

Trading the S&P500 E-Mini With The Robust Regression Velocity Strategy

Copyright © October, 2014 Dennis Meyers

In previous working papers [Ref 4, 5] we examined a trading system that used the velocity of prices fit by a least squares straight line through “N” past prices, to determined buy and sell points. The reasoning behind this type of system was to only trade when the straight line slope or velocity was above a certain threshold. Many times during the day prices meandering around without a notable trend. At these times we do not wish to trade because of the whipsaws 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 threshold value. Thus the velocity system would only issue a trade when certain velocity barriers were crossed.

The Least Squares polynomial is determined by minimizing the sum of the squares of the difference between the N prices and the value of the polynomial line.

$$\text{err}^2(t) = [\text{Price}(t) - (a + b * t)]^2 = \text{error squared}$$

$$\text{Minimize}(a, b) \sum_{t=1}^{t=N} \text{err}^2(t)$$

This mathematical technique has an exact solution and dates back to Gauss in the 1800's.

Recently much work has been done in what is called robust regression and outlier detection techniques, Ref [1]. Robust regression techniques are now defined by a measure called the “breakdown point”. The breakdown point is loosely defined as the smallest amount of bad data points that can cause the regression coefficient solutions to take on values some distance from their true values. Unfortunately the Least Squares technique has a breakdown point of 1/N. In other words only one bad data point can significantly change the computation of the velocity or slope of a straight line. The median of a set of numbers has a breakdown point of 50% . This is because when 50% of the numbers are bad then there is no way of telling which are the bad numbers and which are the good numbers. 50% is the highest breakdown point.

The least absolute deviation (LAD) regression estimator from Ref [1] is

$$\text{Minimize}(a, b) \sum_{i=1}^{i=N} \text{absolute value}[\text{err}(i)]$$

and has a breakdown point of 29.8% . For the LAD this means around ¼ of the price points can be bad before the computations of a and b become erroneous. Siegel Ref[2], in his paper “Robust regression using repeated medians”, introduced a technique for finding the slope that has a 50% breakpoint. The repeated median is also described in Ref [1].

While the repeated median technique may sound complicated it is quite easy to compute. Here's how. For demonstration purposes let's suppose we have 15 data points on an x,y graph such that,

X	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Y	1	2	10	4	5	6	7	8	9	18	11	12	13	18	15	20

We've added two bad Y points at X positions 10 and 14. To calculate the repeated median slope we would take the slope of every pair of y values and then find the median of all the pairs of slopes. For this example we would take

slope	1	$y(2)-y(1)/(2-1) =$	1.00
slope	2	$y(3)-y(1)/(3-1) =$	4.50
slope	3	$y(4)-y(1)/(4-1) =$	1.00
slope	4	$y(5)-y(1)/(5-1) =$	1.00
slope	5	$y(6)-y(1)/(6-1) =$	1.00
slope	6	$y(7)-y(1)/(7-1) =$	1.00
slope	7	$y(8)-y(1)/(8-1) =$	1.00
slope	8	$y(9)-y(1)/(9-1) =$	1.00
slope	9	$y(10)-y(1)/(10-1) =$	1.89
slope	10	$y(11)-y(1)/(11-1) =$	1.00
slope	11	$y(12)-y(1)/(12-1) =$	1.00
slope	12	$y(13)-y(1)/(13-1) =$	1.00
slope	13	$y(14)-y(1)/(14-1) =$	1.31
slope	14	$y(15)-y(1)/(15-1) =$	1.00
slope	14	$y(16)-y(1)/(16-1) =$	1.27
		Median =	1.00

The median slope of the above is 1. The above process is repeated for:

$$(y(2)-y(i))/(2-i), i=1 \text{ to } 15 \ i \neq 2,$$

$$(y(3)-y(i))/(3-i), i=1 \text{ to } 15 \ i \neq 3,$$

.....

.....

$$(y(16)-y(i))/(16-i), i=1 \text{ to } 16 \ i \neq 16.$$

The final slope is then the **median of all the medians calculated above**. While the repeated median looks redundant because the very first calculation produced the correct slope, price data is not so nicely distributed as our example and the extra calculations are needed to assure that the outliers are eliminated.

The mathematical formula for the above is

$$\text{Slope}(t) = \text{median}_i \{ \text{median}_{i \neq j} [\text{price}(t) - \text{price}(t-i) / (i-j)] \}$$

$i=1 \text{ to } N$ $j=1 \text{ to } N$

Figure 1 below shows a plot of the x,y numbers above with the repeated median line and the least squares line on the graph. Notice how the bad points draw the least squares line towards them while the repeated median line is completely unaffected by the outliers. The least Squares line is given by the formula $y = -0.65 + 1.1074 * x$. The true line is given by the formula $y = x$.

From this simple example we can observe how noise has distorted the least squares estimates of **a** and **b**, where $y=a+bx$.

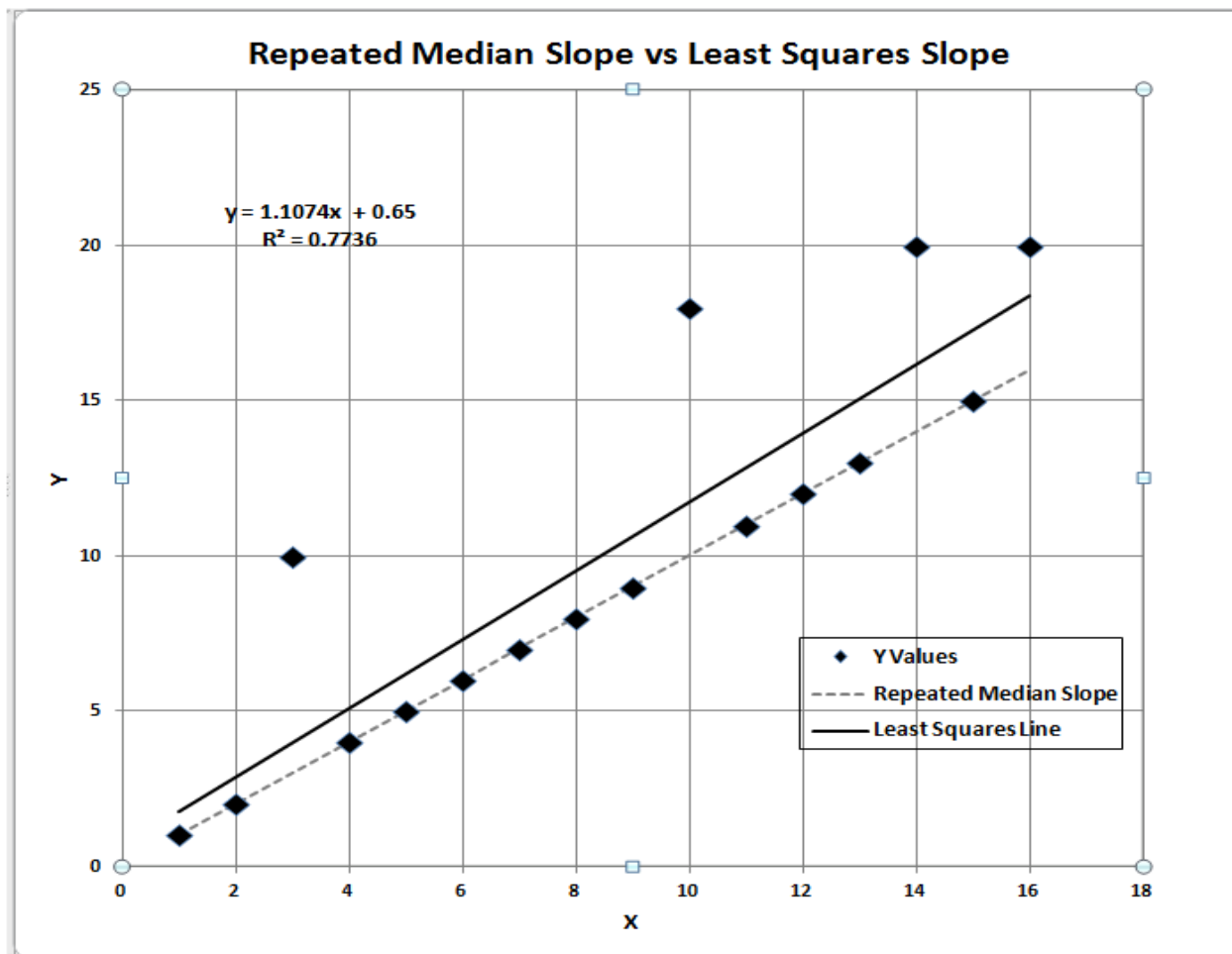


Figure 1 Repeated Median Slope vs Least Squares Slope.

