

Curve Fitting, Data Mining, Strategy Optimization & Walk Forward Analysis Using The Acceleration System

Working Paper October 2004
Copyright © 2004 Dennis Meyers

Table of Contents

	Page
The Dangers of Data Mining and Curve Fitting Real Prices _____	1
Data Mining and Curve Fitting Random Prices _____	9
Walk Forward Analysis Using The Acceleration System _____	15
Appendix A: An Example of The Power Walk Forward Output File _____	22
Appendix B: The Walk Forward Performance Explorer _____	24

The Dangers of Data Mining and Curve Fitting Real Prices

The optimization of the input parameters of a technical analysis system such that they produce the best performance results on past data has become especially easy using today's fast computers and software. Optimization here is defined as the combinatorial search over a range of system input values on price data defined over a fixed number of bars for the cases that produce the best system net profits or some other selected performance variable. This type of optimization is many times dubbed as curve fitting. Unfortunately the input parameters that produce the best results on past data don't necessarily produce good results in real time. Why is that? Optimization is performed on test data consisting of some set of past price data over some fixed number of bars. The price data consists of some deterministic signal or pattern (if any) plus spurious random price movements which we refer here to as noise. The computer optimization of a system's input parameters on the test data produces performance results that fit the signal **and** the noise. It is for certain that the noise process will not be repeated in the same way in the out-of-sample data or real time (out-of-sample data are prices that the system input parameters have not been optimized on). After an optimization run on the test data, it is difficult to determine whether the input variables chosen from the optimization run have curve fit the noise, which will not be repeated, or have successfully modeled the price signal. The profits on out-of-sample data from choosing the input parameters that generate the best performance results from the optimization fit on the test data will usually disappear if anomalous price movements or noise is a large part of the price data.

Optimization and Curve Fitting

One point that many don't realize is that it's almost impossible not to get a good curve fit with almost any indicator over some fixed time segment. If one looks at a curved fitted system's performance and its associated buy and sell signals on a chart, the seemingly astounding results seem to leap off the page giving the **illusion** that future profits from the system will be forthcoming to those who use it. The various mathematical statistics that can be derived from the curve fitted system performance further intensify the illusion by generating statistical measures that validate the test data curve fit. Unfortunately, these statistics do not tell how well the system will perform in the future. The excellent results produced by curve fitted systems only entice a potential user in a modern day "siren call" to believe in their repeatability. The reality is that future profits from curve fitted systems or anecdotal evidence are not assured and cannot be assumed. There is no way to tell if the input parameters chosen from an optimization run are curve fitting the noise which will not be repeated or the signal of the price series test data. The key to determine if a curve fitted system will work is to test it on data it has never been tested on, namely out-of-sample data, and observe how it performs on this data.

Optimization and Curve Fitting

If you think about it, all trading systems are curve fitted in some fashion. Trading systems almost by definition must work on past price data. Who would trade a system that did not work in the past? Curve fitting can be by computer or by astute observation...It's the same. A set of rules define what action one will take on price data. One takes a fixed time data sample called the test data and tries to get some system, or set of defined rules to work on it either by computer or by observing how things work in that time period.

To illustrate the points above we will examine two different systems on two different sets of data and compare the test performance results with the out-of-sample performance results. For the first system we will apply the Noise Channel Breakout System to IBM 5min bars. For the second system we will apply the Velocity System to E-Mini 1min bars..

The Noise Channel Breakout System

The Noise Channel Breakout System is defined as follows:

Buy Rule:

If price crosses above the highest high price of the last **nhi** bars by an amount greater than or equal to **f** then buy at the market. In addition, when short, and when calculating the highest high price (hhp), that hhp cannot be higher than the previously calculated hhp as the **nhi** lookback period moves forward and

previous highs are dropped out of the lookback window. In other words when short the hhp can only stay the same or go lower...it cannot go higher.

Sell Rule:

If price crosses below the lowest low price of last *nlo* bars minus by an amount equal to *f* or greater, **then** sell at the market. In addition, when long and when calculating the lowest low price (llp), that llp cannot be lower than the previous calculated llp as the nlo lookback period moves forward and previous lows are dropped out of the lookback window. In other words when long the llp can only stay the same or go higher...it cannot go lower.

Exit Rule:

Close any position five minutes before the New York Stock Exchange close (no trades are carried overnight).

The Least Squares Velocity System Defined

At each bar we calculate the least squares **velocity** (this is the slope of a straight line fit through the past *n* bars). When the velocity is greater than the threshold amount **vup** we will go long. When the velocity is less than the threshold amount **-vdn** we will go short.

Buy Rule:

IF Velocity is greater than the threshold amount **vup** then buy at the market.

Sell Rule:

IF Velocity is less than the threshold amount **-vdn** then sell at the market.

Intraday Bars Exit Rule:

Close the position 5 minutes before the close (no trades will be carried overnight).

Intraday Bars First Trade of Day Entry Rule:

Ignore all trade signals before 10:00 EST (30 minutes after the open).

For the Noise Channel System

IBM 5 minute bar Data

The test data section consists of one month of IBM 5min bars from May 16, 2001 to June 15, 2001. We will use June 18,2001 through June 22, 2001 as the out-of-sample section to see how the system input parameters and performance variables derived in the test section perform.

Finding The System Parameters Using Optimization

There are three system parameters *nhi*, *nlo* and *f*. We will use TradeStation's optimization module and vary the parameters over the following ranges: *nhi* from 2 to 30 in steps of 2, *nlo* from 2 to 30 in steps of 2 and *f* from 0 to 2 in steps of 0.25 for a total of 2025 cases. For this study, we will have TradeStation save the following system performance statistics for each case: Net Profit (NP), Number of Trades (#T), percent profitable (%P) Profit Factor (PF), Maximum Drawdown (DD), Largest Losing Trade(LLT).

Results

Table 1 presents a table of the best twenty cases ranked by Net Profit(NP). The statistics are based upon trading 1,000 shares of IBM. Slippage and commissions are ignored. Next to the NP column, the out-of-sample net profit (OSNP) column was inserted. Recall the net profit numbers are calculated for the period of May 16, 2001 through June 15, 2001. The out-of-sample net profit (OSNP) is separately calculated for each case from the period outside this test period namely, June 18,2001 through June 22, 2001. Using Excel's cut and paste, the out-of-sample net profit column was inserted into the column next to the test section's Net Profit column for easy comparison.

Figure 1 presents a chart of IBM with the buy and sell signals generated from the best net profits parameters in Table 1. As can be seen from the chart the buy and sell signals from the best net profit curve fit look great. Observing this type of performance on past data is how the siren call of the past performance curve fit tricks us into believing we will be able to repeat this performance in the future

Sadly, this promise of future profitability is just an illusion. As we can observe from Table 1, the Net Profit performance variable in this example is not a good criteria for choosing input parameters that will perform well in the future. The out-of-sample profits are mostly negative. The few cases of positive out-of-sample net profits seem to be distributed at random without defining features with respect to the other statistics.

Table 2 presents a table of the best 20 cases sorted by out-of-sample net profits (OSNP). The case numbers from the Net Profit ranking were retained in column 1 for comparison. As one can see from this table, the top 20 best net profits weren't even close in producing any of the top 20 out-of-sample net profits. As a matter of fact not even the top 10% (202cases) of the net profits produced any of the top 20 out-of-sample net profits.

Figure 2 presents a 3D Histogram of Net Profits vs Out-of-Sample Net profits. The first thing one observes is that there are many more out-of-sample losses than there are gains. The actual count is 278 gains and 1747 losses out of 2025 cases. This indicates that for this time period, the noise channel was unable to perform well. Also as can be seen from the histogram, the number of out-of-sample gains decline as the test data Net Profits increase.. This means that for this system and this time period, the test data Net Profit is not a good performance criteria for selecting input parameters that will produce gains in the out-of-sample period. This histogram shows the essence of the curve fitting of random data. The noise portion of the data did not repeat in the out-of-sample data portion resulting in losses from the input parameters that fit the noise portion of the test data and produced large net profits.

The Profit Factor is often suggested as a good performance statistic for selecting input parameters that will give good out-of-sample gains. Figure 3 presents a 3D Histogram of the Profit Factor vs Out-of-Sample Net profits. As can be seen from the histogram the number of out-of-sample profit gains decline as the Profit Factor increases. This is just the opposite of what is supposed to happen. Larger Profit Factors are supposed to predict larger out-of-sample gains. This means that for this system the profit factor is also not a good performance criteria for selecting parameters that will produce gains in the out-of-sample period. Of particular interest is profit factors above 2. These profit factors produces the smallest number of gains of all. This indicates that higher values of profit factor represent the curve fitting of the noise portion of the test data and thus will be useless in producing good out-of-sample performance.

For the Velocity System

E-Mini 1 minute bar Data

The test data section consists of one month of E-Mini 1 min bars from March 24, 2004 to April 23, 2004. We will use April 26, 2004 through April 30, 2004 as the out-of-sample section to see how the system input parameters and performance variables derived in the test section perform.

Finding The System Parameters Using Optimization

There are three system parameters *len*, *vup* and *vdn*. We will use TradeStation's optimization module and vary the parameters over the following ranges: *len* from 10 to 50 in steps of 2, *vup* and from 0 to 0.50 in steps of 0.05 for a total of 2541 cases.

Results

Figure 4 presents a 3D Histogram of the Net Profits vs Out-of-Sample Net profits for the Velocity system applied to E-min 1min bars. Once again we can see that large test sample Net Profits generate out-of-sample losses. This illustrates the dangers of data mining using the test sample Net Profit . The system has fit the spurious price movements quite well creating big profits in the test section and mostly losses in the out-of-sample section. The number of out-of-sample gains decline as the test data Net Profits increase. This histogram shows the essence of curve fitting a system to past prices. The noise portion of

the data did not repeat in the out-of-sample data section. This resulted in losses from the input parameters that fit the noise portion of the test data and which created large test section net profits.

The Profit Factor is often suggested as a good performance statistic for selecting input parameters that will give good out-of-sample gains. Figure 5 presents a 3D Histogram of the Profit Factor vs Out-of-Sample Net profits. As can be seen from the histogram the number of out-of-sample profit gains decline and losses increase as the Profit Factor increases above 2. This is just the opposite of what is a common believe. Larger Profit Factors from systems are supposed to good. Once again for a completely different system and set of data, the profit factor was not a good performance criteria for selecting parameters that will produce gains in the out-of-sample period. Of particular interest are profit factors above 2. These profit factors produced the smallest number of gains of all. This indicates that higher values of profit factor represent the curve fitting of the noise portion of the test data and thus will be useless in producing good out-of-sample performance. We will use this observation later in constructing a good filter or selection criteria.

