref-3)
4. Particulars with regards to CBOT Ethanol contract specifications can be found in appendix A.1 (Table 8). [↑](#footnote-ref-4)
5. This conversion is explained in more detail in the previous section (section 3). [↑](#footnote-ref-5)
6. In our analysis we used arithmetic returns as opposed to 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 is deemed to be more realistic and more suitable for the purpose of our application. [↑](#footnote-ref-6)
7. These can however be provided on request. [↑](#footnote-ref-7)
8. The dataset used by Dunis *et al.* (2011-b) covered a period from 23/03/2005 until 31/12/2009. [↑](#footnote-ref-8)
9. As such, neural networks are often considered as a ‘black box’ as they fail to show the significance of each input. Furthermore, the way the network weights independent variables to form the forecasted outputs is also unclear. [↑](#footnote-ref-9)
10. 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)). [↑](#footnote-ref-10)
11. Associative recall is the act of associating two seemingly unrelated entities, such as smell and colour. For more information see Karayiannis and Venetsanopoulos (1994). [↑](#footnote-ref-11)
12. 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’. [↑](#footnote-ref-12)
13. 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. [↑](#footnote-ref-13)
14. Calculated using five basis points per contract (round trip) as used by King and Zulauf (2010) for the electronic trading of agricultural futures. In our case every transaction consists of one Corn contract and one Ethanol contract, therefore we calculate the transaction costs based on ten basis points per round trip (buy/sell) transaction. [↑](#footnote-ref-14)
15. We have tested different rolling windows in-sample and found that a 3-month rolling window produces the best results when subjected to our leveraging strategy. [↑](#footnote-ref-15)
16. See Lan *et al.* (2011) for a more detailed account. [↑](#footnote-ref-16)
17. The cost of leverage (interest payments for the additional capital) is calculated at 1.75% p.a. (0.0069% per trading day). [↑](#footnote-ref-17)