The Repeated Median Velocity(RMV) System Defined

Here we will use the repeated median slope to create a trading system. For a straight line the velocity is equal to the slope. The repeated median velocity, also called the **robust velocity**, has the advantage that it is a natural random price noise inhibitor. We can create a system such that unless the repeated median velocity using N past price bars is greater than some threshold value we will not buy or sell. A large percentage of price movements are just noise which generates a lot of back and forth movements of small magnitudes. This back and forth movement creates many false buy and sell signals. However using the repeated median velocity over N past prices, we will attempt to filter out many of the small price noise movements by requiring that the repeated median velocity to be greater than some threshold before we act.

At each price bar we calculate the repeated median velocity (**RMV**) from the formula above. 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.

The Repeated Median Velocity Trading Strategy

Buy Rule:

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

Sell Rule:

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

Intraday Bars Exit Rule:

Close all positions 15 minutes before the ES close (no trades will be carried overnight).

First Trade of Day Entry Rule:

All trade signals before 30 minutes after the open are ignored. We've included this rule because with overnight trading there are often gaps in the open creating immediate strategy 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 the opening

Discussion of S&P500 Index E-Mini Future Prices

The **S&P 500 Index E-Mini Future (ES)** is traded on the CME Futures Exchange and is traded on a 22 hour basis. We have restricted our study to only trading the ES during the stock market hours of 8:30 to 1515 CST. To test this system we will use 1 minute bar prices of the ES E-Mini futures contract for the four years from September 2, 2010 to September 5, 2014

Testing The Repeated Median Velocity System (RMV)

Using Walk Forward Optimization

There are three strategy inputs to determine:

1. *N*, is the lookback period to calculate the **RMV**.
2. *vup*, the threshold amount that RMV has to be greater than to issue a buy signal
3. *vdn*, the threshold amount that RMV has to be less than to issue a sell signal

We will test the RMV strategy with the above ES 1 min bars on a *walk forward basis*, as will be described below.

What Is A Walk Forward Optimization with In-Sample Section and Out-Of-Sample Sections?

Whenever we do a TS optimization on a number of different strategy inputs, TS generates a report of performance metrics (total net profits, number of losing trades, etc) vs these different strategy inputs. If the report is sorted on say the total net profits(*tnp*) performance metric column then the highest *tnp* would correspond to a certain set of inputs. This is called a *in-sample section*. If we choose a set of strategy inputs from this report based upon some

performance metric, we have no idea whether these strategy inputs will produce the same results on future price data or data they have not been tested on. Price data that is not in the in-sample section is defined as *out-of-sample data*. Since the performance metrics generated in the in-sample section are mostly due to “curve fitting” or “data mining” it is important to see how the strategy inputs chosen from the in-sample section perform on out-of-sample data.

Walk forward analysis attempts to minimize the curve fitting of price noise 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 in-sample and out-of-sample sections. Usually for any strategy, one has some performance metric selection procedure, which we will call a *filter*, used to select the input parameters from the optimization run. For instance, a *filter* example 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 in-sample section output and its associated strategy input parameters. Now suppose we ran our optimization on each of our many in-sample sections and applied our filter to each in-sample section output. We would then use the strategy input parameters found by the *filter* in each in-sample section on the out-of-sample section immediately following that in-sample section. The input parameters found in each in-sample section and applied to each out-of-sample section would produce independent net profits or 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, then we will have an estimate of how our system will perform on average. Due to the Central Limit Theorem, as your sample size increases, the spurious noise results in the out-of-sample section performance tend to average out to zero in the limit, leaving us with what to expect from our strategy and filter. *Mathematical note: This assumption assumes that the out-of-sample returns are from probability distributions that have a finite variance.*

Why use the walk forward technique? Why not just perform an optimization on the whole price series and choose the input parameters that give the best total net profits or profit factor? Surely the price noise cancels itself out with such a large number of in-sample prices and trades. Unfortunately, nothing could be farther from the truth! Optimization is a misnomer and should really be called combinatorial search. As stated above, whenever we run a combinatorial search over many different combinations of input parameters on noisy data on a fixed number of prices, *no matter how many*, the best performance parameters found are guaranteed to be due to “*curve fitting*” the noise and signal. What do we mean by “*curve fitting*”? As a simple example, suppose you were taking the Train to work. In the train car you are in suppose you counted the number of blond women in that car and suppose the percent of blond women vs all other women hair colors was 80%. Being that you can't observe what is in the other train cars, you would assume that all the other train cars had the same percentage of blond women. That is an example of curve fitting. The same goes for combinatorial searches. You are observing results from a finite sample of data without knowing the data outside the sample you examined. The price series that we trade consists of random spurious price movements, which we call noise, and repeatable price patterns (*if they exist*). When we run, for example, 5000 different inputs parameter combinations, the best performance parameters will be from those system input variables that are able to produce profits from the price pattern *and* the random spurious movements. While the price patterns will repeat, the same spurious price movements will not. If the spurious price movements that were captured by a certain set of input parameters were a large part of the total net profits, then choosing these input parameters will produce losses when

traded on future data. These losses occur because the spurious price movements will not be repeated in the same way. This is why system optimization or combinatorial searches with no out-of-sample testing cause losses when traded in real time from something that looked great in the in-sample section. Unfortunately it is human nature to extrapolate past performance to project future trading results and thus results from curve fitting give the illusion, a modern “siren call” so to speak, of future trading profits.

In order to gain confidence that our input parameter selection method using the optimization output of the in-sample data will produce profits, we must test the input parameters we found in the in-sample section on out-of-sample data. In addition, we must perform the in-sample/out-of-sample analysis many times. Why not just do the out-of-sample analysis once or just 10 times? Well just as in Poker or any card game, where there is considerable variation in luck from hand to hand, walk forward out-of-sample analysis give considerable variation in week-to-week out-of-sample profit “luck”. That is, by pure chance we may have chosen some input parameter set that did well in the in-sample section data *and* the out-of-sample section data. In order to minimize this type of “luck”, statistically, we must repeat the walk forward out-of-sample (oos) analysis over many (>30) in-sample/out-of-sample sections and take an average over all out-of-sample sections. This average gives us an expected out-of-sample return and a standard deviation of out-of-sample returns which allows us to statistically estimate the expected equity and its range for N out-of-sample periods in the future

Finding The System Parameters Using Walk Forward Optimization

There are three strategy parameters to find N , vup and vdn .

For the test data we will run the TradeStation optimization engine on ES 1min price bars from 9/2/2010 to 9/5/2014 with the following optimization ranges for the repeated median strategy inputs. I will create 205 30 day in-sample periods each followed by a 7 day out-of-sample period (See Figure 1 for the in-sample/out-of-sample periods).

1. N from 20 to 70 in steps of 10
2. vup from 0.2 to 3.6 steps of 0.2
3. vdn from 0.2 to 3.6 in steps of 0.2
4. $Mult=1.6*\sqrt{N}$ Note: this normalizes each N of RMedV to 0 to 3.6 range. Else RMedV would have different ranges for different N .

This will produce 1944 different input combinations or cases of the strategy input parameters for each of the 205 in-sample/out-of-sample files for the two years of 1min bar ES data.

The question we are attempting to answer statistically is which best performance metric or combination of best performance metrics (which we will call a *filter*) applied to the in-sample section will produce strategy inputs that produce statistically valid profits in the out-of-sample section. In other words we wish to find a performance metric *filter* that we can apply to the in-sample section that can give us strategy inputs that will produce, on average, good trading results in the future.

When TS does an optimization over many combinations of inputs, it creates output page that has as its rows each strategy input combination and as its columns various trading performance

measures such as Profit Factor, Total Net Profits, etc. An example of a simple filter would be to choose the strategy input optimization row in the in-sample section that had the highest Net Profit or perhaps a row that had the best Profit Factor with their associated strategy inputs. Unfortunately it was found that this type of simple metric performance filter very rarely produces good out-of-sample results. More complicated metric filters can produce good out-of-sample results minimizing spurious price movement biases in the selection of strategy inputs.