Conclusion

We have shown that when the input parameters of a system are chosen based on the best net profits or the profit factor, out-of-sample losses are likely to occur. The optimization run produces many input parameter cases that have excellent net profits, profit factors and other statistics in the test section but losses in the out-of-sample section. This is how we are tricked by spurious price moves that do not repeat. From just one optimization event we have no idea whether the performance results we've obtained are from curve fitting the noise or an accurate measurement of some signal. Just looking at the test section performance gives one the illusion that future profits will occur. However as we have seen from the tables and figures such is not the case. It is rather difficult to separate out the cases that produce out-of-sample profits from losses using the above statistics from the test data section. More sophisticated statistical techniques are needed to combine the various test data window performance statistics in a way that will select cases that produce out-of-sample profits with greater probability than our simple performance statistics and sorting method above.

**Table 1 Optimum Input Parameters – Noise Channel System on IBM 5min Bars
Best 20 Cases Ranked By Net Profit (NP)**

Case #	nhi	nlo	f	NP	OSNP	#T	%P	PF	DD	LLT
1	12	16	0	\$10,580	(\$4,430)	75	51	1.76	(\$4,010)	(\$740)
2	14	16	0	\$10,510	(\$5,330)	69	51	1.79	(\$4,150)	(\$740)
3	12	18	0	\$9,550	(\$4,070)	73	52	1.69	(\$4,140)	(\$780)
4	16	16	0	\$9,550	(\$4,410)	67	51	1.69	(\$3,810)	(\$840)
5	14	18	0	\$9,160	(\$4,970)	69	49	1.66	(\$4,460)	(\$740)
6	12	14	0	\$8,970	(\$4,050)	83	48	1.59	(\$3,880)	(\$740)
7	16	14	0	\$8,630	(\$4,110)	73	45	1.59	(\$3,340)	(\$840)
8	14	14	0	\$8,490	(\$4,950)	77	47	1.56	(\$3,960)	(\$740)
9	24	10	0.5	\$8,340	\$300	28	61	2.79	(\$1,800)	(\$890)
10	16	18	0	\$8,200	(\$3,930)	67	48	1.56	(\$4,120)	(\$840)
11	18	16	0	\$8,190	(\$3,230)	67	49	1.56	(\$3,870)	(\$840)
12	24	8	0.5	\$8,100	\$560	29	62	2.69	(\$1,610)	(\$850)
13	12	12	0	\$8,070	(\$3,100)	88	47	1.51	(\$3,490)	(\$730)
14	26	10	0.5	\$8,060	\$300	28	61	2.69	(\$1,830)	(\$890)
15	14	20	0	\$7,940	(\$4,490)	65	49	1.56	(\$4,320)	(\$760)
16	24	12	0.5	\$7,940	(\$370)	28	61	2.62	(\$1,990)	(\$960)
17	12	20	0	\$7,850	(\$3,810)	69	49	1.55	(\$4,280)	(\$840)
18	26	8	0.5	\$7,820	\$560	29	62	2.6	(\$1,610)	(\$850)
19	28	10	0.5	\$7,740	\$300	28	61	2.61	(\$1,930)	(\$890)
20	26	12	0.5	\$7,660	(\$370)	28	61	2.53	(\$2,030)	(\$960)

Table 2 Best 20 Cases Ranked By Out-Of-Sample Net Profit (OSNP)

Case #	nhi	nlo	f	NP	OSNP	#T	%P	PF	DD	LLT
415	18	2	0.5	\$1,580	\$3,780	44	41	1.11	(\$4,950)	(\$1,280)
1610	6	2	0.5	(\$2,270)	\$3,710	59	42	0.87	(\$6,300)	(\$1,130)
1205	6	6	0.5	(\$960)	\$3,490	49	41	0.94	(\$5,390)	(\$1,390)
776	20	2	0.5	\$120	\$3,470	44	41	1.01	(\$5,410)	(\$1,280)
349	22	2	0.5	\$2,050	\$3,180	40	43	1.16	(\$4,150)	(\$1,280)
377	12	2	0.5	\$1,790	\$3,170	48	40	1.13	(\$4,600)	(\$1,280)
238	24	2	0.5	\$3,020	\$3,090	37	46	1.27	(\$4,200)	(\$1,280)
274	26	2	0.5	\$2,600	\$3,090	37	46	1.23	(\$4,380)	(\$1,370)
329	28	2	0.5	\$2,220	\$3,090	37	46	1.19	(\$4,650)	(\$1,370)
340	30	2	0.5	\$2,130	\$3,080	37	46	1.18	(\$4,680)	(\$1,370)
2016	4	8	0	(\$9,170)	\$3,000	192	36	0.75	(\$12,750)	(\$1,250)
1077	6	8	0.5	(\$600)	\$2,750	47	43	0.96	(\$5,330)	(\$1,400)
1473	4	2	0.75	(\$1,780)	\$2,750	43	42	0.88	(\$5,640)	(\$1,240)
265	18	4	0.5	\$2,690	\$2,740	39	46	1.23	(\$3,360)	(\$1,150)
2010	4	10	0	(\$8,330)	\$2,720	175	37	0.76	(\$13,260)	(\$1,250)
418	14	2	0.5	\$1,550	\$2,660	46	41	1.11	(\$5,640)	(\$1,280)
1609	22	4	0.25	(\$2,260)	\$2,660	62	39	0.88	(\$6,800)	(\$950)
1651	4	6	0.5	(\$2,450)	\$2,610	54	35	0.86	(\$6,960)	(\$1,200)
1108	8	2	0.5	(\$660)	\$2,540	54	41	0.96	(\$4,570)	(\$1,130)
545	16	2	0.5	\$820	\$2,500	46	39	1.06	(\$5,790)	(\$1,280)

Figure 1 IBM with the buy and sell signals generated from the best net profits parameters in Table 1

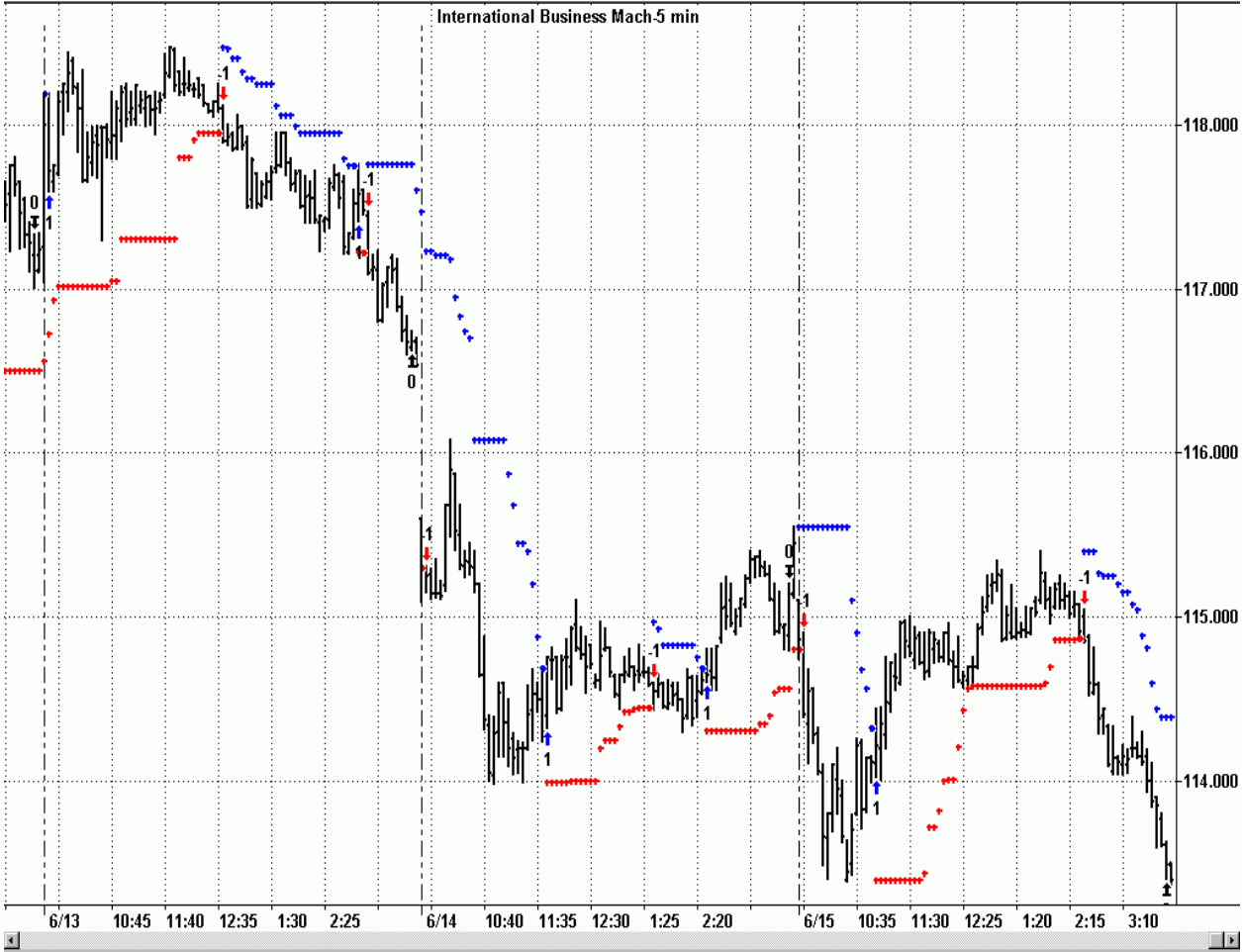


Figure 2 3D Histogram of Noise Channel System on IBM 5min Bars Net Profits vs Out-of-Sample Net profits.

