

Casting neural nets into modern markets

BY MURRAY A. RUGGIERO JR.
MARCH 26, 2013 • REPRINTS

The key to developing intelligent trading systems is to start with a reliable rule-based strategy. The neural network, genetic algorithm, kernel regression or other machine-learning method should be designed to enhance the system's already positive performance.

However, this core system does not have to be the Holy Grail. While this building block may not be tradeable as a standalone system, it should demonstrate an ability to generate a positive return over time, even if that return is lacking in terms of its risk-reward balance. For example, in trend-following markets, moving-average-crossover systems perform well. Similarly, we could employ a classic n-day breakout of highs or lows. Intermarket divergence strategies also are viable, as are those that employ traditional technical indicators or fundamental analysis, such as breadth data for the S&P 500, the Volatility Index or price/earnings ratios, to name a few.

What's important is the trader must have a strong understanding of how a given trading system works, knows its strengths and is aware of its weaknesses. This understanding can provide the foundation for enhancing the strategy using advance technologies.

Starting blocks

Let's take a closer look at some of the better starting points for a technology-enhanced approach.

Consider the moving-average crossover strategy, composed of simple moving average difference oscillators. One of our goals using advanced analysis methods might be to predict these oscillators as a percentage of the shorter moving average some time in the future.

Here's how the process would work:

1. Optimize a moving average crossover system, finding several robust sets of parameters.
2. Test what would happen if you predicted the moving average crossovers a few bars into the future. Also, evaluate the result of being a few bars late. We want the results of slightly late predictions to be reasonable still, and we want early predictions to be far better.
3. When a viable system is identified, develop a model to predict the difference between the moving average or the crossovers a few bars ahead. This is what will generate the signals.

With respect to the channel breakout, we might attempt to use neural networks to predict where prices n bars into the future will fall within the range over a set period. We might only take those breakouts if the market is still in that extreme price range so many bars into the future.

With respect to intermarket systems, which compare the price fluctuations of two related markets with respect to their historical norms, we can improve our performance by filtering out periods when these models are likely to break down. We also may be able to predict periods of stronger correlation.

Predictive indicators, such as the ADX, the relative strength index and stochastics, also can be employed. More complex models also might layer in advance/decline data, put/call ratios and sentiment. Neural networks are effective in combining such seemingly unrelated indicators into a single comprehensive signal.

Another good core approach is a relative strength model. These rank percentage returns over a given period, trading the top n-ranked issues in the basket. For example, in a basket of 100 stocks, we might trade the 10 with the highest returns. These strategies are popular among individual stock and exchange-traded fund traders. Here's how it works:

1. Calculate momentum for each stock in the basket. This momentum needs to be normalized. For example, this is a 50-bar momentum calculation: $Rawreturns = (Close - Close[50]) / Close[50]$. We also could filter out all issues trading below their 200-day moving average.
2. Sort these raw returns and select some number of the highest returning stocks. For example, if we have 100 stocks, we might take the top 10.

We then can use the neural network to predict future returns for each of these top performers. We also can re-balance the portfolio dynamically as the top stocks change.

Adding the net

Regardless of which core strategy you employ, there is a general framework for adding the neural network. The basic approach goes like this:

1. Decide on a classic rule-based trading system.
2. Identify the weaknesses of the system.
3. Decide how to address that weakness, such as filtering, ranking or weighting the output.
4. Determine a target to predict that will accomplish what is identified in step three. This is a critical point and could make or break this effort. The algorithm affects exactly how the target should be created.
5. Check system fault tolerance. For example, if the network fails, does it destroy system performance? Test the effect of late or early predictions. Ideally, late signals should cause minor degradations in performance while early signals should significantly increase performance; this indicates the core logic is sound and that results are not caused by chance.
6. Pre-process the data so that it's comparable across time periods.
7. Design your training and testing scheme for your model, including your retraining period and walk-forward period.
8. Test and analyze your model. Decide if the predictive component improves performance enough and is stable enough to test in live markets.

Case study: Moving average crossover

The moving average crossover system is a classic trend-following method. Two popular variations of this strategy are the dual-moving-average system and the triple-moving-average crossover.

Here's the code for a simple dual-moving-average crossover system:

```
Sub DualMACrossover(SLen As Integer,LLen As Integer)
    Dim ShortAve As BarArray
    Dim LongAve As BarArray
    ShortAve = Average(Close, SLen,0)
    LongAve = Average(Close, LLen,0)
    If ShortAve > LongAve Then
        Buy("BuyEnt",1,0,Market,Day)
    End If
    If ShortAve <LongAve Then
        Sell("SellEnt",1,0,Market,Day)
    End If
End Sub
```

This is a simple trend-following system that has some problems. First, if the short-term moving average is half of the dominant cycle, and the long-term moving average is a full dominant cycle, basic cycle theory says we are 180% out of phase, meaning regardless of how well the system does at other times, we will buy every top and will sell every bottom. This is why a dual-moving-average approach is flawed. It will catch big trends, but it will lose badly when trends are shorter term.