Here is an *example* of a better more complicated *filter* that was used in this paper. We require that the number of trades (**NT**) in the in-sample section be greater or equal to than 10 trades per month. Since there are 21 trading days per month this forces a trade a minimum of half the trading days per month. Not many traders can stay with a strategy that has a large number of losers in a row (**LR**). For this filter we will choose **LR** ≤ 5. This choice of LR is completely arbitrary and is what I feel comfortable with. In addition, from experience, it is known that most curve fitted strategy results have high Profit Factors (PF). So for our filter we will restrict the PF ≤ 4. After using the **NT-LR-PF** filter, as described, there can still be 100's of rows left in the in-sample section. There is a performance metric called **The Median Of The Absolute Deviations of Equity From a Straight Line Fit To The Equity Curve(mDEV)**. Let us choose the 50 rows in the in-sample section that contain the **minimum mDEV** values from the rows that are left from the **NT-LR-PF** screen. In other words we sort *mDEV* from low to high, eliminate the rows that have **NT** < 10, **LR** > 5 and **PF** > 4 and then choose the smallest *mDEV* 50 Rows of whatever is left. This particular filter will now leave 50 cases or rows in the in-sample file that satisfy the above filter conditions. We call this filter **b50mDEV|p<4.0|lr5>10** where **b50 mDEV** means the bottom or minimum 50 *mDEV* rows left *after* the **NT-LR-PF** filter. Suppose for this filter, within the 50 in-sample rows that are left, we want the row that has the highest metric called **The Modified K-Ratio(mkr) = Equity trend/mDEV** in the in-sample section. We abbreviate this final filter as **b50mDEV|p<4.0|lr5>10-mkr**. For each in-sample section this filter leaves only one row in the in-sample section with its associated strategy inputs and out-of-sample net profit in the out-of-sample section using the strategy inputs found in the in-sample section. This particular **b50mDEV|p<4.0|lr5>10-mkr filter** is then applied to each of the 205 in-sample sections which give 205 sets of strategy inputs that are used to produce the corresponding 205 out-of-sample performance results. The average out-of-sample performance is calculated from these 205 out-of-sample performance results. In addition many other important out-of-sample performance statistics for this filter are calculated and summarized. **Figure 3** shows such a computer run along with a small sample of other filter combinations that are constructed in a similar manner. Row 3 of the sample output in Figure 3 shows the results of the filter discussed above.

Bootstrap Probability of Filter Results.

Using modern "Bootstrap" techniques, we can calculate the probability of obtaining our filter's total out-of-sample *net* profits by chance. Here is how the bootstrap technique is applied. Suppose as an example, we have 100 files of in-sample/out-of-sample data. A mirror random filter is created. Instead of picking an out-of-sample net profit (OSNP) from a filter row as before, the mirror filter picks a *random* row's OSNP in each of the 100 files. We repeat this random picking in each of the 100 files 5000 times. Each of the 5000 mirror filters will choose a random row's OSNP of their own in each of the 100 files.. At the end, each of the 5000 mirror filters will have 100 *random* OSNP's picked from the rows of the 100 files. The sum of the 100 random OSNP picks for each mirror filter will generate a random total out-of-sample net profit

(tOnpNet) or final random equity. The average and standard deviation of the 5000 mirror filter's different random tOnpNets will allow us to calculate the chance probability of our above chosen filter's tOnpNet. Thus given the mirror filter's bootstrap random tOnpNet average and standard deviation, we can calculate the probability of obtaining our chosen filter's tOnpNet by pure chance alone. Figure 3 lists the 5000 mirror filter's bootstrap average for our 205 out-of-sample files of **(\$16284)** with a bootstrap standard deviation of **\$10476**. The probability for obtaining our filters net profit of **\$27963** is $1.20 \cdot 10^{-5}$ which is **4.22** standard deviations from the bootstrap average. For our filter, in row 3 in Figure 3, the expected number of cases that we could obtain by pure chance that would match or exceed the **\$27963** is $23064 \times 1.20 \cdot 10^{-5} = 0.277$ where 23064 is the total number of different filters we looked at in this run. This number is much less than 1, so it is improbable that our result was due to pure chance.

Results

Table 1 below presents a table of the 205 in-sample and out-of-sample windows, the **Filter** selected, strategy inputs and the weekly out-of-sample profit/loss results using the filter described above.

Figure 1 presents a graph of the equity curve generated by using the filter on the 205 weeks ending 10/8/10 – 9/5/14 (note the first month starting 9/2/10 was part of the first 30 day in-sample period). The equity curves is plotted from Equity and Net Equity columns in Table 1. Plotted on the equity curves is the 2nd Order Polynomial curve. The blue line is the equity curve without commissions and the red dots on the blue line are new highs in equity. The brown line is the net equity curve with commissions and the green dots are the new highs in net equity.

Figure 2 presents the out-of-sample 1 minute bar chart of ES for 8/6/14 to 8/8/14 with the RMV Indicator and all the buy and sell signals for those dates.

Discussion of System Performance

In Figure 3 Row 3 of the spreadsheet filter output are some statistics that are of interest for our filter. An interesting statistic is **Blw**. Blw is the maximum number of weeks the OSNP equity curve failed to make a new high. Blw is 14 weeks for this filter. This means that 14 weeks was the longest time that the equity for this strategy failed to make a new equity high.

To see the effect of walk forward analysis, take a look at **Table 1**. Notice how the input parameters **N**, **vup**, **vdn** take sudden jumps from high to low and back . This is the walk forward process quickly adapting to changing volatility conditions in the in-sample sample. In addition, notice how often **N** changes from 20 to 60. When the data gets very noisy with a lot of spurious price movements, the lookback period, **N**, has to be higher. During other times when the noise level is not as much **N** can be lower to get onboard a trend faster.

In Figure 1, which presents a graph of the equity curve using the filter on the 205 weeks of out-of-sample data, notice how the equity curve follows the 2nd order polynomial trend line with an R^2 of 0.98. This R^2 dropped to 0.97 for the net equity curve.

Using this filter, the strategy was able to generate \$27963 net equity after commissions and slippage of \$20 trading one ES contract for 205 weeks. This period of time from 10/8/10 to 9/5/14 was a volatile market. Yet the RMV strategy was able to adapt quite well. From Table 1, the largest losing week was -\$1475 on the week ending 9/30/11 a very wild financial time and market week. The largest drawdown was -\$2638 from the week ending on 9/16/11 to 10/4/11.. However this drawdown only lasted four weeks and completely recovered and made a new equity high in another four weeks. The longest time between new equity highs was 14 weeks.

In observing Table 1 we can see that this strategy and filter made trades from a low of 0 or no trades/week to a high of 25 trades/week with an average of 4.3 trades/week. For the no trade weeks, the inputs found by the filter in the in-sample section generated no trades in the out-of-sample section.

Given 23 hour trading of the ES, restricting the strategy to trade only from 830am to 3:00pm CT caused the strategy to miss many profitable trends opportunities when Asia and then Europe opened trading in the early morning. Further research will include the A.M. time zones.

Disclaimer

The strategies, methods and indicators presented here are given for educational purposes only and should not be construed as investment advice. Be aware that the profitable performance presented here is based upon hypothetical trading with the benefit of hindsight and can in no way be assumed nor can it be claimed that the strategy and methods presented here will be profitable in the future or that they will not result in losses.

References

1. Rousseeuw, P.J., Leroy, A.M., (1987) "Robust Regression and Outlier Detection", New York, John Wiley & Sons.
2. Siegel, A.F. (1982), "Robust Regression using Repeated Medians." *Biometrika*. 69, pp242-244.
3. Efron, B., Tibshirani, R.J., (1993), "An Introduction to the Bootstrap", New York, Chapman & Hall/CRC.
4. Meyers, Dennis (2013), "The British Pound Cubed, Redux", Technical Analysis of Stocks & Commodities, Volume 31: January
5. Meyers, Dennis (2005) "The Polynomial Velocity System Applied To E-Mini 1min Bars using Walk Forward, Out-Of-Sample Analysis", Working Paper Sept/2005, www.meyersanalytics.com/articles.php
6. Meyers, Dennis (1998) "Surfing The Linear Regression Curve", Technical Analysis of Stocks & Commodities, Volume 16: May

Figure 2 Walk Forward Out-Of-Sample Performance Summary ES-Mini 1 min bars Robust Regression Velocity Strategy

ES-1 min bars 9/1/2010 - 9/5/2014 using the below filter on each in-sample segment. The input values *N*, *vup*, and *vdn* are the values found from applying the filter to the in-sample sample's optimization run.

In-sample Section Filter: $b50mDev|p<40|lr5|>10-mKr$

Where:

osnp = Weekly Out-of-sample net profit from strategy inputs chosen by In-sample Section filter

ollt = out-of-sample largest losing trade for that week from strategy inputs chosen by In-sample Section filter.

odd = Out-of-Sample closing trade drawdown for that week

ont = The number of trades in the out-of-sample week from strategy inputs chosen by In-sample Section filter.