Noise Channel System - Net Profits vs Out-Of-Sample Net Profits

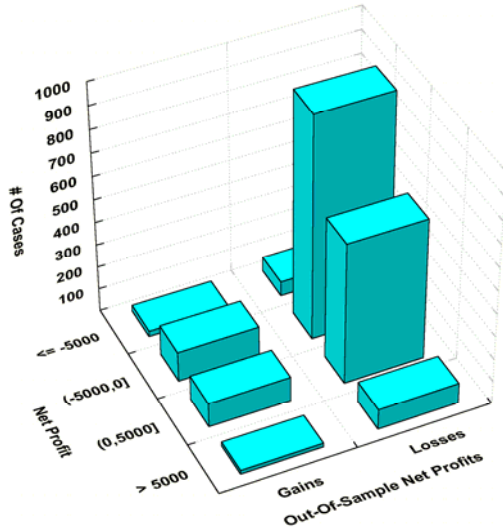


Figure 3 3D Histogram of Noise Channel System on IBM 5min Bars Profit Factor vs Out-of-Sample Net profits.

Noise Channel System - Profit Factor vs Out-Of-Sample Net Profits

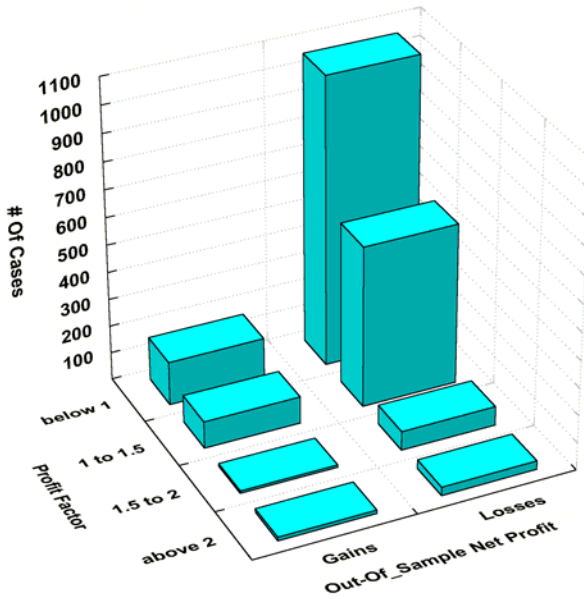


Figure 4 3D Histogram of Velocity System on E-Mini 1min Bars Net Profits vs Out-of-Sample Net profits.

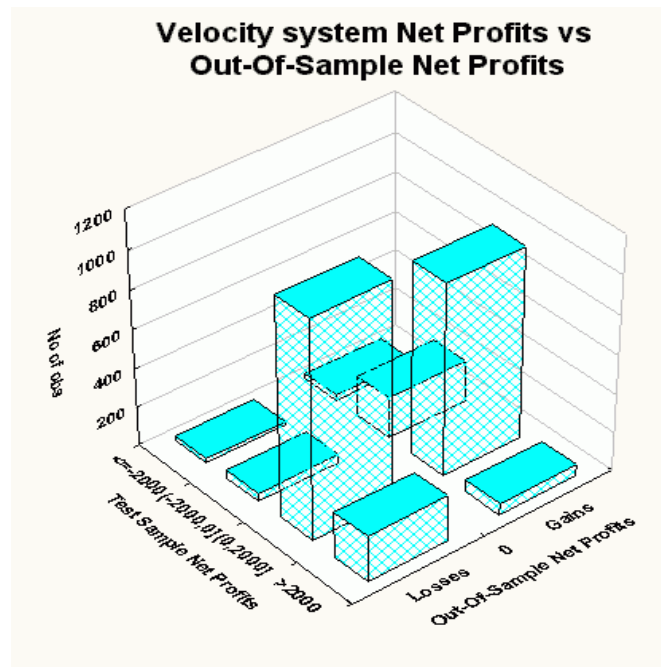
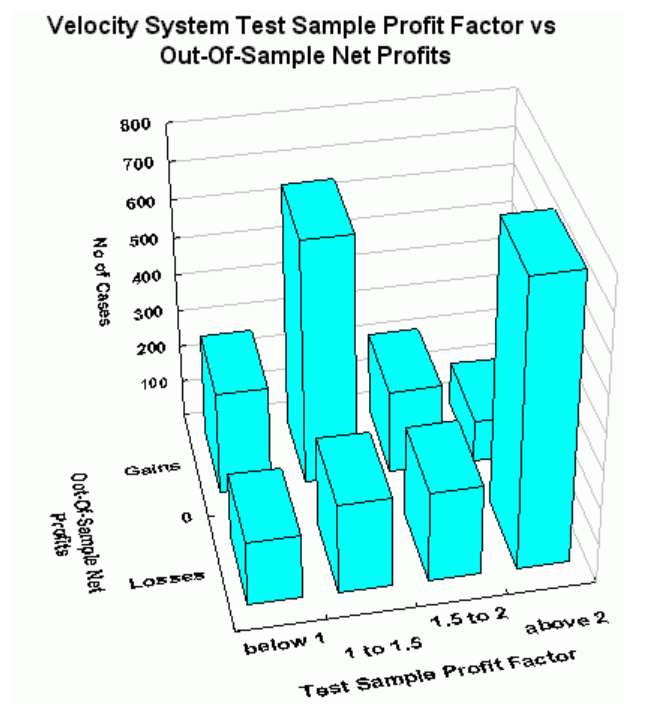


Figure 5 3D Histogram of the Velocity System on E-Mini 1min Bars Profit Factor vs Out-of-Sample Net profits.



Data Mining and Curve Fitting Random Prices

Optimization of technical analysis systems has become wide spread and very easy on today's trading software. By optimization we mean having the software run many variations of the system's input parameters of a system on data defined over a fixed number of bars and displaying the performance results for each variation or case. The performance results, being Net Profits or Profit Factor are then sorted from best to worse and the input parameters that produce the best performance results or combination of performance results are chosen and used in trading.

This type of combinatorial optimization is many times dubbed as curve fitting, data mining or data snooping. Unfortunately the input parameters that produce the best results on past data don't necessarily produce good results in real time. Why is that? Optimization is performed on test data. That is some past price data over some fixed number of bars. The price data consists of some deterministic signal or pattern(if any) plus random price spurious movements which we refer to here as noise. The computer optimization of a system's input parameters on the test data produces performance results that fit the signal **and** the noise. It is for certain that the noise process will not be repeated in the same way in the out-of-sample data or real time (out-of-sample data are prices that the system input parameters have not be optimized on).

Optimization and Curve Fitting

As we stated in the previous section, one point that many don't realize is that it's almost impossible not to get a good curve fit with almost any indicator and past price data over some fixed time segment. If one looks at the optimized system's performance and it's associated buy and sell signals on a chart, the seemingly astounding results seem to leap off the page giving the **illusion** that future profits from the system will be forthcoming to those who use it. As we will show in this section, the above even applies to random data. That is, given almost any indicator or system it is almost impossible not to get excellent performance results from the optimization procedure of a system on random data.

To demonstrate these points we will construct two different random price series. The first series will be a random price series of 3,968 five minute bars. The price series will start at a price level of 992 and the price changes of each bar from the previous bar will have the same standard deviation as the S&P500 futures 5 minute bars so as mimic a real price series in terms of the magnitude of price changes. The second series will be a random price series of 11,341 one minute bars. The price series will start at a price level of 1100 and the price changes of each bar from the previous bar will have the same standard deviation as the E-Mini futures 1 minute bars so as mimic a real price series in terms of the magnitude of price changes. We will add real dates and times to each generated random price point so that TradeStation can import this series onto a chart. We will apply the Channel Breakout System described in the previous section to the random 5 minute bar data. We will apply the Velocity System described in the previous section to the random 1 minute bars. We will then use the TradeStation optimization module to optimize the input parameters of these two systems over the same range of values described in the previous section. We will then examine the optimization output to see the performance results of these two systems on random data.

Results

Table 3 presents the results the best twenty cases ranked by Net Profit(NP) for the Channel System on the 5min random data . The performance statistics are based upon trading 1 contract of the random price series with each contract point equal to \$250. Slippage and commissions are ignored.

As we can see from Table 3 there are many cases that look like they present good input parameters to use. The Profit Factors in table 1 look good. The Profit Factors are not too high to alert one to curve fitting and not too low to prevent assumed profitable future trading. If I didn't know that this was random data then case 3 looks particularly interesting. The average trade profit is \$294, the profit factor is 1.81, the largest losing trade is only 4% of the Net Profits and the drawdown is only 16% of the net profits. In addition case 3's Net Profit was based on 133 trades. Many traders believe that if your system shows a profit on over 100 trades then the input parameters will be successful in future trading. This number of trades criteria is obvious not correct here. Remember this *is* random data so there are no good input parameters to use

on future random data because this random data pattern will not be repeated in the future. The only fly in the ointment to alert one not to trade this system in that there are no losers in a row(LR) less than 5. Usually in real data with a good system you will see LR's of 3 and 4. However there are many good trading systems with LR's of 5 or higher so it is not at all obvious that the table 1 results were based upon random data.

Figure 6 presents a chart of 5min Random data with the buy and sell signals generated from the input values of case 3 in Table 1. As can be seen from the chart the buy and sell signals from the case 3 curve fit look great. Observing this type of performance on past data is how the siren call of the past performance curve fit fools us into believing we will be able to repeat this performance in the future. Sadly, this promise of future profitability is just an illusion.

Table 4 presents the results for the best twenty cases ranked by Net Profit(NP) for the Velocity System on the 1 min random data. This table uses the spreadsheet output from the Power Walk Forward Optimizer described later in the next section. The performance statistics are based upon trading 1 contract of the random price series with each contract point equal to \$50. Slippage and commissions are ignored. In this table the test section statistics were generated from 9316 random 1min bars from 3/24/04 to 4/23/03 and the out-of sample section statistics were generated from 2025 random 1min bars immediately following the test section. Once again we can observe that the velocity system produced excellent Net Profits(tnp column) on the random data in the test section. There is little hint from observing these top 20 tnp's that these results are from applying the system to completely random data. Even the Losers in a Row (lr) stats are excellent. However, moving across any given row to the out-of-sample Net Profit (osnp) column we can see that the osnp's are all losses except for line 15.

Figure 7 presents a 3D Histogram of the Net Profits vs Out-of-Sample Net profits for the Velocity system applied to the random 1min bars. The most out-of-sample number of gains were when the test sample net profits were losses. This histogram indicates the randomness of the association between test net profits and out-of-sample net profits.

Figure 8 presents a 3D Histogram of the Profit Factor vs Out-of-Sample Net profits for the Velocity system applied to the random 1min bars. For any given profit factor range the out-of-sample gains and losses are about equal. If we compare this with the Velocity System on real data histogram of Figure 5 there are some interesting differences that we can exploit later in the walk forward section.