Our tests will use the following basket of markets: Australian dollar, cotton, dollar index, euro, Japanese yen, Swiss franc, sugar and natural gas. Our analysis covers Jan. 4, 1989, to Jan. 29, 2013. The results of the optimization, with \$75 per trade deducted for slippage and commissions, are shown in "First run" (below). There is a nice area of performance around the parameters 38 and 60, clearly the best set. That the parameters on each side of this set are also viable suggests that this is a robust combination.

FIRST RUN

The optimized results of our dual-moving-average trading system show some promises of being effective — if we can filter out those periods that inevitably will bring it down when traded on out-of-sample data.

Parameters	NetProfit	Trades	Win%	AveTrade	W/L	DD	PF
34, 50	\$531,193.40	655	32.37	\$810.98	3.16	(\$165,565.80)	1.51
34, 60	\$592,584.10	593	31.87	\$999.30	3.51	(\$110,809.60)	1.64
34, 70	\$666,178.70	537	33.15	\$1,240.56	3.77	(\$114,672.50)	1.87
36, 50	\$578,965.60	633	33.49	\$914.64	3.14	(\$140,356.60)	1.58
36, 60	\$701,398.10	551	33.76	\$1,272.95	3.7	(\$95,918.00)	1.89
36, 70	\$674,626.90	515	34.76	\$1,309.96	3.58	(\$106,914.30)	1.91
38, 50	\$639,991.50	611	33.55	\$1,047.45	3.35	(\$113,545.40)	1.69
38, 60	\$711,556.50	525	34.1	\$1,355.35	3.77	(\$98,380.50)	1.95
38, 70	\$649,824.30	499	34.87	\$1,302.25	3.49	(\$107,269.40)	1.87
40, 50	\$642,476.50	589	33.11	\$1,090.79	3.45	(\$108,227.20)	1.71
40, 60	\$695,073.50	519	35.07	\$1,339.26	3.58	(\$123,050.30)	1.93
40, 70	\$677,399.60	491	34.62	\$1,379.63	3.7	(\$105,756.60)	1.96
42, 50	\$587,490.10	587	32.71	\$1,000.83	3.37	(\$116,200.80)	1.64
42, 60	\$684,739.20	519	34.49	\$1,319.34	3.63	(\$110,661.30)	1.91
42, 70	\$668,937.30	471	35.03	\$1,420.25	3.63	(\$91,071.90)	1.96
44, 50	\$690,982.30	555	33.33	\$1,245.01	3.71	(\$101,794.90)	1.86
44, 60	\$668,096.70	503	35.39	\$1,328.22	3.47	(\$109,575.70)	1.9

The next step is to see how we can improve this using a neural network. For example, we can predict the crossover as a percentage of the short-term average. Before we implement that approach, however, we need to do a quick test of its effectiveness.

“Back to the past” (below) shows this analysis on the 38, 60 parameter pair. It shows the performance of the moving average cross five bars into the future and five bars late. We only lose about 30% being five bars late but gain 250% being five bars early. When we break this down further and look at the results on a market-by-market basis, we see much of the same. A short delay doesn't destroy the system, and a short lead improves it significantly. This is the type of delay/lead profile we are seeking. If we can use neural networks to improve our lead time, our results should follow. Because our short-term moving average is 38, predicting even five bars into the future is a reasonable goal.

BACK TO THE PAST

As we hoped, an earlier prediction (negative parameter numbers) would improve results significantly, while a late prediction (positive parameter numbers) only would temper them.

Parameters	NetProfit	Trades	Win%	AveTrade	W/L	DD
38, 60, -5	\$1,890,431.10	537	62.01	\$3,520.36	5.73	(\$52,187.20)
38, 60, -4	\$1,746,579.10	537	56.98	\$3,252.48	5.8	(\$52,187.20)
38, 60, -3	\$1,512,426.60	546	48.9	\$2,770.01	5.55	(\$55,166.50)
38, 60, -2	\$1,245,267.30	530	44.34	\$2,349.56	4.56	(\$56,528.20)
38, 60, -1	\$917,176.50	522	38.12	\$1,757.04	3.93	(\$82,153.10)
38, 60, 0	\$711,556.50	525	34.1	\$1,355.35	3.77	(\$98,380.50)
38, 60, 1	\$694,847.20	523	36.14	\$1,328.58	3.39	(\$134,627.10)
38, 60, 2	\$623,293.70	533	36.21	\$1,169.41	3.13	(\$128,866.50)
38, 60, 3	\$618,611.90	531	35.59	\$1,164.99	3.23	(\$101,288.30)
38, 60, 4	\$551,174.10	522	36.02	\$1,055.89	2.97	(\$126,971.50)
38, 60, 5	\$502,622.40	519	36.42	\$968.44	2.77	(\$119,793.60)

Case study: Keltner channel

Now let's break down another example. We will turn our attention to a Keltner channel breakout trend-following system on the same sample basket, using the same dates and the same \$75 for slippage and commissions.