Equity = running sum of the weekly out-of-sample profits(Osnp)

NetEq = running sum of weekly out-of-sample profits minus \$20*Ont

Note: Blank rows indicate that no out-of-sample trades were made that week

Week	In-Sample Dates		Out-Of-Sample Dates		osnp	Equity	NOnp\$20	NetEq	ollt	odd	ont	N	vup	vdn
1	09/01/10	to 10/01/10	10/04/10	to 10/08/10	825	825	745	745	(38)	(38)	4	20	1.6	1.4
2	09/08/10	to 10/08/10	10/11/10	to 10/15/10	275	1100	235	980	(25)	(25)	2	30	1.4	3.2
3	09/15/10	to 10/15/10	10/18/10	to 10/22/10	588	1688	408	1388	(275)	(463)	9	30	1.2	1.4
4	09/22/10	to 10/22/10	10/25/10	to 10/29/10	(600)	1088	(640)	748	(438)	(600)	2	70	1.4	3
5	09/29/10	to 10/29/10	11/01/10	to 11/05/10	838	1926	698	1446	(325)	(325)	7	20	1.2	1.6
6	10/06/10	to 11/05/10	11/08/10	to 11/12/10	588	2514	468	1914	(213)	(213)	6	30	1.2	1.4
7	10/13/10	to 11/12/10	11/15/10	to 11/19/10	(538)	1976	(878)	1036	(300)	(1113)	17	40	0.2	1
8	10/20/10	to 11/19/10	11/22/10	to 11/26/10	0	1976	(140)	896	(350)	(350)	7	30	1.2	1.8
9	10/27/10	to 11/26/10	11/29/10	to 12/03/10	200	2176	100	996	(488)	(488)	5	50	1.6	2.6
10	11/03/10	to 12/03/10	12/06/10	to 12/10/10		2176		996				20	1.6	2.4
11	11/10/10	to 12/10/10	12/13/10	to 12/17/10	125	2301	105	1101	0	0	1	20	1.6	3
12	11/17/10	to 12/17/10	12/20/10	to 12/24/10		2301		1101				20	1.4	3.2
13	11/24/10	to 12/24/10	12/27/10	to 12/31/10		2301		1101				30	1.2	1.8
14	12/01/10	to 12/31/10	01/03/11	to 01/07/11	(25)	2276	(205)	896	(400)	(800)	9	50	0.2	1.6
15	12/08/10	to 01/07/11	01/10/11	to 01/14/11	663	2939	543	1439	(188)	(188)	6	50	0.2	2.2
16	12/15/10	to 01/14/11	01/17/11	to 01/21/11	(375)	2564	(455)	984	(338)	(463)	4	20	0.6	2.6
17	12/22/10	to 01/21/11	01/24/11	to 01/28/11	550	3114	410	1394	(338)	(338)	7	40	0.2	2.4
18	12/29/10	to 01/28/11	01/31/11	to 02/04/11	1338	4452	1258	2652	0	0	4	60	0.8	2.6
19	01/05/11	to 02/04/11	02/07/11	to 02/11/11	300	4752	220	2872	(225)	(225)	4	60	1	2.2
20	01/12/11	to 02/11/11	02/14/11	to 02/18/11	88	4840	68	2940	0	0	1	50	1.2	3
21	01/19/11	to 02/18/11	02/21/11	to 02/25/11	(563)	4277	(663)	2277	(913)	(938)	5	60	1.2	2.6
22	01/26/11	to 02/25/11	02/28/11	to 03/04/11	988	5265	868	3145	(125)	(225)	6	70	1	1.4
23	02/02/11	to 03/04/11	03/07/11	to 03/11/11	38	5303	(82)	3063	(588)	(850)	6	30	2.4	1.4
24	02/09/11	to 03/11/11	03/14/11	to 03/18/11	125	5428	(35)	3028	(325)	(375)	8	40	2.2	1.4
25	02/16/11	to 03/18/11	03/21/11	to 03/25/11	(25)	5403	(45)	2983	0	0	1	40	1.6	3.6
26	02/23/11	to 03/25/11	03/28/11	to 04/01/11		5403		2983				40	2.6	1.6
27	03/02/11	to 04/01/11	04/04/11	to 04/08/11		5403		2983				40	2.2	2.4
28	03/09/11	to 04/08/11	04/11/11	to 04/15/11	(463)	4940	(483)	2500	0	0	1	50	1.8	2.4
29	03/16/11	to 04/15/11	04/18/11	to 04/22/11		4940		2500				30	1.6	3
30	03/23/11	to 04/22/11	04/25/11	to 04/29/11	138	5078	98	2598	(88)	(88)	2	30	1.2	3
31	03/30/11	to 04/29/11	05/02/11	to 05/06/11	(363)	4715	(923)	1675	(300)	(825)	28	20	1	1
32	04/06/11	to 05/06/11	05/09/11	to 05/13/11	950	5665	830	2505	(338)	(338)	6	70	1.4	0.8
33	04/13/11	to 05/13/11	05/16/11	to 05/20/11	(88)	5577	(168)	2337	(225)	(425)	4	50	2.6	1.2