Conclusion

We have shown that system performance on random data can produce excellent optimization results. How are to determine that the results we obtain from optimization are due to repeatable price dynamics and not from randomness? One way is called walk forward analysis or out-of-sample testing. In out of sample testing you would run your optimization on some test data, pick your input parameters based upon some criteria and then run the system on data that the system has not been tested on namely out-of-sample data. If this Test Data/Out-Of-Sample data procedure is performed enough times and the average returns on these runs are satisfactory then you can gain confidence that your system is capturing some repeatable price patterns that you can trade.

Table 3 Optimum Input Parameters - Best 20 Cases Ranked By Net Profit

Case #	nhi	nlo	NetPrft	AvgTrd	%Prft	PFact	#Trds	#WTrds	#LTrds	LLT	Avg W/L	WR	LR	MaxDD
1	4	14	42648	151.8	40	1.55	281	113	168	-1045	2.31	6	8	-9533
2	14	14	42178	283.1	47	1.83	149	71	78	-1663	2.01	5	7	-6348
3	18	14	39180	294.6	45	1.81	133	60	73	-1663	2.20	6	6	-6348
4	16	14	38568	273.5	44	1.74	141	63	78	-1663	2.16	5	7	-6733
5	4	16	36938	140.4	39	1.49	263	105	158	-1045	2.25	6	8	-11955
6	20	14	36848	285.6	44	1.76	129	58	71	-1663	2.15	6	6	-7688
7	14	16	36008	255.4	44	1.70	141	63	78	-1663	2.11	4	7	-7203
8	8	14	35903	168.6	45	1.52	213	96	117	-1440	1.86	6	5	-8845
9	6	14	35548	150.0	43	1.49	237	102	135	-1353	1.97	6	8	-8983
10	4	10	33573	97.9	37	1.36	343	128	215	-1045	2.29	6	10	-13478
11	4	18	33523	134.6	38	1.46	249	95	154	-1045	2.37	5	8	-12070
12	14	18	32393	243.6	46	1.62	133	62	71	-1663	1.86	4	7	-7203
13	18	16	32205	253.6	41	1.63	127	53	74	-1663	2.27	4	6	-8148
14	8	16	32083	161.2	45	1.50	199	91	108	-1530	1.78	6	6	-10373
15	4	12	31978	100.2	36	1.36	319	118	201	-1045	2.32	6	9	-12288
16	6	16	31663	143.3	43	1.46	221	97	124	-1353	1.86	5	5	-11198
17	16	16	31213	231.2	41	1.57	135	56	79	-1663	2.22	4	7	-8093
18	20	16	30768	250.1	42	1.61	123	52	71	-1663	2.19	4	6	-9488
19	12	14	30728	175.6	42	1.49	175	74	101	-1663	2.04	5	7	-10375
20	18	18	30340	255.0	42	1.60	119	51	68	-1663	2.13	4	7	-7373

Figure 6 Chart of Channel System Applied to Random 5 min Data



**Table 4 Optimum Input Parameters - Best 20 Cases Ranked By Net Profit
For the Velocity System Applied to 1 min Bars of Random Data**

1	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
len	vup	vdn	xt	mtnp	tnp	nT	pP	PF	std	t	mlb	lbr	mwb	wbr	awl	wlb	mru	mdd	
2	30	0.12	0.26	1610	165	3000	20	60	3.0	348	1.93	325	2576	332	3450	1.02	1.339	145	311
3	32	0.14	0.22	1610	110	2786	19	63	3.8	288	2.22	288	2111	263	2870	0.91	1.36	145	237
4	40	0.12	0.22	1610	128	2745	17	65	3.6	328	2.03	280	1875	278	2832	0.99	1.51	117	362
5	34	0.14	0.22	1610	115	2739	18	61	3.8	293	2.21	286	2101	165	2400	0.58	1.142	117	234
6	38	0.12	0.22	1610	98.5	2727	17	65	3.7	322	2.05	281	1880	280	2847	1.00	1.514	117	362
7	38	0.12	0.24	1610	136	2663	17	65	3.6	327	1.98	281	1892	280	2830	1.00	1.496	117	362
8	30	0.12	0.24	1610	4	2622	23	57	2.4	344	1.59	262	2355	291	3674	1.11	1.56	151	266
9	30	0.12	0.28	1610	0	2607	18	56	2.7	366	1.68	325	2576	366	3179	1.13	1.234	145	311
10	30	0.12	0.3	1610	0	2607	18	56	2.7	366	1.68	325	2576	366	3179	1.13	1.234	145	311
11	32	0.14	0.26	1610	110	2585	17	65	3.7	305	2.05	277	1845	267	2839	0.96	1.539	145	347
12	32	0.14	0.24	1610	76	2535	18	61	3.3	307	1.94	277	1845	267	2844	0.96	1.541	117	237
13	30	0.12	0.22	1610	0	2509	27	52	2.1	325	1.48	101	2351	291	3703	2.88	1.575	151	230
14	28	0.16	0.26	1610	180.5	2482	17	65	3.5	287	2.10	279	1912	243	2767	0.87	1.447	145	347
15	32	0.16	0.22	1610	176.5	2460	16	63	3.6	290	2.12	288	1845	165	2223	0.57	1.205	117	237
16	38	0.12	0.28	1610	199	2454	14	57	3.4	357	1.84	281	1899	351	2564	1.25	1.35	117	368
17	38	0.12	0.3	1610	199	2454	14	57	3.4	357	1.84	281	1899	351	2564	1.25	1.35	117	368
18	34	0.14	0.26	1610	115	2432	16	63	3.5	314	1.94	276	1837	165	2364	0.60	1.287	117	237
19	40	0.12	0.28	1610	150.5	2425	14	57	3.2	365	1.78	280	1892	349	2554	1.25	1.35	117	368
20	40	0.12	0.3	1610	150.5	2425	14	57	3.2	365	1.78	280	1892	349	2554	1.25	1.35	117	368
21	38	0.12	0.26	1610	199	2412	15	53	3.2	349	1.79	281	2049	351	2564	1.25	1.251	145	368

1	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM
wr	lr	mpft	mlos	p/l	dd	llt	eqB0	eqR2	eqDev	eqB1	eqDy	eqB2	eqR22	mKr	osnp	odd	ollt	ont	aosp	
2	3	2	281	-143	2.0	-593	-369	219	86.1	304	216	167	-1.2	85.1	0.72	(139)	-604	-604	5	-27.8
3	6	2	214	-135	1.6	-310	-310	211	84.2	424	160	200	1.1	81.1	0.50	(355)	-731	-582	7	-50.6
4	6	2	257	-179	1.4	-369	-369	243	85.1	412	112	299	5.5	80.7	0.59	(154)	-703	-615	4	-38.4
5	6	2	229	-110	2.1	-310	-310	219	85.0	397	153	224	2.0	81.9	0.55	(304)	-817	-596	6	-50.6
6	6	2	223	-188	1.2	-345	-345	241	85.3	402	125	282	4.6	80.9	0.60	(56)	-596	-596	5	-11.2
7	6	2	223	-188	1.2	-383	-383	237	84.3	408	139	263	3.6	80.4	0.58	(119)	-640	-596	4	-29.6
8	3	4	275	-123	2.2	-980	-409	165	71.3	559	278	0	-6.0	79.6	0.30	(188)	-684	-604	7	-26.9
9	3	2	329	-143	2.3	-593	-369	226	84.9	181	208	168	-1.1	82.6	1.25	(139)	-604	-604	5	-27.8
10	3	2	329	-143	2.3	-593	-369	226	84.9	181	208	168	-1.1	82.6	1.25	(139)	-604	-604	5	-27.8
11	6	2	277	-176	1.6	-322	-322	236	85.8	406	87	310	6.6	81.1	0.58	(258)	-614	-582	5	-51.6
12	6	2	277	-110	2.5	-459	-390	216	82.6	442	146	219	2.0	79.0	0.49	(206)	-582	-582	6	-34.3
13	3	4	275	-143	1.9	-803	-460	161	79.1	554	102	155	1.0	73.4	0.29	(230)	-725	-604	7	-32.8
14	7	2	263	-122	2.2	-333	-333	219	86.7	364	105	272	4.9	83.2	0.60	(280)	-682	-557	7	-39.9
15	6	2	226	-135	1.7	-371	-346	229	86.4	323	132	257	3.9	82.6	0.71	820	-143	-143	5	163.9
16	5	2	329	-188	1.8	-385	-385	287	88.2	230	81	388	10.9	82.3	1.25	(119)	-640	-596	4	-29.6
17	5	2	329	-188	1.8	-385	-385	287	88.2	230	81	388	10.9	82.3	1.25	(119)	-640	-596	4	-29.6
18	6	2	221	-188	1.2	-358	-358	239	85.9	322	75	332	8.0	81.9	0.74	(122)	-624	-596	4	-30.4
19	5	2	329	-179	1.8	-385	-385	290	88.1	249	55	414	12.8	81.8	1.17	(154)	-703	-615	4	-38.4
20	5	2	329	-179	1.8	-385	-385	290	88.1	249	55	414	12.8	81.8	1.17	(154)	-703	-615	4	-38.4
21	5	3	329	-110	3.0	-385	-385	263	85.9	261	129	297	5.6	79.9	1.01	(119)	-640	-596	4	-29.6

Column Definition For The Test Section:

mtnp=median TrdProfit; **tnp**=Net Profits; **nT**=# of Trades; **pP**=% profitable; **PF**=Profit Factor; **std**=Standard Deviation of TrdProfits
t=Student t statistic for Net Profits; **mlb**=median losing bars; **lbr**=Total Losing Bars; **mwb**=median of winning bars; **awl**=mwb/mlb
wlb=wbr/lbr; **mru**=median(max\$TrdRU-TrdPft); **mdd**=median(TrdPft-Max\$TrdDD); **wr**=Max winners in a row; **lr**=Max losers in a row
mpft=median of winning trades; **mlos**=median of losing trades; **p/l**=mpft/mlos; **dd**=drawdown; **llt**=largest losing trade;
eqB0=slope of equity curve; **eqR2**=R² of equity curve; **eqDev**=median deviation of equity from straight line fit to equity curve
eqB1=slope of 2nd order polynomial fit to equity curve; **eqDy**=velocity of 2nd order polynomial fit; **eqB2**=acceleration of 2nd order fit.
EqR22=R² of 2nd order fit; **mKr**=Modified k-ratio=eqB0/Dev.