Here's the code:

```
Sub KeltnerBandBKSimple(SLen,BandMult,RfiltLB,RfiltMult)
Dim MAave As BarArray
Dim StdVal As BarArray
Dim virprofit As BarArray
MAave=Average(Close,SLen,0)
StdVal=Average(Range,SLen,0)
If Close>MAave+BandMult*StdVal And Range<RfiltMult*Average(Range,RfiltLB,0) Then Buy("LE",1,0,Market,Day)
If Close<MAave-BandMult*StdVal And Range<RfiltMult*Average(Range,RfiltLB,0) Then Sell("SE",1,0,Market,Day)
If MarketPosition=1 And Close<MAave Then ExitLong("LX","LE",1,0,Market,Day)
If MarketPosition=-1 And Close>MAave Then ExitShort("SX","SE",1,0,Market,Day)
End Sub
```

Our best performing parameters are the 80, 3, 20, 1 combination, which earns \$538,656.50 on 518 trades with a 38.8% win percentage. The average trade makes \$1,039.88. The worst drawdown over the period is \$79,086.10. Our next step is to take this original system and add a predictive filter of some type. Toward that end, we develop the following forward-looking indicator:

```
Function ExcursionRatio(Offset,LB)
Dim HighPrice As BarArray
Dim LowPrice As BarArray
Dim ClosePrice As BarArray
Dim PostiveEx As BarArray
Dim NegativeEx As BarArray
HighPrice=tsgetdatastream(0,"High",Offset)
```

```

LowPrice= tsgetdatastream(0,"Low", Offset)
ClosePrice=tsgetdatastream(0,"Close", Offset)
PostiveEx=Highest(HighPrice, LB, 0)-ClosePrice[LB]
NegativeEx=ClosePrice[LB]-Lowest(LowPrice, LB, 0)
If NegativeEx<>0 Then
ExcursionRatio=PostiveEx/ NegativeEx
Else
ExcursionRatio=999999
End If
End Function

```

This indicator looks at the excursion from the first price in the window to the highest and lowest price in the future. It compares the ratio of price excursion. If positive excursion in the future is bigger than negative excursion, that's bullish and the ratio will be greater than 1.00. If the ratio is less than 1.00, then that's bearish. Using our excursion indicator and looking five bars into the future, we about double profits over our standard system (see "Time bandits," below).

TIME BANDITS

Here are the results of looking forward with our Keltner channel system. The results are substantially better. However, developing a filter that approaches these results will be a challenge.

Parameters	NetProfit	Trades	Win%	AveTrade	W/L	DD
80, 3, 20, 1, -5, 5, 0, 0.6	\$1,032,662.30	375	52.53	\$2,753.77	3.66	(\$54,257.00)
80, 3, 20, 1, -5, 5, 0, 0.8	\$1,027,178.10	374	51.87	\$2,746.47	3.75	(\$52,267.00)
80, 3, 20, 1, -5, 5, 0.1, 0.6	\$1,026,426.40	374	52.67	\$2,744.46	3.62	(\$53,982.00)
80, 3, 20, 1, -5, 5, 0.2, 0.6	\$1,023,147.00	371	52.29	\$2,757.81	3.67	(\$55,278.00)
80, 3, 20, 1, -5, 5, 0.1, 0.8	\$1,021,487.20	373	52.01	\$2,738.57	3.72	(\$51,992.00)
80, 3, 20, 1, -5, 5, 0.2, 0.8	\$1,018,207.80	370	51.62	\$2,751.91	3.77	(\$53,288.00)
80, 3, 20, 1, -5, 5, 0, 0.4	\$1,016,509.50	377	52.25	\$2,696.31	3.59	(\$52,839.50)
80, 3, 20, 1, -5, 5, 0.1, 0.4	\$1,010,273.60	376	52.39	\$2,686.90	3.56	(\$52,564.50)
80, 3, 20, 1, -5, 5, 0, 0.2	\$1,010,141.00	381	51.44	\$2,651.29	3.67	(\$53,624.50)
80, 3, 20, 1, -5, 5, 0.2, 0.4	\$1,006,994.20		52.01	\$2,699.72	3.6	(\$53,860.50)
80, 3, 20, 1, -5, 5, 0.1, 0.2	\$1,003,980.10	380	51.58	\$2,642.05	3.64	(\$53,349.50)
80, 3, 20, 1, -5, 5, 0.2, 0.2	\$1,000,700.70	377	51.19	\$2,654.38	3.68	(\$54,645.50)
80, 3, 20, 1, -5, 5, 0, 0	\$956,868.10	387	50.39	\$2,472.53	3.57	(\$53,504.50)

Unfortunately, making such predictions in real life is easier to synthesize than actually to do. In the case of the moving average crossover, predicting five bars into the future for a 38-60 crossover is effectively eliminating lag; although not easy by any means, it's a viable target. The target we are using for this Keltner channel system, however, is far less stable and much harder to predict. Based on this initial research, the moving-average crossover is the better prospect for improvement. The five-day lag elimination increases profits by 2.5 times, while the excursion prediction for the Keltner channel is much harder to accomplish and only doubles profits.

The next step is to develop a hybrid system neural network model based on this analysis and examine some additional case studies that round out our understanding of this approach.

Murray A. Ruggiero Jr. is the author of "Cybernetic Trading Strategies" (Wiley). E-mail him at ruggieroassoc@aol.com.