Week	In-Sample Dates		Out-Of-Sample Dates		osnp	Equity	NOnp\$20	NetEq	ollt	odd	ont	N	vup	vdn
34	04/20/11	to 05/20/11	05/23/11	to 05/27/11	(375)	5202	(435)	1902	(525)	(525)	3	50	2.6	1
35	04/27/11	to 05/27/11	05/30/11	to 06/03/11	975	6177	895	2797	(250)	(250)	4	50	2	1
36	05/04/11	to 06/03/11	06/06/11	to 06/10/11	(188)	5989	(228)	2569	(288)	(288)	2	20	2.4	1.8
37	05/11/11	to 06/10/11	06/13/11	to 06/17/11	513	6502	453	3022	(288)	(288)	3	70	2	2.2
38	05/18/11	to 06/17/11	06/20/11	to 06/24/11	(88)	6414	(228)	2794	(475)	(838)	7	30	2	1
39	05/25/11	to 06/24/11	06/27/11	to 07/01/11	1000	7414	920	3714	(175)	(175)	4	40	2	1.4
40	06/01/11	to 07/01/11	07/04/11	to 07/08/11	25	7439	(15)	3699	(150)	(150)	2	70	1.8	2.2
41	06/08/11	to 07/08/11	07/11/11	to 07/15/11	88	7527	(112)	3587	(375)	(600)	10	30	1.6	1.6
42	06/15/11	to 07/15/11	07/18/11	to 07/22/11	125	7652	65	3652	(263)	(263)	3	40	2.6	1.6
43	06/22/11	to 07/22/11	07/25/11	to 07/29/11	(613)	7039	(713)	2939	(588)	(1088)	5	60	2.6	2.2
44	06/29/11	to 07/29/11	08/01/11	to 08/05/11	2538	9577	2318	5257	(638)	(1925)	11	50	3.2	1.2
45	07/06/11	to 08/05/11	08/08/11	to 08/12/11	2288	11865	1888	7145	(663)	(2250)	20	30	3.2	2.6
46	07/13/11	to 08/12/11	08/15/11	to 08/19/11	625	12490	505	7650	(675)	(838)	6	40	3	3
47	07/20/11	to 08/19/11	08/22/11	to 08/26/11	100	12590	0	7650	(700)	(700)	5	60	3.4	2.8
48	07/27/11	to 08/26/11	08/29/11	to 09/02/11	588	13178	508	8158	(375)	(400)	4	60	2.6	2.6
49	08/03/11	to 09/02/11	09/05/11	to 09/09/11	525	13703	465	8623	(863)	(863)	3	60	2.8	3
50	08/10/11	to 09/09/11	09/12/11	to 09/16/11	800	14503	740	9363	0	0	3	30	3.6	3.4
51	08/17/11	to 09/16/11	09/19/11	to 09/23/11	(750)	13753	(850)	8513	(1113)	(1700)	5	30	3.6	3.4
52	08/24/11	to 09/23/11	09/26/11	to 09/30/11	(1475)	12278	(1575)	6938	(1513)	(1813)	5	30	3.6	3.6
53	08/31/11	to 09/30/11	10/03/11	to 10/07/11	(325)	11953	(465)	6473	(2013)	(2450)	7	30	2.8	3.6
54	09/07/11	to 10/07/11	10/10/11	to 10/14/11	(88)	11865	(128)	6345	(250)	(250)	2	30	2.4	3.4
55	09/14/11	to 10/14/11	10/17/11	to 10/21/11	25	11890	(75)	6270	(950)	(950)	5	30	2.4	3.4
56	09/21/11	to 10/21/11	10/24/11	to 10/28/11	1725	13615	1625	7895	(188)	(188)	5	70	2.6	2.2
57	09/28/11	to 10/28/11	10/31/11	to 11/04/11	700	14315	620	8515	(300)	(300)	4	60	3.2	3.6
58	10/05/11	to 11/04/11	11/07/11	to 11/11/11	1613	15928	1533	10048	0	0	4	70	2.6	3.6
59	10/12/11	to 11/11/11	11/14/11	to 11/18/11	125	16053	85	10133	(100)	(100)	2	70	2.6	3.6
60	10/19/11	to 11/18/11	11/21/11	to 11/25/11	413	16466	373	10506	0	0	2	70	3.2	2.8
61	10/26/11	to 11/25/11	11/28/11	to 12/02/11	(250)	16216	(310)	10196	(663)	(663)	3	70	2.6	3.6
62	11/02/11	to 12/02/11	12/05/11	to 12/09/11	488	16704	408	10604	(575)	(875)	4	60	3.2	2.8
63	11/09/11	to 12/09/11	12/12/11	to 12/16/11	1425	18129	1325	11929	(13)	(13)	5	40	3.2	1.4
64	11/16/11	to 12/16/11	12/19/11	to 12/23/11	488	18617	468	12397	0	0	1	70	3.6	2.6
65	11/23/11	to 12/23/11	12/26/11	to 12/30/11		18617		12397				70	3.6	2.4
66	11/30/11	to 12/30/11	01/02/12	to 01/06/12	(663)	17954	(683)	11714	0	0	1	40	2.4	2.2
67	12/07/11	to 01/06/12	01/09/12	to 01/13/12	(450)	17504	(490)	11224	(238)	(450)	2	70	3.2	2
68	12/14/11	to 01/13/12	01/16/12	to 01/20/12	175	17679	115	11339	(363)	(363)	3	50	1	1.8
69	12/21/11	to 01/20/12	01/23/12	to 01/27/12	13	17692	(147)	11192	(413)	(563)	8	70	0.6	1.8
70	12/28/11	to 01/27/12	01/30/12	to 02/03/12	438	18130	318	11510	(200)	(200)	6	70	0.2	2.6
71	01/04/12	to 02/03/12	02/06/12	to 02/10/12	750	18880	650	12160	0	0	5	20	0.2	2.4
72	01/11/12	to 02/10/12	02/13/12	to 02/17/12	375	19255	275	12435	(500)	(500)	5	70	0.2	2.4
73	01/18/12	to 02/17/12	02/20/12	to 02/24/12	(50)	19205	(90)	12345	(213)	(213)	2	40	1.2	2
74	01/25/12	to 02/24/12	02/27/12	to 03/02/12	(238)	18967	(298)	12047	(613)	(638)	3	50	1.4	3.2
75	02/01/12	to 03/02/12	03/05/12	to 03/09/12	(200)	18767	(280)	11767	(138)	(200)	4	30	1.2	2
76	02/08/12	to 03/09/12	03/12/12	to 03/16/12	963	19730	943	12710	0	0	1	70	1.6	1.8
77	02/15/12	to 03/16/12	03/19/12	to 03/23/12		19730		12710				20	1.6	3
78	02/22/12	to 03/23/12	03/26/12	to 03/30/12	75	19805	35	12745	(325)	(325)	2	50	1.6	2
79	02/29/12	to 03/30/12	04/02/12	to 04/06/12	475	20280	435	13180	0	0	2	50	1.6	1.6
80	03/07/12	to 04/06/12	04/09/12	to 04/13/12	263	20543	223	13403	(250)	(250)	2	20	1.2	3.2
81	03/14/12	to 04/13/12	04/16/12	to 04/20/12	(338)	20205	(418)	12985	(588)	(875)	4	60	1.6	2
82	03/21/12	to 04/20/12	04/23/12	to 04/27/12	(188)	20017	(228)	12757	(250)	(250)	2	50	1.8	2.4
83	03/28/12	to 04/27/12	04/30/12	to 05/04/12	500	20517	380	13137	(200)	(213)	6	70	2.8	0.6
84	04/04/12	to 05/04/12	05/07/12	to 05/11/12	175	20692	75	13212	(325)	(325)	5	30	1.4	1.8
85	04/11/12	to 05/11/12	05/14/12	to 05/18/12	2088	22780	1988	15200	0	0	5	70	3.2	1.2
86	04/18/12	to 05/18/12	05/21/12	to 05/25/12	(750)	22030	(810)	14390	(525)	(750)	3	50	2.8	2