Column Definition For Out-Of-Sample Section:

Osnp=Out-Of-Sample Net Profit; **odd**=OOS drawdown; **ollt**=OOS largest losing trade; **ont**=OOS # of trades;
aosp=ave OOS trade profit.

Figure 7 3D Histogram of the Net Profits vs Out-of-Sample Net profits.

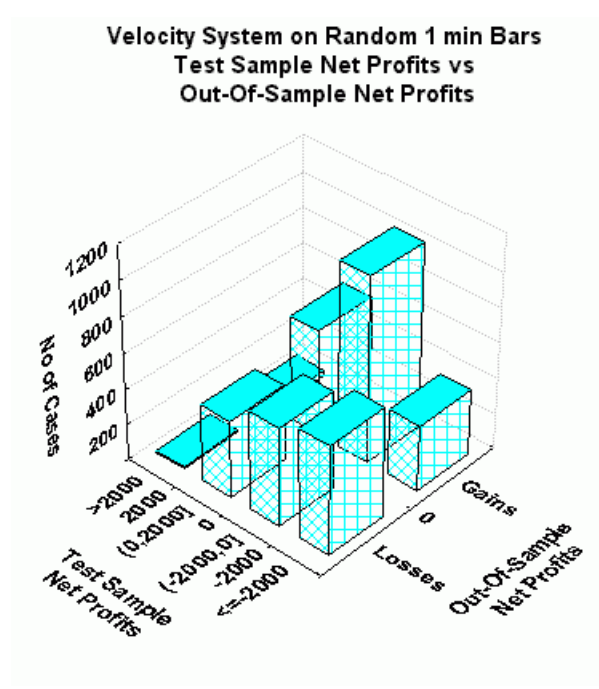
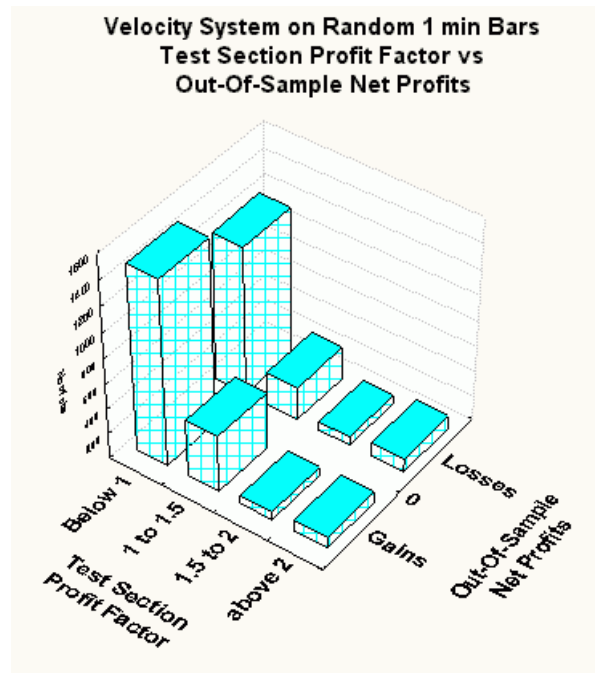


Figure 8 3D Histogram of the Profit Factor vs Out-of-Sample Net profits.



Walk Forward Analysis Using The Acceleration System

The Acceleration System

In an article in the July 2003 issue of Active Trader Magazine, I examined a trading system that used the velocity of prices fit by a straight line, through “N” past prices, to determine buy and sell points. The reasoning behind this type of system was to only trade when the price trend velocity was above a certain threshold. Many times prices meander around without a notable trend. At these times we do not wish to trade because of the whipsaw losses that occur from this type of price action. When a price trend finally starts, the velocity of that price trend moves above some minimum value. Thus the velocity system would only issue a trade when certain velocity barriers were crossed. Acceleration is the change in velocity per unit time. In order for velocity to reach a certain value the price changes must start accelerating. Thus acceleration should be an even earlier indicator than velocity that prices are starting to trend.

Unfortunately to find acceleration we cannot use the straight line formula of the previous article. Straight lines have only constant velocities and no acceleration. We have to use a 2nd order polynomial to fit N past prices in order to find the price acceleration.

The Best Fit 2nd Order Polynomial Line.

Let us imagine a set of closing prices on a graph with time as the horizontal axis and price as the vertical axis. Let us further suppose that we have only twenty closing price dots at twenty time intervals. How can we draw a 2nd order polynomial (also called a parabolic) through those twenty prices such that the sum of all the squared differences between the prices at each time interval and the parabolic line that is being fit to the data is minimized? This is called the “Least Squares Fit” of the data. This mathematical technique is available in most of today’s technical analysis software.

The formula for the 2nd order polynomial is:

$$p(t)_{EST} = a + b*t + c*t^2$$

Where $p(t)_{EST}$ is the estimated price found from the least squares fit at time t , a is the initial value of the line, b is the slope of the line, $2c$ is the acceleration and t is the time of the bar. Recall that velocity is defined as the change of position per unit time and acceleration is defined as the rate of change of velocity per unit time. Using the formula above acceleration would be the second derivative of $p(t)_{EST}$ and would be $2c$.

We can show this by using simple algebra on successive data bars

$$\text{Velocity}(t) = p(t)_{EST} - p(t-1)_{EST} = [a+b*t+ct^2] - [a+b*(t-1)+c(t-1)^2] = b+2c*t-c$$

$$\text{Acceleration}(t) = \text{Velocity}(t) - \text{Velocity}(t-1) = [b+2c*t-c] - [b+2c*(t-1)-c] = 2c$$

If you are fitting a 2nd order polynomial to N prices then the Least Squares Fit of the acceleration c are obtained from the solution of the following matrix equation.

$$\begin{bmatrix} T & \sum t & \sum t^2 \\ \sum t & \sum t^2 & \sum t^3 \\ \sum t^2 & \sum t^3 & \sum t^4 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} \sum p(t) \\ \sum (p(t)*t) \\ \sum (p(t)*t^2) \end{bmatrix}$$

where

$p(T)$ is today’s price, $p(T-1)$ is yesterday’s price and $p(1)$ is the price T days ago.

$\sum p(t)$ is the summation of prices from $t=1$ to T days
 $\sum p(t)*t$ is the summation of prices times t from $t=1$ to T days
 $\sum t$ is the summation of the integer t from $t=1$ to T days
 $\sum t^2$ is the summation of the integer t squared from $t=1$ to T days
etc.

Programs that solve for the solution to matrix equations can be found in the book "Numerical Recipes" by W. Press, et. al. which is currently available in book stores. The programs in this book can be translated into TradeStation's EasyLanguage™.

Where all summations are from $t=1$ to N . $2c(t)$ is the acceleration at point t and N is the number of prices we are using to calculate the coefficients.

While these matrix equations are correct, the implementation of them may cause a few computational problems. The first problem is slow speed. The computation of the polynomial coefficients at each bar using the matrix equations above is very slow. The second problem is floating point overflow. The summations in the polynomial coefficients can become very large causing floating point overflow and computational errors. To solve these problems I developed a fast efficient algorithm, similar to the Lagrange coefficients method discussed in the Nth Order Polynomial papers, that solves the above noted numerical floating point overflow problems and is super fast.

Here we will use the acceleration formula of the least squares 2nd order polynomial line to create a system. The least squares acceleration has the advantage that it is a natural random price noise inhibitor. We can create a system such that unless the acceleration from fitting N past price bars with a 2nd order polynomial is greater than some threshold we will not buy or sell. A large percentage of price noise generates a lot of back and forth movements of small magnitudes over a short period of time. With a lot of systems this back and forth movement creates many false buy and sell signals. However using the least squares acceleration over N past prices, we can filter out many of the small price noise movements by requiring that the acceleration be greater than some threshold before we act.

The Least Squares Acceleration System Defined

At each bar we calculate the least squares **acceleration** or $2c$ from the formula above. When the acceleration is greater than the threshold amount **aup** we will go long. When the acceleration is less than the threshold amount **-adn** we will go short.

Buy Rule:

IF Acceleration is greater than the threshold amount **aup** then buy at the market.

Sell Rule:

IF Acceleration is less than the threshold amount **-adn** then sell at the market.

Intraday Bars Exit Rule:

Close the position 5 minutes before the E-Mini close (no trades will be carried overnight).

Intraday Bars First Trade of Day Entry Rule:

Ignore all trade signals before 10:00 EST (30 minutes after the open). Buy and Sell rules above we have included a first trade of the day entry rule. We've included this rule because often there are gaps in the open creating immediate system buys and sells. Many times these gaps are closed creating a losing whipsaw trade. In order to avoid the opening gap whipsaw trade problem we've delayed the first trade of the day for 30 minutes until after 10:00 EST

Testing The Least Squares Acceleration System Using Walk Forward Optimization

There are three system parameters to find:

1. **N**, is the lookback period to calculate the **Acceleration = 2c**. Note that c is a very small number on the order of 0.001. In order to scale c we multiply it by 2000 instead of 2.
2. **aup**, the threshold amount that acceleration has to be greater than to issue a buy signal
3. **adn**, the threshold amount that acceleration has to be less than to issue a sell signal

To test this system we will use 1 minute bar prices of the E-Mini futures contract traded on the CME known by the symbol ES.

Walk Forward Analysis

Walk forward optimization will be used here because of the changing nature of intraday markets. Intraday price dynamics are constantly changing due to current economic surprises, events and trader sentiment. Also the time of year changes the nature of intraday markets, such as the seasons, holidays, vacation time, etc. As such, optimizations on intraday data performed three months ago may no longer be representative of today's intraday price dynamics. In addition to the above, spurious price movements (noise) in any test sample make it difficult to find the input parameters from any single optimization run that represent the repeatable price dynamics only.

What is walk forward analysis?. Optimization is a misnomer on today's trading platforms. Optimization should really be called a combinatorial search. That is, we have the software output certain performance parameters for every combination of a range of input parameters. We then sort that range on some performance parameter. Whenever you run an optimization on noisy data on a fixed time interval, the best performance will almost always be due to curve fitting the noise and signal. This curve fitting is also called "Data Mining" or "Data Snooping". A fixed number of prices on a fixed time interval has many spurious movements which are also called noise. When we run many different combinations of the input parameters, the best performance will be from those cases that are able to capture profits from the spurious movements and the price or signal dynamics. While the signal dynamics, if there, will repeat, the same spurious price movements will not. If the spurious movements that were captured by a certain set of input parameters were a large part of the net profits, then choosing these input parameters would produce losses when traded on future data. These losses occur because the spurious movements will not be repeated in the same way. The input parameters chosen from the test section performance are "cherry picked" to perform well on only those exact same spurious movements. This is why curve fitted systems with no out-of-sample testing cause losses in the out-of-sample section from something that looked great in the test section.

Walk forward analysis attempts to minimize curve fitting by using the law of averages from the Central Limit Theorem on the out-of-sample performance. In walk forward analysis the data is broken up into many test and out-of-sample sections. Usually for any system one has some performance parameter selection procedure or filter used to select the input parameters from the optimization run. For instance, a filter might be all cases that have a profit factor (PF) greater than 1 and less than 3. For the number of cases left we might select the cases that had the best percent profit. This procedure would leave you with one case in the test section output and it's associated input parameters. Now suppose we ran our optimization on each of our many test sections and applied our filter to each test section output. We would then use the input parameters found by the filter in each test section on the out-of-sample section immediately following that test section. The input parameters found in each test section and applied to each out-of-sample section would produce independent net profits and losses for each of the out-of-sample sections. Using this method we now have "x" number of independent out-of-sample section profit and losses from our filter. If we take the average of these out-of-sample section net profits and losses than we will have an estimate of how our system will perform on average. Due to the Central Limit Theorem, as our sample size increases, the spurious noises results in the out-of-sample section performance should average out to zero in the limit leaving us with what to expect from our system and filter. *Mathematical note:* This assumption assumes that the out-of-sample returns are from probability distributions that have a finite variance.

Finding The Acceleration System Parameters Using Walk Forward Optimization

There are three system parameters to find **N**, **aup** and **adn**.

For each test section we will run a TradeStation optimization on:

1. *aup* from 4 to 20 in steps of 2 (*c* is multiplied by 2000)
2. *adn* from 4 to 20 in steps of 2
3. *N* from 10 to 40 in steps of 2.

This will produce 1296 different cases or combinations of the input parameters.

As discussed above, one thing we do know about data mining is that the curve fit of the signal and noise will always produce the best performance results. Thus if we eliminate the optimization cases with the very best performance results we are sure to eliminate many of the data mining system input parameters that fitted the past spurious movements. We know from experience that very few good intraday systems can sustain profit factors above 3.0 over time. If we eliminate, from our optimization, all cases that have profit factors greater than 3.0 we will probably eliminate most of the cases that are due to curve fitting the noise. In addition, since in real time it is tough to sustain more than five losses in a row and still keep trading, we will eliminate all those cases that have more than five losses in a row.

Application of Walk Forward Analysis to the Acceleration System.

The procedure to use walk forward testing is as follows. In order to use the walk forward testing method, for this example, break the test data up into twenty-seven 30 calendar day test sections, as shown in Table 5 Page 20, and run a TradeStation optimization on each of those test sections. Then use some defined input parameter selection procedure, like the one described above, on the TradeStation optimization output on each of those twenty-seven test sections to find the system input parameters ***N***, ***aup***, and ***adn***. Next use the input parameters found in each test section on the out-of-sample section on the week directly following the test section to determine the out-of-sample performance. The results from each oos section are tabulated and the statistical out-of-sample performance of the input parameter selection procedure can be calculated.

The walk forward method described above is very time consuming and laborious. I use a TradeStation add-on product called The Power Walk Forward Optimizer (PWFO) in combination with a software product called the Walk Forward Performance Explorer (WFPE). With the PWFO, you tell it the test section length and the out-of-sample section length and it generates all the necessary test section and out-of-sample section files in an Excel csv format. For instance to set up, I input the E-mini data into a TradeStation chart from 12/03/03 to 07/09/04. I add the PWFO as a strategy in the TradeStation Strategy window along with the Acceleration Strategy. I tell the PWFO that I want a test period of 30 calendar days and a out-of-sample period of 7 calendar days. When the TradeStation optimization run is finished the PWFO generates 27 test/out-of-sample files. All files have the output format shown in Appendix A. I then use the Walk Forward Performance Explorer (WFPE) software on the PWFO optimization output on each of those 27 test sections to find the input parameters ***N***, ***aup***, and ***adn*** that will be used in the out of sample section immediately following each test section. The WFPE eliminates all cases in the PWFO test files that do not meet certain Profit Factor (PF), and Losers in a row (LR) criteria, as discussed above, and chooses the input parameters from the cases that is left. The WFPE explores a range of PF and LR criteria generating a number of searches in one run and automatically displays the best average PF-LR filter in Excel .

Below are the best ten results of the WFPE run sorted by the Student t statistic. The t statistic is calculated from the weekly out-of-sample net profits generated by the given PF-LR filter. The Student t statistic gives us what the probability is of the average out-of-sample profit for each PF-LR filter being due to pure chance. The higher the t statistic the lower the probability that the results are due to chance.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	es1fixma30xm	1040109	1040709	27														
2	Filter	Var	tOnp	mOnp	aOnp	aanp	aOnT	B0	%P	t	std	llw	eqDD	lr	#	b00	Blw	BE
3	pf20-4:16	mKr	10905	362	419	62.3	6.7	-0.6	0.73	3.93	544	-400	-400	2	26	438	4	6.7
4	pf25-a	nT	4127	176	172	59.0	2.9	3.0	0.67	2.42	348.6	-488	-488	1	24	176	6	16.4
5	pf25-4	nT	4127	176	172	59.0	2.9	3.0	0.67	2.42	348.6	-488	-488	1	24	176	6	16.4
6	pf25-3	nT	4127	176	172	59.0	2.9	3.0	0.67	2.42	348.6	-488	-488	1	24	176	6	16.4
7	pf25-5	nT	4127	176	172	59.0	2.9	3.0	0.67	2.42	348.6	-488	-488	1	24	176	6	16.4
8	pf20-a:16	mKr	10842	362	417	58.9	7.1	-1.4	0.73	3.98	534.5	-400	-400	2	26	437	4	6.6
9	pf20-5:16	mKr	10842	362	417	58.9	7.1	-1.4	0.73	3.98	534.5	-400	-400	2	26	437	4	6.6
10	pf25-3	lbr	5028	219	210	55.9	3.8	1.2	0.71	2.88	355.9	-513	-726	2	24	217	4	11.5

From this WFPE output we will select the filter on row 3.

Filter: PF>1 and PF<=2 and LR<=4, #Trds>=16 and highest mKr.

Where:

PF = Profit Factor in Test optimization section

LR=Maximum consecutive loses in a row in test optimization section

#Trds = The number of trades in the test run. We want our system to trade almost every day. There are 20 to 22 trading days a month, so we only want to look at input parameters that are able to generate at least 16 trades in the one month test sample.

mKr = Modified k-ratio = slope of equity curve/median deviation of equity from straight line fit to equity curve.

For the definitions of the WFPE columns please see Appendix B.

Results

Table 5 presents a table of the 27 test and out-of-sample windows and the selected optimum parameters and out-of-sample results using the WFPE filter described above.

Table 6 presents a specialized percentage trade by trade summary of the last four weeks of out-of-sample data segments from 4/12/04 and 5/7/04. where the out-of-sample input values for those out-of-sample weeks were taken from Table 5.

Discussion of System Performance

To see the effect of walk forward analysis take a look at Figure 5. Notice how the out-of-sample week 13 has only two trades and the out-of-sample week 23 has no trades. These were very narrow trading range weeks. The acceleration system did it's job by initiating few trades and avoiding whipsaw losses during the narrow trading range weeks. In out-of-sample week number 7 notice how the input parameters changed for that week based on the test section price data from 1/14/03 to 2/13/04. The aup changed from 6 in the test period 12/31/-3 to 1/30/04 to 20 in the test period 1/14/03 to 2/13/04. This change means the model is getting more bearish by increasing the level necessary for a buy signal and decreasing the level necessary for a sell signal. Looking at the out-of-sample profits column we can see that the walk forward system adapts quite rapidly to the different trading patterns that occurred during this twenty-seven week period.

At the bottom of Figure 5 are some statistics that are of interest. The first statistic ave=419.4 is the average weekly net profit for the 27 oos weeks. Std=544 is the standard deviation of the weekly return. Given a process that generates the above average and standard deviation, it would be interesting to know, statistically, the probabilistic outcome from trading this system for 13 weeks (one quarter). There is a statistical technique that we can use to determine this called bootstrap Ref [1]. Using the bootstrap technique we would randomly choose 13 of the twenty-seven weekly profits in Table 5, with replacement and take the sum of those 13 randomly chosen oos net profits. "With replacement" means we don't eliminate the randomly chosen weekly profit from being chosen again. Let us suppose that we repeat this random choosing of 13 weeks and summing the results 200 times. We would then have 200 different 13

week net profit summations. If we take the average and standard deviation of those 200 different 13 week net profit summations we would have an estimate of what to expect from this system by trading it for 13 weeks.

For this system the 200 bootstrap average of 13 week net profits is \$5453 and the standard deviation is \$1856. This means that at two standard deviations we can expect our 13 week return from this system to be between \$1741 and \$9165, 95% of the time. Please note that slippage and commissions were not taken into account.

References

1. Efron, B., Tibshirani, R.J., (1993), "An Introduction to the Bootstrap", New York, Chapman & Hall/CRC.

Table 5 Walk Forward Out-Of-Sample Performance Summary for E-Mini Least Squares Acceleration System

ES-1 min bars 12/3/2003 - 7/9/2004 using the below filter on each test segment. The input values N, aup, and adn are the values found from applying the filter to the PWFO files.

Filter= PF>=1; PF<=2.0; LR<=4; #Trds>=16; Largest modified K ratio.