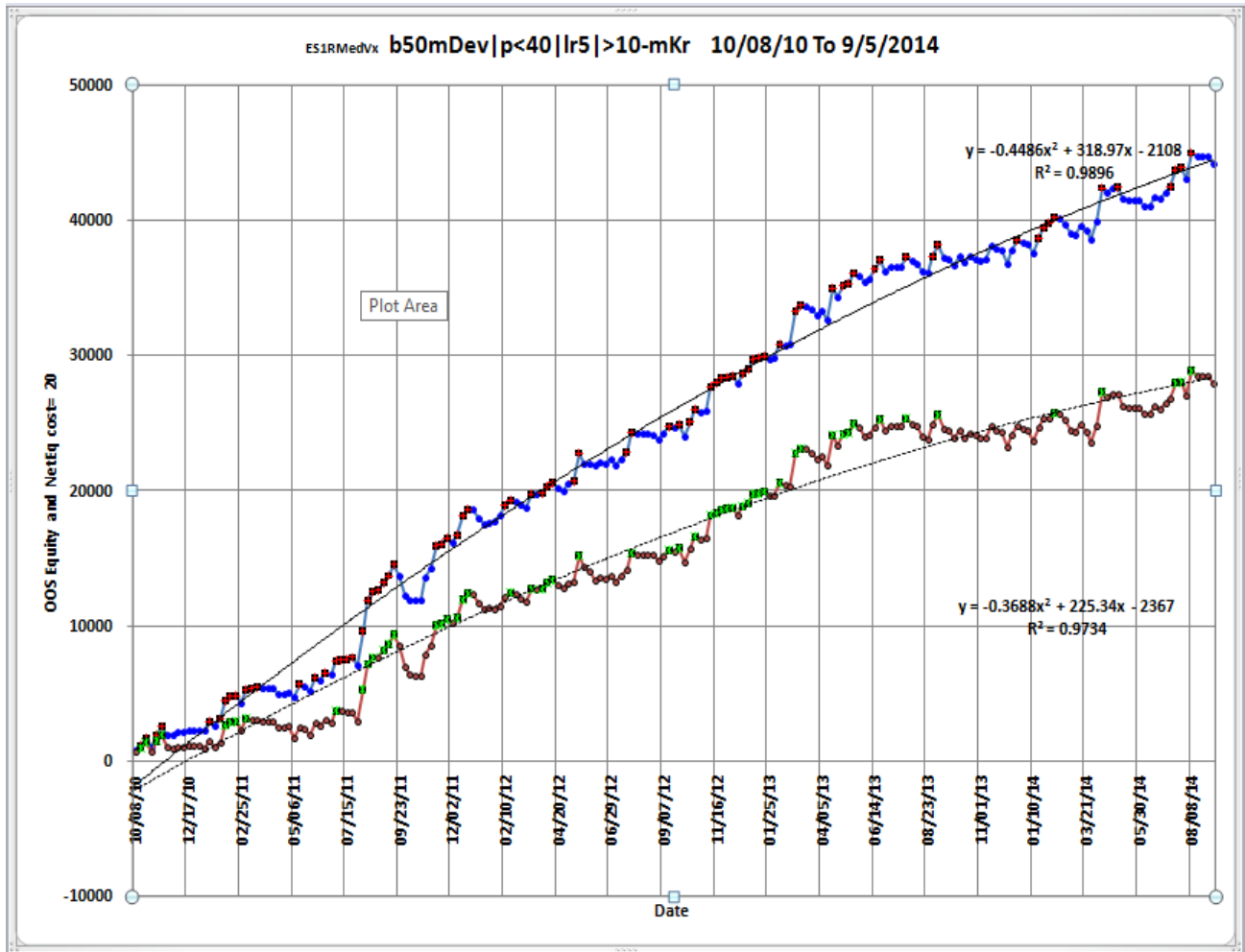
Week	In-Sample Dates		Out-Of-Sample Dates		osnp	Equity	NOnp\$20	NetEq	oilt	odd	ont	N	vup	vdn
87	04/25/12	to 05/25/12	05/28/12	to 06/01/12	(63)	21967	(363)	14027	(300)	(763)	15	20	1.6	1.2
88	05/02/12	to 06/01/12	06/04/12	to 06/08/12	(113)	21854	(613)	13414	(238)	(613)	25	40	0.8	0.4
89	05/09/12	to 06/08/12	06/11/12	to 06/15/12	200	22054	180	13594	0	0	1	70	2.4	2.6
90	05/16/12	to 06/15/12	06/18/12	to 06/22/12	(25)	22029	(65)	13529	(163)	(163)	2	60	2.2	3.2
91	05/23/12	to 06/22/12	06/25/12	to 06/29/12	263	22292	223	13752	0	0	2	20	1.8	3.6
92	05/30/12	to 06/29/12	07/02/12	to 07/06/12	(438)	21854	(458)	13294	0	0	1	20	2	2.8
93	06/06/12	to 07/06/12	07/09/12	to 07/13/12	513	22367	433	13727	(200)	(200)	4	40	1.2	2.6
94	06/13/12	to 07/13/12	07/16/12	to 07/20/12	450	22817	390	14117	(100)	(100)	3	50	1.6	2.8
95	06/20/12	to 07/20/12	07/23/12	to 07/27/12	1450	24267	1250	15367	(200)	(200)	10	40	0.8	1.6
96	06/27/12	to 07/27/12	07/30/12	to 08/03/12	(50)	24217	(90)	15277	(200)	(200)	2	30	1.6	3
97	07/04/12	to 08/03/12	08/06/12	to 08/10/12		24217		15277				20	1.8	3.4
98	07/11/12	to 08/10/12	08/13/12	to 08/17/12		24217		15277				20	2	2.4
99	07/18/12	to 08/17/12	08/20/12	to 08/24/12	(38)	24179	(58)	15219	0	0	1	30	1.8	1.8
100	07/25/12	to 08/24/12	08/27/12	to 08/31/12	(350)	23829	(390)	14829	(300)	(350)	2	40	2.6	1.2
101	08/01/12	to 08/31/12	09/03/12	to 09/07/12	425	24254	365	15194	(63)	(63)	3	20	1.2	3.6
102	08/08/12	to 09/07/12	09/10/12	to 09/14/12	438	24692	378	15572	(338)	(338)	3	20	1.2	3.6
103	08/15/12	to 09/14/12	09/17/12	to 09/21/12	(50)	24642	(70)	15502	0	0	1	20	1.6	3.6
104	08/22/12	to 09/21/12	09/24/12	to 09/28/12	263	24905	223	15725	0	0	2	20	1.4	3.6
105	08/29/12	to 09/28/12	10/01/12	to 10/05/12	(913)	23992	(993)	14732	(638)	(913)	4	40	1.4	3.4
106	09/05/12	to 10/05/12	10/08/12	to 10/12/12	1063	25055	983	15715	(38)	(38)	4	30	3.6	1
107	09/12/12	to 10/12/12	10/15/12	to 10/19/12	925	25980	885	16600	0	0	2	70	3.2	1.2
108	09/19/12	to 10/19/12	10/22/12	to 10/26/12	(200)	25780	(260)	16340	(188)	(200)	3	20	2.8	1.8
109	09/26/12	to 10/26/12	10/29/12	to 11/02/12	175	25955	155	16495	0	0	1	20	3.6	1.8
110	10/03/12	to 11/02/12	11/05/12	to 11/09/12	1750	27705	1690	18185	0	0	3	20	3.4	1.6
111	10/10/12	to 11/09/12	11/12/12	to 11/16/12	250	27955	130	18315	(650)	(988)	6	60	3.4	1
112	10/17/12	to 11/16/12	11/19/12	to 11/23/12	325	28280	265	18580	(438)	(438)	3	70	1.4	2
113	10/24/12	to 11/23/12	11/26/12	to 11/30/12	88	28368	68	18648	0	0	1	30	3	2
114	10/31/12	to 11/30/12	12/03/12	to 12/07/12	75	28443	55	18703	0	0	1	70	2	2.4
115	11/07/12	to 12/07/12	12/10/12	to 12/14/12	(475)	27968	(515)	18188	(450)	(475)	2	60	1.8	2
116	11/14/12	to 12/14/12	12/17/12	to 12/21/12	688	28656	628	18816	(113)	(113)	3	70	2	1.8
117	11/21/12	to 12/21/12	12/24/12	to 12/28/12	313	28969	233	19049	(163)	(325)	4	70	2	1.8
118	11/28/12	to 12/28/12	12/31/12	to 01/04/13	713	29682	673	19722	0	0	2	30	1.8	2
119	12/05/12	to 01/04/13	01/07/13	to 01/11/13	113	29795	73	19795	(113)	(113)	2	70	1.2	2.4
120	12/12/12	to 01/11/13	01/14/13	to 01/18/13	150	29945	130	19925	0	0	1	60	1.4	2.6
121	12/19/12	to 01/18/13	01/21/13	to 01/25/13	(225)	29720	(265)	19660	(300)	(300)	2	50	1.4	3
122	12/26/12	to 01/25/13	01/28/13	to 02/01/13	138	29858	38	19698	(225)	(438)	5	40	0.4	3.4
123	01/02/13	to 02/01/13	02/04/13	to 02/08/13	988	30846	868	20566	(50)	(50)	6	70	0.6	1.6
124	01/09/13	to 02/08/13	02/11/13	to 02/15/13	(88)	30758	(168)	20398	(138)	(163)	4	60	0.8	2.6
125	01/16/13	to 02/15/13	02/18/13	to 02/22/13	50	30808	(30)	20368	(175)	(325)	4	50	1.2	1.8
126	01/23/13	to 02/22/13	02/25/13	to 03/01/13	2475	33283	2375	22743	(188)	(188)	5	60	1.2	2.2
127	01/30/13	to 03/01/13	03/04/13	to 03/08/13	375	33658	355	23098	0	0	1	20	1.4	2.4
128	02/06/13	to 03/08/13	03/11/13	to 03/15/13		33658		23098				20	1.4	2.2
129	02/13/13	to 03/15/13	03/18/13	to 03/22/13	(225)	33433	(285)	22813	(250)	(350)	3	40	1.4	1.8
130	02/20/13	to 03/22/13	03/25/13	to 03/29/13	(450)	32983	(490)	22323	(625)	(625)	2	60	1.6	3.2
131	02/27/13	to 03/29/13	04/01/13	to 04/05/13	338	33321	278	22601	(88)	(100)	3	30	1.2	2
132	03/06/13	to 04/05/13	04/08/13	to 04/12/13	(725)	32596	(765)	21836	(375)	(725)	2	30	1.8	1.6
133	03/13/13	to 04/12/13	04/15/13	to 04/19/13	2363	34959	2283	24119	0	0	4	40	2	1.4
134	03/20/13	to 04/19/13	04/22/13	to 04/26/13	(663)	34296	(723)	23396	(575)	(663)	3	30	2.2	1.4
135	03/27/13	to 04/26/13	04/29/13	to 05/03/13	825	35121	745	24141	(38)	(38)	4	40	1.2	2.4
136	04/03/13	to 05/03/13	05/06/13	to 05/10/13	188	35309	108	24249	(288)	(288)	4	50	1	3.2
137	04/10/13	to 05/10/13	05/13/13	to 05/17/13	750	36059	690	24939	0	0	3	20	1.6	3.6
138	04/17/13	to 05/17/13	05/20/13	to 05/24/13	(138)	35921	(278)	24661	(513)	(650)	7	50	1.4	2.2
139	04/24/13	to 05/24/13	05/27/13	to 05/31/13	(500)	35421	(620)	24041	(700)	(1113)	6	70	1.8	1.8
140	05/01/13	to 05/31/13	06/03/13	to 06/07/13	225	35646	105	24146	(863)	(863)	6	60	1.6	1.4