Week #	Test Dates	oos Dates			osnp	Equity	ollt	odd	ont	avosnp	N	aup	adn
1	12/03/03	01/02/04	01/05/04	01/09/04	550	550	-225	-400	11	50	30	8	18
2	12/10/03	01/09/04	01/12/04	01/16/04	750	1300	-588	-600	15	50	32	4	18
3	12/17/03	01/16/04	01/19/04	01/23/04	-50	1250	-425	-538	8	-6.3	36	4	16
4	12/24/03	01/23/04	01/26/04	01/30/04	-338	912	-525	-1200	15	-22.5	36	6	20
5	12/31/03	01/30/04	02/02/04	02/06/04	325	1237	-188	-188	7	46.4	40	6	20
6	01/07/04	02/06/04	02/09/04	02/13/04	-50	1187	-300	-300	4	-12.5	40	12	20
7	01/14/04	02/13/04	02/16/04	02/20/04	213	1400	-175	-313	5	42.6	38	20	14
8	01/21/04	02/20/04	02/23/04	02/27/04	488	1888	0	0	3	162.7	34	20	20
9	01/28/04	02/27/04	03/01/04	03/05/04	400	2288	-413	-475	6	66.7	40	12	16
10	02/04/04	03/05/04	03/08/04	03/12/04	2225	4513	-263	-263	6	370.8	40	20	8
11	02/11/04	03/12/04	03/15/04	03/19/04	488	5001	-263	-338	7	69.7	40	18	14
12	02/18/04	03/19/04	03/22/04	03/26/04	213	5214	-338	-663	18	11.8	32	20	16
13	02/25/04	03/26/04	03/29/04	04/02/04	-250	4964	-275	-275	2	-125	40	18	16
14	03/03/04	04/02/04	04/05/04	04/09/04	463	5427	-13	-13	3	154.3	36	20	14
15	03/10/04	04/09/04	04/12/04	04/16/04	188	5615	-388	-450	8	23.5	36	20	12
16	03/17/04	04/16/04	04/19/04	04/23/04	1050	6665	-550	-550	6	175	40	6	20
17	03/24/04	04/23/04	04/26/04	04/30/04	788	7453	-388	-388	13	60.6	38	18	16
18	03/31/04	04/30/04	05/03/04	05/07/04	775	8228	-138	-175	6	129.2	40	18	16
19	04/07/04	05/07/04	05/10/04	05/14/04	1025	9253	-175	-175	5	205	40	16	20
20	04/14/04	05/14/04	05/17/04	05/21/04	788	10041	-138	-138	5	157.6	40	16	14
21	04/21/04	05/21/04	05/24/04	05/28/04	288	10329	0	0	1	288	38	20	18
22	04/28/04	05/28/04	05/31/04	06/04/04	0	10329	-175	-175	3	0	40	16	12
23	05/05/04	06/04/04	06/07/04	06/11/04		10329					36	20	18
24	05/12/04	06/11/04	06/14/04	06/18/04	-50	10279	-200	-200	3	-16.7	36	16	18
25	05/19/04	06/18/04	06/21/04	06/25/04	263	10542	-63	-63	2	131.5	34	14	20
26	05/26/04	06/25/04	06/28/04	07/02/04	763	11305	-213	-313	9	84.8	32	20	14
27	06/02/04	07/02/04	07/05/04	07/09/04	-400	10905	-225	-438	4	-100	38	12	18
			OSNP AVE =	419.4									
			OSNP STD =	544									
		200 Boot	13wk Sum AVE =	5453.6									
		200 Boot	13wk Sum STD =	1856.6									

Where

PF = Profit Factor in Test optimization section

LR=Maximum consecutive loses in a row in test optimization section

osnp = Out-of-sample net profit for that week.

Equity = running sum of the weekly out-of-sample net profits

ollt = Largest losing trade in that week

odd = largest close drawdown in that week

ont = The number of trades in the out-of-sample week.

Avosnp = the average out-of-sample trade profit in that week.

Table 6 Specialized Trade By Trade Summary
E-Mini 1 min Bars Least Squares Acceleration System 4/12/2004 - 5/7/2004

Trade #	Entry Date	Entry Time		Entry Price	Exit Time	Exit Price	Bars InTrd	Trade P&L	Equity	Trade Max Pft	Time	Trade Max DD	Time
1	04/12/04	1001	Sell	1143.50	1610	1145.00	369	(\$75)	(\$75)	\$63	1326	(\$125)	1036
2	04/13/04	1001	Sell	1142.25	1610	1129.00	369	\$663	\$588	\$775	1509	\$0	1001
3	04/14/04	1001	Buy	1125.75	1023	1126.75	22	\$50	\$638	\$175	1013	(\$13)	1001
4	04/14/04	1023	Sell	1126.75	1610	1128.50	347	(\$88)	\$550	\$288	1517	(\$238)	1108
5	04/15/04	1003	Sell	1128.75	1542	1128.75	339	\$0	\$550	\$475	1412	(\$63)	1016
6	04/15/04	1542	Buy	1128.75	1556	1127.75	14	(\$50)	\$500	\$13	1549	(\$50)	1555
7	04/15/04	1556	Sell	1127.75	1610	1126.25	14	\$75	\$575	\$75	1610	(\$25)	1556
8	04/16/04	1001	Sell	1126.25	1610	1134.00	369	(\$388)	\$188	\$13	1002	(\$475)	1251
9	04/19/04	1020	Buy	1129.75	1610	1134.25	350	\$225	\$413	\$275	1531	(\$50)	1042
10	04/20/04	1124	Buy	1134.50	1540	1123.50	256	(\$550)	(\$138)	\$25	1126	(\$575)	1536
11	04/20/04	1540	Sell	1123.50	1610	1115.00	30	\$425	\$288	\$488	1604	(\$13)	1541
12	04/21/04	1003	Buy	1121.00	1610	1122.75	367	\$88	\$375	\$163	1425	(\$313)	1139
13	04/22/04	1001	Buy	1123.50	1610	1138.00	369	\$725	\$1,100	\$925	1459	(\$163)	1029
14	04/23/04	1023	Buy	1136.75	1610	1139.50	347	\$138	\$1,238	\$200	1438	(\$150)	1132
15	04/26/04	1020	Sell	1139.25	1610	1136.25	350	\$150	\$1,388	\$400	1512	(\$50)	1023
16	04/27/04	1015	Buy	1145.75	1031	1143.50	16	(\$113)	\$1,275	\$0	1015	(\$125)	1024
17	04/27/04	1031	Sell	1143.50	1610	1137.75	339	\$288	\$1,563	\$413	1447	(\$38)	1339
18	04/28/04	1001	Buy	1130.25	1610	1122.50	369	(\$388)	\$1,175	\$25	1005	(\$500)	1526
19	04/29/04	1007	Buy	1123.25	1030	1126.00	23	\$138	\$1,313	\$225	1016	\$0	1007
20	04/29/04	1030	Sell	1126.00	1445	1111.75	255	\$713	\$2,025	\$788	1437	(\$75)	1102
21	04/29/04	1445	Buy	1111.75	1608	1113.00	83	\$63	\$2,088	\$138	1556	(\$238)	1522
22	04/29/04	1608	Sell	1113.00	1610	1113.25	2	(\$13)	\$2,075	\$0	1608	(\$25)	1608
23	04/30/04	1001	Sell	1114.50	1051	1114.75	50	(\$13)	\$2,063	\$213	1034	(\$125)	1008
24	04/30/04	1051	Buy	1114.75	1140	1113.00	49	(\$88)	\$1,975	\$138	1120	(\$100)	1137
25	04/30/04	1140	Sell	1113.00	1543	1108.75	243	\$213	\$2,188	\$350	1525	(\$125)	1343
26	04/30/04	1543	Buy	1108.75	1604	1105.25	21	(\$175)	\$2,013	\$88	1551	(\$175)	1603
27	04/30/04	1604	Sell	1105.25	1610	1105.00	6	\$13	\$2,025	\$38	1609	\$0	1604
28	05/04/04	1426	Buy	1119.00	1513	1121.75	47	\$138	\$2,163	\$375	1500	(\$113)	1427
29	05/04/04	1513	Sell	1121.75	1556	1118.50	43	\$163	\$2,325	\$338	1539	\$0	1513
30	05/04/04	1556	Buy	1118.50	1610	1115.75	14	(\$138)	\$2,188	\$13	1559	(\$138)	1610
31	05/06/04	1001	Buy	1115.00	1031	1114.25	30	(\$38)	\$2,150	\$175	1004	(\$50)	1026
32	05/06/04	1031	Sell	1114.25	1610	1113.50	339	\$38	\$2,188	\$475	1244	(\$50)	1532
33	05/07/04	1009	Sell	1109.00	1610	1096.75	361	\$613	\$2,800	\$613	1609	(\$200)	1048

Appendix A: An Example of The Power Walk Forward Optimizer Output File

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
I	pw	N	aup	adn	xop	xt	mntp	trp	nT	pP	PF	std	t	mlb	lbr	mwb	wbr	awl	wlb	mru	mdd	wr	lr
2	2	38	18	16	30	1610	62.5	2275	26	69	3.22	214	2.09	21	797	243	3580	11.57	4.49	125	163	9	3
3	2	34	16	18	30	1610	50.0	2988	47	66	2.35	210	2.08	26	1018	51	4564	1.96	4.48	100	138	16	5
4	2	24	18	8	30	1610	0.0	3100	225	51	1.43	109	1.89	11	3394	24	4307	2.18	1.27	88	50	8	7
5	2	24	18	10	30	1610	-12.5	2938	222	50	1.41	107	1.84	12	3418	25	4250	2.08	1.24	88	50	8	7
6	2	32	20	20	30	1610	25.0	2513	39	64	2.31	222	1.81	25	1350	50	3098	2.00	2.30	100	125	7	3
7	2	32	18	20	30	1610	37.5	2588	43	65	2.18	219	1.80	25	1707	50	3479	2.00	2.04	100	150	9	3
8	2	40	14	14	30	1610	75.0	2138	34	68	2.37	204	1.80	49	1256	84	3540	1.71	2.82	113	138	6	2
9	2	24	20	10	30	1610	-12.5	2900	194	49	1.46	116	1.80	12	3570	24	4090	2.00	1.15	88	50	10	7
10	2	38	20	16	30	1610	50.0	1763	23	61	2.70	221	1.66	45	1135	245	2759	5.44	2.43	150	163	3	3
11	2	38	14	16	30	1610	50.0	2100	38	68	2.08	205	1.66	23	930	76	3876	3.30	4.17	100	150	15	5
12	2	24	20	8	30	1610	0.0	2700	195	51	1.41	117	1.65	12	3851	22	3844	1.83	1.00	88	50	6	7
13	2	24	18	6	30	1610	-12.5	2713	230	49	1.35	110	1.63	11	3605	22	4101	2.00	1.14	88	50	11	7
14	2	34	16	20	30	1610	50.0	2350	37	73	2.13	238	1.62	18	1258	76	4286	4.22	3.41	88	188	9	3
15	2	28	20	12	30	1610	0.0	2363	110	53	1.50	144	1.56	15	3627	34	3763	2.27	1.04	113	63	5	6
16	2	40	12	14	30	1610	37.5	1775	44	68	1.89	172	1.56	23	1274	76	4144	3.30	3.25	113	125	7	2
17	2	28	20	4	30	1610	0.0	2600	117	56	1.50	157	1.53	12	3384	28	4288	2.33	1.27	100	50	6	6
18	2	38	16	16	30	1610	37.5	1750	30	63	2.23	210	1.52	22	896	242	3500	11.00	3.91	113	125	7	3
19	2	40	20	14	30	1610	25.0	1575	22	59	2.45	221	1.52	57	1501	271	2755	4.75	1.84	150	113	3	2
20	2	40	14	16	30	1610	62.5	1725	30	67	2.10	215	1.46	23	1084	82	2975	3.57	2.74	113	150	9	2
21	2	26	20	6	30	1610	0.0	2400	151	54	1.40	134	1.46	12	3587	23	4094	1.92	1.14	88	50	10	8