Week	In-Sample Dates		Out-Of-Sample Dates		osnp	Equity	NOnp\$20	NetEq	oilt	odd	ont	N	vup	vdn
141	05/08/13	to 06/07/13	06/10/13	to 06/14/13	700	36346	520	24666	(313)	(938)	9	60	1.6	1.2
142	05/15/13	to 06/14/13	06/17/13	to 06/21/13	700	37046	620	25286	(638)	(638)	4	20	3.6	2.6
143	05/22/13	to 06/21/13	06/24/13	to 06/28/13	(763)	36283	(823)	24463	(375)	(763)	3	20	3.6	2.2
144	05/29/13	to 06/28/13	07/01/13	to 07/05/13	300	36583	280	24743	0	0	1	20	2.8	3.6
145	06/05/13	to 07/05/13	07/08/13	to 07/12/13		36583		24743				20	2.8	3.6
146	06/12/13	to 07/12/13	07/15/13	to 07/19/13		36583		24743				20	2.6	3.6
147	06/19/13	to 07/19/13	07/22/13	to 07/26/13	663	37246	563	25306	(225)	(338)	5	30	0.6	3.6
148	06/26/13	to 07/26/13	07/29/13	to 08/02/13	(288)	36958	(388)	24918	(513)	(663)	5	40	0.8	3.2
149	07/03/13	to 08/02/13	08/05/13	to 08/09/13	(113)	36845	(153)	24765	(88)	(113)	2	50	1	3.4
150	07/10/13	to 08/09/13	08/12/13	to 08/16/13	(588)	36257	(708)	24057	(263)	(838)	6	70	0.2	2.6
151	07/17/13	to 08/16/13	08/19/13	to 08/23/13	(175)	36082	(275)	23782	(700)	(700)	5	50	0.6	2.6
152	07/24/13	to 08/23/13	08/26/13	to 08/30/13	1225	37307	1145	24927	0	0	4	60	2	1
153	07/31/13	to 08/30/13	09/02/13	to 09/06/13	850	38157	690	25617	(113)	(150)	8	30	1.6	1
154	08/07/13	to 09/06/13	09/09/13	to 09/13/13	(925)	37232	(1025)	24592	(513)	(925)	5	50	2	0.8
155	08/14/13	to 09/13/13	09/16/13	to 09/20/13	(75)	37157	(115)	24477	(213)	(213)	2	50	1.8	2.2
156	08/21/13	to 09/20/13	09/23/13	to 09/27/13	(463)	36694	(583)	23894	(250)	(463)	6	30	1.8	1.6
157	08/28/13	to 09/27/13	09/30/13	to 10/04/13	650	37344	590	24484	0	0	3	20	1.6	3.2
158	09/04/13	to 10/04/13	10/07/13	to 10/11/13	(425)	36919	(565)	23919	(375)	(888)	7	20	2.2	2
159	09/11/13	to 10/11/13	10/14/13	to 10/18/13	388	37307	348	24267	0	0	2	40	2.6	1.4
160	09/18/13	to 10/18/13	10/21/13	to 10/25/13	(138)	37169	(158)	24109	0	0	1	20	2.6	2.2
161	09/25/13	to 10/25/13	10/28/13	to 11/01/13	(188)	36981	(248)	23861	(200)	(200)	3	20	3	2
162	10/02/13	to 11/01/13	11/04/13	to 11/08/13	175	37156	75	23936	(1113)	(1213)	5	40	0.6	3.4
163	10/09/13	to 11/08/13	11/11/13	to 11/15/13	963	38119	883	24819	(138)	(213)	4	20	0.8	2.8
164	10/16/13	to 11/15/13	11/18/13	to 11/22/13	(263)	37856	(363)	24456	(475)	(950)	5	20	0.8	3.6
165	10/23/13	to 11/22/13	11/25/13	to 11/29/13	(38)	37818	(78)	24378	(25)	(38)	2	20	1.2	1.8
166	10/30/13	to 11/29/13	12/02/13	to 12/06/13	(975)	36843	(1115)	23263	(525)	(1363)	7	60	3	1.4
167	11/06/13	to 12/06/13	12/09/13	to 12/13/13	900	37743	840	24103	(25)	(25)	3	20	2.2	1.2
168	11/13/13	to 12/13/13	12/16/13	to 12/20/13	775	38518	675	24778	(475)	(475)	5	40	2.4	1.2
169	11/20/13	to 12/20/13	12/23/13	to 12/27/13	(125)	38393	(165)	24613	(138)	(138)	2	30	2.2	0.6
170	11/27/13	to 12/27/13	12/30/13	to 01/03/14	(100)	38293	(160)	24453	(363)	(363)	3	70	1.2	2.2
171	12/04/13	to 01/03/14	01/06/14	to 01/10/14	(713)	37580	(773)	23680	(325)	(713)	3	70	2.2	1.4
172	12/11/13	to 01/10/14	01/13/14	to 01/17/14	1100	38680	1020	24700	(75)	(125)	4	50	3	0.4
173	12/18/13	to 01/17/14	01/20/14	to 01/24/14	775	39455	675	25375	(263)	(488)	5	70	2.4	0.2
174	12/25/13	to 01/24/14	01/27/14	to 01/31/14	313	39768	33	25408	(425)	(975)	14	30	1.4	1.8
175	01/01/14	to 01/31/14	02/03/14	to 02/07/14	400	40168	320	25728	(838)	(1338)	4	30	2.8	1.8
176	01/08/14	to 02/07/14	02/10/14	to 02/14/14		40168		25728				70	3	2.2
177	01/15/14	to 02/14/14	02/17/14	to 02/21/14	(488)	39680	(528)	25200	(888)	(888)	2	20	2.2	3.6
178	01/22/14	to 02/21/14	02/24/14	to 02/28/14	(663)	39017	(723)	24477	(300)	(663)	3	40	2.2	3.6
179	01/29/14	to 02/28/14	03/03/14	to 03/07/14	(50)	38967	(110)	24367	(375)	(375)	3	60	2.8	2
180	02/05/14	to 03/07/14	03/10/14	to 03/14/14	625	39592	525	24892	(500)	(500)	5	50	3.4	1.4
181	02/12/14	to 03/14/14	03/17/14	to 03/21/14	(338)	39254	(518)	24374	(500)	(1350)	9	70	2.2	0.8
182	02/19/14	to 03/21/14	03/24/14	to 03/28/14	(650)	38604	(810)	23564	(413)	(1113)	8	70	2.4	1
183	02/26/14	to 03/28/14	03/31/14	to 04/04/14	1300	39904	1280	24844	0	0	1	30	3.6	1.8
184	03/05/14	to 04/04/14	04/07/14	to 04/11/14	2488	42392	2428	27272	0	0	3	40	3.6	2.2
185	03/12/14	to 04/11/14	04/14/14	to 04/18/14	(288)	42104	(348)	26924	(663)	(725)	3	20	3.6	2.6
186	03/19/14	to 04/18/14	04/21/14	to 04/25/14	263	42367	243	27167	0	0	1	50	3.2	2.8
187	03/26/14	to 04/25/14	04/28/14	to 05/02/14	75	42442	35	27202	0	0	2	20	3.6	2.2
188	04/02/14	to 05/02/14	05/05/14	to 05/09/14	(875)	41567	(915)	26287	(788)	(875)	2	20	3.6	2.2
189	04/09/14	to 05/09/14	05/12/14	to 05/16/14	(113)	41454	(133)	26154	0	0	1	20	2.6	3
190	04/16/14	to 05/16/14	05/19/14	to 05/23/14		41454		26154				20	2	2.4
191	04/23/14	to 05/23/14	05/26/14	to 05/30/14		41454		26154				30	2	2.4
192	04/30/14	to 05/30/14	06/02/14	to 06/06/14	(388)	41066	(408)	25746	0	0	1	20	1.8	2.4
193	05/07/14	to 06/06/14	06/09/14	to 06/13/14	(25)	41041	(45)	25701	0	0	1	50	1.2	2
194	05/14/14	to 06/13/14	06/16/14	to 06/20/14	638	41679	578	26279	(150)	(150)	3	30	1.4	1.6

Week	In-Sample Dates		Out-Of-Sample Dates		osnp	Equity	NOnp\$20	NetEq	ollt	odd	ont	N	vup	vdn
195	05/21/14	to 06/20/14	06/23/14	to 06/27/14	(125)	41554	(265)	26014	(375)	(375)	7	20	1.4	1.8
196	05/28/14	to 06/27/14	06/30/14	to 07/04/14	525	42079	485	26499	0	0	2	60	1.8	1.4
197	06/04/14	to 07/04/14	07/07/14	to 07/11/14	413	42492	313	26812	(175)	(225)	5	20	1.4	3.6
198	06/11/14	to 07/11/14	07/14/14	to 07/18/14	1213	43705	1133	27945	(100)	(113)	4	60	2	1.4
199	06/18/14	to 07/18/14	07/21/14	to 07/25/14	150	43855	90	28035	(75)	(75)	3	60	1.4	1.2
200	06/25/14	to 07/25/14	07/28/14	to 08/01/14	(800)	43055	(1020)	27015	(488)	(1675)	11	70	1	1.4
201	07/02/14	to 08/01/14	08/04/14	to 08/08/14	1913	44968	1853	28868	0	0	3	20	2.6	2.2
202	07/09/14	to 08/08/14	08/11/14	to 08/15/14	(275)	44693	(335)	28533	(188)	(325)	3	50	2	2.6
203	07/16/14	to 08/15/14	08/18/14	to 08/22/14		44693		28533				30	3	2.2
204	07/23/14	to 08/22/14	08/25/14	to 08/29/14		44693		28533				20	3.2	2
205	07/30/14	to 08/29/14	09/01/14	to 09/05/14	(550)	44143	(570)	27963	0	0	1	50	2	2.8

Figure 1 Graph of Robust Regression Velocity Strategy Net Equity Applying the Walk Forward Filter Each Week On ES 1min Bar Prices 10/8/2010 to 9/5/2014

Note: The blue line is the equity curve without commissions and the red dots on the blue line are new highs in equity. The brown line is the equity curve with commissions and the green dots are the new highs in net equity.