	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO
I	mpft	mlos	p/l	dd	llt	eqB0	eqR2	eqDev	eqB1	eqDy	eqB2	eqR22	mKr	osmp	odd	ollt	ont	aomp
2	138	-113	1.22	-388	-388	102	92	150	125	67	-1.1	93	68.2	163	-313	-213	12	13.5
3	125	-88	1.43	-638	-488	68	86	335	133	-4	-1.5	95	20.2	350	-500	-350	17	20.6
4	75	-50	1.50	-938	-288	11	77	268	19	3	0.0	82	4.2	(150)	-1475	-313	66	-2.3
5	75	-50	1.50	-788	-288	12	81	233	19	4	0.0	84	5.1	(25)	-1425	-300	66	-0.4
6	125	-88	1.43	-613	-488	66	90	196	104	26	-1.0	93	33.7	325	-375	-300	16	20.3
7	125	-75	1.67	-575	-488	66	92	191	98	30	-0.8	94	34.5	1100	-375	-300	16	68.8
8	100	-88	1.14	-413	-388	66	93	123	83	49	-0.5	93	53.3	(625)	-775	-663	11	-56.8
9	88	-38	2.33	-700	-288	16	91	211	11	17	0.0	89	7.4	(263)	-1288	-338	55	-4.8
10	138	-88	1.57	-388	-388	86	89	125	111	56	-1.2	91	69.2	450	-338	-225	9	50.0
11	125	-113	1.11	-575	-388	58	79	232	142	-26	-2.2	93	24.8	(925)	-1113	-663	16	-57.8
12	75	-50	1.50	-838	-288	14	87	237	7	16	0.0	83	5.7	(388)	-1325	-338	55	-7.0
13	75	-50	1.50	-913	-288	9	68	304	19	-1	0.0	76	3.1	(125)	-1225	-288	66	-1.9
14	100	-100	1.00	-613	-488	75	90	202	93	51	-0.6	91	37.0	375	-400	-300	17	22.1
15	113	-75	1.50	-813	-288	19	83	208	22	15	0.0	83	9.2	(725)	-1063	-288	45	-16.1
16	125	-75	1.67	-613	-388	34	69	203	120	-46	-1.9	91	16.8	(200)	-775	-650	15	-13.3
17	100	-63	1.60	-925	-350	19	69	349	11	24	0.1	67	5.4	(600)	-1250	-275	45	-13.3
18	138	-75	1.83	-425	-388	59	87	150	87	33	-0.9	88	39.4	138	-300	-225	12	11.5
19	138	-75	1.83	-388	-388	74	89	113	54	85	0.7	89	65.4	538	-425	-288	7	76.8
20	88	-175	0.50	-525	-388	59	83	185	111	9	-1.7	88	32.2	(600)	-938	-675	9	-66.7
21	75	-63	1.20	-900	-325	17	85	227	6	25	0.1	82	7.5	(763)	-1463	-288	49	-15.6

Column Definition For The Test Section:

Sorted on Student t column

Len, aup, adn, xop, xt = Input Parameters.

mntp=median TrdProfit;

Column Definition For The Test Section Continued:

tnp=Net Profits; nT=# of Trades;
pP=% profitable;
PF=Profit Factor;
std=Standard Deviation of TrdProfits
t=Student t statistic for Net Profits;
mlb=median losing bars;
lbr=Total Losing Bars;
mwb=median of winning bars;
awl=mwb/mlb; **wlb**=wbr/lbr; **mr**=median(max\$TrdRU-TrdPft); **mdd**=median(TrdPft-Max\$TrdDD);
wr=Max winners in a row;
lr=Max losers in a row;
mpft=median of winning trades;
mlos=median of losing trades;
p/l=mpft/mlos;
dd=drawdown;
llt=largest losing trade;
eqB0=slope of equity curve;
eqR2=R² of equity curve;
ended=median deviation of equity from straight line fit to equity curve;
eqB1=slope of 2nd order polynomial fit to equity curve;
eqDy=velocity of 2nd order polynomial fit;
eqB2=acceleration of 2nd order fit.;
EqR22=R² of 2nd order fit;
mKr=Modified k-ratio=eqB0/Dev.

Column Definition For Out-Of-Sample Section:

Osnp=Out-Of-Sample(OOS) Net Profit;
odd=OOS drawdown;
ollt=OOS largest losing trade;
ont=OOS # of trades;
aonsp=ave OOS trade profit.

Note:

For the PWFO we use the median instead of the average whenever we can. The median is a much more robust statistic than the average. As an example look at row 5 on the above spreadsheet. The median trade profit is minus **-\$12.50** while the average trade profit is a plus **\$13.23**. The average is controlled by a few outliers while the median is immune to these few outliers.

Appendix B: The Walk Forward Performance Explorer.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	es1fixma30xm	1040109	1040709	27														
2	Filter	Var	tOnp	mOnp	aOnp	aanp	aOnT	B0	%P	t	std	llw	eqDD	lr	#	b00	Blw	BE
3	pf20-4:16	mKr	10905	362	419	62.3	6.7	-0.6	0.73	3.93	544	-400	-400	2	26	438	4	6.7
4	pf25-a	nT	4127	176	172	59.0	2.9	3.0	0.67	2.42	348.6	-488	-488	1	24	176	6	16.4
5	pf25-4	nT	4127	176	172	59.0	2.9	3.0	0.67	2.42	348.6	-488	-488	1	24	176	6	16.4
6	pf25-3	nT	4127	176	172	59.0	2.9	3.0	0.67	2.42	348.6	-488	-488	1	24	176	6	16.4
7	pf25-5	nT	4127	176	172	59.0	2.9	3.0	0.67	2.42	348.6	-488	-488	1	24	176	6	16.4
8	pf20-a:16	mKr	10842	362	417	58.9	7.1	-1.4	0.73	3.98	534.5	-400	-400	2	26	437	4	6.6
9	pf20-5:16	mKr	10842	362	417	58.9	7.1	-1.4	0.73	3.98	534.5	-400	-400	2	26	437	4	6.6
10	pf25-3	lbr	5028	219	210	55.9	3.8	1.2	0.71	2.88	355.9	-513	-726	2	24	217	4	11.5

The WFPE Columns are defined as follows

- **Row 1** is the PWFO Stub, File Start Date, File End Date and number of weeks or number of PWFO files processed.
- **Filter Definition.** The filter is abbreviated in the following manner.
 - The first part of the filter is the profit factor filter designated by pf{ (2 digit number with decimal place after first number) or xx }. The number after pf means <=num1.num2. Thus pf35 would mean pf<=3.5. pfx would mean any or all pf's. That is, the xx means that the filter doesn't filter for pf.
 - The second part of the filter is the losers in a row (LR) filter. This is denoted by a -LR after the pf filter part as in pf35-5. pf35-5 filters is pf<=3.5 and LR<=5. if LR is set to "a" as in pf35-a then the filter doesn't filter for LR and all LR's are included in the filter. A filter of pfx-a would mean that there is no pf or LR filter and the filter is just the maximum of the VAR column.
 - The third part of the filter is the Min # Trds. This is designated by a ":number" after the LR portion of the filter as in pf35-5:19. Here the ":19" means that the minimum number of trades in the test section has to be greater than or equal to 19.
 - The fourth part of the filter is the performance variables(Var). After the PF and LR filters are applied to a PWFO output file there are anywhere from 0 to 100's of rows or cases left. If there are 0 cases left after the application of the PF-LR filter, this means that none of the cases satisfied the filter requirements. Excluding the PF and LR columns there are 28 columns of performance variables. Out of the rows that are left the program finds the maximum or minimum (see below for minimum variables) for each of the 28 columns. This creates 28 different filters for any given PFLR filter. As an example the first filter above is pf20-3:19 Var=r22. This means that for the filter pf20-3:19 that the maximum value in the r22 column is chosen leaving only one case. So the filter should be called pf20-3:19-maxR22. Note if there is more than one row of the maximum value for a variable the tie is broken with the row that has the highest "t" value.
 - The variables that use a minimum rather than a maximum are: nT, std, mlb, lbr, mru, mlos, dd, llt, dev. This variables are defined on pages II-1 and II=2 of the PWFO manual.
- **Var** = The PWFO variable. For this PFLR filter the maximum or minimum of this variable produced the following average statistics on this line.
- **tOnp** = Total out-of-sample(oos) net profit for the 27 weeks.
- **mOsp** = median oos net profit for the 27 weeks
- **aOsp** = Average oos net profit for the 27 weeks

- **aanp** = Average oos profit per trade
- **aOnT** = Average number of oos trades per week
- **B0** = The 27 week trend of the out-of-sample weekly profits
- **%P** = The percentage of oos weeks that were profitable
- **t** = The student t statistic for the 27 weekly oos profits. The higher the t statistic the higher the probability that this result was not due to pure chance
- **std** = The standard deviation of the 27 weekly oos profits
- **llw** = The largest losing oos week
- **eqDD** = The oos equity drawdown
- **lr** = The largest number of losing oos weeks in a row
- **#** = The number of weeks this filter produced a case. Note for some weeks there can be no cases that satisfy a given filter's criteria.
- **b00** = The straight line trend of the oos equity curve in \$/week.
- **Blw** = The maximum number of weeks the oos equity curve was below the straight line equity fit.
- **BE** = Break even weeks. Assuming the average and standard deviation are from a normal distribution, this is the number of weeks you would have to trade to have a 99% probability that your oos equity is above zero.