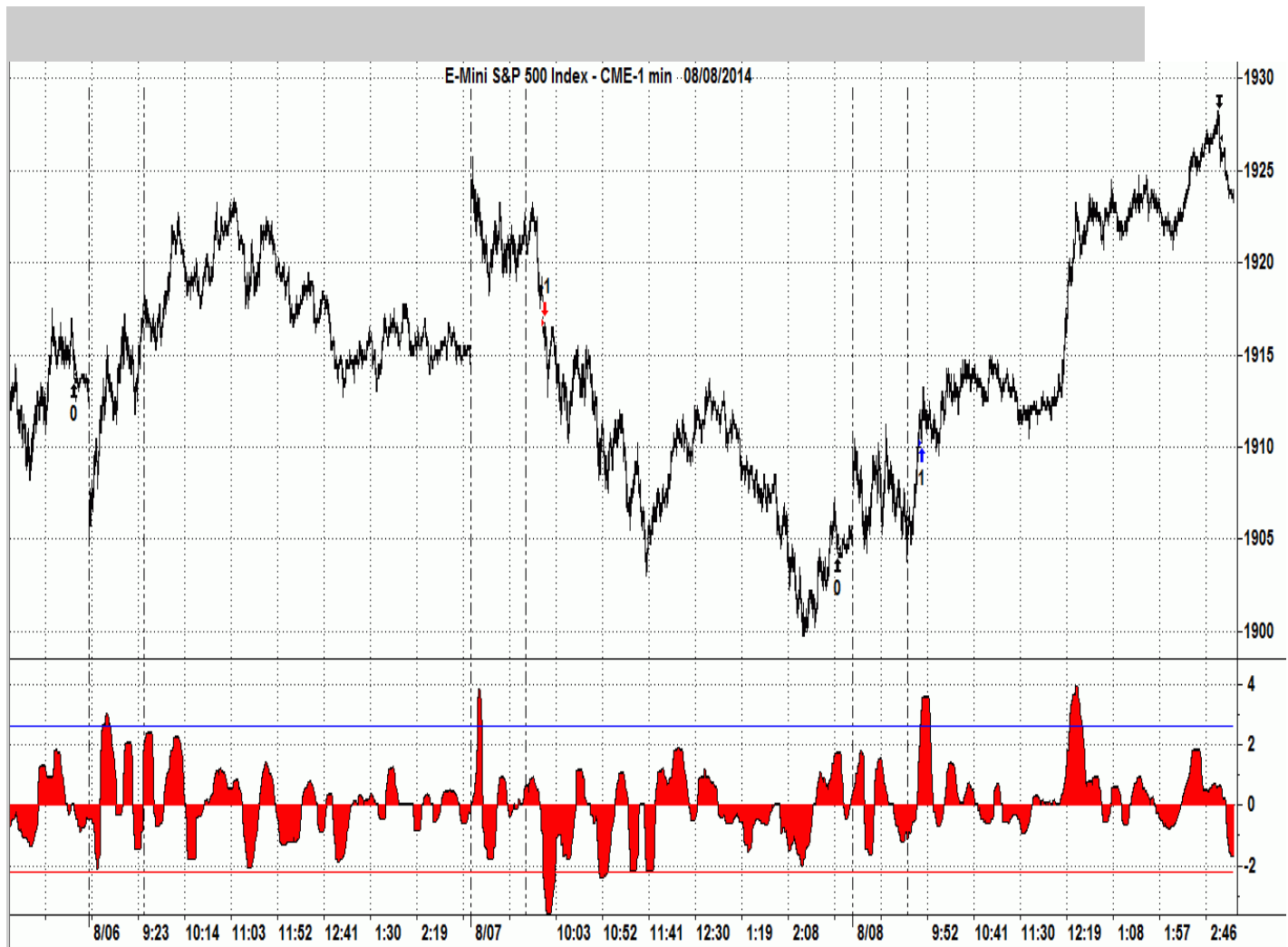


Figure 2 Walk Forward Out-Of-Sample Performance Summary for ES1 Robust Regression Velocity Strategy 1 minute bar chart from 5/1/13-5/3/2013

Figure 3 Partial output of the Walk Forward Metric Performance Explorer (WFME)
ES-Mini 1 min bars Repeated Median Velocity System

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
1	ES1RMedVx	s10/08/10	e09/05/14	#205	AnyTnp				a[16284]	s10476	f23064										c=\$20	
2	Filter-Metric	tOnp	aOnp	aOTrd	aO#T	B0	%P	t	std	LLp	eqDD	lr	#	eqTrn	eqV^3	eqR2	Dev^2	Blw	BE	eff	tOnpNet	Prob
3	b50mDev p<40 lr5 >10-mkr	44143	236	54.6	4.3	(0.8)	59	4.62	699	-1475	-2638	4	187	227	43	98	1970	14	35.1	0.63	27963	1.20E-05
4	t50eq2b1 >10-mkr	43362	228	56	4.1	(1.2)	61	3.67	857	-2288	-5577	5	190	207	63	90	4143	37	56.4	0.47	27882	1.24E-05
5	b50mDev p<40 lr3 >10-mkr	41159	220	55	4	(1.2)	59	4.21	715	-1475	-2638	4	187	196	72	96	2283	14	42.2	0.57	26199	2.50E-05
6	t50mLb p<30 lr5-eq2A	34941	228	77.1	3	(1.1)	58	3.35	843	-1713	-3088	5	153	191	166	92	3358	23	54.5	4.59	25881	2.85E-05
7	t50eq2R2 p<20 lr3-mkr	36897	235	66.2	3.5	(1.6)	59	3.57	824	-1575	-3038	4	157	205	27	92	3576	21	49.2	0.9	25757	3.00E-05

The WFME Filter Output Columns are defined as follows:

Row 1 ES1RMedVx is the strategy abbreviation, First OOS Week End Date(10/8/110), Last OOS Week End Date(9/5/14), **Number of weeks**(#205) **a**=average of bootstrap random picks. **s**= standard deviation of bootstrap random picks. **f**=number of different filters examined. **c**= slippage and round trip trade cost(c=\$20).

Filter = The filter that was run. Row 3 filter **b50mDev|p<40|lr5|>10-mkr**

The **b50mDev|p<40|lr5|>10-mkr** filter produced the following average 205 week statistics on row 3.

tOnp = Total out-of-sample(oos) net profit for these 205 weeks.

aOsp = Average oos net profit for the 205 weeks

aOTrd = Average oos profit per trade

aO#T = Average number of oos trades per week

B0 = The 205 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 205 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 205 weekly oos profits

llp = The largest losing oos period(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 weekly result. Note for some weeks there can be no strategy inputs that satisfy a given filter's criteria.

eqTrn = The straight line trend of the oos gross profit equity curve in \$/week.

eqV^3 = The ending velocity of 3rd order polynomial that is fit to the equity curve

EqR2 = The correlation coefficient(r^2) of a straight line fit to the equity curve

Dev^2 = A measure of equity curve smoothness. The square root of the average [(equity curve minus a straight line)²]

Blw = The maximum number of weeks the oos equity curve failed to make a new high.

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 98% probability that your oos equity is above zero.

eff = Efficiency. The average daily out-of-sample profit divided by the average daily in-sample profit.

tOnpNet = Total out-of-sample net profit(tOnpNet) minus the total trade cost.
$$tOnpNet = tOnp - (\text{Number of trade weeks}) * aOnT * \text{Cost}.$$

Prob = the probability that the filter's tOnpNet was due to pure chance.